Génie lexico-sémantique multilingue contributif

Mathieu Mangeot

Mémoire en vue de l'habilitation à diriger des recherches

Présenté à
L'Université Savoie Mont Blanc

Spécialité
Informatique

Soutenue par
Mathieu MANGEOT
Université Savoie Mont Blanc, Laboratoire d'Informatique de Grenoble

Le 5 décembre 2019

Génie lexico-sémantique multilingue contributif

Composition du Jury

Christian Boitet	Université Grenoble Alpes – LIG	Examinateur
Antoine Chalvin	INALCO – CEE	Examinateur
Annie Foret	Université Rennes 1 – IRISA	Rapporteure
Olivier Kraif	Université Grenoble Alpes – LIDILEM	Examinateur
Denis Maurel	Université de Tours – LIFAT	Rapporteur
Christophe Roche	Université Savoie Mont Blanc – LISTIC	Examinateur
Max Silberztein	Université de Franche Comté – ELLIADD	Rapporteur

Avant-propos

La tradition scientifique veut que, dans leurs écrits, les auteurs soient objectifs et n'évoquent pas leur histoire personnelle. Cependant, il m'est apparu nécessaire de donner quelques détails qui pourront aider le lecteur à comprendre mon fort intérêt, voire ma passion pour les environnements informatiques de manipulation de dictionnaires.

Je me suis très tôt intéressé à cet objet si particulier, si riche, jouant le rôle de passerelle vers de nouvelles connaissances.

Ma passion pour l'étude des langues étrangères est née en 1983, lors de mes années de Cours Moyen 1ère et 2ème année puisque j'ai eu la chance d'assister à des cours d'italien inclus dans le temps scolaire. Ces cours étaient mis en place par l'état italien à destination des écoliers savoyards, département marqué par une forte concentration d'émigrés italiens et par une histoire partagée à travers le royaume de Piémont-Sardaigne jusqu'en 1860, année du rattachement de la Savoie à la France. Depuis ce jour, ma passion pour les dictionnaires jusqu'à présent monolingues s'est étendue aux dictionnaires bilingues puis aux ressources lexicales de manière générale. À tel point que mon instituteur de CM2 m'avait surnommé « Monsieur Dictionnaire » !

À cette époque, j'ai été initié au khmer par un camarade de classe d'origine cambodgienne. Par rapport à l'italien, j'ai déjà pu constater la difficulté d'apprendre une langue très éloignée du français avec un alphabet radicalement différent.

Mon autre passion pour l'informatique s'est déclenchée quelques années après, en 1986 lorsque mes parents ont acheté (à crédit) un ordinateur Macintosh Plus. J'ai commencé à y programmer de manière autodidacte de petits algorithmes en Pascal pour résoudre des équations mathématiques.

Au collège, puis au lycée, j'ai choisi l'anglais en première langue puis l'espagnol en deuxième langue. Ce choix était avant tout utilitaire : les pays anglophones et hispanophones sont majoritaires. Avec mes connaissances actuelles en linguistique, j'aurais certainement choisi d'autres langues plus éloignées du français et moins accessibles comme l'allemand ou le russe.

Pendant deux ans, j'ai également étudié le latin que j'ai abandonné pensant qu'il était inutile d'étudier une langue morte. La suite de mes apprentissages linguistiques m'a prouvé que c'était une grosse erreur !

L'été de mes 16 ans, je suis parti seul un mois en Roumanie. Sur place, j'ai appris les rudiments de roumain en échangeant de manière informelle. Le fait que ce soit une langue latine m'a permis de faire de rapides progrès.

En deuxième année d'université, un ami iranien m'a initié au persan. Je me suis familiarisé avec l'alphabet arabe et j'ai pu constater, notamment au niveau de la grammaire, des similitudes propres aux langues indo-européennes.

L’année suivante, j’ai étudié le grec moderne dans un cours organisé par le pope de Grenoble. Nous étions installés dans le sous-sol de l’église orthodoxe grecque bâtie par les premiers émigrants il y a environ 60 ans.

Pendant l’année de maîtrise (bac+4) lors de mes études supérieures en informatique, je suis parti étudier un an à l’Université Polytechnique de Catalogne, à Barcelone en Espagne avec le programme Erasmus. La Catalogne est une région bilingue entre l’espagnol (castillan) et le catalan. Avant mon départ, on m’avait expliqué que, pour chaque cours, je pourrais en choisir un en castillan. À mon arrivée, j’ai constaté que c’est en fait l’enseignant qui décide quelle langue il souhaite utiliser dans son cours. J’ai dû alors apprendre le catalan en urgence afin de suivre mes cours. Pour le logement, nous étions 6 colocataires dont 4 Italiennes. Je me suis donc remis à l’italien en m’appuyant sur des méthodes telles que l’*Assimil*.

J’avais alors l’occasion de pratiquer à plein temps mes deux passions mais toujours séparément.

Et c’est à la fin de mes études supérieures en informatique, en DEA, que j’ai pu les conjuguer en travaillant sur des outils informatiques pour lexicographes. Cette année-là, je me suis inscrit à un cours d’arabe littéraire dispensé par l’association *Amal*.

Pendant mes études de doctorat, j’ai convaincu ma collègue de bureau de nous enseigner le hongrois. J’ai pu constater la difficulté d’étudier une langue agglutinante non indo-européenne. La production d’une phrase même simple demande un effort particulier selon le déterminisme de l’objet.

Lors de ma dernière année de doctorat, dès que mon projet de postdoc au Japon a pris forme, je me suis inscrit à un cours de japonais à l’université. J’ai alors mis en œuvre les premiers outils rudimentaires de construction de dictionnaires à usage personnel.

De retour en France après mon post-doctorat en 2004, j’espérais continuer la découverte d’autres langues, mais les quelques tentatives effectuées sur le mandarin et l’allemand m’ont fait comprendre qu’il est nécessaire d’avoir un objectif tangible et du temps de cerveau disponible !

C’est finalement à l’occasion d’un séjour de recherche d’un mois au Brésil à l’automne 2011 que les conditions étaient réunies. Avant le départ, j’ai travaillé avec l’*Assimil* du portugais brésilien et une fois sur place, j’ai pratiqué tant que j’ai pu.

Je suis en attente de nouvelles situations et opportunités qui me permettront d’améliorer mes connaissances sur des langues déjà étudiées ou d’étudier de nouvelles langues.

Remerciements

Je remercie tout d’abord Christian et Valérie, mes proches collègues du GETALP qui m’ont encouragé chacun à leur manière ces dernières années à rédiger ce mémoire. Je remercie aussi Stéphane qui était à mes côtés pour le début et qui a accepté de relire mon manuscrit, ainsi que Thibaut qui a compris que j’avais besoin de champ libre pour arriver à mes fins. Je pense à tous mes collègues du département MMI qui ont permis sans le savoir que je puisse rédiger au calme. Et enfin, je remercie ma famille et particulièrement Michiko, Manao et Mitsuki pour leur patience.

Table des matières

Liste des figures

Liste des tables

Introduction

Situation

Pour mieux comprendre un texte et ainsi augmenter sa compétence dans une langue donnée, le dictionnaire est un élément central. L'outil informatique ouvre le champ des possibles. Il révolutionne les usages et les pratiques lexicographiques. La première révolution fut l'apport des corpus informatisés pour concevoir de nouveaux dictionnaires.

Par la suite, le succès du projet collaboratif Wikipédia aurait pu nous amener à penser qu'il en serait de même pour les dictionnaires. Or, le succès n'est visiblement pas au rendez-vous. Il y a malgré tout une dizaine de projets Wiktionary qui dépassent les 200 000 entrées, mais ce sont des dictionnaires monolingues. Il existe également quelques projets de dictionnaires bilingues collaboratifs qui sont la plupart du temps l'œuvre de passionnés avec peu de connaissances en linguistique et en lexicographie. En effet, autant il est possible de contribuer à quelques articles d'une encyclopédie dans les domaines où l'on est expert, autant il n'est pas possible d'être expert sur quelques mots seulement de la langue.

Il faut souligner également que les coûts de construction d'un dictionnaire bilingue à large couverture sont prohibitifs et le prix de vente ne peut même pas refléter ces coûts. La plupart du temps, les maisons d'édition se contentent de nouvelles éditions de leurs dictionnaires favoris et évitent de se lancer dans un nouveau projet. Il faut dire que la concurrence est rude. Les logiciels de traduction automatique statistique et maintenant neuronale remplacent facilement les dictionnaires pour la plupart des utilisateurs qui ne cherchent qu'une traduction.

Un autre écueil est celui de la disponibilité des données. Il existe des dictionnaires bilingues provenant de grandes maisons d'édition en consultation gratuite sur le Web, mais elles ne proposent pas leurs données au téléchargement. Et même les projets contributifs permettent rarement un accès libre à leurs données[1].

Il existe pourtant un nombre très important de dictionnaires bilingues, réalisés pour la plupart au début du XX^{ème} siècle, et qui n'ont pas été informatisés. Il devrait être possible de récupérer ces données pour la plupart libres de droit et de les mettre à disposition du public. Cela permettrait de valoriser une deuxième fois le fruit d'un important travail lexicographique que nous n'avons plus les moyens de financer.

Intérêt

Mon objectif est de concevoir des environnements de gestion de bases lexicales accessibles au plus grand nombre et permettant de valoriser des ressources existantes, de créer de nouvelles ressources, de les enrichir en y ajoutant de nouvelles données ou en corrigeant les données existantes, de les

1 Voir par exemple Google translate qui permet de proposer une meilleure traduction et qui anime des communautés de contributeurs. Pourtant, il n'est pas possible d'accéder aux données qu'ils ont améliorées !

consulter sur le Web et de les utiliser dans d'autres applications, principalement pour la compréhension de textes et l'apprentissage des langues.

Le défi sociétal auquel je veux répondre est celui de la valorisation et du partage de ressources lexicales existantes, de la création d'outils collaboratifs permettant la construction de nouvelles ressources et leur usage à distance.

Le défi scientifique principal posé par cet objectif est celui de trouver comment manipuler des ressources lexicales avec des macrostructures (organisation et liens entre les volumes) et microstructures (structures des articles) différentes en utilisant le même environnement. Il s'agit de consulter ces ressources, de les éditer, de les convertir dans d'autres structures, etc. En effet, pour les dictionnaires, contrairement à d'autres ressources comme les bases terminologiques ou les corpus, il n'existe pas de standard qui puisse représenter toutes les structures existantes car celles-ci dépendent de théories linguistiques différentes et de choix éditoriaux variés en fonction des besoins.

Un autre défi intéressant qui découle du premier est celui de l'automatisation de la gestion des ressources tout au long de leur cycle de vie, de façon à faciliter le travail des administrateurs de ces environnements.

En ce qui concerne le cycle de vie des ressources, un troisième défi se pose : celui de la récupération de données existantes, dans différents formats, y compris à partir de versions imprimées, afin de pouvoir les réutiliser pour construire de nouvelles ressources.

Au delà des défis qu'elle pose, l'intérêt de cette recherche se situe également dans son positionnement fort pour une reproductibilité intégrale :

- les projets et programmes mis en œuvre sont accessibles via un site web ;
- les sources des programmes sont disponibles via le système de gestion de code source *Git* sur mes comptes *GitHub*[2] ou *GitLab*[3] ;
- les données produites sont disponibles en téléchargement sur chaque site de projet selon une licence *Creative Commons*[4] ;
- les plateformes sont utilisables directement grâce à des images virtuelles sur notre compte *Docker*[5].

Présentation

Dans la première partie de ce mémoire, je définis le vocabulaire du domaine puis je présente mon objet d'étude : les ressources lexicales. Je termine par les normes de représentation de ces ressources.

2 https://github.com/mangeot
3 https://gricad-gitlab.univ-grenoble-alpes.fr/mmang
4 https://fr.wikipedia.org/wiki/Licence_Creative_Commons
5 https://hub.docker.com/r/mangeot

La deuxième partie sera consacrée à la présentation de iPoLex et Jibiki, des environnements de manipulation de ces ressources, qui constitue le point d'appui de toutes mes recherches.

Dans la troisième partie, j'évoque les principaux projets de construction de dictionnaires auxquels j'ai participé : Papillon, GDEF, MotÀMot, DiLAF et Jibiki.fr.

Dans la quatrième partie, je détaillerai les techniques de récupération de données lexicales existantes depuis la lecture optique d'un exemplaire imprimé jusqu'à la conversion de structures XML.

La cinquième partie traitera d'un champ de recherche s'appuyant fortement sur les environnements de gestion de ressources lexicales : l'outillage de l'accès aux textes à travers la lecture active et les environnements d'apprentissage des langues.

La dernière partie sera l'occasion d'aborder les perspectives de recherche les plus saillantes à l'heure actuelle issues de mon travail passé : le passage à l'échelle pour les bases lexicales à structure pivot, l'automatisation de la récupération de ressources existantes et la lecture active enrichie.

1 Description de mon objet d'étude : la lexicographie computationnelle et les ressources lexicales

1.1 Vocabulaire du domaine

Les définitions que je donne dans cette partie sont personnelles et se situent entre des définitions formelles linguistiques et des définitions pratiques informatiques.

1.1.1 Qu'est-ce qu'un dictionnaire ?

Un **dictionnaire** est tout d'abord décrit par sa macrostructure. La **macrostructure** d'un dictionnaire est l'organisation des volumes composant le dictionnaire et des liens entre ces volumes.

La macrostructure d'un dictionnaire monolingue est composée d'un seul volume, tout comme celle d'un dictionnaire bilingue monodirectionnel. Celle d'un dictionnaire bilingue bidirectionnel est composée de deux volumes : un volume langue A → langue B et un volume langue B → langue A. La macrostructure d'un dictionnaire multilingue mono-volume est appelée multi-cible ou furcoïde : langue A → langue B et langue C. Il existe également des macrostructures plus élaborées fondées sur une structure pivot ou en étoile : un volume pivot ou interlingue au centre et des volumes monolingues autour. Le volume pivot contient les liens entre les articles des volumes monolingues.

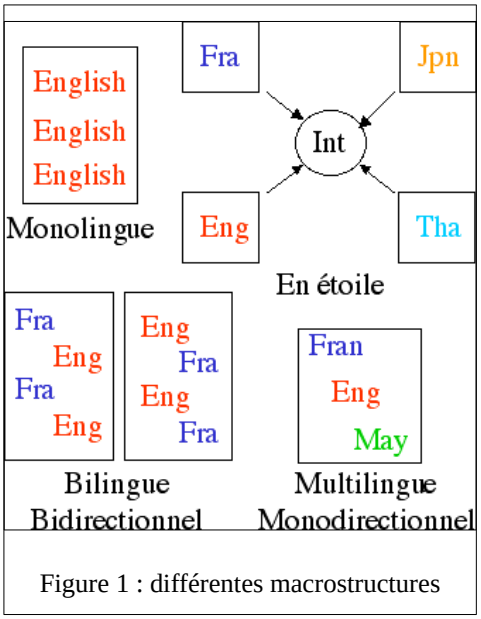


Figure 1 : différentes macrostructures

Un **volume** est un ensemble d'articles. Il est identifié par une langue source et éventuellement des langues cibles. La langue source est la langue des mots-vedette des articles.

Un **article** est identifié a minima par un mot-vedette et décrit par sa microstructure. Le terme d'« entrée » de dictionnaire est fréquemment employé dans le domaine de la dictionnairique. Nous éviterons pourtant ce terme car il est ambigu. Il peut correspondre à un article, à un mot-vedette ou

à un enregistrement de base de données. Le terme de notice est également parfois utilisé pour désigner un article.

Un **mot-vedette** est dans la plupart des cas un lemme. Un **lemme** est la forme canonique d'un lexème ou d'une locution. En français, la forme canonique des verbes est l'infinitif, celle des noms est le singulier et celle des adjectifs est le masculin singulier.

Du point de vue informatique, les lemmes sont le résultat d'un processus appliqué à un texte avec un outil appelé lemmatiseur. Du point de vue linguistique, la différence entre lemme et lexème est sujette à caution[6].

Un **lexème** est un regroupement de mots-formes qui ne se distinguent que par la flexion. Du point de vue informatique, un **mot-forme** est une chaîne de caractères trouvée dans un texte qui peut éventuellement contenir des séparateurs (ex : pommes de terre). Du point de vue linguistique, un mot-forme est un signe linguistique qui possède une certaine autonomie de fonctionnement et qui possède une certaine cohésion interne (Polguère, 2002, p 31.). Nous n'utiliserons pas le terme « mot » qui est trop ambigu (Polguère, 2002, p 33).

Une **locution** est un regroupement de constructions linguistiques qui ne se distinguent que par la flexion. Une locution possède un certain degré de figement. Elle ne respecte pas le principe de compositionnalité sémantique qui veut que le sens d'un énoncé est la résultante de la composition du sens des éléments qui le constituent (Polguère, 2002, p 135).

Plusieurs articles d'un même volume peuvent être identifiés par le même mot-vedette. Pour les distinguer, l'identification s'effectue en associant la catégorie grammaticale au mot-vedette. Par exemple, en français, « manger, nom masculin » et «manger, verbe ». Du point de vue linguistique, le terme « catégorie grammaticale » est critiquable car il implique un ensemble de valeurs mutuellement exclusives, ce qui est rarement le cas. Certains lui préfèrent les termes de « classe grammaticale », « catégorie syntaxique » ou « partie du discours » (Polguère, 2002, p 73). Nous utiliserons cependant ce terme car il est le plus répandu dans le domaine de la dictionnairique.

Une **lexie**, aussi appelée « unité lexicale » est soit un lexème, soit une locution. Chaque lexie est associée à un sens de mot donné, que l'on retrouve dans le signifié de chacun des signes auxquels elle correspond (Polguère, 2002, p 41).

Un **vocable** est un regroupement de lexies qui sont associées aux mêmes signifiants et qui ont un lien sémantique évident (Polguère, 2002, p 42). Un vocable est **polysème** s'il regroupe plusieurs lexies.

Si deux lexies distinctes ont le même lexème alors qu'elles n'entretiennent aucune relation de sens, alors chacune sera regroupée dans un vocable différent. Ces deux vocables sont **homonymes**.

Certains dictionnaires distinguent les vocables homonymes. Dans un tel cas, chaque vocable homonyme est décrit par un article distinct. En plus du mot-vedette et de la catégorie grammaticale,

6 Voir https://fr.wikipedia.org/wiki/Lex%C3%A8me_(linguistique)#Terminologie

les articles sont identifiés par un numéro d'homonyme. Par exemple, le Trésor de la Langue Française[7] distingue deux vocables « voler 1, verbe» (se soutenir et se déplacer dans l'air) et « voler 2, verbe transitif » (s'emparer frauduleusement de ce qui appartient à autrui).

Du point de vue informatique, il n'est pas possible de démontrer « un lien sémantique évident » entre deux lexies. D'ailleurs, il est fréquent de constater que deux dictionnaires de la même langue ne s'accordent pas sur ce lien sémantique évident et distinguent un nombre différent de vocables pour le même lexème. Cela dépend en effet du point de vue du lexicographe qui a rédigé ces articles. C'est pourquoi, lors de la création d'une nouvelle ressource lexicale, je préfère regrouper les vocables homonymes dans un seul article.

Dans certains cas, le mot-vedette n'est pas un lemme. Par exemple, pour le français, un lexique de formes fléchies utilisera des mots-forme pour mots-vedette. Chaque article décrira le mot-forme, sa flexion et le lemme qui correspond. Voir par exemple le dictionnaire DELAF (Courtois, 1990). Pour l'arabe littéral, la tradition lexicographique majoritaire jusqu'à une époque récente utilise les racines des mots (les consonnes) comme mot-vedette. Chaque article décrit alors une famille de mots. Pour خبز [khbz], on trouvera خباز [khoubz] (pain), [khabAz] (boulanger), مخبز [makhbaza] (boulangerie), etc. Pour les dictionnaires japonais, il est fréquent que le mot-vedette soit constitué uniquement d'une forme graphique même si celle-ci regroupe plusieurs lexèmes qui se prononcent de manière différente ou qui ont différentes catégories grammaticales. Par exemple, « 山上 » se prononce « yamakami » ou « sanjô ».

La **microstructure** des articles est une structure commune à chacun des articles d'un volume. Elle contient a minima un mot-vedette. De manière générale, un article est composé d'un **bloc forme** qui regroupe les information relatives aux formes que peut prendre le mot-vedette dans un texte : prononciation (avec l'alphabet phonétique international ou dans une notation spécifique), transcriptions dans d'autres alphabets (ex : *pinyin*[Error: Reference source not found] pour le mandarin, *romaji*[32] pour le japonais, etc.), étymologie, etc. puis d'un corps. Le **corps** d'un article est soit un bloc grammatical dans le cas où l'article n'est identifié que par le mot-vedette, soit un bloc sémantique dans le cas où l'article est identifié par le mot-vedette et sa catégorie grammaticale.

Le **bloc grammatical** est composé d'une catégorie grammaticale et d'un bloc sémantique.

Le **bloc sémantique** regroupe les sens de mot (ou lexies), les collocations (ou expressions semi figées) et les locutions (ou expressions idiomatiques) lorsqu'elles ne font pas l'objet d'un article distinct.

Un **sens de mot** est décrit par une définition ou par une traduction ou une définition en langue cible dans le cas des dictionnaires bilingues. Lorsque ce sens de mot est un terme d'un **domaine** particulier, celui-ci peut être mentionné (anatomie, biologie, chimie, droit, linguistique, mathématiques, etc.). Le sens de mot est parfois illustré par des **exemples** d'usage. Ceux-ci sont

7 Voir http://atilf.atilf.fr/

forgés à la main ou tirés d'un corpus. Dans ce cas, la référence dans le corpus peut être indiquée (auteur, titre de l'ouvrage, page).

La **nomenclature** d'un dictionnaire est la liste des mots-vedettes qui composent ces volumes et l'ordre lexicographique de ces mots-vedettes (au moins dans une version imprimée). La liste des mots-vedettes à ajouter dans un dictionnaire fait toujours l'objet d'un choix. Le vocable décrit doit apparaître au moins une fois dans un corpus (on dit qu'il est attesté) ou dans un autre dictionnaire. Lorsque les dictionnaires n'existaient qu'en version imprimée, le coût d'édition et plus précisément le nombre final de pages était pris en compte. À l'heure actuelle, même s'il n'y a plus de contrainte physique de place, il reste toujours un choix à faire, notamment en ce qui concerne les noms propres car rares sont ceux qui n'en contiennent pas.

L'**ordre lexicographique** est l'ordre dans lequel sont triés les mots-vedettes d'un volume de dictionnaire. Pour la plupart des langues, il s'agit de l'ordre alphabétique des lettres de cette langue avec quelques particularités notamment pour les signes diacritiques (ex : lettres accentuées en français) et les digrammes ou trigrammes.

Le tri lexicographique du français est finalement peu connu. Pour les mots-vedette comportant des signes diacritiques, un premier tri est effectué en les ignorant. Puis, pour les homographes, la discrimination s'effectue à rebours, en partant de la fin du mot et en reculant. Par exemple, pour le cas « cote », cela donne cote, côte, coté, côté, coter[8].

Dans certaines langues, les digrammes correspondent à un seul phonème mais sont considérés comme deux lettres dans l'ordre alphabétique. Par exemple, en français, le digramme « ch » se prononce /ʃ/ mais le lemme « chien » se classe entre « cela » et « cil ». Autre exemple, en haoussa, l'ordre alphabétique comporte 9 digrammes : a b ɓ c d ɗ e f fy g gw gy h i j k kw ky ƙ ƙw ƙy l m n o p r s sh t ts u w y y̆ z (Enguehard et al., 2011).

1.1.2 Types de ressources

Un **dictionnaire** est un inventaire des lexies ou sens de mot d'une ou plusieurs langues. Les lexies sont décrites dans une démarche sémasiologique : partir des lexèmes pour aller vers la détermination des sens de mot. Dans un dictionnaire monolingue, les lexies sont décrites à l'intérieur de la langue et en principe caractérisées par une définition. Dans un dictionnaire bilingue ou plurilingue, elles sont décrites soit par leur équivalent dans une autre langue (une ou plusieurs traductions), soit par une définition dans une autre langue. Généralement, un dictionnaire a l'objectif de décrire l'ensemble de toutes les lexies d'une langue, mais cet objectif ne peut être que théorique car les limites de cet ensemble sont floues et évoluent dans le temps. En principe, les dictionnaires évitent de décrire des noms propres. S'ils en décrivent, on parle de **dictionnaire encyclopédique**.

8 Voir http://www-clips.imag.fr/geta/gilles.serasset/tri-du-francais.html

Un **lexique** ou un **glossaire** est un dictionnaire limité soit dans les lexies qu'il décrit (un domaine, une catégorie grammaticale particulière, un niveau de langue, etc.), soit dans sa microstructure qui est simplifiée. Un lexique bilingue peut ne comporter pour chaque mot-vedette que sa traduction.

Une **base lexicale** est une extension des dictionnaires permise par l'outil informatique. Par exemple, une macrostructure pivot ne peut être représentée simplement par une version imprimée d'un dictionnaire. Il est par contre possible à partir des éléments contenus dans une base lexicale d'obtenir des vues partielles correspondant à des dictionnaires.

Une **encyclopédie** donne pour chacun des mots-vedettes décrits des informations non linguistiques.

Une **base terminologique** est un répertoire des termes d'un domaine constitué selon une démarche onomasiologique : partir du concept pour aller vers les différentes réalisations du terme dans les différentes langues de la base. Même si la démarche est inverse de celle des dictionnaires, leurs macrostructures et microstructures sont semblables. C'est pourquoi nous incluons les bases terminologiques dans les types de ressources que nous manipulons.

D'un point de vue linguistique, un **réseau lexical** est un « ensemble de mots qui désignent des réalités ou des idées appartenant au même thème (le champ lexical) auxquels s'ajoutent tous les mots qui à cause du contexte, et de certaines de leurs connotations, évoquent aussi ce thème »[9]. D'un point de vue informatique, le terme de **réseau lexical** est plus général. Il peut représenter tous les mots d'une langue et même de plusieurs langues. Chaque nœud du réseau est une information : mot-vedette, catégorie grammaticale, etc. Les notions de macrostructure, de volume (et avec elle celle de voisinage alphabétique) et de microstructure disparaissent (Zhang et al., 2014). En principe, une ressource de ce type ne peut être directement consultée par des humains. Il faut des programmes sophistiqués de visualisation et de navigation. Par contre, il est très utilisé par des machines. Le format de représentation des réseaux lexicaux le plus répandu est le LLOD pour Linguistic Linked Open Data.

Un **thésaurus** est un ensemble structuré de termes d'un domaine. Chaque terme est un descripteur aussi peu ambigu que possible et préféré à ses voisins synonymes. Les termes synonymes sont regroupés. Par exemple, dans WordNet (Miller et al., 1990), ces groupes sont appelés « synsets ».

Une **ontologie** est un ensemble de concepts structurés sous forme de graphe dont les relations sont sémantiques, de composition ou d'héritage.

Le terme de **ressource lexicale** est un terme générique qui regroupe les lexiques, les dictionnaires, les bases lexicales et terminologiques, les encyclopédies, les thésaurus et les réseaux lexicaux.

Un **corpus** est un ensemble fini d'énoncés écrits ou enregistrés, constitué en vue de leur analyse linguistique. D'un point de vue informatique, un corpus écrit est généralement enrichi avec le

9 Voir http://www.lettres.org/files/reseau_lexical.html À noter que le serveur était injoignable lors de notre consultation le 18 avril 2019 et que nous avons extrait la définition du résumé de la page fournie par le moteur de recherche.

résultat d'analyses morphologiques, syntaxiques ou sémantiques. Il est également consultable via une interface permettant des recherches précises.

Les dictionnaires et les corpus sont les deux principales ressources représentant une langue. Ils sont interdépendants : le dictionnaire utilise le corpus pour attester de la présence d'un lexème et pour trouver des exemples d'usage ; le corpus utilise le dictionnaire pour analyser morphologiquement le texte et éventuellement pour le traduire.

1.1.3 Description du contexte

Un **entrepôt de données** est un espace où les données numériques sont déposées puis stockées sans modification. Les données sont décrites pour pouvoir les retrouver plus facilement.

Un **environnement de manipulation de données** est un ensemble d'outils permettant de manipuler ces données. Les données sont d'abord importées dans l'environnement puis éventuellement transformées et enfin exportées hors de l'environnement. Lorsque les données peuvent être modifiées par des contributeurs, on parle d'environnement contributif ou **plateforme contributive**. Ces environnements ont la plupart du temps un système de gestion d'utilisateurs et de droits associés : le ou les **administrateurs** peuvent gérer les utilisateurs et manipuler des données (import/ export) ; les **contributeurs** peuvent modifier des données existantes ou créer de nouvelles données ; les autres **utilisateurs** peuvent consulter les données.

La **lexicographie** est l'activité ou le domaine d'étude visant la construction de dictionnaires (Polguère, 2002, p 176). Un **lexicographe** est une personne qui rédige des articles de dictionnaire. Un **lexicographe en chef** est une personne chargée de définir et de faire respecter la nomenclature d'un dictionnaire lorsque plusieurs lexicographes travaillent sur le même ouvrage.

La dictionnairique a été définie par Bernard Quemada10 comme la discipline dont le principal objectif est la publication d'un dictionnaire. Nous étendrons cette définition pour y ajouter l'étude des dictionnaires et des éléments qui les composent (macro et microstructures, nomenclature, etc.).

1.2 Formats de ressources lexicales

Historiquement, les dictionnaires sont d'abord imprimés. Avec les débuts de l'informatique, on utilise des logiciels de traitement de texte pour rédiger puis imprimer les dictionnaires. Puis, on s'attache à représenter explicitement les informations à l'aide de formats adéquats tels que le balisage.

1.2.1 Les fichiers issus de logiciels de traitement de texte

Dès l'apparition des logiciels de traitement de texte, ceux-ci sont utilisés pour rédiger des dictionnaires. Malgré l'apparition de formats plus adaptés à la représentation de l'information structurée contenue dans les dictionnaires, il est encore relativement fréquent de nos jours de trouver de tels fichiers. Par rapport aux versions imprimées, ils ont l'avantage de permettre une récupération de toutes les informations écrites. Par contre, ils sont destinés à l'impression.

L’information est donc marquée de manière implicite à travers les informations de style : le mot-vedette sera en gras, la catégorie grammaticale encadrée par des crochets, les exemples en italique, etc. Pour pouvoir manipuler ces ressources avec nos environnements, il faut d’abord marquer explicitement toutes les informations et les convertir au format XML.

1.2.2 Les fichiers en colonne : csv

Lorsque la microstructure d’un ressource lexicale est simple (lexique, base terminologique) et qu’elle ne comporte pas ou peu de sous-blocs, il est possible de la représenter sous forme de fichier en colonnes ou csv pour « comma separated value » où chaque ligne du fichier représente un article. Ces fichiers sont facilement manipulables par un éditeur de texte basique ou un tableur. Ils sont également facilement convertibles au format XML.

1.2.3 Les langages de balisage : l’avènement de XML

Au tout début de l’informatique, chaque constructeur d’imprimante met au point son propre langage de commandes destinées à la mise en forme de textes avant impression. C’est le degré zéro de l’interopérabilité. Face à cette situation, un comité se met en place dirigé par Charles Goldfarb et met au point le standard SGML pour Standard Generalized Markup Language qui sera adopté comme standard ISO 8879 en 1986. Ce langage de balisage est rapidement adopté par les grandes maisons d’édition de dictionnaires (Oxford University Press). Il permet de représenter les documents de manière structurée. Il a cependant un défaut majeur : dans certains cas, il est possible d’omettre des balises fermantes. L’analyse d’un document SGML requiert donc une grammaire dépendante du contexte et peut déboucher sur plusieurs arbres d’analyse pour un même document.

C’est pourquoi est sorti en 1998 XML pour eXtended Markup Language comme recommandation du W3C, le World Wide Web consortium. XML est un sous-ensemble de SGML qui rend obligatoires les balises fermantes. De ce fait, l’analyse est plus rapide et ne donne en résultat qu’un seul arbre.

Dès la sortie de la recommandation, les acteurs de la dictionnairique s’en sont emparés pour encoder les ressources lexicales. Pourquoi ce rapide succès ? XML est particulièrement bien adapté à la représentation de dictionnaires. En effet, sa structure arborescente convient très bien à la plupart des microstructures. De plus, l’encodage par défaut est UTF-8 pour Universal Character Set Transformation Format1 - 8 bits, qui représente la table de codage Unicode. Cet encodage permet donc d’encoder la grande majorité des langues écrites dans le monde tout en étant compatible avec les machines codant les caractères sur un seul octet, soit la majorité des machines à l’époque.

Avec l’arrivée de XML, un autre changement important se produit. Jusqu’ici, la version de référence d’un dictionnaire était la version imprimée, et ce, malgré l’utilisation de SGML qui permet de séparer le fond (les informations) de la forme (le style) en codant ces informations dans une feuille de style à part. Petit à petit, les versions de référence des ressources lexicales deviennent des fichiers électroniques au format XML. Les versions imprimées ne sont que des vues de cette version de référence. Et c’est dont naturellement que l’on en vient à coder les informations de

manière explicite plutôt qu'avec des instructions de style. Associé à XML, le langage de transformation d'arbres XSLT[10] et les feuilles de style CSS[11] permettent de produire plus facilement des versions imprimées (PDF[12]) ou à lire sur écran (HTML[13]).

Cette évolution coïncide avec l'expansion du Web (inventé en 1991) et de l'ordinateur personnel dont les premières version grand public sont sorties en 1977. L'usage des ordinateurs portables s'est répandu également à cette époque (le PowerBook d'Apple est sorti également en 1991). La technique permettait alors d'envisager une version de référence entièrement électronique pour un dictionnaire.

1.3 Efforts de standardisation

1.3.1 La Text Encoding Initiative

La TEI, en français « initiative pour l'encodage du texte » est une communauté académique internationale dans le champ des humanités numériques visant à définir des recommandations pour l'encodage de documents textuels[14].

La première version de la TEI, TEI P1, a été publiée en 1987 et utilise le langage de balisage SGML.

La TEI P3, publiée en 1994 est considérée comme la première version complète. Le chapitre 12 de cette version est dédiée aux dictionnaires. Cependant, le groupe de travail en charge de ce chapitre a conclu qu'il n'était pas possible de proposer une Définition de Type de Document (DTD) qui puisse représenter tous les dictionnaires. Le problème majeur identifié est la contradiction entre la généralité de la description, qui doit être applicable à un grand nombre de dictionnaires, et le pouvoir descriptif, c'est à dire la possibilité de décrire avec précision la structure de n'importe quel dictionnaire.

La TEI P4, parue en 2001, introduit XML à côté de SGML.

La TEI P5 est parue en 2007. Elle abandonne définitivement SGML au profit de XML. Le chapitre 9[15] est consacré aux dictionnaires. Pour contourner le problème cité plus haut, la recommandation utilise deux éléments différents pour coder les articles de dictionnaire : l'élément <*entry*> dont la microstructure est très rigide et très codifiée et l'élément <*entryFree*> qui, comme son nom l'indique, permet de coder librement une microstructure. Dans la pratique, il est très difficile de se contenter de <*entry*> pour coder un dictionnaire existant. De ce fait, on est obligé d'avoir recours à <*entryFree*> et ce faisant, on perd les avantages d'un standard, notamment pour l'échange de données.

10 https://fr.wikipedia.org/wiki/Extensible_Stylesheet_Language_Transformations
11 https://fr.wikipedia.org/wiki/Feuilles_de_style_en_cascade
12 https://fr.wikipedia.org/wiki/Portable_Document_Format
13 https://fr.wikipedia.org/wiki/Hypertext_Markup_Language
14 https://fr.wikipedia.org/wiki/Text_Encoding_Initiative
15 https://tei-c.org/release/doc/tei-p5-doc/fr/html/DI.html

Une fonctionnalité intéressante de la TEI est la description des textes dans un en-tête relativement riche de méta-données. La structure des en-têtes est partagée avec tous les types de texte, y compris les dictionnaires. Je pense toutefois qu'il est préférable de stocker les métadonnées séparément des données afin de ne stocker qu'un seul type d'information par fichier.

Malgré l'engouement de la TEI pour un grand nombre de types de texte, y compris les corpus, le chapitre sur les dictionnaires n'a pas été très utilisé. À part les quelques exemples cités dans la littérature, nous n'avons jamais rencontré de dictionnaire encodé avec la TEI sur la centaine dont nous avons gardé la trace.

1.3.2 Lexical Markup Framework

Le cadre de balisage lexical LMF est un standard de l'ISO conçu pour les lexiques utilisés en traitement automatique des langues. Le projet a démarré en 2004 sous l'égide du comité technique TC 37 et la version finale du standard est parue en 2008. L'objectif est de fournir un modèle commun pour la création et l'utilisation des ressources langagières, de gérer l'échange des données entre ces ressources et de permettre la fusion d'un grand nombre de ressources électroniques afin de constituer un vaste réseau de descriptions linguistiques[16].

Pour éviter l'écueil de la TEI avec une microstructure soit trop rigide, soit trop lâche, la partie normative de LMF est fondée sur un méta-modèle séparant les parties lexicale, grammaticale et sémantique (voir figure 2). Elle n'impose aucune syntaxe particulière. De plus, le méta-modèle est modulaire avec un noyau qui décrit la structure de base d'une « entrée lexicale » (qui correspond à notre définition d'**article**) et des extensions pour décrire des ressources lexicales spécifiques.

La classe principale est une ressource lexicale (*Lexical Resource)*. Elle contient une classe décrivant la méta-information sur le lexique (*Global Information)* et un ou plusieurs lexiques (*Lexicon)*. Le lexique contient une ou plusieurs entrées lexicales (*Lexical Entry)*. Une entrée lexicale contient une ou plusieurs formes de l'entrée (*Form)* ainsi qu'un ou plusieurs sens (*Sense)*. Les formes contiennent des variantes orthographiques *Form representation*. Les sens peuvent à leur tour contenir des sens de manière récursive. Les sens peuvent contenir des définitions (*Definition)* qui contiennent des contenus textuels (*Text Representation)* et des descriptions narratives (*Statement)*. La classe (*representation)* permet de faire un lien entre les variantes orthographiques (*Form Representation)* et leurs occurrences dans un texte (*Text Representation)*. Ce méta-modèle est devenu une norme ISO sous le numéro 24613:2008 en novembre 2008 (Francopoulo et al., 2009).

16 https://fr.wikipedia.org/wiki/Lexical_Markup_Framework#Objectifs_de_LMF

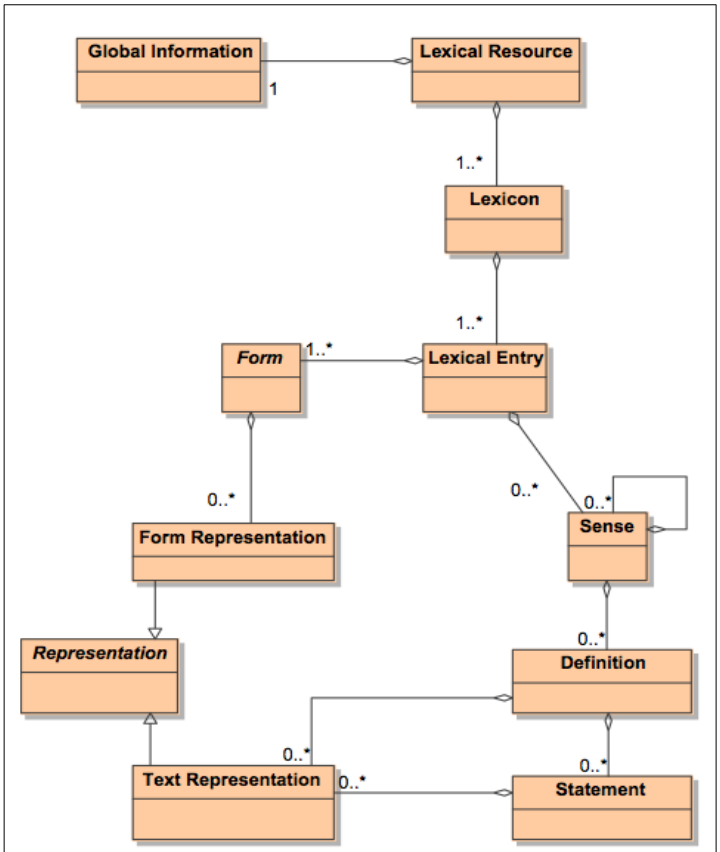


Figure 2 : méta-modèle noyau de LMF

La partie informative propose une syntaxe pour représenter des ressources lexicales au format LMF. Elle est représentée sous forme d'une structure de traits. Les objets structurants de l'article sont des éléments XML n'ayant que des éléments fils. Les objets informatifs de l'article sont des éléments vides feat (pour feature ou trait) ou mono-balises et portent l'information dans un attribut (voir figure 3).

```
<LexicalResource dtdVersion="15">
  <GlobalInformation>
    <feat att="languageCoding" val="UTF-8"/>
  </GlobalInformation>
  <Lexicon>
    <feat att="language" val="kau"/>
    <LexicalEntry id="bannadu2">
    <Lemma>
        <feat att="writtenForm" val="bannadu"/>
        <feat att="phoneticForm" val="bànnàḍú"/>
    </Lemma>
    <feat att="partOfSpeech" val="kkye3."/>
    <Sense id="1">
        <Equivalent>
            <feat att="language" val="fra"/>
            <feat att="writtenForm" val="éduquer(mal)"/>
        </Equivalent>
        <Definition>
            <feat att="writtenForm" val="Diwiro yal alamdu."/>
        </Definition>
        <Context>
            <TextRepresentation>
                <feat att="language" val="kau"/>
                <feat att="writtenForm" val="Gənanjun bannaje, ku tadanju rakce kəlanju
rojiwawo."/></TextRepresentation>
            <TextRepresentation>
                <feat att="language" val="fra"/>
                <feat att="writtenForm" val="Durant son jeune âge il l'a mal éduqué, aujourd'hui son fils n'arrive pas à se prendre en
charge."/>
            </TextRepresentation>
        </Context>
        <SenseRelation targets="làṇḍú">
            <feat att="type" val="synonym"/>
        </SenseRelation>
    </Sense>
    </LexicalEntry>
  </Lexicon>
</LexicalResource>
```

Figure 3 : article « bannadu 2 » du dictionnaire DiLAF kanouri-français au format LMF

Malgré son méta-modèle souple, LMF souffre à son tour de plusieurs défauts :

- Les parties normatives et informatives sont distinctes mais cette distinction n'a finalement peu été comprise. Lorsque l'on produit une ressource au format LMF, c'est presque toujours avec la syntaxe de la partie informative.
- La partie informative a une syntaxe difficile à lire pour un humain et ne respecte pas les habituelles conventions de codage d'un texte en XML. Les éléments parents regroupent en général des informations du même type. Dans la figure 3, l'élément <Lexicon> comporte un élément fils <feat> qui indique la langue source de la ressource ainsi que des éléments <LexicalEntry> qui sont des articles. D'autre part, généralement, le texte libre est encadré par des balises mais dans la figure 3, on peut voir que le texte est à chaque fois placé en attribut d'un élément <*feat*>.
- Il était prévu au départ de standardiser les listes fermées de valeurs comme la catégorie grammaticale. Un projet ISOCat avait démarré au sein du comité technique TC 37 mais le mandat initial n'a jamais été rempli et finalement été jugé inutile17.

Il faudrait selon moi clarifier la distinction entre les parties normative et informative, en permettant d'une part à chaque utilisateur d'élaborer une microstructure XML qui convienne à ses usages mais qui soit certifiée compatible avec le standard LMF, et d'autre part en fournissant un outil permettant de convertir automatiquement ce XML vers la syntaxe de la partie informative de LMF qui fonctionne alors comme un format d'échange. Il serait également préférable de revoir la partie informative pour qu'elle soit plus lisible directement par un humain.Il faudrait aussi qu'il y ait un volume spécial de « méta-descriptions » donnant le sens de tous les traits et valeurs.

De même que pour la TEI, je n'ai manipulé qu'une seule ressource encodée avec LMF. Il s'agit de Morphalou (Romary et al., 2004) qui a été conçue justement pour illustrer LMF.

1.3.3 Notre solution : les pointeurs Common Dictionary Markup

Si l'on veut manipuler une ressource existante sans perdre d'information, il n'est pas possible de la transformer dans une nouvelle structure qui imposera nécessairement ses contraintes. D'un autre côté, il n'est pas envisageable de manipuler des ressources avec des outils génériques si l'on doit systématiquement adapter ces outils à la structure de chaque ressource.

Pourtant, lorsque l'on manipule des ressources lexicales, on trouve presque toujours le même type d'information, certes encodé différemment (avec un élément ou un attribut, avec des noms de balise différents, etc.), mais facilement identifiable : l'article, le mot-vedette, la prononciation, la catégorie grammaticale, la définition, la traduction, l'exemple, etc.

Je propose de manipuler ces ressources à l'aide de pointeurs dans les microstructures XML exprimés à l'aide de la norme XPath[17]. Je définis un ensemble de balises communes appelé CDM pour Common Dictionary Markup pour les informations que l'on retrouve dans plusieurs ressources. Les pointeurs associés à ces balises identifient des objets structurants dans chaque ressource considérée : volume, article, bloc sémantique, bloc d'exemples, ainsi que des objets informatifs : identifiant de l'article, mot-vedette, numéro d'homographe, prononciation, lecture, écriture, catégorie grammaticale, domaine, définition, traduction, exemple, locution. L'ensemble de ces pointeurs est appelé CDMptr.

La table 1 montre les valeurs des pointeurs CDM pour 6 ressources différentes :

- Les langues sources des ressources sont décrites à l'aide du code 3 lettres de la norme ISO 639-3 (eng = anglais, est = estonien, fra = français, kau = kanouri, khm = khmer, jpn = japonais, etc.)[18]. Nous utilisons cette convention pour le nommage des langues tout au long de nos travaux. Si une langue ne possède pas de code ISO 639-3, un code est choisi dans la plage qaa-qtz réservée à un usage local. Par exemple, j'ai choisi « qno » pour le nouchi19.

17 https://fr.wikipedia.org/wiki/XPath

18 Lors de la création de l'ISO 639-2, l'espagnol avait un statut à part. Le code « esp » pouvait être utilisé encore pendant 5 ans puis devait être remplacé par « spa ». Nous avons gardé « esp » car il correspond à la philosophie du standard de garder un code en rapport avec la langue source.

- DiLAF kau LMF est le dictionnaire kanouri-français du projet DiLAF au format LMF (voir figure 3). Le début du pointeur `/LexicalResource/Lexicon[feat/(@att='language' and @val='kau')]` a été abrégé en … pour les pointeurs suivant le mot-vedette.

Balise CDM	Ressource	Valeur du pointeur CDM exprimée selon la norme XPath
Mot-vedette `cdm-headword`	Papillon fra	`/p:volume/p:lexie/p:headword`
	GDEF est	`/g:volume/g:article/g:vedette/g:mot`
	MotÀMot khm	`/m:volume/m:entry/m:head/m:headword`
	DiLAF kau	`/dilaf/article/kalma`
	DiLAF LMF	`/LexicalResource/Lexicon[feat/(@att='language' and @val='kau')]/LexicalEntry/Lemma/feat[@att='writtenForm']/@val`
	Jibiki.fr jpn	`/volume/article/forme/vedette/vedette-jpn`
Prononciation `cdm-pronunciation`	Papillon fra	`/p:volume/p:lexie/p:pronunciation`
	MotÀMot khm	`/m:volume/m:entry/m:head/m:pronunciation`
	DiLAF kau	`/dilaf/article/bowodu`
	DiLAF LMF	`.../LexicalEntry/Lemma/feat[@att='phoneticForm']/@val`
	Jibiki.fr jpn	`/volume/article/forme/vedette/vedette-hiragana`
Cat. gram. `cdm-pos`	Papillon fra	`/p:volume/p:lexie/p:pos`
	GDEF est	`/g:volume/g:article/g:vedette/g:bloc-gram/g:cat-gram`
	MotÀMot khm	`/m:volume/m:entry/m:head/m:pos`
	DiLAF kau	`/dilaf/article/naptu_curo_nahauyen`
	DiLAF LMF	`.../LexicalEntry/feat[@att='partOfSpeech']/@val`
	Jibiki.fr jpn	`/volume/article/sémantique/bloc-gram/étiquettes/gram`
Définition `cdm-definition`	Papillon fra	`/p:volume/p:lexie/p:semantic-formula`
	DiLAF kau	`/dilaf/article/bəri/maana`
	DiLAF LMF	`.../LexicalEntry/Sense/Definition/feat[@att='writtenForm']/@val`
Domaine `cdm-domain`	GDEF est	`/g:volume/g:article/g:vedette/g:domaine-vedette`
	MotÀMot khm	`/m:volume/m:entry/m:sense/m:domain`
	Jibiki.fr jpn	`/volume/article/sémantique/bloc-gram/étiquettes/domaine`
Traduction `cdm-translation`	Papillon fra	`/p:volume/p:lexie/p:refaxie`
	GDEF est	`//g:mot-princ`
	MotÀMot khm	`/m:volume/m:entry/m:sense/m:refaxie`
	DiLAF kau	`/dilaf/article/bəri/kalakta[@təlam='fa']`
	DiLAF LMF	`.../LexicalEntry/Sense/Equivalent[feat/(@att='language' and @val='fra')]/feat[@att='writtenForm']/@val`
	Jibiki.fr jpn	`/volume/article/sémantique/bloc-gram/sens/texte-sens`
Exemple `cdm-example`	Papillon fra	`/p:volume/p:lexie/p:examples/p:example`
	GDEF est	`/g:volume/g:article/g:bloc-gram/g:bloc-semantique/g:bloc-`

		`exemples/g:exemple/g:exemple-est`
	MotÀMot khm	`/m:volume/m:entry/m:sense/m:examples/m:example`
	DiLAF kau	`/dilaf/article/bəri/misal/version[@təlam='ka']`
	DiLAF LMF	`.../LexicalEntry/Sense/Context/TextRepresentation[feat/(@att='language' and @val='kau')]/feat[@att='writtenForm']/@val`
	Jibiki.fr jpn	`/volume/article/sémantique/exemples/exemple/kanji`

Table 1 : pointeurs CDM pour 6 ressources lexicales différentes

Cet ensemble n'est pas figé. Si, lors de nouvelles manipulations de ressources, la même information est retrouvée dans plusieurs ressources différentes, alors un nouveau pointeur CDM est créé et ajouté à l'ensemble. Pour des informations spécifiques à une seule ressource, il est toujours possible de définir des pointeurs spécifiques à cette ressource.

Article associé au chapitre 1

Mathieu Mangeot & Chantal Enguehard (2013) *Des dictionnaires éditoriaux aux représentations XML standardisées.* Chapitre 8, livre "Ressources Lexicales : contenu, construction, utilisation, évaluation", Eds. Nuria Gala & Michael Zock, Linguisticae Investigationes Supplementa, John Benjamins Publishing, Amsterdam, Pays-Bas, 24 p. https://hal.archives-ouvertes.fr/hal-00959229

2 Environnements de gestion de ressources lexicales

Dans cette partie, nous abordons le cœur de notre recherche : comment manipuler des ressources lexicales de manière générique. Pour la partie informatique, il s'agit de concevoir des outils permettant à un spécialiste lexicographe non informaticien d'importer, de stocker, de transformer, de consulter des ressources lexicales hétérogènes avec le minimum de configuration et d'adaptation entre les ressources lexicales. L'objectif ultime est d'automatiser intégralement la chaîne de traitement.

2.1 Entrepôt de données lexicales avec « devin de microstructures »

En premier lieu, nous devons prendre soin des données lexicales que nous manipulons. Une organisation précise est nécessaire afin de stocker une ressource, de la retrouver facilement, de l'échanger avec d'autres utilisateurs, etc.

Le défi scientifique de cette sous-partie est le suivant : automatiser l'analyse et la description des ressources lexicales hétérogènes, cette dernière s'effectuant via la génération de fichiers de métadonnées décrivant ces ressources.

Le défi sociétal est de concevoir un outil simple et robuste, disponible en source ouverte, facile à installer et à manipuler, et qui réponde au défi scientifique.

2.1.1 Notion de signature dictionnairique

Lorsque l'on souhaite importer une ressource existante dans nos environnements afin de la manipuler, il faut tout d'abord la décrire, ce qui nécessite entre autres de définir sa microstructure à l'aide de pointeurs CDMptr (voir précédemment le paragraphe 1.3.3). Même si la syntaxe des pointeurs XPath est relativement simple à comprendre avec le fichier XML correspondant en regard, elle est source d'erreurs de la part du rédacteur. Cela a été confirmé par mon expérience. Avec le temps, il m'a semblé nécessaire de fournir une aide à la rédaction, voire une automatisation complète du calcul de ces pointeurs CDMptr. Pour cela, une analyse fine de la microstructure et du contenu de la ressource à importer est nécessaire.

Afin de répondre à ces besoins, nous avons inventé le concept de signature dictionnairique. Le but est de caractériser toute ressource lexicale d'un point de vue structurel. Il s'agit donc d'une description synthétique présentant un ensemble d'indicateurs quantitatifs pour chaque élément de texte repérable par le même mécanisme. Pour chaque mécanisme, les valeurs calculées sont principalement le nombre de tels éléments et le nombre moyen de caractères des éléments textuels ainsi repérés.

Le fichier à importer est analysé avec un parseur XML et chaque partie d'information est enregistrée : élément, attribut, contenu textuel. Des calculs sont ensuite effectués pour obtenir leur fréquence, le respect de l'ordre alphabétique, etc. Ces informations sont ensuite affichées de manière condensée avec une syntaxe proche de celle d'un fichier XML où chaque élément

n’apparaît qu’une fois et où seule la balise ouvrante est affichée, ce qui permet un gain de place. Les éléments fils sont indentés par rapport à leur élément père. C’est la signature dictionnairique (voir figure 4).

```
<dilaf:1 src=1≠≥1(dje:1) trg=1≠≥1(fra:1)>
  <article:6914>chars:1.0;words:1;2≠2≥3(,:1,.:2)
    <sanniize:6914 lamba=10≠2080≥3338(1:383,10:1,2:377,3:122,4:61,5:24,6:14,7:6,8:6,9:5)>cl
    <ciiyaŋ:6914>chars:8.2;words:1;759≠6507≥6917
    <kanandi:6914>chars:3.9;words:1;25≠5070≥6884(alteeb.:6,bsif.:1,ceey.:5,dab.:3,damb.:2,
    <bareyaŋ:6914>chars:18.6;words:3;918≠3519≥6864
    <feeriji:6253>chars:40.4;words:9;980≠3125≥6263
    <silmaŋ:6044>
      <version:12088 lang=2≠6044≥12088(dje:500,fra:499)>chars:48.3;words:10;999≠3800≥7379
    <himacare:1928 lamba=1≠≥1(2:1)>chars:7.6;words:1;855≠1006≥1928
    <f:4017>chars:7.4;words:1;844≠3519≥4018
    <b:3632>chars:7.9;words:1;797≠3251≥3630
    <di:810 lamba=7≠79≥125(1:40,2:49,3:15,4:8,5:9,6:3,7:1)>chars:6.6;words:1;605≠484≥810
    <cidumi:311>chars:6.9;words:1;256≠224≥311
    <wayc:157>chars:7.5;words:1;137≠82≥157
    <complément:113>chars:30.4;words:5;26≠79≥113(catégorie:2,Cet article est vide:1,Erreur
    <soŋay:1>chars:5.0;words:1;1≠≥1(galci:1)
  <complémént:1>chars:7.0;words:1;1≠≥1(exemple:1)
```

Figure 4 : signature dictionnairique du dictionnaire DiLAF zarma-français

L’exemple de la figure 4 se lit de la manière suivante :

- l’élément « dilaf » est l’élément racine de la ressource. Il n’apparaît qu’une seule fois dans toute la ressource à cette place dans l’arbre XML (logique pour un élément racine…). Il contient deux attributs. L’attribut src n’apparaît qu’une fois (là aussi c’est logique puisque l’élément n’apparaît qu’une fois !) et sa valeur est dje. L’attribut trg n’apparaît aussi qu’une seule fois (idem) et sa valeur est fra.

- l’élément « article » est le premier élément fils de l’élément dilaf rencontré dans l’arbre XML car il est indenté de deux espaces vers la droite. Il apparaît 6 914 fois dans toute la ressource à cette place dans l’arbre XML. Il n’a pas d’attribut. Son contenu textuel est d’en moyenne 1 caractère. Il contient 2 valeurs différentes qui sont classées par ordre alphabétique. Il contient en tout 3 valeurs. Les valeurs différentes sont une virgule « , » qui apparaît une fois et un point « . » qui apparaît deux fois.

- l’élément « saniize » est le premier élément fils de l’élément article rencontré dans l’arbre XML. Il apparaît également 6 914 fois dans toute la ressource à cette place dans l’arbre XML. Il a un attribut lamba qui a 10 valeurs différentes et dont 2080 valeurs sont classées par ordre alphabétique sur un total de 3 338. La liste des valeurs différentes est ensuite affichée avec le nombre de valeurs pour chaque : « 1 » apparaît 383 fois, « 10 » apparaît une fois, « 2 » apparaît 337 fois, etc. Son contenu textuel est d’en moyenne 6,2 caractères et 1 mot. Il a 704 valeurs différentes et 6 865 valeurs sont classées par ordre alphabétique sur 6 914.

Les autres éléments se décrivent selon le même principe.

2.1.2 Pré-calcul des pointeurs CDM de description de microstructures

La signature dictionnairique donne des informations permettant d'émettre des hypothèses pour chaque pointeur CDM. Ces hypothèses sont appliquées à tous les éléments de la microstructure. L'hypothèse obtenant le meilleur score est retenue puis présentée à l'utilisateur qui peut ensuite la corriger.

Je détaille maintenant les heuristiques servant à calculer le pointeur CDM du mot-vedette.

- Les éléments sélectionnés doivent contenir du texte.
- Le score final est augmenté s'il y a autant de mots-vedette que d'articles ou si leur nombre est proche.
- Il est augmenté si le nom de l'élément contient les mots « headword » ou « vedette ».
- Il est augmenté s'il y a en moyenne peu de mots (de 1 à 3).
- Il est augmenté s'il y a beaucoup de valeurs différentes (proche du nombre total de valeurs).
- Il est diminué selon le rang dans sa fratrie. Le mot-vedette est souvent le premier de sa fratrie.

Ensuite, le score final est calculé en divisant le score par le niveau de l'élément dans l'arbre XML. Plus l'élément est profond dans l'arbre, plus son score est faible.

```
Pour la signature lexicale de la figure 4, la meilleure hypothèse est la suivante : /dilaf/article/sanniize
```

Voici les hypothèses servant à calculer le score du pointeur CDM de la catégorie grammaticale :

- Les éléments sélectionnés doivent contenir du texte et avoir un nombre de valeurs proche ou supérieur à celui du mot-vedette.
- Le score est augmenté si le nom de l'élément contient les mots «pos», « cat » ou « gram ».
- Il est augmenté s'il y a en moyenne peu de mots contenus dans l'élément (de 1 à 3).
- Il est augmenté s'il y a peu de valeurs différentes (une liste fermée de valeurs).

Ensuite, le score final est calculé en divisant le score par le niveau de l'élément dans l'arbre XML.

Pour la signature lexicale de la figure 4, la meilleur hypothèse est la suivante :

```
/dilaf/article/kanandi
```

2.1.3 Entrepôt de données lexicales avec la plateforme iPoLex

Dès le début de mes recherches dans le domaine des ressources lexicales, j'ai fixé une nomenclature pour les noms de fichiers et répertoires servant au stockage des données lexicales. Celle-ci a donné naissance à la première version d'un entrepôt de données lexicales (Mangeot, 1999).

Chaque ressource est décrite dans un répertoire dont le nom est composé du nom abrégé de la ressource, suivi des langues sources et cibles dans l'ordre alphabétique. Celles-ci sont décrites avec un code à 3 lettres selon la norme ISO-639-3. Pour les ressources de la table 1, le répertoire de chaque ressource est décrit dans la table 2.

Papillon_axi-deu-eng-fra-jpn-	Base lexicale multilingue

msa-zho/	Papillon
GDEF_est-fra/	Grand Dictionnaire Estonien-Français
MotÀMot_fra-khm/	Dictionnaire MotÀMot français-khmer
DiLAF_bam-dje-fra-hau-kau-qno-tmh-wol/	Dictionnaires Langue Africaine-Français
Cesselin_fra-jpn/	Dictionnaire Cesselin

Table 2 : nom des répertoires des ressources décrites dans la table 1.

À l'intérieur de son répertoire, chaque ressource est décrite par un ensemble de fichiers XML :

- les métadonnées décrivant la ressource : Nom_metadata.xml

Puis pour chaque volume composant la ressource :

- les métadonnées décrivant le volume avec la liste des pointeurs CDM : Nom_src-metadata.xml ;
- le schéma XML décrivant la microstructure du volume : nom_src.xsd ;
- un article vide au format XML servant de modèle : nom_src-template.xml ;
- une feuille de style pour afficher les données au format HTML : Nom_src-view.xsl (src étant le code de la langue source) ;
- le fichier des données : nom_src.xml

Une première amélioration a consisté à permettre l'accès à distance à cet entrepôt de données lexicales. Celui-ci est géré par un serveur Web mettant en œuvre le protocole WebDAV[19].

Malgré le respect de la nomenclature, l'import d'une ressource restait laborieux et réservé aux utilisateurs avertis. Nous avons alors mis en œuvre une architecture légère en PHP[20] permettant de générer automatiquement tous les fichiers décrits plus haut à partir de formulaires HTML décrivant la ressource et ses volumes.

L'étape suivante a consisté à intégrer à la plateforme le calcul de la signature dictionnairique et des hypothèses concernant les pointeurs CDM. L'ajout d'une nouvelle ressource dans l'entrepôt ne prend alors plus que quelques minutes, au lieu d'une demi-heure dans les conditions précédentes. Surtout, cet ajout est complètement automatisé. L'utilisateur n'a plus qu'à rectifier les pointeurs CDM dont les hypothèses sont erronées.

Le code source de la plateforme est disponible sur notre compte github[21]. Une image Docker permettant d'automatiser l'installation de la plateforme est également disponible[22].

19 https://fr.wikipedia.org/wiki/WebDAV
20 https://fr.wikipedia.org/wiki/Php
21 https://github.com/mangeot/ipolex
22 https://hub.docker.com/r/mangeot/ipolex/

Une première perspective d'enrichissement de la plateforme concerne l'intégration de la gestion automatisée des différentes versions d'une même ressource.

Une seconde perspective concerne la généralisation du concept d'entrepôt à services pour d'autres types d'objets. Dans cette direction, nous avons commencé à mettre en œuvre une plateforme iPoCorp[23] permettant de gérer les corpus bilingues.

Article associé au chapitre 2.1

> Mathieu Mangeot & Valérie Bellynck (2018) *Un devin de microstructures pour importer ou normaliser des ressources lexicales.* Actes de la conférence Lexicologie Terminologie Traduction LTT 2018, Grenoble, France, 27-28 septembre 2018. https://hal.archives-ouvertes.fr/hal-02063815

2.2 Gestion de bases lexicales à structures hétérogènes

Les ressources lexicales étant stockées et décrites de manière claire et précise, je peux me concentrer sur le cœur de cette recherche : permettre un accès à ces ressources au plus grand nombre afin que les utilisateurs puissent les consulter et les enrichir via des contributions.

Le défi scientifique de cette sous-partie est le suivant : importer, consulter et afficher des ressources lexicales hétérogènes sans modifier leur structure (macro et micro) et avec le minimum d'intervention humaine.

Le défi sociétal est la valorisation de ressources lexicales en les rendant accessibles à tous à l'aide d'outils disponibles gratuitement et dont le code est en source ouverte.

2.2.1 Principes de conception

Mon travail de conception et de mise en œuvre de solutions informatiques a été et reste guidé par une série de principes que j'énonce ici :

- la **généricité** est le principe moteur de notre travail de recherche. Il consiste à concevoir à partir d'observations des outils qui puissent s'adapter au plus grand nombre de situations. C'est également un principe qui distingue un travail de recherche en informatique d'un travail d'ingénierie pure.
- l'**automaticité** consiste à automatiser le plus de tâches possibles tout en laissant le contrôle à l'utilisateur.
- l'**intégrité**, principalement celle des données, doit être respectée. Lors d'une manipulation de données, il faut garder toutes les informations originales, y compris celles qui ne servent pas a priori.
- la **traçabilité** des actions est nécessaire afin de pouvoir revenir à un état antérieur en cas d'erreur.
- la **reproductibilité** de notre recherche est également un principe central, surtout pour une science jeune comme l'informatique. Il ne s'agit pas seulement de partager des algorithmes, mais aussi l'ensemble de la chaîne de traitement, afin que d'autres puissent reproduire nos expériences.

23 https://github.com/mangeot/ipocorp/

2.2.2 Gestion de ressources lexicales en ligne avec la plateforme Jibiki

La plateforme Jibiki[24] (Mangeot, 2006) est un environnement de gestion de bases lexicales en ligne permettant d'importer, de consulter et de modifier tout type de ressource lexicale au format XML.

D'un point de vue technique, la plateforme est construite avec la structure logicielle Enhydra, serveur web d'objets java. Elle est fondée sur une architecture à trois tiers comportant :

- une couche données compatible avec SQL25 utilisant Data Object Design Studio, un outil de conversion objet/relationnel, et le système de gestion de bases de données Postgres ;
- une couche métier programmée exclusivement en Java et
- une couche présentation livrant des pages HTML.

La partie développée spécifiquement pour Jibiki contient 279 classes et plus de 82 000 lignes de code. À l'usage, nous avons dû recoder directement une grande partie des interactions avec la base de données afin de pouvoir optimiser le code.

D'un point de vue fonctionnel, les données sont stockées comme suit : une table stocke les métadonnées de chaque ressource ; une autre table stocke les métadonnées de chaque volume ; puis, pour chaque volume, trois tables sont créées dynamiquement lors de l'import : une table pour les données avec chaque article enregistré directement au format XML comme chaîne de caractères, une table pour les index calculés grâce aux pointeurs CDM et une table de liens entre les articles (voir figure 5). Les consultations fonctionnent comme suit : les paramètres de recherche sont convertis en requête SQL. Le résultat est une liste d'articles. Chaque article est traité ensuite par un algorithme de calcul de liens puis le résultat est envoyé à l'algorithme final qui affiche les données.

24 « jibiki » signifie « dictionnaire » en japonais.

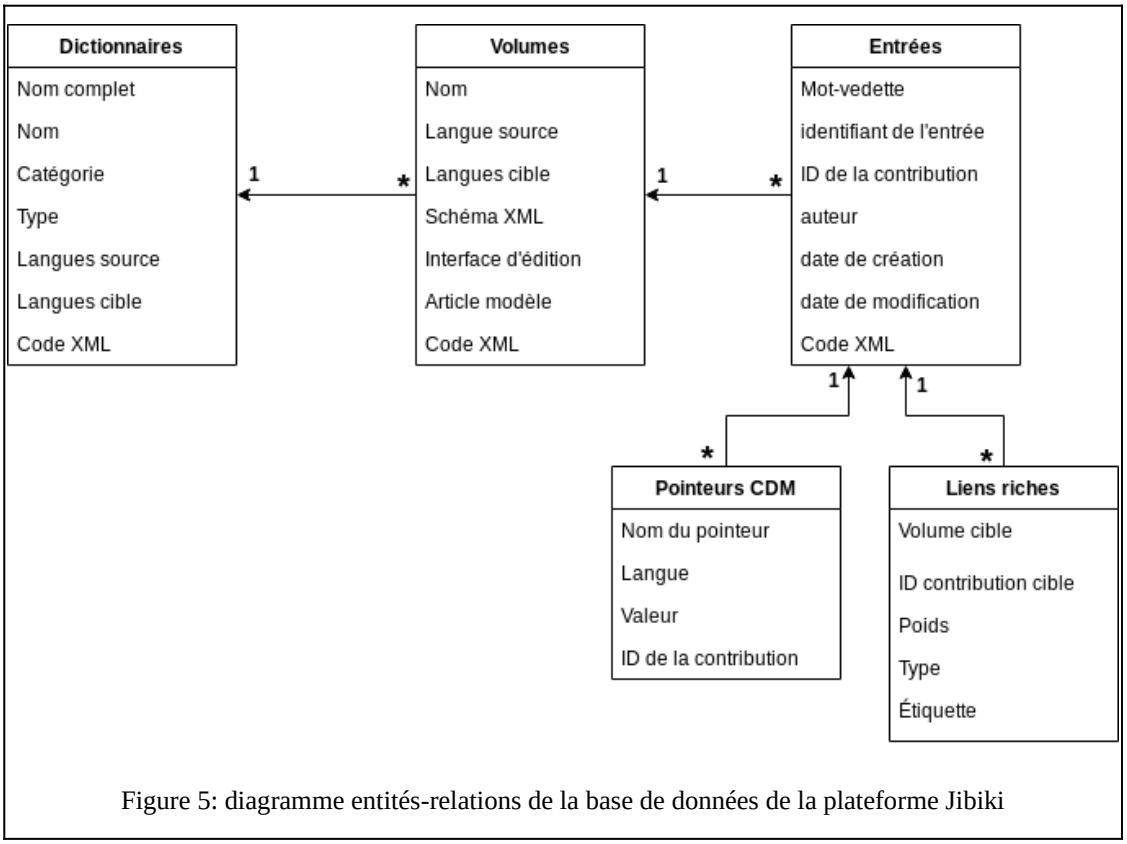


Figure 5: diagramme entités-relations de la base de données de la plateforme Jibiki

D’un point de vue pratique, l’administrateur de la plateforme accède à une interface lui permettant d’importer une ressource. Pour cela, il spécifie l’URL du fichier de métadonnées de la ressource généré précédemment avec l’outil iPoLex. Ensuite, tous les autres fichiers de métadonnées et de données sont chargés automatiquement, puis les données sont indexées en calculant les pointeurs CDM pour chaque article. L’administrateur peut également gérer les droits des comptes utilisateurs de la plateforme. Les utilisateurs ont accès à deux interfaces de consultation : une interface simple permettant de faire des recherches sur les mots-vedettes et une interface avancée permettant de combiner les critères de recherche sur tous les pointeurs CDM (voir figure 6). Lors de l’affichage, un ascenseur infini est affiché à gauche des articles. Il permet de naviguer par ordre alphabétique dans les vedettes du dictionnaire (voir par exemple figure 14). Les applications clientes accèdent à la plateforme en utilisant une interface de programmation (API)25 qui suit le protocole REpresentational State Transfer26 (REST). L’API est décrite en annexe 1 p 83.

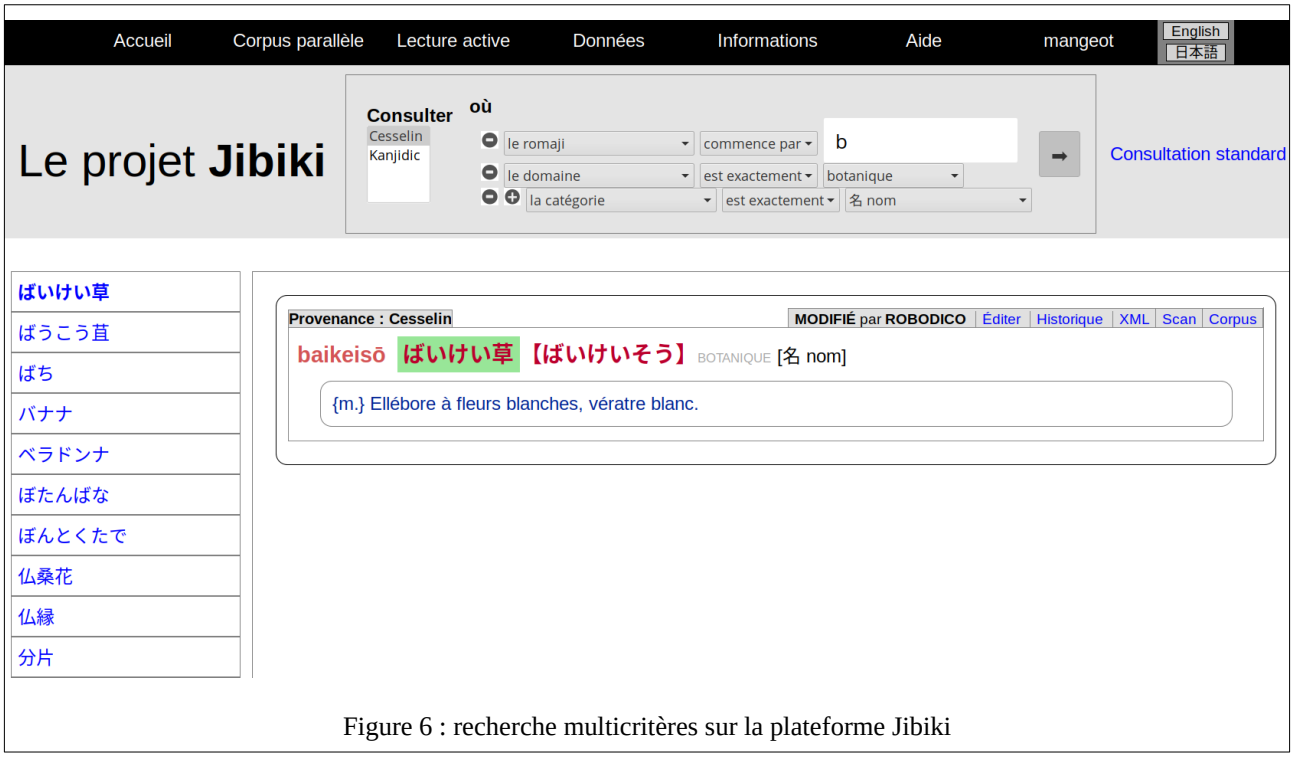


Figure 6 : recherche multicritères sur la plateforme Jibiki

Historiquement, les fondations du premier prototype ont été lancées en 2001 par Gilles Sérasset. Le code était géré par Concurrent Version System25 (CVS). Il a été publié sous licence LGPL. En 2003, j'ai lancé une première version fonctionnelle à l'occasion du début du projet GDEF (voir 3.2). En 2005, CVS a été remplacé par Subversion25 (SVN). C'est aussi l'année où Francis Brunet-Manquat a programmé la recherche avancée multicritères pour le projet LexAlp (Sérasset, 2008). En 2014, j'ai implémenté la gestion de liens riches (voir section suivante) en collaboration avec Ying Zhang, doctorante que je codirigeais avec Christian Boitet et Valérie Bellynck. En 2017, le passage de SVN à Git25 ainsi que la création d'une image Docker26 a permis de simplifier grandement l'installation de la plateforme par des non spécialistes du domaine. Le code source de la plateforme est disponible sur mon compte github[25] ainsi que l'image Docker[26].

En perspective, je souhaite m'orienter vers une architecture à micro-services en recodant le noyau en tant que service disponible uniquement à travers l'API. La gestion des utilisateurs et l'affichage des articles seront pris en charge par d'autres services spécialisés.

2.2.3 Gestion des macrostructures à liens riches

Dans l'introduction, nous avons vu qu'il existe différents types de macrostructures (voir 1.1.1). Nous avons mentionné entre autres la macrostructure pivot. Celle-ci nécessite déjà plusieurs calculs pour réaliser un affichage complet d'un article et de ses articles reliés par le volume pivot. Au cours de nos recherches, nous avons élaboré et expérimenté des macrostructures plus complexes avec

25 https://github.com/mangeot/jibiki
26 https://hub.docker.com/r/mangeot/jibiki/

plusieurs niveaux de liens et des liens de natures différentes. La première implémentation de la gestion des liens dans Jibiki a dû être revue pour tenir compte de ces cas.

Le défi scientifique de cette sous partie est de pouvoir manipuler des macrostructures complexes avec plusieurs niveaux d'objets linguistiques reliés entre eux par des liens riches.

La macrostructure Pivax (Nguyen et al., 2007) a été définie par Christian Boitet et élaborée par Hong-Thai Nguyen durant sa thèse (Nguyen, 2009). Elle comporte trois niveaux : lexie, axème et axie. Les axèmes sont les acceptions monolingues et regroupent des lexies monolingues ayant la même signification. Les axies reroupent des axèmes synonymes de différentes langues dans un pivot central. Dans certaines situations, une base de données lexicale a plusieurs volumes pour une seule langue. Par exemple, lorsqu'il y a plusieurs éditions, ou lorsque la ressource lexicale est créée pour un système de traduction automatique : on peut avoir un volume provenant de Systran, un de Ariane/Héloïse, un de IATE, etc. La distinction est opérée entre informations propriétaires et publiques, le but étant le partage de ressources lexicales entre constructeurs de systèmes de traduction automatique. Cette macrostructure permet de gérer plusieurs volumes dans la même langue. Étant donné une langue, il existe un ou plusieurs volumes de lexies et un seul volume d'axèmes. Pour toute base de données Pivax, il n'y a qu'un seul volume d'axies (voir figure 7).

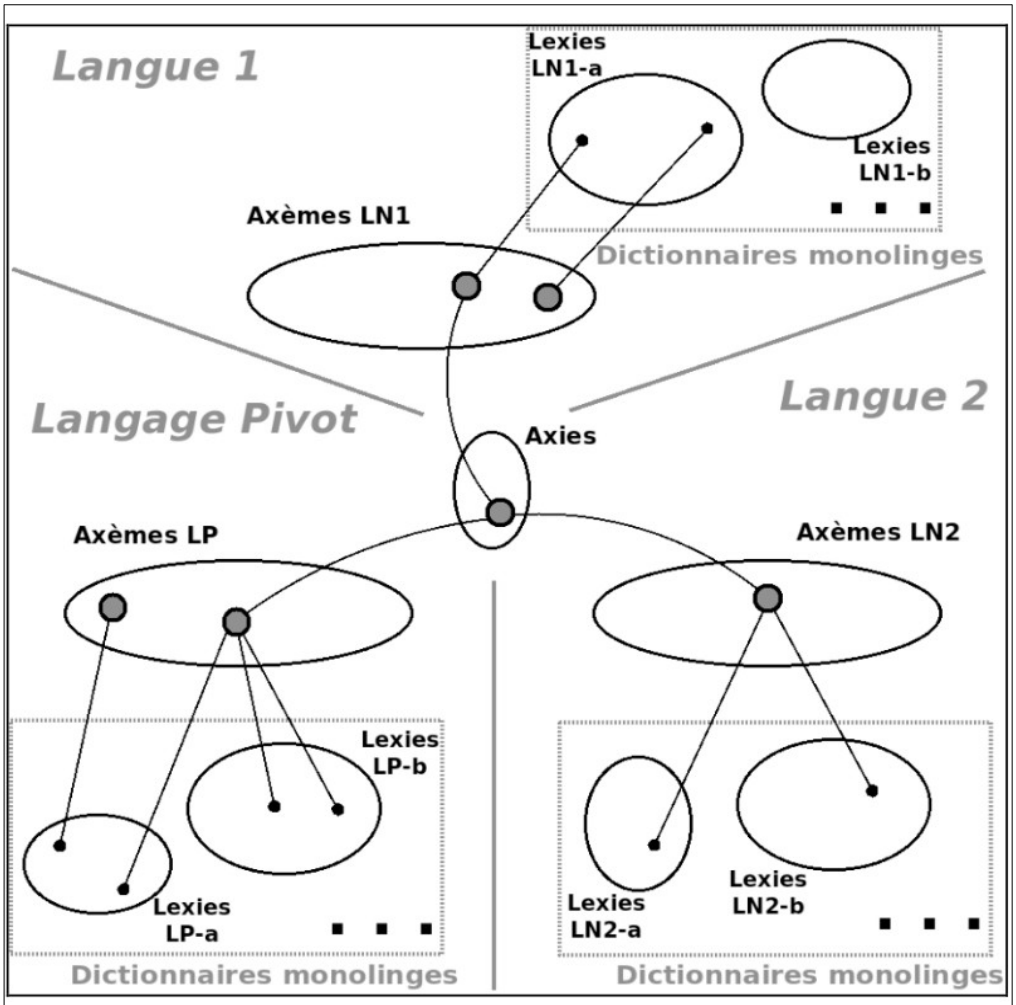


Figure 7 : macrostructure Pivax

La macrostructure ProAxie (Zhang et al., 2014) a été élaborée par Ying Zhang durant sa thèse (Zhang, 2016).

Elle vise à résoudre le problème de la mise en relation de plusieurs termes faisant référence au même référent avec différents degrés de « situement », notamment pour la gestion des acronymes (Zhang, Mangeot, 2013). Dans cette macrostructure, il y a deux couches différentes.

La couche de base se compose de deux types de volume : les volumes de lexèmes et les volumes d'axies. Les axies sont utilisés pour relier les lexies qui sont des traductions exactes les unes des autres, c.à.d. de même nature. Par exemple, on traduit "ONU" par "UN" (voir figure 4) du français vers l'anglais.

La couche "Pro" permet de proposer aux utilisateurs des traductions ayant la même signification référentielle. Cette couche comprend les volumes de prolexèmes (Tran, 2006) et un volume de proaxies. Un prolexème relie des lexies ayant la même signification avec une étiquette (aka, acronyme, définition, etc.). Une proaxie connecte des prolexèmes de différentes langues. Si l'on ne

trouve pas les traductions directement à l'aide de la couche inférieure, on obtiendra les traductions proposées par la couche "Pro".

Par exemple, pour "Organisation des Nations Unies", des traductions comme "United Nations" et "UN" seront proposées, avec l'étiquette "alias".

Pour chaque ressource lexicale suivant la macrostructure ProAxie, il y a un volume d'axies et un volume de proaxies. Pour chaque langue présente dans la ressource, il y a un ou plusieurs volumes de lexies, et un seul volume de prolexèmes. Cela donne trois niveaux de traduction, classés en fonction de la précision obtenue.

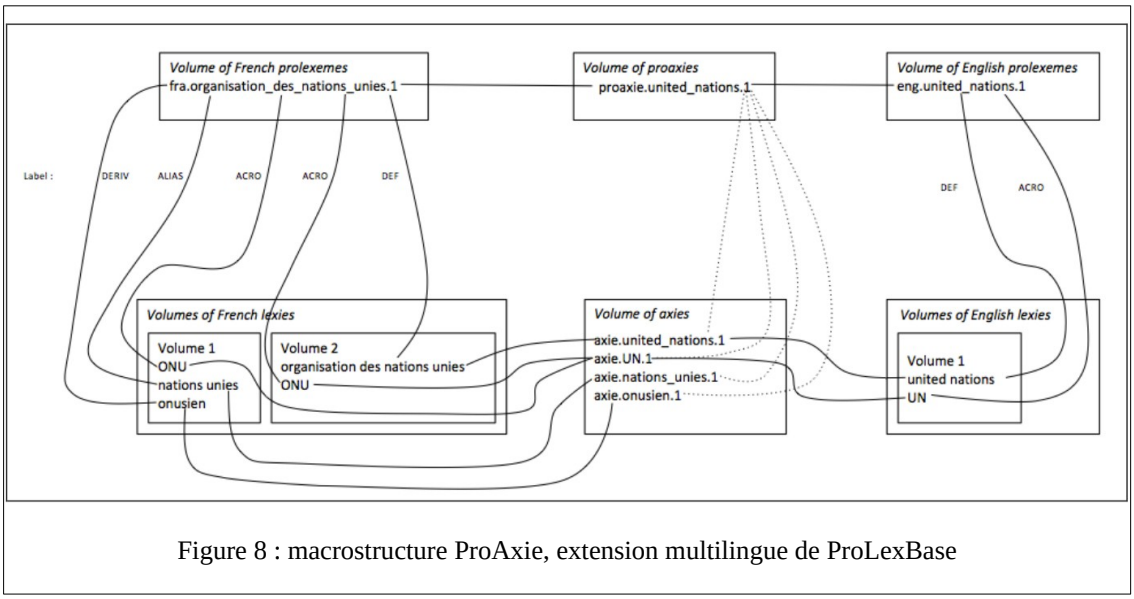


Figure 8 : macrostructure ProAxie, extension multilingue de ProLexBase

Ces macrostructures sont organisées sous forme d'arbre : les feuilles sont les lexies sur lesquelles les recherches s'effectuent ; les branches sont les objets linguistiques intermédiaires (axèmes et prolexèmes) et les racines sont la dernière couche (axies et proaxies). Lors de la recherche d'un article suivie de son affichage, ces macrostructures nécessitent une étape intermédiaire consistant à rechercher les objets linguistiques reliés pour pouvoir tout afficher à la fin.

Les liens relient des objets de différente nature et dans des volumes différents. Ils peuvent porter des informations spécifiques comme des étiquettes.

Nous avons donc mis en œuvre une gestion de liens « riches » qui indexe pour chaque lien :

- l'identifiant des objets source et cible,
- l'identifiant de l'élément XML de l'objet source contenant le lien pour retrouver avec précision l'origine du lien,
- le nom du lien,
- la langue et le volume cible,
- le type de lien,
- une étiquette dont le texte est arbitraire et un poids dont la valeur doit être un nombre réel.

Ces informations sont stockées pour chaque volume source dans une table à part, ce qui permet une recherche plus rapide.

Article associé au chapitre 2.2

> Ying Zhang, Mathieu Mangeot, Valérie Bellynck & Christian Boitet (2014) *Jibiki-LINKS: a tool between traditional dictionaries and lexical networks for modelling lexical resources.* Proceedings of the 4th Workshop on Cognitive Aspects of the Lexicon (CogALex) 2014 (Eds. Michael Zock, Reinhard Rapp, Chu-Ren Huang), Dublin, Ireland, 23 August 2014, 12 p. https://hal.archives-ouvertes.fr/hal-01107544

2.3 Édition en ligne d'articles de dictionnaires

Dans cette sous-partie, nous décrivons plus précisément une fonctionnalité primordiale d'un environnement de gestion de bases lexicales : l'édition des articles.

Le défi scientifique consiste à éditer des articles de dictionnaire de structures différentes sans configuration préalable de l'environnement.

Le défi sociétal est de concevoir des interfaces simples et pratiques afin de ne pas décourager les contributeurs potentiels.

2.3.1 Édition démocratique avec un traitement de texte

Dans un premier temps, nous revenons sur nos débuts dans ce domaine avec l'édition « démocratique » d'articles de dictionnaires.

Lors du projet FeM (Gut et al., 1996), Internet n'était pas encore très répandu. Il n'était pas possible d'envisager que les contributeurs puissent travailler en étant connectés en permanence à un serveur. Il fallait trouver une solution facile à mettre en œuvre. Christian Boitet, organisateur de ce projet a (en 1991) d'abord proposé une gestion dans le système de gestion de bases de données 4D, mais ce logiciel n'était disponible que sur Mac, et surtout les lexicographes n'aimaient pas travailler fiche par fiche. Il a alors proposé une représentation en Word/RTF, chaque ligne correspondant à un élément « syntaxique » (mot-vedette, classe, prononciation, etc.) que l'on retrouve plus tard dans CDM. Word est alors utilisé comme « éditeur pseudo-syntaxique ». Le dictionnaire fut alors développé sous forme de 25 fichiers Word.

En 1995, Mathieu Lafourcade construisit des programmes d'import dans une base de données, ce qui permet de transformer la microstructure fra→eng→msa en fra→eng +fra→msa et, pour l'impression papier, de transformer la transcription phonétique « à la française » en Alphabet Phonétique International À partir de la même base de données, il construisit un outil autonome, ALEX, (programmée en Common Lisp) pour la consultation « multi-niveaux ».

Durant mon DEA (Mangeot, 1997), j'ai repris et amélioré la technique d'import des fichiers en base de données pour le projet de construction du dictionnaire UNL-français. J'y ai ajouté la génération

des fichiers RTF à partir de contenu existant dans la base ainsi que la vérification de la structure du fichier en cours d'édition à l'aide de macros Word.

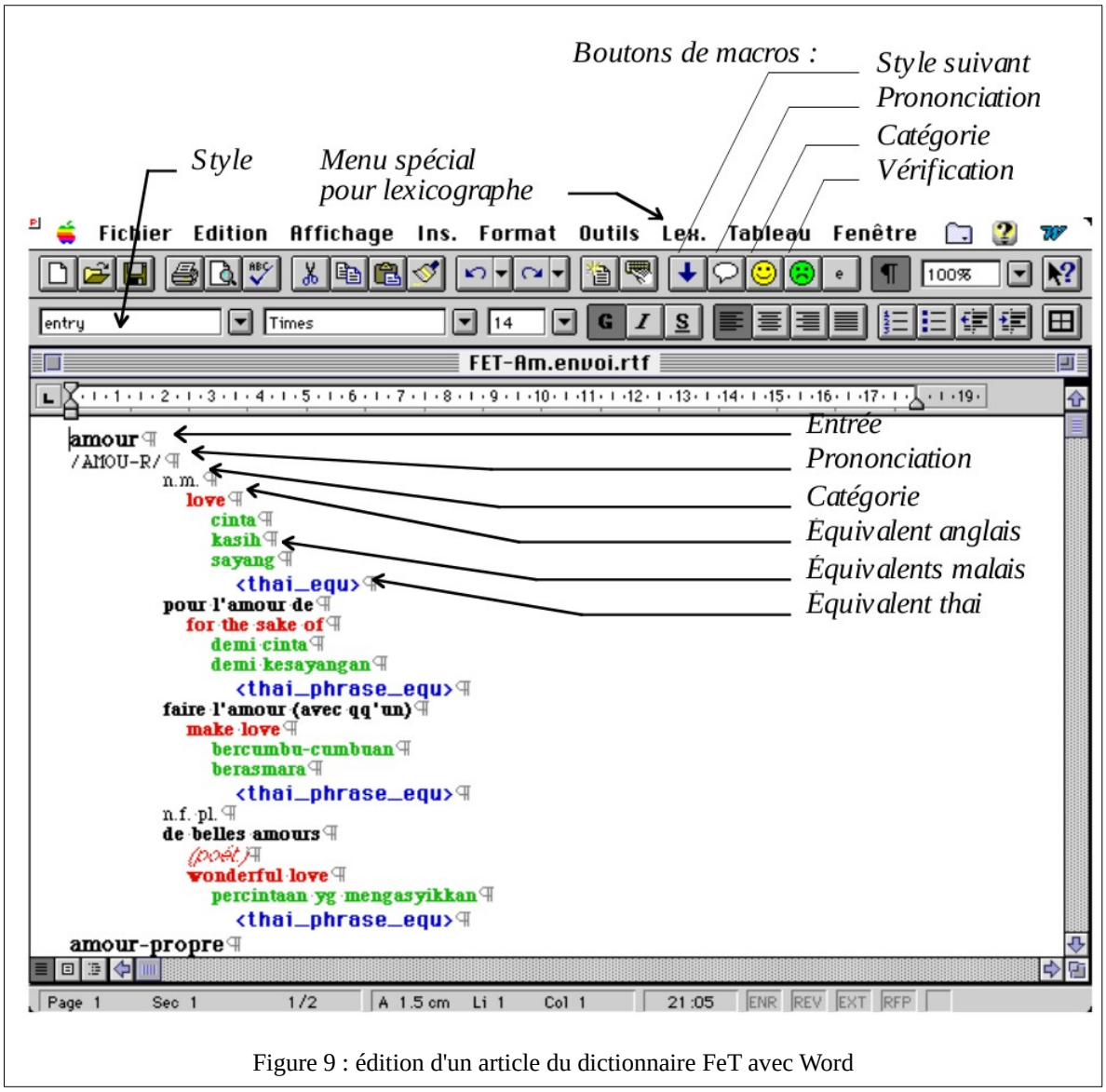


Figure 9 : édition d'un article du dictionnaire FeT avec Word

Cette technique a permis de construire rapidement le dictionnaire. L'inconvénient majeur réside dans le fait que les rédacteurs pouvaient modifier une ligne pour qu'elle ressemble à un style sans que ce style soit réellement affecté à la ligne. Des erreurs de ce type ont été détectées lors de l'import des fichiers de contribution.

2.3.2 Édition en ligne d'un article complet

L'avènement d'Internet et surtout du Web a rendu possible voire nécessaire la conception d'un éditeur accessible avec un simple navigateur Web. Il s'agit donc d'utiliser les formulaires HTML qui permettent une interaction avec les utilisateurs. Ils sont par contre limités au niveau des interacteurs : champs texte, cases à cocher et menus déroulants. Il n'existe par contre pas d'interacteur de liste pour ajouter ou supprimer un élément.

Lors de mon séjour post-doctoral à l'Institut National d'Informatique, à Tokyo, Japon (de septembre 2001 à mars 2004), avec David Thévenin, spécialiste de l'adaptabilité des interfaces, nous avons donc créé de nouveaux interacteurs en combinant les interacteurs de base (voir figure 10, les boutons « + » et « - » permettent de gérer une liste), puis nous avons programmé un générateur d'interfaces qui utilise le schéma XML de la microstructure de chaque article ainsi que les interacteurs élaborés pour générer automatiquement une interface Web utilisant un formulaire HTML permettant d'éditer en ligne un article de dictionnaire au format XML (Mangeot, Thevenin, 2004).

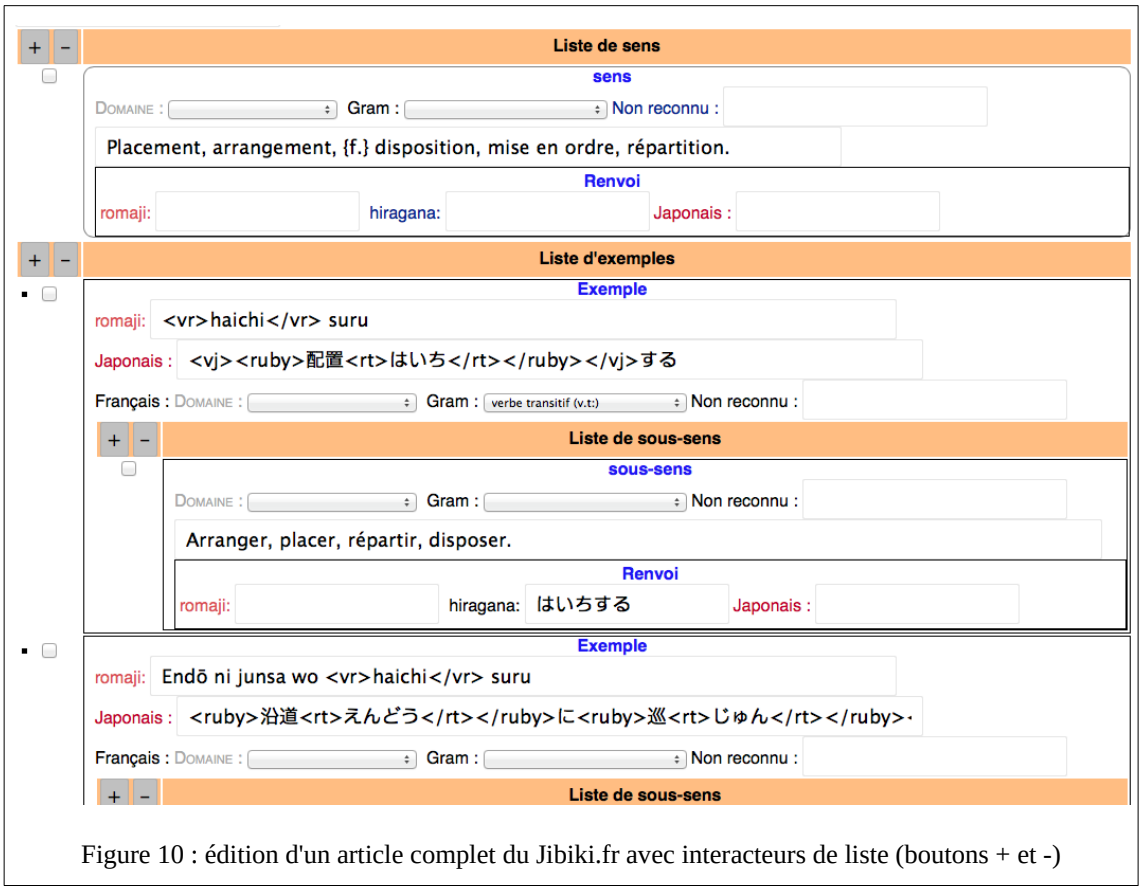


Figure 10 : édition d'un article complet du Jibiki.fr avec interacteurs de liste (boutons + et -)

Cette solution est tout à fait fonctionnelle mais son affichage est radicalement différent de celui de l'article en mode consultation. Les contributeurs peuvent être perdus lors de l'édition d'un article un peu long. De plus, le changement de mode de consultation vers l'édition demande d'afficher une nouvelle page, ce qui gêne la navigation.

2.3.3 Édition rapide pour correction

Lors du projet Jibiki.fr (Mangeot-Nagata, 2016), la majeure partie des données produites étaient issues de lecture optique. Il y avait donc un certain nombre d'erreurs à corriger. La plupart du temps, cela ne nécessite pas de restructurer les articles. Une édition de texte suffit.

Pour répondre aux problèmes posés par l'édition complète et ne pas décourager les éventuels contributeurs, j'ai élaboré une solution plus simple qui permet d'éditer du texte sans modifier l'affichage de consultation. Il s'agit d'un module javascript/jquery qui remplace chaque zone de texte par un champ texte éditable lorsque l'utilisateur a double cliqué dessus (voir figure 11). Le texte édité est ensuite envoyé au serveur accompagné du pointeur XPath[15] de l'élément XML à modifier dans l'article à l'aide d'une requête vers l'API REST[Error: Reference source not found] (voir 2.2.2).

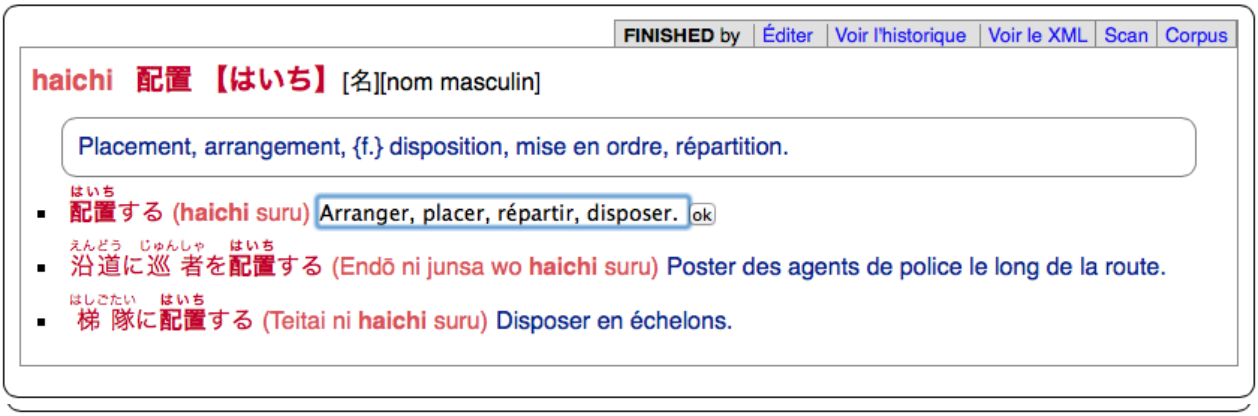


Figure 11 : édition rapide d'une zone de texte dans le dictionnaire Jibiki.fr

En perspective, je souhaite combiner les deux approches de l'édition complète et de l'édition rapide pour concevoir un mode d'édition sur place qui respecte l'affichage de consultation et qui permet d'éditer la structure complète de l'article.

Article associé au chapitre 2.3

Mathieu Mangeot & David Thevenin (2004) *Online Generic Editing of Heterogeneous Dictionary Entries in Papillon Project.* Proc. of COLING 2004, ISSCO, Université de Genève, Switzerland, 23-27 August 2004, vol 2/2, pp 1029-1035. https://hal.archives-ouvertes.fr/hal-00968639

3 Projets de création de ressources lexicales

Les projets présentés dans cette partie forment le terrain d'expérimentation principal de mes recherches. Ce sont les principaux projets de création de dictionnaires auxquels j'ai participé et qui utilisent l'environnement Jibiki pour consulter et éditer les données. Les micro et macrostructures ainsi que la méthodologie de construction varient beaucoup d'un projet à l'autre. Ils confirment l'utilité de Jibiki en tant que plateforme de développement de ressources lexicales hétérogènes.

3.1 Base multilingue à structure pivot : le projet Papillon

3.1.1 Présentation du projet

Le projet démarra en 2000 à l'initiative de François Brown De Colstoun, alors attaché scientifique à l'Ambassade de France au Japon et Emmanuel Planas, en séjour postdoctoral à NTT takanohara, Japon et Mutsuko Tomokiyo, chercheur invitée au GETA-CLIPS. Il fait suite au constat de la communauté francophone au Japon : il n'existe pas de dictionnaire japonais-français informatisé en accès libre. Les apprenants francophones doivent se rabattre sur des dictionnaires japonais-anglais et multiplient de ce fait les contresens.

Le défi sociétal est donc de construire une ressource lexicale comprenant au moins le japonais, l'anglais et le français à large couverture et de bonne qualité accessible gratuitement sur le Web.

Les défis scientifiques sont nombreux : expérimenter une base lexicale à structure pivot, utiliser la lexicographie explicative et combinatoire à grande échelle et surtout concevoir un environnement générique pour la gestion de bases lexicales en ligne.

À la suite de notre soutenance, nous sommes parti à l'Institut National d'Informatique (NII), à Tokyo pour travailler sur ce projet lors d'un postdoc financé par la JSPS27.

Dès le début du projet, nous avons organisé un atelier annuel pour discuter de toutes les questions ouvertes par les défis. Le premier atelier a eu lieu au NII en 2002, le suivant à Sapporo en 2003, ensuite à Genève en 2004 et enfin à Chiang Mai en 2005. Des acteurs importants de la lexicographie informatique bilingue japonaise y ont assisté comme Jean-Marc Desperrier, auteur d'un premier dictionnaire français-japonais sur Internet (Desperrier, 2002); Jim Breen, auteur du dictionnaire japonais-anglais JMdict (Breen, 2004) ou encore Ulrich Apel, auteur du dictionnaire japonais-allemand WaDokuJiten (Apel, 2002).

3.1.2 Structures des bases lexicales

Le projet Papillon distingue deux bases lexicales. La première, appelée Papillon-CDM regroupe toutes les ressources lexicales disponibles contenant du japonais, du français et de l'anglais. Elle contient plus de deux millions d'articles en 8 langues mais pas d'axies (liens interlingues) ni même de lexies. La deuxième base lexicale, appelée Papillon NADIA, fonde sa macrostructure (Mangeot, Sérasset, 2001) sur la notion d'acception interlingue. Chaque volume monolingue est vu comme un

ensemble d'acceptions d'une langue. Les liens entre les langues sont établis grâce à un volume pivot au centre qui contient un ensemble d'acceptions interlingues (que nous appelons axies).

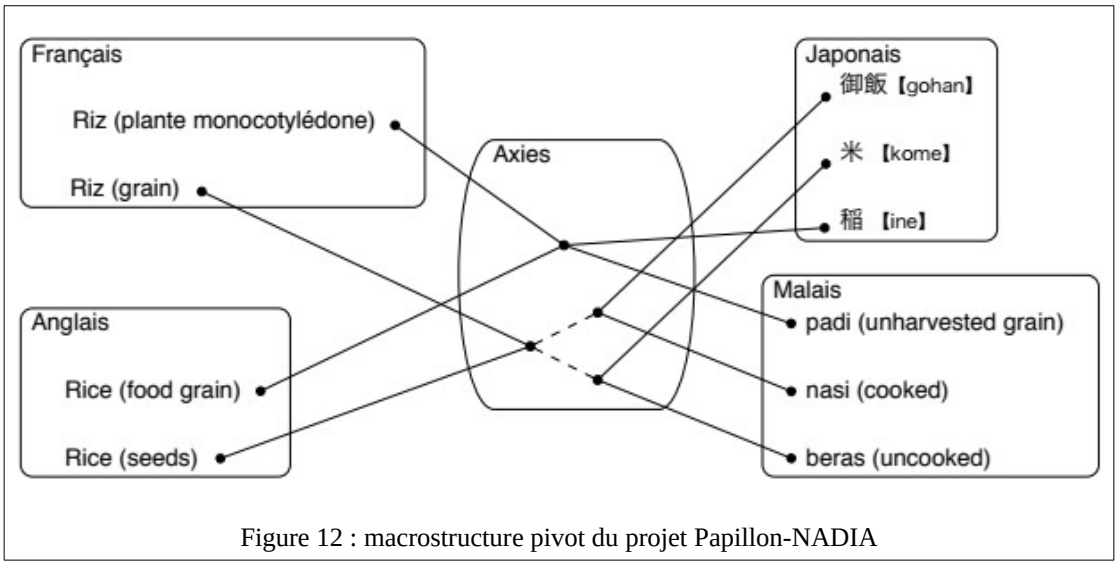


Figure 12 : macrostructure pivot du projet Papillon-NADIA

Les problèmes contrastifs d'équivalence lexicale sont représentés grâce à un lien entre axies. Ce phénomène se retrouve dans l'exemple de la traduction du mot « riz » dans 4 langues (voir figure 12). Pour cet exemple, nous avons utilisé les acceptions définies dans des dictionnaires du commerce.

Chaque langue définit une acception correspondant au sens français désignant le riz comme une plante céréalière mais ni le français, ni l'anglais ne définissent d'acceptions différentes suivant que le « riz » (considéré comme aliment) est cuit ou non alors que le japonais et le malais font cette distinction.

La microstructure des articles est reprise de celle du DEC (Polguère, 2000). C'est une application de la lexicographie explicative et combinatoire, partie de la théorie sens-texte (Mel'čuk, 1995). Cette théorie est indépendante des langues, ce qui nous permet de conserver la même microstructure quelle que soit la langue. L'unité lexicale à la base d'un article est la lexie (appelée également sens de mot). Elle est équivalente à l'acception monolingue du paragraphe précédent.

Nom de l'unité lexicale	Meurtre.1
Propriétés grammaticales	nom, masc
Formule sémantique	action de tuer : PAR L'individu X DE L'individu Y
Régime	X = I = de N, A-poss Y = II = de N, A-poss
Fonctions lexicales	QSyn assassinat, homicide#1 ; crime V0 tuer A0 meurtrier-adj S1 auteur [de ART]//meurtrier-n /* Nom pour X*/
Exemples	La mésentente pourrait être le mobile du meurtre.
Idiomes	_appel au meurtre_, _crier au meurtre_

Figure 13 : exemple d'article de la base du projet Papillon-NADIA

La méthodologie de construction de la base consiste à récupérer des dictionnaires existants pour créer chaque volume monolingue de la base, puis à utiliser des dictionnaires bilingues pour créer des axies. Ensuite, de nouveaux liens sont créés soit par des contributeurs soit par calcul à l'aide de vecteurs conceptuels entre les lexies.

3.1.3 Résultats

Contributeurs du projet Papillon	**2000**	**2001**	**2002**	**2003**	**2004**	**2005**	**TOTAL**
François Brown de Colstoun	20 h	20 h	30 h	30 h	10 h	10 h	120 h
Christian Boitet	10 h	10 h	30 h	30 h	10 h	10 h	100 h
Guy Lapalme				40 h			40 h
Mathieu Mangeot	50 h	280 h	1800 h	1700 h	50 h	30 h	3910 h
Sitii Khaotijah Mohammad				300 h			300 h
Emmanuel Planas	30 h	20 h	30 h	30 h	10 h		120 h
Gilles Sérasset	40 h	100 h	100 h	100 h	50 h	30 h	420 h
Mutsuko Tomokiyo	30 h	150 h	150 h	150 h	10 h	10 h	500 h
TOTAL	180 h	580 h	2140 h	2380 h	140 h	90 h	5510 h

Table 3 : estimation des heures passés par les principaux contributeurs sur le projet Papillon

Les données collectées pour être importées dans la structure Papillon-CDM sont conséquentes : plus d'un million d'entrées issues d'une vingtaine de ressources ont été collectées. Certaines sont disponibles sur une instance de la plateforme Jibiki[27].

Les articles finalement construits suivant la microstructure Papillon-NADIA sont très peu nombreux. Pour le français, il existe 813 lexies dont 631 converties depuis le DEC par Guy

27 https://papillon.imag.fr/

Lapalme (Lapalme, Sérasset, 2003) et 202 rédigées par Mutsuko Tomokiyo. Pour l'anglais, on trouve 183 lexies et pour le japonais, 105 lexies toutes rédigées également par Mutsuko. Pour le malais, on trouve 134 lexies rédigées par Sitii Khaotijah Mohammad. Le volume interlingue contient 42 axies créées manuellement entre l'anglais, le français et le japonais. Ces données servent de preuve de concept pour présenter la macrostructure pivot utilisée pour cette base par acceptions.

Les défis scientifiques du projet étaient certainement trop nombreux pour être relevés rapidement. Si l'on observe les suites du projet, on constate par contre qu'il a donné lieu à de nombreux succès. La plateforme Jibiki a été et est toujours utilisée dans de nombreux projets de création de ressources lexicales. L'architecture interlingue a été reprise avec succès par Gilles Sérasset pour le projet LexAlp (Sérasset, 2008) et par Hong-Thai Nguyen pour le projet PIVAX (via des systèmes de traduction automatique hétérogènes) en 2006-2009 (Nguyen, 2009). Face aux difficultés que rencontrent les contributeurs devant une microstructure si détaillée, l'idée de les faire contribuer sous forme de jeu est apparue au cours du projet. Elle a été concrétisée avec succès par Mathieu Lafourcade dans son projet JeuxDeMots (Lafourcade, Joubert, 2008).

Quant au défi sociétal du projet, il a été relevé par le projet Jibiki.fr (voir 3.5). En perspective de recherche, nous souhaitons également expérimenter une architecture pivot pour construire des bases lexicales multilingues à large couverture (voir 6.1).

Article associé au chapitre 3.1

> Mathieu Mangeot, Gilles Sérasset & Mathieu Lafourcade (2003) *Construction collaborative de données lexicales multilingues, le projet Papillon.* Revue TAL, édition spéciale, Les dictionnaires électroniques : pour les personnes, les machines ou pour les deux ? (Electronic dictionaries: for humans, machines or both?) Ed. Michael Zock & John Carroll, Vol. 44:2/2003, pp. 151-176. https://hal.archives-ouvertes.fr/hal-00340723

3.2 Dictionnaire estonien-français : le projet GDEF

3.2.1 Présentation du projet

Le projet de Grand Dictionnaire Estonien Français28 démarra en 2003 à l'initiative d'Antoine Chalvin, enseignant-chercheur de langue estonienne à l'INALCO. L'adhésion de l'Estonie à l'Union européenne (effective le 1er mai 2004) et le développement de l'enseignement du français dans ce pays firent apparaître comme une urgente nécessité la réalisation d'un nouveau dictionnaire estonien-français. Il n'existait en effet aucun grand dictionnaire estonien-français répondant aux exigence de la lexicographie moderne et reflétant l'état actuel de la langue estonienne, dont le lexique a considérablement évolué depuis le retour à l'indépendance de l'Estonie en 1991, au cours des quinze dernières années. Le dernier dictionnaire d'une certaine ampleur (Kann, Kaplinski, 1979) réalisé à l'époque soviétique, est aujourd'hui en grande partie périmé. Il est en outre entaché

de graves défauts de conception, auxquels il ne semble pas possible de remédier par une simple mise à jour.

Le défi sociétal était donc de répondre à un besoin urgent de dictionnaire estonien-français moderne.

Les défis scientifiques sont en premier lieu d'utiliser la plateforme Jibiki pour la création d'une autre ressource lexicale en ligne que celle pour laquelle elle a été conçue, ce qui nous a permis de tester l'aspect générique ; ensuite d'utiliser une macrostructure avec deux volumes séparés et enfin de mettre en œuvre un processus d'édition à plusieurs niveaux.

3.2.2 Structures de la base lexicale

Le dictionnaire étant bilingue mais monodirectionnel estonien→ français, la macrostructure pourrait n'être constituée que d'un seul volume. Nous avons cependant décidé de séparer le volume estonien du volume français. Cela nous permet de garder une cohérence dans les articles français. Un autre avantage de cette structuration est de faciliter la création d'un squelette de dictionnaire en sens inverse français→ estonien. Lors de l'édition d'un article estonien, une interface de recherche dans le volume français s'affiche à la place de l'équivalent français. Le rédacteur sélectionne un article français et le lien est ensuite sauvegardé dans la structure XML de l'article estonien. Ce lien servira ensuite pour l'affichage complet de l'article. Il sera remplacé par le mot-vedette de l'article français.

Le dictionnaire étant destiné aux traducteurs, la microstructure des articles estoniens est très détaillée. L'article est composé d'un bloc forme, d'une suite de blocs grammaticaux, d'un bloc phraséologique, et d'indications sur la fréquence de la vedette dans des corpus. Le bloc forme regroupe les éléments suivants : vedette, variante, type, particule, numéro d'homographe, registre, domaine ainsi qu'un bloc morphologique qui détaille la flexion et les formes. Le bloc grammatical est composé d'une catégorie grammaticale, d'un registre, d'un domaine et d'une suite de blocs sémantiques. Le bloc sémantique est composé d'une indication sémantique (glose), d'un registre, d'un domaine, d'une suite de sous-blocs sémantiques et d'un bloc d'exemples. Le sous-bloc sémantique est composé d'une indication sémantique et d'un bloc contextuel. Celui-ci est composé d'une indication contextuelle (glose) et d'un bloc équivalent. Celui-ci est composé de l'équivalent français, de mots se trouvant avant et après cet équivalent (voir « de lecture » après « livre » dans la figure 14), du registre et des rections pour les verbes. Le bloc d'exemples est composé d'une suite d'exemples. Chaque exemple est composé de l'exemple estonien, de son registre, de son domaine et d'un bloc de traduction de l'exemple. Celui-ci est composé de la traduction française de l'exemple, d'un registre, d'une rection, d'une indication sémantique (glose) et d'une locution si l'exemple se traduit par une locution. Le bloc phraséologique est composé d'une suite d'unités phraséologiques. Chacune est composée comme un exemple.

Il existe donc quatre niveaux de détail (bloc sémantique, sous-bloc sémantique, bloc contextuel et bloc équivalent) avant de pouvoir décrire l'équivalent français de la vedette estonienne. Cela permet de bien détailler les contextes de traduction. L'article de la figure 14 contient 3 blocs sémantiques.

Le premier bloc sémantique contient un sous-bloc sémantique qui contient un bloc contextuel qui contient deux blocs équivalents pour « abécédaire » et « livre ».

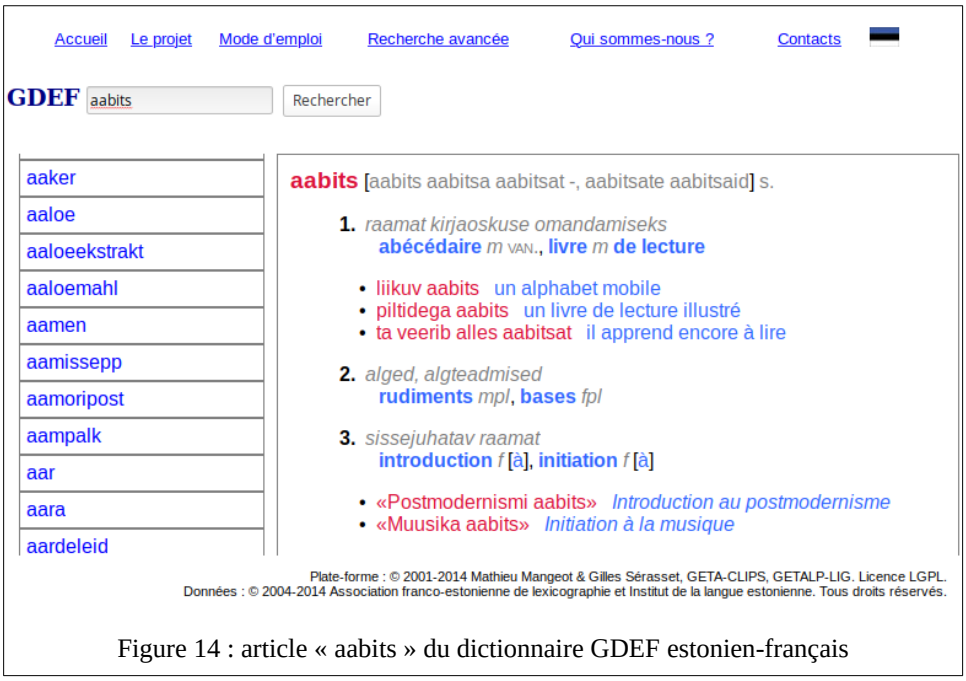


Figure 14 : article « aabits » du dictionnaire GDEF estonien-français

La méthodologie de construction fut de commencer par créer un squelette pour chaque volume. Pour l'estonien, nous avons utilisé le grand dictionnaire estonien-russe (EVS)28 (Eesti Keele Instituut, 1997). Nous en avons extrait les mots vedettes et leurs informations morphologiques, les subdivisions sémantiques de chaque article avec les indications sémantiques correspondantes, les indications de domaine de spécialité et de registre, les exemples et les locutions.

Environ 6 330 articles estoniens ont en outre été pourvus d'une indication de fréquence tirée du Dictionnaire des fréquences de l'estonien écrit (Kaalep, Muischnek, 2002). Cette information nous permet de rédiger en priorité les articles associés aux mots les plus fréquents, afin de maximiser l'utilité du dictionnaire en cours d'élaboration qui est immédiatement consultable sur Internet.

Pour le français, nous avons utilisé la base Morphalou28, dictionnaire des formes fléchies du français élaboré par le laboratoire ATILF et comprenant environ 66 000 lemmes (à l'époque). Nous en avons extrait également la catégorie grammaticale, le genre des substantifs, ainsi que les pluriels et féminins « irréguliers ». Nous avons ensuite ajouté d'autres données (h aspiré, classe de conjugaison, mots composés, etc.).

Ensuite, chaque article passe par un processus en 3 étapes : un premier contributeur rédige l'article, un deuxième le révise et enfin un troisième le valide. Les deux premiers contributeurs sont constitués de couples francophone + estonophone pour permettre une révision optimale. La validation est effectuée par les responsables du projet.

Un corpus bilingue[28] aligné de 5 millions de mots (littératures française et estonienne, législation européenne, débats du parlement européen, Bible, textes divers) a également été constitué en parallèle aux projet de dictionnaire. Il permet aux rédacteurs de chercher notamment des exemples.

3.2.3 Résultats

Contributeurs	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18	19	Total
Martin Carayol						5	220	300	100	15	30							670
Antoine Chalvin	360	360	360	180	180	180	180	180	180	180	180	180	180	180	180	180	180	3 600
Madis Jurviste		360	360	180	180	180	180	180	180	180	180	180	180	180	180	180	180	3 240
Mathieu Mangeot	300	200	100	60	30	20	10	10	10	10	10	10	10	10	10	10	10	820
Ülo Siirak						40	340	270	150	300	500	40	50	70	50	5		1 815
Eva Toulouze			300	600	125	40	5	3	10									1 083
+71 contributeurs rétribués						170	300	310	230	220	120	15	130	180	220	240	80	2 215
TOTAL	660	920	1120	1020	515	635	1235	1253	860	905	1020	425	550	620	640	615	450	13 443

Table 4 : estimation des heures passées par les principaux contributeurs sur le projet GDEF

Avec une estimation basse de ¼ d'heure par contribution, les contributeurs ont passé au total plus de 13 000 heures sur le projet !

Le dictionnaire contient actuellement 40 614 articles dont :

- 24 523 articles validés,
- 6 543 articles révisés en attente de validation,

9 548 articles finis en attente de révision.

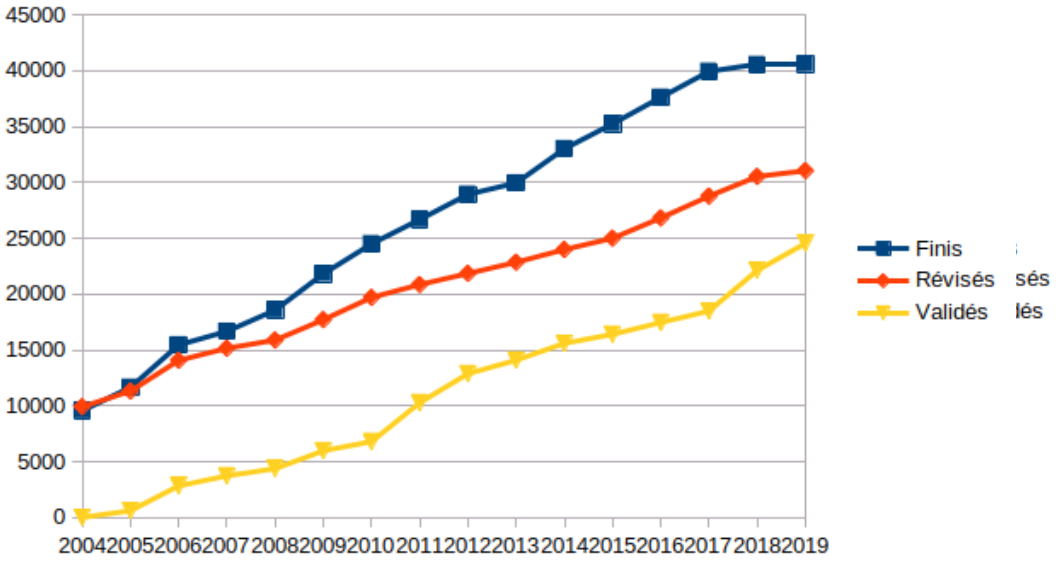


Figure 15: nombre d'articles finis, révisés et validés par an pour le dictionnaire GDEF

28 https://corpus.estfra.ee/

Après 16 ans d'existence, le projet est toujours actif. Il comptabilise 80 contributeurs depuis le début. Pour le mois de juin 2019, 59 articles ont été révisés et 271 articles validés. La rédaction avance au gré des financements, qui ne sont pas des plus abondants pour ce type de projet.

Le projet GDEF a validé l'utilité de la plateforme Jibiki pour construire collectivement et consulter un dictionnaire en ligne.

Article associé au chapitre 3.2

> Antoine Chalvin & Mathieu Mangeot (2006) *Méthodes et outils pour la lexicographie bilingue en ligne : le cas du Grand Dictionnaire Estonien-Français.* Actes d'EURALEX 2006, Turin, Italie, 6-9 septembre 2006, 8 p. https://hal.archives-ouvertes.fr/hal-00959238

3.3 Dictionnaire khmer-français : le projet MotÀMot

3.3.1 Présentation du projet

Le projet MotÀMot envisage au départ de créer une base lexicale multilingue pour des langues d'Asie du Sud-Est : vietnamien, khmer, thaï, malais ainsi que le français et l'anglais (Mangeot, Hong Thai Nguyen, 2009). Ce projet, lancé en 2008 a été partiellement financé jusqu'en 2010 par l'Agence Universitaire de la Francophonie au travers de l'appel à projets « action de recherche en réseau ».

Les pays de l'ancienne Indochine ont encore gardé des liens étroits avec la francophonie. Mais les langues de ces pays sont souvent peu dotées informatiquement (Berment, 2004). De plus, comme dans le cas du japonais, seuls des dictionnaires vers l'anglais sont informatisés sauf dans de rares cas (Max Reinhorn, Berment, 2013).

Lors de nos recherche préliminaire de données avec Vanra Yeng, stagiaire cambodgien participant au projet, nous avons contacté Denis Richer, ethnolinguiste français installé à Siem Reap au Cambodge et président de l'association « Pays perdu ». Son équipe avait rédigé un dictionnaire français-khmer au format Word de la fin des années 1990 à 2006. Il nous a autorisé à utiliser ces données dans notre projet. Nous avons tout de suite saisi cette opportunité.

Le défi sociétal de ce projet est sensiblement le même que pour le projet Papillon : il n'existe pas de pas de dictionnaire informatisé français-khmer mais uniquement anglais-khmer.

Le défi scientifique se situe au niveau de l'utilisation d'une macrostructure pivot pour la création, l'affichage et l'édition des articles avec des niveaux de qualité différents.

3.3.2 Structures de la base lexicale

La macrostructure de la base est une structure pivot du type de celle de Papillon-NADIA (voir 3.1.2). J'ai également rajouté des liens directs entre volumes monolingues lorsqu'il n'est pas possible de créer un lien de sens à sens. Dans ce cas, un lien direct d'un sens de mot d'une langue

vers un article (vocable) d'une autre langue. Ce lien sera marqué comme étant à raffiner par la suite (voir figure 17).

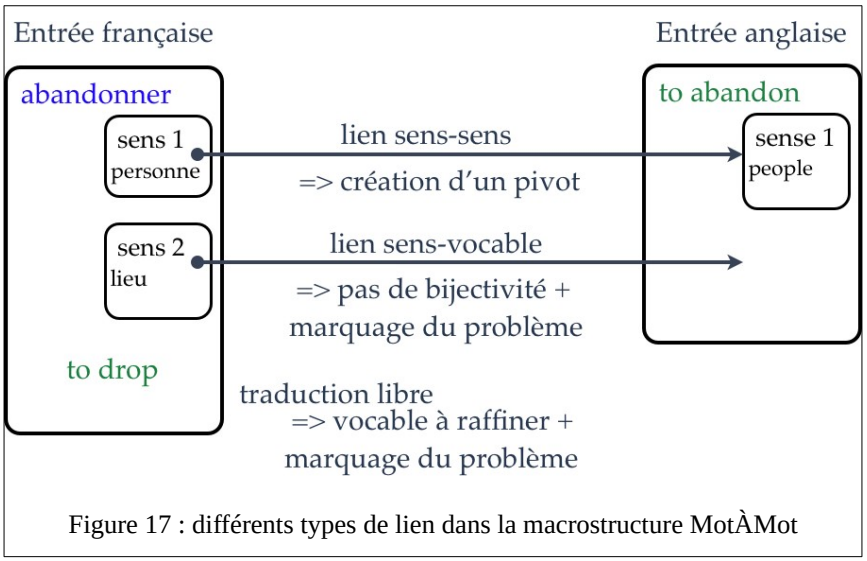


Figure 17 : différents types de lien dans la macrostructure MotÀMot

La microstructure était au départ calquée sur celle du projet Papillon-NADIA, puis nous l'avons modifiée pour qu'elle corresponde aux données récupérées du dictionnaire français-khmer de « Pays perdu » : chaque article est un vocable contenant une liste de sens de mot.

La méthodologie de construction a consisté d'abord à récupérer les données des fichiers Word (voir 4.2.2). Les mots khmer du dictionnaire français-khmer étaient transcrits uniquement en alphabet phonétique international et non en alphabet khmer. Nous avons programmé un transducteur à états finis afin de convertir l'API en khmer (Mangeot, Touch, 2010). Malheureusement, dans de nombreux mots, il reste des racines de sanskrit qui ne se prononcent pas. Les sorties en khmer doivent être impérativement corrigées.

Les données ont ensuite été mises en ligne pour consultation et contributions par la communauté d'utilisateurs.

Pour indiquer la qualité des données, nous avons mis en œuvre un système de notation de 1 à 5 étoiles. Une étoile correspond à un brouillon lorsque la qualité des données réutilisées n'est pas connue. 5 étoiles correspond à un article certifié par un traducteur expert. De manière similaire, nous avons expérimenté la gestion automatique des contributions dans (Courtin et al., 2010). Si le travail d'un contributeur est systématiquement validé sans corrections, alors celui-ci augmentera d'un niveau à partir d'un certain seuil (par exemple 10 contributions).

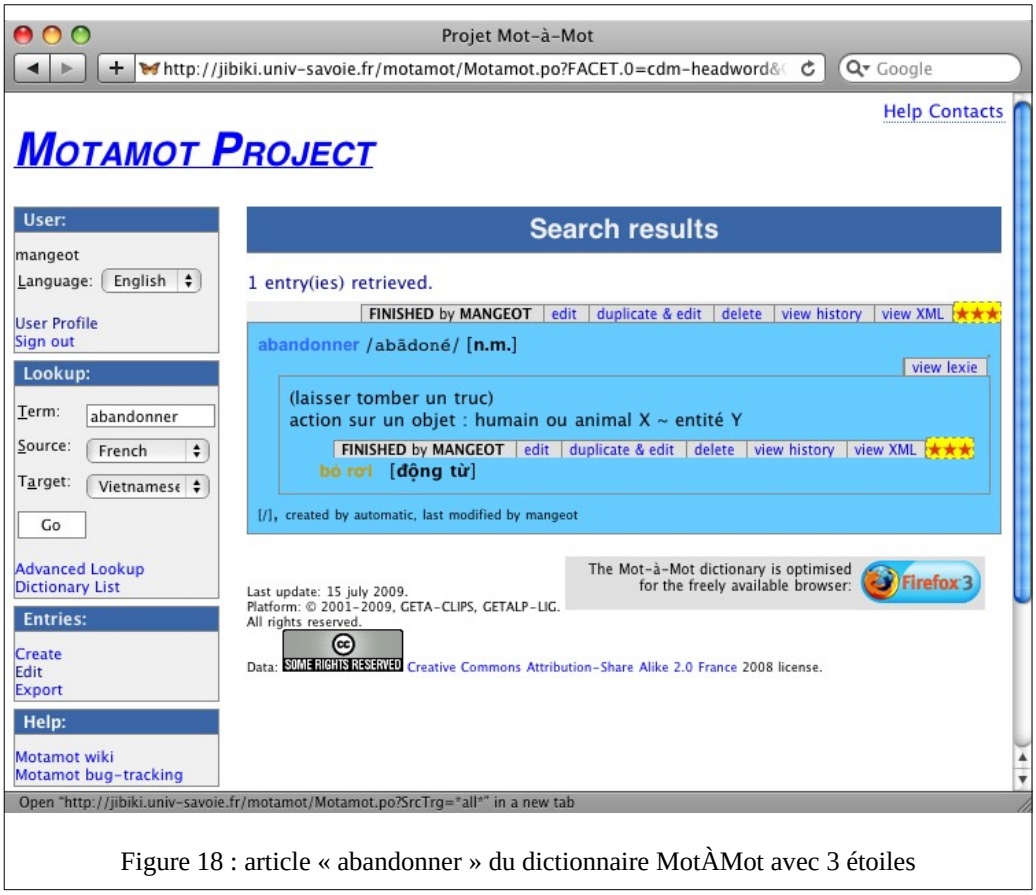


Figure 18 : article « abandonner » du dictionnaire MotÀMot avec 3 étoiles

De manière similaire, un niveau est assigné à chaque contributeur. Si un contributeur de niveau 3 révise un article de niveau inférieur, alors l'article obtient ensuite le niveau 3 (voir figure 18).

3.3.3 Résultats

Contributeurs du projet MotÀMot	**2009**	**2010**	**TOTAL**
Vincent Gros	350 h		350 h
Mathieu Mangeot	450 h	450 h	900 h
Hong-Thai Nguyen	80 h		80 h
Sereysethi Touch	50 h		50 h
Ieng Vanra		450 h	450 h
TOTAL	930 h	900 h	1830 h

Table 5 : estimation des heures passées par les principaux contributeurs sur le projet MotÀMot

Le dictionnaire MotÀMot est disponible en consultation sur le Web[29]. La version actuelle contient :

- 68 344 articles français dont 13 234 articles et 32 236 sens de mot reliés à des axies du volume pivot.
- 23 766 articles khmer tous reliés à des axies via 32 402 sens de mot.
- 32 402 axies reliant les sens de mot français et khmer.

29 https://motamot.imag.fr/

Ce projet m' a permis une fois de plus de valider l'utilité de la plateforme Jibiki. Avec Yeng Vanra, nous avons pu également mettre en valeur le dictionnaire construit par l'association « Pays perdu » (Mangeot, Hong-Thai Nguyen, 2009). Une perspective intéressante serait d'ajouter à cette base lexicale les données du dictionnaire français-lao numérisé et complété par Vincent Berment (Max Reinhorn, Berment, 2013).

Nous avons cependant rencontré deux écueils. Pour être juste, le système de notation des articles devrait pouvoir être étendu à chaque partie d'information. En effet, certains contributeurs peuvent par exemple corriger le khmer, d'autres auront une expertise sur les traductions françaises, etc. Mais la gestion d'un tel système est complexe. Il s'agit de trouver une bonne granularité d'information. De plus, il faut également afficher ces informations sans perturber l'utilisateur.

Une autre difficulté est apparue sur la capacité à motiver une communauté de contributeurs. Nous avons tenté de faire connaître le projet via des forums de la communauté khmérophone française mais sans vraiment de succès. Un tel projet de contribution ne peut réussir que s'il y a des animateurs prêts à s'investir ou si la contribution lexicale est un sous-produit d'une autre activité, comme l'apprentissage d'une langue ou de traduction ou de rédaction en langue seconde (L2).

Article associé au chapitre 3.3

> Mathieu Mangeot (2014) MotÀMot *project: conversion of a French-Khmer published dictionary for building a multilingual lexical system.* Proc. of LREC 2014, Reykjavik, Island, 28-30 May 2014, pp. 1024-1031. https://hal.archives-ouvertes.fr/hal-00994800

3.4 Base lexicale pour langues africaines : projets DiLAF et iBaatukaay

3.4.1 Présentation des projets

Le projet de Dictionnaires Langues Africaines-Français (DiLAF)[30] vise à informatiser des dictionnaires bilingues de langues nationales d'Afrique de l'Ouest et français. Ces dictionnaires existent en version électronique au format Word destiné à l'impression. Il s'agit dans un premier temps de les restructurer puis de les mettre à disposition en consultation sur le Web et en téléchargement gratuit au format XML sous licence Creative Commons puis, dans un deuxième temps, avec le projet iBaatukaay31, de les utiliser pour construire une base lexicale multilingue contributive à structure pivot.

Le défi sociétal concerne la reconnaissance des langues nationales des pays d'Afrique de l'Ouest qui sont des langues peu dotées (langues π) (Berment, 2004) et la lutte contre l'analphabétisme. La plupart des personnes sont locuteurs mais pas lettrés (Mangeot, Enguehard, 2011).

Les défis scientifiques sont d'une part de concevoir une méthodologie simple pour la récupération des données (Mangeot, Enguehard, 2013) et d'autre part d'expérimenter le passage à l'échelle de la macrostructure pivot (Khoule et al., 2016).

30 http://www.dilaf.org/

Le projet DiLAF est piloté par Chantal Enguehard, enseignante-chercheuse de l'Université de Nantes. Il a été financé à ses débuts de 2010 à 2013 par le Fonds Francophone des Inforoutes de l'Organisation Internationale de la Francophonie. Il regroupe des collègues linguistes du Burkina-Faso, du Mali et du Niger. Chantal et moi-même assurons la partie informatique (Enguehard et al., 2011).

Le projet iBaatukaay est piloté par Mouhamadou Khoulé dans le cadre de sa thèse en cours. Il vise principalement les langues nationales du Sénégal les plus utilisées telles que le wolof, le sérère, le pulaar, etc.

3.4.2 Structures de la base lexicale

La macrostructure des dictionnaires bilingues est toujours constituée d'un seul volume langue africaine → français. La macrostructure de la base iBaatukaay est calquée sur celle de Papillon-NADIA : un volume monolingue pour chaque langue et un volume pivot au centre. Les sens de mot de chaque article monolingue sont reliés aux axies du volume pivot. La microstructure des axies est constituée d'une liste de liens retour vers les lexies. Nous utilisons les traductions françaises pour proposer des axies entre langues africaines aux contributeurs qui devront les valider.

Les premiers dictionnaires bilingues traités proviennent du projet SouTeBa de soutien à l'éducation de base lancé par le ministère de l'éducation au Niger. Il s'agit de dictionnaires pour les écoles élémentaires qui s'ouvrent en langue nationale :

- dictionnaire haoussa-français de 7 823 articles, 2008, Soutéba,
- dictionnaire kanouri-français de 5 994 articles, 2004, Soutéba,
- dictionnaire soŋay zarma-français de 6 916 articles, 2007, Soutéba,
- dictionnaire tamajaq-français de 5 205 articles, 2007, Soutéba,
- dictionnaire fulfuldé-français de 4 305 articles, 2007, Soutéba

Ces dictionnaires ont sensiblement la même microstructure. Ce sont avant tout des dictionnaires monolingues avec une traduction en français (voir figure 19).

Figure 19 : article « boko » du dictionnaire DiLAF haoussa-français

Le dictionnaire bambara-français a été rédigé par le Père Charles Bailleul et publié en 1996. Il comporte plus de 10 000 articles. Sa microstructure est plus riche et détaillée et comporte des exemples traduits en français (voir figure 20). Il est destiné au lecteur francophone qui souhaite comprendre le bambara.

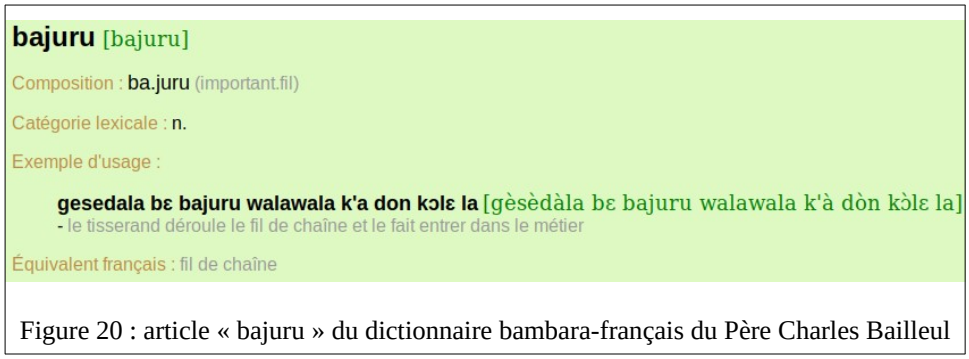


Figure 20 : article « bajuru » du dictionnaire bambara-français du Père Charles Bailleul

Récemment, un nouveau dictionnaire nouchi-français est venu enrichir la liste. Une première version a été publiée en 2016 par Jean-Baptiste Atsé N'cho.

Pour le projet iBaatukaay, nous avons également fusionné deux dictionnaires wolof-français. L'un provient d'une base de 8 167 entrées produite par un projet financé par l'AUF et dirigé par Thierno Cissé (Cissé et al., 2007). L''autre, le wolof fondamental contient 3 000 mots les plus fréquents et résulte d'un projet financé par la fondation Ford et dirigé par Chérif Mbodj du Centre de Linguistique Appliquée de Dakar, tous deux de l'Université Cheick Anta Diop au Sénégal.

La méthodologie de conversion des données depuis les documents Word vers des fichiers XML respectant le standard LMF est détaillée section 4.2.2. Au niveau de la microstructure, nous avons regroupé les vocables homographes en un seul article, nous avons créé des blocs sémantiques et ajouté un identifiant unique pour tous les articles.

La méthodologie de construction de la base iBaatukaay consiste dans un premier temps à transformer les données résultant du projet DiLAF vers les macro et microstructures du projet en réifiant les liens bilingues et en fusionnant les volumes monolingues de même langue (Khoulé et al., 2018). Ces opérations sont détaillées dans la section 4.3.3. Dans un deuxième temps, les étudiants en linguistique des universités sénégalaises seront sollicités pour contribuer en ligne sur la base lexicale (Khoule et al., 2016).

3.4.3 Résultats

Contributeurs du projet DiLAF	2010	2011	2012	TOTAL
Chantal Enguehard	450 h	450 h	450 h	1350 h
Soumana Kane	70 h	30 h	70 h	170 h
Modi Issouf	100 h	30 h	100 h	230 h
Mai Moussa Mai	70 h	30 h	70 h	170 h

Mathieu Mangeot	450 h	450 h	450 h	1350 h
Radji	70 h	30 h	70 h	170 h
Rakia	70 h	30 h	70 h	170 h
Malamine Sanogo	70 h	30 h	100 h	200 h
TOTAL	1350 h	1080 h	1380 h	3810 h

Table 6 : estimation des heures passées par les principaux contributeurs sur le projet DiLAF

Le site du projet DiLAF propose déjà gratuitement 7 dictionnaires en consultation et au téléchargement. Les données sont disponibles sous licence Creative Commons. Il constitue une référence dans le domaine.

Le projet iBaatukaay est en cours de réalisation. Il est prévu que Mouhamadou Khoulé, le responsable du projet, soutienne son doctorat prochainement à l'Université Gaston Berger à Saint-Louis du Sénégal. Il pourra ensuite monter un projet d'envergure avec le soutien de la communauté scientifique sénégalaise.

Article associé au chapitre 3.4

Chantal Enguehard, Mathieu Mangeot (2014) *Computerization of African languages-French dictionaries.* Proc. of Collaboration and Computing for Under Resourced Languages in the Linked Open Data Era (CCURL), LREC 2014 workshop, Reykjavik, Island, 27 May 2014, pp. 44-51. https://hal.archives-ouvertes.fr/hal-00994821

3.5 Dictionnaire japonais-français : Jibiki.fr

3.5.1 Présentation du projet Jibiki.fr

Le projet Jibiki.fr[31] est la suite directe du projet Papillon en ce qui concerne l'objectif premier de fournir une ressource lexicale bijective bilingue français-japonais à large couverture (Mangeot, 2014). J'ai lancé ce projet au départ seul en 2014, comme but de mon Congé de Recherche et de Conversion Thématique (CRCT), qui s'est déroulé à l'Université Hosei à Tokyo, financé par la bourse Hosei International Fellowship. Une fois le site Web contributif opérationnel, nous avons mis en place un comité de suivi du projet avec Mutsuko Tomokiyo, collègue de l'équipe GETALP et Nicolas Mollard, collègue actuellement enseignant-chercheur à l'Université Lyon III et spécialiste de la littérature japonaise du XIX^{ème} siècle.

Au début du projet, alors que j'étais à la recherche de ressources à réutiliser pour ce projet, Nicolas Mollard m'a orienté vers le dictionnaire japonais→ français du révérend père Gustave Cesselin (Cesselin, 1940) qui contient plus de 82 000 articles sur 2 365 pages. Il a trois avantages essentiels : sa bonne qualité associée à sa large couverture, et le fait qu'il est à présent libre de droits. Ses inconvénients résident dans le fait qu'il n'existe qu'en version imprimée et que son vocabulaire date

31 https://jibiki.fr/

d'avant la deuxième guerre mondiale, et donc avant l'occupation américaine du Japon, qui a vu un apport conséquent de nouveau vocabulaire provenant de l'anglais.

Malgré les risques d'une mauvaise qualité des résultats de lecture optique (surtout avec des sinogrammes de la première moitié du XX^{ème} siècle), et devant le fait que nous n'avions pratiquement aucune ressource japonais-français exploitable, nous avons tenté l'expérience de la récupération du Cesselin pour notre projet. Les objectifs de la ressource à construire se sont alors adaptés à la structure du Cesselin. Nous avons notamment repoussé à plus long terme le projet de constituer une base lexicale bijective entre les lexies japonaises et françaises pour nous concentrer sur l'élaboration d'un dictionnaire japonais→français. J'ai donc changé le nom du projet en Jibiki.fr.

Le défi sociétal principal du projet Jibiki.fr reprend celui du projet Papillon : il n'existait pas de dictionnaire français-japonais informatisé, de qualité et en accès libre. Un nouveau défi s'est ajouté, celui de la sauvegarde et de la valorisation du patrimoine existant en tant que dictionnaire imprimé de qualité.

Le défi scientifique est de construire une ressource lexicale de qualité et à large couverture à partir de ressources imprimées et électroniques existantes.

3.5.2 Structures de la base lexicale

La macrostructure du dictionnaire Cesselin est constituée d'un seul volume japonais→ français. Nous l'avons reprise dans un premier temps. Nous envisageons plus tard de la convertir vers une macrostructure pivot (voir section 6.1).

Figure 21 : article « zutto » du dictionnaire japonais-français Jibiki.fr

La microstructure des articles est assez riche. Chaque mot-vedette est d'abord transcrit en *romaji*[32] puis écrit en japonais. Nous avons également ajouté une prononciation en *hiragana*[33]. Le corps de l'article est constitué de blocs grammaticaux puis d'une suite d'exemples, expressions figées ou collocations. Chaque bloc grammatical contient une suite de blocs sémantiques décrivant chacun un sens de mot avec une ou plusieurs traductions en français (voir figure 21).

La méthode de construction comporte quatre étapes principales :

- Lecture optique (OCR) d'un exemplaire imprimé du dictionnaire Cesselin avec corrections automatiques et détection d'erreurs potentielles (voir section 4.1) ;
- Restructuration explicite des données (voir section 4.2.2) ;
- Ajout d'articles provenant des dictionnaires JMdict34 japonais-anglais (Breen, 2004) et des liens Wikipedia japonais-français ou japonais-anglais afin de compléter les données avec le vocabulaire récent tout en suivant la nomenclature du dictionnaire monolingue japonais Daijirin35 (Matsumura, 1995) afin de ne pas « polluer » les données avec des articles inutiles ;
- Mise en ligne pour consultation et correction.

3.5.3 Résultats

Contributeurs	**2014**	**2015**	**2016**	**2017**	**2018**	**TOTAL**
Mathieu Mangeot	900 h	900 h	100 h	100 h	100 h	2100 h
Nicolas Mollard	15 h	50 h	60 h	60 h	60 h	245 h
Mutsuko Tomokiyo		60 h	180 h	180 h	60 h	480 h
Autres contributeurs		10 h	10 h	10 h	10 h	40 h
TOTAL	915 h	1020 h	350 h h	350 h	230 h	2865 h

Table 7 : estimation des heures passées par les principaux contributeurs sur le projet Jibiki.fr

Le dictionnaire actuel contient 153 150 articles au total, dont :

- 82 033 articles provenant du Cesselin dont :
  - 7 737 articles (9,43 %) dont les mots-vedette n'ont pas encore été vérifiés manuellement; ex : tōbinan **東微南【とうびなん】**
  - 11 125 articles (13,56 %) dont les mots-vedette ont une prononciation incorrecte; ex : nomikarasu **飲涸す【のみからす】**
- 47 630 articles provenant du JMdict
- 23 486 articles provenant de Wikipedia
- 45 696 articles (29,84 %) dont les traductions sont en anglais (donc à traduire en français !)

Le site Web du projet a été lancé le 27 juillet 2016. En 3 ans, au total, 84 289 modifications d'articles ont été effectuées dont :

- 10 329 mots-vedette en *kanji*[34] ajoutés,

32 Transcription du japonais en alphabet romain : https://fr.wikipedia.org/wiki/Hanyu_pinyin
33 Syllabaire japonais : https://fr.wikipedia.org/wiki/Hiragana
34 https://fr.wikipedia.org/wiki/Kanji

- 5 185 mots-vedette en *kanji* vérifiés,
- 6 108 mots-vedette en *kanji* corrigés,
- 2 249 traductions en français.

Les 10 329 mots-vedette ajoutés l'ont été pour des articles dont la lecture optique avait échoué sur le mot-vedette en japonais. Ce travail a été effectué manuellement par 3 personnes et a duré 2 ans et demi de juillet 2015 à février 2018. D'autres corrections ont été effectuées, dont certaines manuellement (Tomokiyo et al., 2017), et d'autres automatiquement à l'aide d'un robot (Mangeot, Tomokiyo, 2018). À l'heure actuelle, le projet accueille 20 contributeurs dont la moitié est très active.

Le défaut majeur de la microstructure du dictionnaire Cesselin se situe au niveau des exemples. D'une part, ils sont listés à la fin de l'article sans être rattachés à un sens précis, et d'autre part, il n'est pas possible de distinguer un exemple d'une expression figée ou d'une collocation. Nous essayons de traiter ce problème en utilisant plusieurs heuristiques linguistiques et informatiques (Tomokiyo et al., 2018).

Ce projet montre que, malgré les erreurs provenant de la lecture optique, il est possible de créer un dictionnaire à partir d'un exemplaire imprimé et que celui-ci est tout à fait exploitable par des humains. Cela ouvre des perspectives intéressantes pour tous les dictionnaires imprimés de bonne qualité et surtout libres de droits.

Article associé au chapitre 3.5

Mathieu Mangeot (2016) *Collaborative Construction of a Good Quality, Broad Coverage and Copyright Free Japanese-French Dictionary.* International Journal of Lexicography, Volume 31, Issue 1, 1 March 2018, Pages 78–112; doi: 10.1093/ijl/ecw035; 35 p. https://hal.archives-ouvertes.fr/hal-01712271

4 Réutilisation de données existantes

Lors de la construction de nouvelles ressources, la réutilisation de données lexicales existantes est très utile pour construire des squelettes de dictionnaires. En effet, d'expérience, nous avons observé qu'il est bien plus facile pour un contributeur de corriger des données erronées que de créer de nouvelles données à partir de zéro.

En particulier, la nomenclature contenant la liste des articles est un élément central du dictionnaire. Elle est à fixer si possible en premier lieu et pour cela, la réutilisation de ressources existantes est très utile.

Nous distinguons la récupération de données, qui consiste à convertir des données d'un format autre que XML (fichiers CSV, traitement de texte, exemplaires imprimés) vers XML, et la transformation de données, qui convertit des données XML d'une structure vers une autre.

Il est à noter que, bien que les formats récents de traitement de texte comme OpenDocument[35] ou Office Open XML[36] utilisent en interne XML pour stocker les documents, nous considérons que les ressources lexicales créées avec des outils de traitement de texte utilisant ce format ne sont pas à proprement parler des ressources au format XML car elles n'ont pas été conçues pour cela.

Dans cette partie, nous suivrons une progression logique : récupération de données issues de lecture optique, récupération de données issues de traitement de texte et transformation de données XML d'une structure vers une autre.

4.1 Récupération de données issues de lecture optique

Malgré de récents progrès, la lecture optique (en anglais OCR pour Optical Character Recognition) n'est pas exempte d'erreurs et ne le sera probablement jamais, d'autant plus pour un dictionnaire qui alterne les langues. Il faut donc n'employer cette technique qu'en dernier recours, lorsqu'aucune ressource électronique (dans n'importe quel format) n'a été trouvée.

Nous avons utilisé cette méthode sur 2 ressources différentes : le dictionnaire japonais→français Cesselin (Cesselin, 1940) avec 2 340 pages et le dictionnaire français→japonais Raguet-Martin (Raguet, Martin, 1953) avec 1 467 pages.

Le défi sociétal de cette section est la sauvegarde et la valorisation du patrimoine dictionnairique existant sous forme imprimée.

Le défi scientifique est de récupérer des données lexicales issues de lecture optique en minimisant le nombre d'erreurs de façon qu'un humain puisse lire le résultat.

35 https://fr.wikipedia.org/wiki/OpenDocument
36 https://fr.wikipedia.org/wiki/Office_Open_XML

4.1.1 Remarques générales sur la lecture optique

Le résultat de lecture optique dépend en premier lieu de la qualité des données reçues en entrée. Nous avons testé plusieurs solution pour scanner un livre. Le scanner à plat introduit une bande noire au milieu de la page et le scanner avec caméra sur le dessus évite ce problème mais conserve une courbure. La meilleure solution que j'ai utilisée pour numériser le Cesselin consiste à sacrifier un exemplaire en coupant sa reliure. Les pages sont ensuite scannées automatiquement une par une.

Le choix du logiciel de lecture optique est également crucial. Les techniques évoluant rapidement et les besoins étant différents, nous conseillons de tester une dizaine de logiciels avant de se lancer. Les logiciels payant permettent très souvent de tester quelques pages avant de payer une licence.

Les logiciels permettant d'entraîner la lecture optique sur une partie du texte sont très intéressants. Ils permettent de s'adapter à n'importe quelle police de caractères y compris une écriture manuelle. Dans le cas du projet Jibiki.fr, nous avions besoin de lire des sinogrammes. Le logiciel que nous avions retenu, *Abbyy FineReader*, ne permettait malheureusement pas un entraînement pour les sinogrammes. Même si celui-ci était capable de reconnaître du texte avec plusieurs langues (ici le français et le japonais), nous avons choisi d'effectuer deux passes : une passe réglée sur le français avec entraînement (mais celle-ci ne reconnaissait pas les sinogrammes) et une passe réglée sur les sinogrammes (qui comportait des erreurs principalement sur les diacritiques du français). Nous avons ensuite programmé un script qui fusionne le résultat de ces deux passes.

Une autre fonctionnalité intéressante de certains logiciels de lecture optique est de garder la trace de l'endroit où chaque mot a été lu. Cela permet de mieux localiser les erreurs sur la version PDF de la page pour les corrections ultérieures. Nous n'avons malheureusement pas pu bénéficier de cet avantage pour l'instant.

4.1.2 Localisation des mots-vedettes

Un dictionnaire est un objet très structuré dont l'objet de base est l'article. La première tâche lors de la structuration des informations après lecture optique est la détection des articles. Celle-ci se traduit par la localisation des mots-vedette dans la page, qui constituent les bornes inférieures et supérieures des articles.

La première étape consiste à lister pour chaque page l'intervalle des mots-vedettes s'y trouvant. Dans beaucoup de dictionnaires, le premier et le dernier mot-vedette de la page sont notés dans l'en-tête de la page. Nous avons donc extrait ces listes puis nous les avons contrôlées manuellement lorsque l'ordre alphabétique n'était pas respecté.

La deuxième étape consiste à comparer chaque début de ligne suivant un ordre alphabétique strict avec les vedettes de l'en-tête, puis entre les mots-vedettes détectés : vedette en-tête 1 < vedette détectée 1 < vedette détectée 2 < vedette détectée 3, etc < vedette en-tête 2. Dans la figure 22, les mots-vedettes suivants sont extraits : haichai, haichi, haichi, haichüritsu, haidan.

La troisième étape consiste à effectuer une comparaison approximative de chaque début de ligne en utilisant l'algorithme de Wagner et Fischer (Wagner, Fischer, 1974) pour calculer la distance de Levenshtein, ce qui permet de prendre en compte les erreurs de lecture optique. Dans la figure 22, les mots-vedette « iichi » and « haicbutsu » sont extraits et automatiquement corrigés pour devenir « haichi » and « haichutsu ».

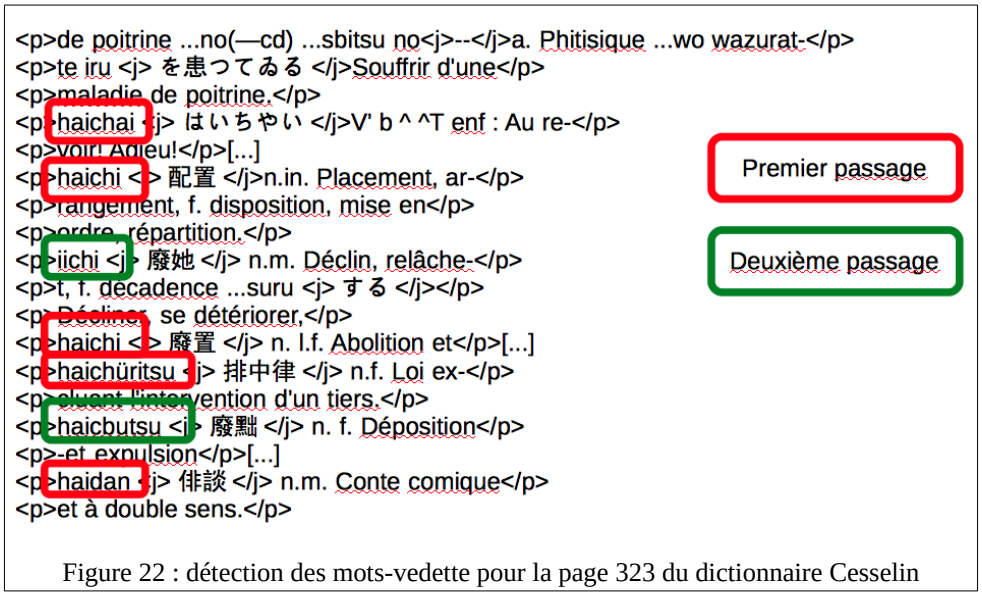


Figure 22 : détection des mots-vedette pour la page 323 du dictionnaire Cesselin

Il reste un problème de sur-détection lorsque certaines parties d'article (locutions, exemples) commencent au début d'une ligne avec un mot très proche du mot-vedette. Dans notre cas, cela concernait principalement des formes conjuguées de verbes. Par exemple, sur la page 1 038 du Cesselin, « kuwadate iru » a été détecté comme vedette alors que c'est un exemple de l'article « kuwadateru », et « Kuwashiku monoshiberu » a été détecté comme vedette alors que c'est une locution de l'article « kuwashii ».

Suite à cette approche, nous avons constaté en moyenne 1 mot-vedette non détecté par page de 30 vedettes, soit 3,33 % de vedettes non détectées.

4.1.3 Corrections et détections d'erreurs potentielles

Suite à la lecture optique, il est possible de corriger certaines erreurs en fonction des systèmes d'écriture utilisés. Nous donnons ici les corrections effectuées sur des dictionnaires japonais-français.

Pour le français, celles-ci se concentreront sur les diacritiques : Â + ' ' ⇒ À; Etre ⇒ Être; ç[^àaou] ⇒ c; etc.

Pour le *romaji*, les corrections sont faciles puisqu'il n'existe que les macrons comme diacritiques : ā,ī,ū,ē,ō. Les autres diacritiques sont automatiquement convertis : à ⇒ a; äâ ⇒ ā, etc. Certaines combinaisons de lettres n'existent pas en romaji. Elles sont automatiquement converties : lt + [aiueo] ⇒ h + [aiueo]; rn + [aiueo] ⇒ m +[aiueo], etc.

Pour le japonais, les caractères latins ne font pas partie des caractères utilisés (sinogrammes, kanas, ponctuation). Si l'un d'eux est trouvé dans une chaîne, l'erreur est marquées pour correction future.

Nous avons également rencontré des erreurs fréquentes à la frontière entre deux chaînes de caractères de systèmes d'écriture différents. Il semble que le logiciel de lecture optique ait des difficultés à détecter l'emplacement exact de ces changements. Nous avons effectué les corrections ce-dessous pour les fins de segment de japonais (marqué <j>) :

9 ⇒ り ; ｜c ⇒ に; = ⇒ニ

v') ⇒ い　Ex : <j>と忙はし</j>v') ⇒ <j>と忙はしい</j>)

't) ⇒ す　Ex : <j>心を越</j>'t)⇒ <j>心を越す</j>)

Pour les erreurs restantes, nous avons d'abord pensé à utiliser un correcteur orthographique, mais il les corrections étaient souvent erronées du fait des hapax[37] que l'on trouve dans un dictionnaire et qui ne sont pas inclus dans le correcteur orthographique. Nous avons finalement opté pour l'utilisation d'un lemmatiseur pour le français : TreeTagger[38]. Celui-ci couvre plus de 20 langues. Si le mot n'est pas détecté par le lemmatiseur, il est surligné en jaune pour correction future. Il se peut que le mot surligné soit malgré tout correct mais que la forme analysée soit inconnue du lemmatiseur (c'est le cas du mot « Veuillez » de la figure 21). Dans ce cas, la correction est immédiate. L'utilisateur clique sur le segment en jaune et le valide (voir figure 11).

Pour le japonais, la notion de faute d'orthographe n'existe pas en tant que telle. Pour détecter les erreurs potentielles, nous avons analysé chaque segment japonais à l'aide du lemmatiseur Mecab[39] afin de le transcrire en *hiragana*. Ensuite, nous avons également transcrit le segment *romaji* en *hiragana* puis nous avons comparé les deux. Si une différence est détectée, la partie de segment concernée est surlignée en orange. Les kanjis ayant plusieurs prononciations possibles, certaines différences ne sont pas des erreurs (voir « kita no hō ni » figure 21).

Les étapes suivantes sont consacrées à la structuration des articles. Elles sont décrites dans la section suivante.

Article associé au chapitre 4.1

Mathieu Mangeot (2016) *Collaborative Construction of a Good Quality, Broad Coverage and Copyright Free Japanese-French Dictionary.* International Journal of Lexicography, Volume 31, Issue 1, 1 March 2018, Pages 78–112; doi: 10.1093/ijl/ecw035; 35 p. https://hal.archives-ouvertes.fr/hal-01712271

4.2 Récupération de fichiers issus de traitement de texte

Dans cette section, nous abordons deux techniques de récupération de données lexicales issues de logiciels de traitement de texte ayant servi à rédiger puis imprimer des dictionnaires éditoriaux. Ces

37 https://fr.wikipedia.org/wiki/Hapax
38 https://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/
39 https://taku910.github.io/mecab/

techniques servent également pour convertir des données depuis des formats différents vers XML. Cependant, depuis l'invention de XML, les autres formats se raréfient.

Le défi sociétal est de mettre au point une méthodologie simple de récupération de données lexicales pour que des linguistes non-informaticiens puissent la mettre en œuvre.

Le défi scientifique est de trouver comment convertir les données d'un format vers un autre sans perdre d'information.

4.2.1 Outil de récupération automatique : RÉCUPDIC

Dans sa thèse (Doan-Nguyen, 1998), Hai Doan-Nguyen a élaboré un outil générique de récupération de données lexicales : RÉCUPDIC. L'utilisateur définit d'abord la grammaire du format de la ressource qu'il souhaite récupérer (voir figure 23 pour la grammaire et 24 pour la ressource) puis l'outil se charge du reste.

```
#grammar babel-glossary /* Acquisition du glossaire BABEL de I. Kind */

#syntax-rules   :1:   babel-entry(;entry)   ->   >hwd(;hwd)   body(;body)   --
(#!babel((trim-whites hwd) body); entry).
:2: body(;body) -> sense(;S1) sense*(;S*) -- cons(S1 S*; body).
:3:  sense(;S)->  >exps(;exps)  expl?(;expl)  subj?(;subj)  --  (#!sense((trim-
whites exps) (if expl (trim-whites expl)) (if subj (trim-whites subj))); S).
:4: expl?(;expl) -> "(" >to-cparen(;expl) ")".
:5: expl?(;expl) -> §(nil;expl).
:6: subj?(;subj) -> "[" >to-cbrak(;subj) "]".
:7: subj?(;subj) -> §(nil; subj).
:8: sense*(;S*) -> "+" sense(;S1) sense*(;S*1) -- cons(S1 S*1; S*).
:9: sense*(;S*) -> §(nil; S*).

#start-symbol babel-entry /*symbole de départ de la grammaire*/

#lexical-rules hwd -> _¨10. /* Headword prend 10 caractères */
exps -> ![+([]*. to-cparen -> >()]. to-cbrak -> >[\]].

#lexical-order ("+" "(" ")" "[" "]" hwd exps expl subj)

#working-code (sia-defclass babel () (hwd body))
/* classe définitions */
(sia-defclass sense () (exps expl subj))
(defun trim-whites (string) (string-trim '(#\Space #\Tab #\Newline) string))
```

Figure 23 : grammaire de récupération du lexique BABEL pour l'outil RÉCUPDIC

```
.COM    Command (file name extension) +
        Commercial Business (Domain Name) [Internet]
```

Figure 24 : article « .COM » du lexique BABEL avant récupération

L'outil est puissant. Il a servi à récupérer plus de 1 650 000 articles. Mais on aperçoit rapidement ses limites. En effet, cet outil fonctionne bien lorsque la ressource à récupérer est déjà relativement bien structurée et normalisée (ex : pas de blanc en gras dans une suite de mots non gras) (voir figure

24). Par contre, les données issues de traitement de texte contiennent souvent des exceptions dues au fait que le rédacteur est libre d'écrire ce qu'il veut. Il faut alors soit écrire une règle de grammaire pour chaque exception soit effectuer une passe de normalisation manuelle sur tout le dictionnaire soit corriger à la main les articles refusés et les réanalyser avec l'outil.

L'autre inconvénient est qu'il faut une bonne connaissance de la programmation en LISP[40] pour écrire une grammaire et faire fonctionner l'outil. Il ne peut pas être utilisé par un linguiste sans une solide formation en programmation. C'est pour ces deux raisons que nous avons finalement mis au point une méthodologie plus simple pour nos projets ultérieurs.

4.2.2 Méthodologie de récupération fondée sur des expressions régulières

Nous avons raffiné cette méthodologie au fil de nos projets de création de dictionnaires qui nécessitaient la récupération de ressources existantes. C'est lors du projet DiLAF (voir 3.4.1) que nous l'avons formalisée[41] (Enguehard, Mangeot, 2014). Travaillant avec des collègues linguistes africains, il était important de concevoir une méthodologie simple à mettre en œuvre, n'utilisant que des outils gratuits et en source ouverte.

Nous avons appliqué cette méthodologie à plus d'une vingtaine de ressources dont celles des projets MotÀMot, DiLAF et Jibiki.fr.

La ressource à récupérer doit être convertie depuis son format d'origine (*Original format* de la figure 25) au format OpenDocument[35] ou Office Open XML[36] avec un éditeur de la famille *OpenOffice*. Il faut ensuite extraire le fichier de données qui est déjà au format XML. Il va falloir ensuite le structurer.

Tout d'abord, nous utilisons une feuille de style XSLT pour simplifier le balisage du format *OpenXML*. Celui-ci indique les différents styles utilisés en attributs. Nous remplaçons le nom de chaque balise par le nom du style porté en attribut :

```
<office:document-content[...]><office:body><office:text><text:p text:style-
name="P1"><text:span text:style-name="T1">abalone</text:span>
```

```
→ <volume><P1><T1>abalone</T1>
```

Cela rend le XML plus lisible pour la suite des opérations.

40 https://fr.wikipedia.org/wiki/Lisp
41 http://pagesperso.ls2n.fr/~enguehard-c/DiLAF/dicos/DiLAF_methodologie.zip

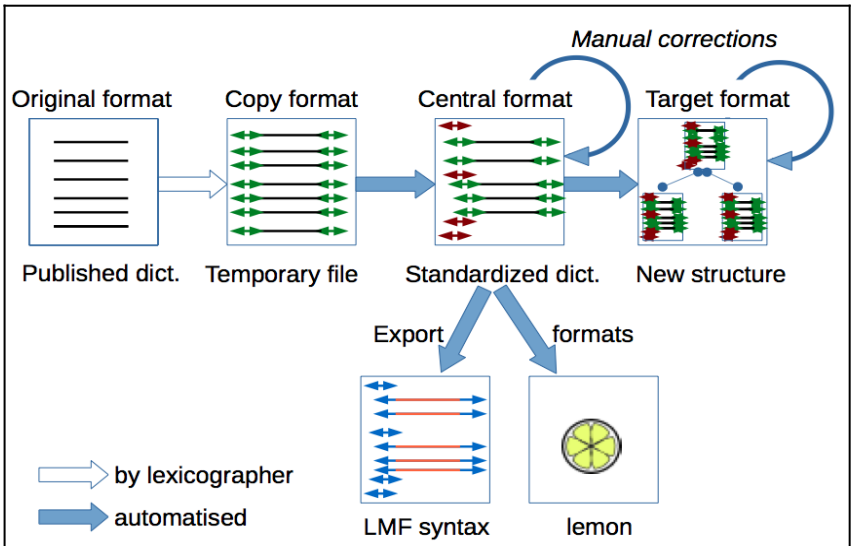


Figure 25 : processus de conversion d'un dictionnaire éditorial

La deuxième étape consiste à marquer chaque partie d'information telle que le mot-vedette, la prononciation, la catégorie grammaticale, la définition, les traductions, les exemples, etc. à l'aide d'expressions régulières en s'appuyant sur les informations de style disponibles (gras, italique, saut à la ligne, etc.). Sur la figure 26, les balises en vert ont été ajoutées lors de cette étape. Un éditeur de texte basique tel que *gedit* ou *NotePad++* peut faire l'affaire. Nous obtenons le format copie (Copy format de la figure 25).

```
<sanniize>abiyanso</sanniize>
<ciiyaŋ>[àbìyànsôo]</ciiyaŋ>
<kanandi>m.</kanandi>
<bareyaŋ>aéroport<bareyaŋ>
<feeriji>batama kaŋ ra abiyey ga zumbu.</feeriji>
<silmaŋ>Tilbeeri nda Dooso sinda abiyanso kaŋ ra abiyo beeri ga zumbu</silmaŋ>
<f>abiyansa</f><b>abiyansey</b>
```

Figure 26 : article « abiyanso » du dictionnaire DiLAF zarma-français au format copie

La troisième étape consiste à expliciter la structure des articles, toujours à l'aide d'expressions régulières. Nous conseillons de suivre la structuration du noyau LMF (voir 1.3.2) avec pour chaque article une partie décrivant la forme du mot-vedette (transcriptions, prononciation, etc.) et une partie sémantique constituée d'une liste de sens de mot (Enguehard, Mangeot, 2013). Sur la figure 27, les balises en rouge ont été rajoutées lors de cette étape. Dans cette étape, il est également conseillé de vérifier les valeurs des listes fermées (la catégorie grammaticale, le domaine, etc.) car de nombreuses erreurs peuvent s'y trouver. Pour cela, il est possible d'utiliser la signature dictionnairique décrite au 2.1.1. Nous obtenons le format central (*Central format* de la figure 25).

Figure 27 : article « abiyanso » du dictionnaire DiLAF zarma-français au format central

Chaque étape nécessite plusieurs remplacements à l'aide d'expressions rationnelles. Il s'agira de conserver chaque version de fichier afin de faciliter les retours en arrière en cas d'erreur. Il faut également vérifier que chaque version soit bien formée du point de vue XML avec un parseur. Cette fonctionnalité existe dans la plupart des navigateurs web. À la fin de la structuration, il est possible de définir une Définition de Type de Document (DTD) puis vérifier la validité du fichier XML produit.

Une fois la ressource convertie au format central, il est possible de l'enrichir via des contributions manuelles à l'aide de la plateforme Jibiki. Il est également possible de l'exporter dans d'autres formats tels que la syntaxe LMF (voir figure 3).

Au sein d'un projet de construction d'une nouvelle ressource, la prochaine étape consiste à transformer la ressource au format central vers le format cible du projet. Cette transformation peut être automatisée. C'est l'objet de la section suivante.

Article associé au chapitre 4.2

Mathieu Mangeot & Chantal Enguehard (2018) *Dictionaries for Under-Resourced Languages: from Published Files to Standardized Resources Available on the Web.* Rapport de recherche LIG, 22 juillet 2018, 38 p. https://hal.archives-ouvertes.fr/hal-02056905

4.3 Réutilisation de données XML

Lorsque l'on gère des ressources lexicales, il est très fréquent de devoir transformer une structure dans une autre, soit pour réutiliser une ressource existante dans un nouveau projet, soit pour faire évoluer une ressource.

Le défi sociétal de cette partie est de proposer un outil de transformation accessible à des non-informaticiens.

Le défi scientifique de cette partie est d'automatiser la transformation de ressources d'une structure vers une autre.

4.3.1 Outil de transformation automatique : PRODUCDIC

L'outil PRODUCDIC a également été élaboré par Hai Doan-Nguyen pendant sa thèse (Doan-Nguyen, 1998). Même si à l'époque, le format utilisé n'était pas XML mais fondé sur LISP, il m'a paru intéressant de le mentionner, car il a inspiré mes recherches.

Cet outil ensembliste du type langage de programmation est construit à partir de fonctions LISP qui permettent des manipulations sur les structures profondes récupérées. Il permet 7 types d'opération sur les dictionnaires :

1. La sélection d'un sous-ensemble B des éléments de l'ensemble A qui satisfont un prédicat P ;
2. L'extraction qui permet de créer un objet à partir d'une donnée a $\in$ A quelconque ;
3. Le regroupement qui partitionne un ensemble selon certains critères et transforme chaque sous-ensemble en un objet ;
4. L'inversion qui se compose de deux étapes : le regroupement et la division.
5. L'enchaînement est utilisé pour composer un dictionnaire *langue A-langue C* à partir de deux dictionnaires *langue A-langue B* et *langue B-langue C*.
6. La combinaison parallèle. Pour combiner en parallèle deux dictionnaires Dict1 et Dict2 pour obtenir Dict3, on passe par deux étapes : création des articles de Dict3 à partir de Dict1, puis intégration des articles de Dict2 à Dict3.
7. La combinaison en étoile est une généralisation de l'enchaînement et de la combinaison parallèle Elle peut être implémentée avec les opérations présentées précédemment.

Environ 543 000 articles ont été produits avec PRODUCDIC. L'outil a les mêmes défauts que RÉCUPDIC : il nécessite des connaissances de programmation en LISP et n'a pas été prévu pour traiter des structures XML. D'autre part, il ne peut transformer que des dictionnaires de macrostructures et microstructures relativement simples. La question de la correspondance des catégories de données (par exemples la fusion de deux dictionnaires dont la liste de catégories grammaticales est différente) n'est pas réglée. C'est pourquoi, malgré l'existence de cet outil, et à cause de l'évolution de nos besoins vers des structures XML, nous avons dû pour nos premiers projets convertir des structures manuellement.

4.3.2 Conversion de structures *ad hoc*

Pour la conversion de structures XML, nous utilisons la plupart du temps des feuilles de style XSLT[10]. Ce langage de transformation d'arbres est verbeux puisqu'il faut également utiliser XML pour écrire les transformations. Par contre, il oblige à transformer systématiquement tout l'arbre XML. D'autre part, le résultat est obligatoirement du XML bien formé.

Lors du projet GDEF, pour l'estonien, nous avons transformé le grand dictionnaire estonien-russe (EVS)[42] (Eesti Keele Instituut, 1997) en supprimant le russe et en l'adaptant à la microstructure du GDEF. Pour le français, nous avons repris la base Morphalou (Romary et al., 2004) en ne gardant que les mots-vedettes et en transformant la microstructure vers celle du GDEF.

Lors du projet MotÀMot, après avoir converti la ressource depuis Word vers le format central, nous avons séparé le volume français-khmer en un volume français et un volume khmer. Ensuite, nous avons réifié les liens bilingues pour obtenir un volume pivot. Nous avons également repris la prononciation et les mots-vedettes supplémentaires provenant du dictionnaire FeM (Gut et al., 1996). Il est à noter que la réification des liens bilingues ne peut se faire avec une seule feuille de style XSLT car il n'est pas possible d'obtenir plusieurs fichiers de sortie. Nous avons donc programmé un script Perl[43] pour ces opérations.

Lors du projet Jibiki.fr, après avoir obtenu le format central du projet suite à la lecture optique, nous avons ajouté des articles du dictionnaire JMdict (Breen, 2004) et Wikipedia en suivant la nomenclature du dictionnaire monolingue japonais Daijirin (Matsumura, 1995).

4.3.3 Transformation automatisée de structures

Pour tous les projets précédents iBaatukaay, nous avons programmé des scripts ad hoc pour transformer les structures XML nécessaires. Le projet iBaatukaay demande un certain nombre d'opérations de transformation pour construire une base lexicale multilingue à partir de dictionnaires bilingues. Au début du projet, nous avions déjà 5 dictionnaires à convertir. Il devenait donc urgent de travailler sur un outil générique de transformation automatisée de structures.

Le cahier des charges était simple : il fallait pouvoir paramétrer les transformations uniquement en se servant des fichiers produits par iPolex pour chaque ressource et notamment les pointeurs CDM.

Dans le cadre de la thèse de Mouhamadou Khoulé, nous avons donc élaboré un outil de transformation de structures intégré à iPoLex. Il se compose de 4 opérations principales, chacune étant implantée à l'aide d'un script Perl. Les interfaces sont réalisées en PHP et donc accessibles depuis un navigateur web. Tous les paramètres d'entrée des scripts existent déjà pour les ressources lexicales importées dans iPoLex.

La **fusion interne** fusionne les articles ayant le même mot-vedette et, de manière optionnelle, la même catégorie grammaticale. Les sens de mot du deuxième article sont fusionnés avec ceux du premier. Les sens de mot sont également renumérotés. Les paramètres d'entrée sont le fichier de métadonnées contenant les pointeurs CDM et le fichier de données du volume à fusionner.

La **transformation** transforme les articles organisés selon la microstructure d'un volume de départ en articles organisés selon la microstructure d'un volume d'arrivée. Les paramètres d'entrée sont le fichier de métadonnées et le fichier de données du volume de départ, ainsi que le fichier de métadonnées et l'article modèle du volume d'arrivée. Chaque information identifiée par un pointeur

42 https://www.eki.ee/dict/evs/
43 https://fr.wikipedia.org/wiki/Perl_(langage)

CDM dans l'article de départ est transformée en information à l'aide du pointeur CDM du volume d'arrivée. Si une donnée de l'article de départ n'est pas décrite par un pointeur CDM, elle est recopiée au même niveau dans la structure d'arrivée. L'article de départ est finalement recopié dans son intégralité à la fin de l'article d'arrivée, ce qui permet de ne pas perdre d'information. Il est possible ensuite de retravailler certaines informations pour les intégrer dans l'article d'arrivée. Pour les catégories de données telles que les catégories grammaticales ou les domaines, la correspondance entre les listes du volume de départ et d'arrivée est spécifiée sous forme de tableau JSON dans le fichiers de métadonnées du volume d'arrivée.

La **fusion externe**, soit la fusion de deux volumes A et B, est réalisée ensuite en effectuant d'abord une transformation du volume A vers la microstructure du volume B, puis une fusion du volume produit avec le volume B, et enfin une fusion interne du volume B enrichi avec les articles du volume A.

La **création de liens** d'un volume bilingue mono-volume langue A → langue B génère deux volumes monolingues langue A et langue B reliés entre eux par des liens bidirectionnels. Ces liens peuvent être des liens de traduction mais pas nécessairement. Les paramètres d'entrée sont les fichiers de métadonnées, les articles modèles des deux volumes A et B ainsi que le fichier de données du volume A.

La **réification** des liens existant entre deux volumes monolingues langue A et langue B consiste à créer un troisième volume C qui contiendra les liens entre les volume A et C, puis entre les volumes C et B. Comme pour la création de liens, ces liens peuvent être de traduction ou non. Cette opération est utilisée par exemple lors de la création d'un volume pivot. Les paramètres d'entrée sont les fichiers de métadonnées, les articles modèles des trois volumes A, B et C ainsi que le fichier de données des volumes A et B.

La **sélection** de certains articles correspondant à des critères précis est implantée dans la plateforme Jibiki. Un administrateur de dictionnaire peut exporter dans un fichier XML une requête qu'il a effectuée avec l'interface de recherche multicritères (voir la figure 6).

Pour le projet iBaatukaay (Khoulé et al., 2018) , nous avons réalisé avec succès :

- la transformation de 5 dictionnaires mono-volumes : 2 dictionnaires wolof-français, 1 dictionnaire bambara-français, 2 dictionnaires fulfulde-français ;
- la fusion des 2 dictionnaires wolof et des 2 dictionnaires fulfulde ;
- la création de liens de traduction langue nationale → français, avec 2 volumes séparés pour chacune des langues bambara, wolof et fulfulde ;
- la réification de tous les liens bilingues en créant un seul volume pivot regroupant tous les liens.

5 Utilisation de bases lexicales pour l'outillage de l'accès aux textes.

Jusqu'à présent, je me suis essentiellement concentré sur les utilisateurs humains des bases lexicales. Mais l'avantage de ces bases réside dans le fait qu'elles sont informatisées. Elles peuvent donc être utilisées par d'autres applications et faire partie d'écosystèmes plus vastes.

C'est à partir de 2015, suite au projet Jibiki.fr lors duquel nous avons produit une quantité de données suffisante pour être utilisée avec intérêt dans d'autres contextes, que je me suis intéressé à l'interaction entre les bases lexicales et d'autres applications avec deux thèmes principaux que j'aborde ici.

5.1 Exploitation de bases lexicales dans un environnement d'apprentissage des langues

Le dictionnaire est un outil incontournable de l'apprentissage des langues. L'intérêt de pouvoir utiliser directement un outil de gestion de bases lexicales dans un environnement d'apprentissage des langues en ligne est évident. Nous l'avons observé notamment lors du projet IDEFI Innovalangues initié en 2012 et auquel nous avons participé en 2015.

Le défi sociétal de cette partie est d'améliorer l'apprentissage du lexique d'une nouvelle langue en proposant aux utilisateurs des outils innovants et faciles d'accès.

Le défi scientifique est d'intégrer un environnement de bases lexicales dans un environnement d'apprentissage des langues en gardant l'avantage principal de l'outil, à savoir sa généricité.

5.1.1 Description de la problématique

L'un des principaux objectifs du projet Innovalangues est de mettre à disposition un Environnement Numérique Personnalisé d'Apprentissage des langues (ENPA). Outre les parcours d'apprentissage, l'écosystème numérique proposé par Innovalangues vise à fournir aux apprenants des outils pour soutenir l'apprentissage : des jeux, des générateurs d'exercices, un tchat, un test de positionnement, etc.

Le projet est divisé en plusieurs lots dont l'objectif est de produire des outils répondant à des problématiques didactiques et liés ou intégrés à l'ENPA.

L'une des tâches fondamentales de l'apprentissage des langues est de travailler le lexique. Dès lors, il est rapidement apparu que la réalisation des divers outils conduisait au développement d'autant de bases lexicales avec des contraintes spécifiques pour chacune, sans qu'il n'y ait de lien pratique entre eux. De plus, les apprenants peuvent souhaiter également créer leurs propres lexiques d'apprentissage. Les attentes en termes d'outillage du lexique s'expriment différemment : pour certains, les besoins principaux portent sur le contenu informationnel, alors que d'autres utilisateurs potentiels sont plus tournés vers les fonctionnalités offertes par le système.

Cette situation a rapidement posé un problème de mise en œuvre informatique (opérationnalisation), ce qui a mené à la définition d'un sous projet transversal au projet Innovalangues (« chantier interlot »), appelé LexInnova par ses fondateurs : Yoann Goudin, responsable du lot SELF pour le mandarin, Mathieu Loiseau, responsable du lot GAMER (Loiseau et al., 2015) et Emmanuelle Eggers, enseignante d'espagnol. L'équipe GETALP a été contactée en 2014 pour notre expertise sur le sujet. Valérie Bellynck, Ying Zhang et moi-même avons alors rejoint le groupe de travail LexInnova. À mon retour du Japon en 2015, j'ai été financé par le projet Innovalangues pour produire un prototype intégrable à l'ENPA.

5.1.2 Cahier des charges de l'outil souhaité

Pour définir précisément les besoins, nous avons d'abord mis au point des scénarios utilisateurs. Celui-ci est conçu par rapport au cours d'espagnol d'Emmanuelle Eggers. Le voici.

> Anne étudie l'espagnol débutant depuis deux mois. Son enseignant a choisi de faire créer un lexique de groupe à tour de rôle par des étudiants en tandem. Pour ce faire, cette semaine, Anne et son partenaire travaillent sur l'ENPA. Ils relisent les notes prises durant le cours à la lumière des ressources proposées par l'enseignant et choisissent les mots qu'ils veulent inclure dans le lexique de groupe. Dans certains cas, ils ajoutent à la notice de l'entrée un commentaire qui a été fait pendant la classe (ex : « Attention, en espagnol, 'padres' signifie non seulement 'pères', mais aussi 'parents', ex : "Llamo a mis padres cada semana." ; (voir aussi hermanos, tíos) » dans l'article 'padre').
>
> Ils profitent également de ce travail à destination du groupe pour mettre à jour leur propre lexique et cliquent sur les mots qu'ils souhaitent rajouter dans leur lexique respectif (lexique personnel qui alimente le lexique de groupe et vice-versa). Anne souhaite par exemple s'approprier le mot « tío » (oncle). Elle clique sur ce mot, un menu déroulant s'ouvre. Elle peut alors consulter les définitions ainsi que les exemples correspondant à ce mot, puis importer ces données telles quelles ou en les modifiant. Anne décide par exemple de modifier l'exemple associé à « tío » pour son lexique personnel ; son oncle s'appelant Bernard, pour retenir « tío », elle va donc choisir de mettre comme exemple personnel « Mi tío se llama Bernard » (besoin fonctionnel) (voir figure 30). Anne, pour réviser le lexique de groupe et son lexique personnel, peut lancer des générateurs d'exercices (lien avec l'ENPA).
>
> Eunice, bilingue français-portugais, est inscrite dans le même cours. Elle va consulter les mots nouvellement arrivés dans le lexique de groupe et faire des exercices à partir de ces mots (besoin fonctionnel). Elle choisira de rajouter une information spécifique dans son lexique personnel « Attention : accent sur le "i" de "tío" » (contrairement au portugais).

Ce scenario met en relief des statuts différents de certaines données de la base lexicale :

- données relatives à l'utilisateur (choisies explicitement ou collectées du fait de son activité) ;

- données relatives à des groupes d'utilisateurs ;
- données issues de ressources externes.

Cela nous conduit à formaliser ces statuts, définissant ainsi une macrostructure fonctionnelle pour le prototype. En effet, le système doit offrir un point de vue sur les ressources lexicales qui soit personnalisé tout en permettant que certaines informations soient mutualisées.

Nous avons ainsi défini une architecture multi-niveau qui permet de traiter les différents regards sur le lexique, les droits de modification sont différents selon les niveaux et chaque utilisateur perçoit les informations contenues selon le prisme des niveaux successifs proposés (voir figure 28) :

- Le dictionnaire de référence pour une langue donnée est la somme des informations de la base lexicale qui a été validée, que ce soit par le biais d'une institution externe à l'instance ou par le biais des mécanismes de validation du système.
- Un lexique institutionnel a pour vocation de synthétiser le point de vue sur le lexique d'une institution. Une institution pourra être tout à fait externe à l'instance déployée (ex : Conseil de l'Europe) ou un groupement d'utilisateurs. Ce lexique constitue une forme de référentiel, il n'a pas vocation à évoluer selon le déroulement d'un cours.
- Un lexique de groupe est un lexique dynamique qui peut être modifié par un ensemble d'utilisateurs. La configuration de base sera un enseignant propriétaire d'un lexique de groupe, qui y donne accès (y compris en écriture) à un groupe d'apprenants.
- Un lexique personnel est le lexique d'un apprenant, qui seul pourra modifier son contenu et décider de sa visibilité. Au sein de ce lexique personnel, les données pourront être étiquetées pour faire la différence entre les entrées que l'apprenant a explicitement ajoutées à son lexique et celles qui y ont été ajoutées pour une raison ou pour une autre par le système.

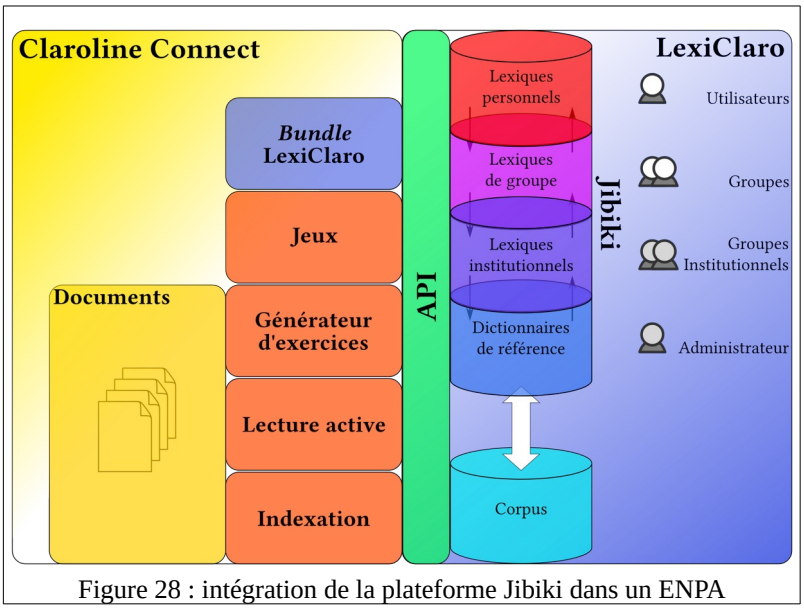


Figure 28 : intégration de la plateforme Jibiki dans un ENPA

5.1.3 Réalisation d'un prototype basé sur Jibiki

Nous avons mis en œuvre un prototype de module de gestion de bases lexicales au sein d'un ENPA. Nous avons choisi d'intégrer le prototype à Claroline44 (voir figure 29) puisque c'est

l’environnement utilisé dans le projet Innovalangues, mais l’architecture modulaire fondée sur l’utilisation d'APIs pour communiquer avec la base Jibiki fait qu’il est tout à fait possible de développer des modules pour d’autres environnements d’apprentissage (même s’ils ne sont pas dédiés à l’apprentissage des langues) tels que Moodle45 ou Chamilo46.

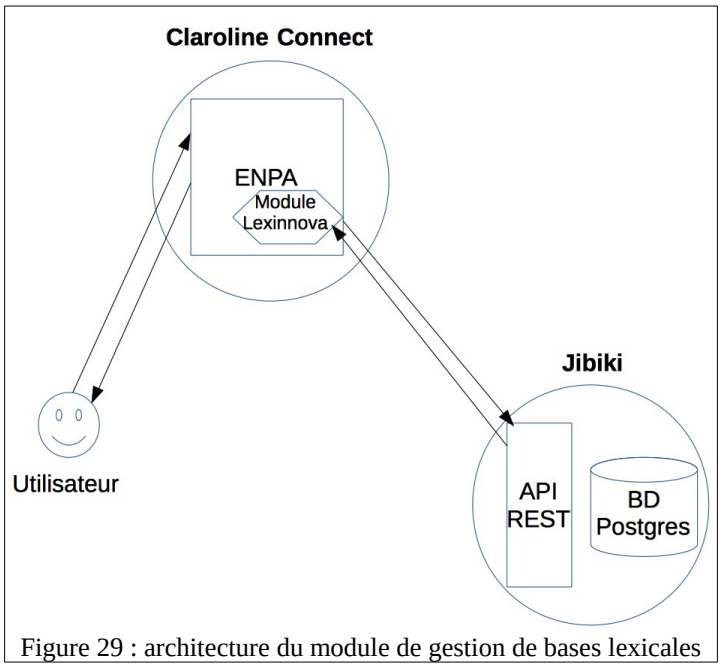


Figure 29 : architecture du module de gestion de bases lexicales

Le module est programmé en PHP[20] et accessible en ligne[44]. Il communique avec Jibiki par l’intermédiaire d’une API pour gérer les données (voir Annexe 1) et une autre pour les utilisateurs (voir Annexe 2).

Chaque utilisateur de l’ENPA peut consulter les dictionnaires de références ainsi que les lexiques institutionnels installés sur la plateforme. Il peut consulter et modifier les lexiques de groupe auxquels il a accès. Il peut enfin créer, modifier et supprimer autant de lexiques personnels qu’il le souhaite (voir figure 30). Il peut aussi partager ses lexiques personnels en lecture et/ou en écriture avec d’autres utilisateurs de la plateforme.

44 https://totoro.imag.fr/Lexm

Figure 30 : modification de l'exemple dans le lexique personnel de l'ENPA

Nous avons également ajouté pour l'espagnol un module de lecture active (voir section suivante) ainsi qu'un générateur d'exercices à partir des exemples du lexique personnel (voir figure 31).

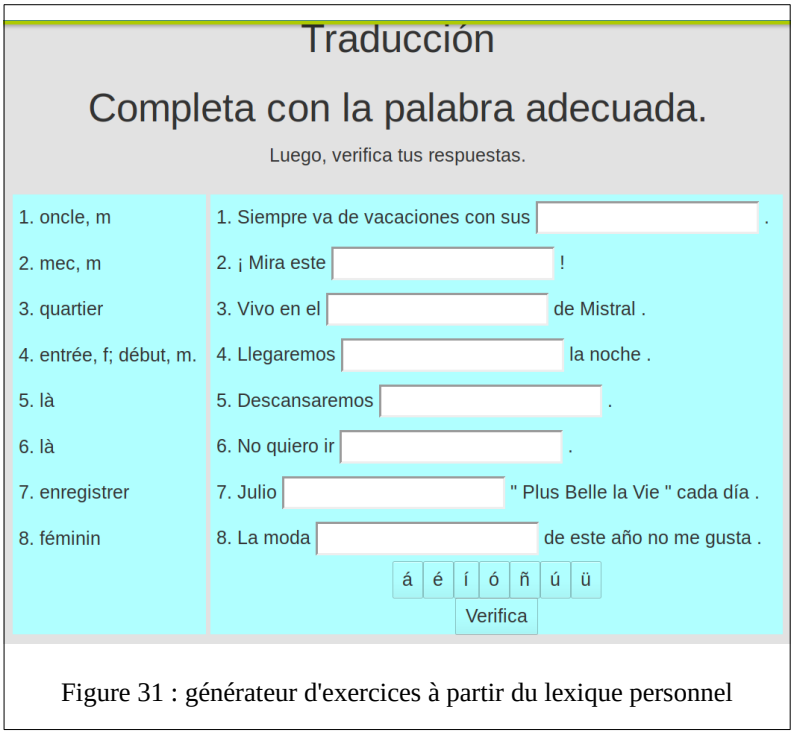


Figure 31 : générateur d'exercices à partir du lexique personnel

Lors des différentes présentations de ce travail, les enseignants de langues ont manifesté un vif intérêt. Nous avons recruté un stagiaire de master 2 pour travailler sur le projet. Il était prévu de continuer avec une thèse CIFRE avec l'entreprise Claroline.com, mais celle-ci a fait faillite.

En perspective, nous prévoyons d'expérimenter ce prototype lors d'un cours de langue. Emmanuelle Eggers est volontaire pour le tester dans un cours d'espagnol. Ensuite, nous souhaitons intégrer le prototype dans un ENPA.

Article associé au chapitre 5.1

Mathieu Mangeot, Valérie Bellynck, Emmanuelle Eggers, Mathieu Loiseau & Yoann Goudin (2016) *Exploitation d'une base lexicale dans le cadre de la conception de l'ENPA Innovalangues.* Actes de l'atelier Enseignement des langues et Traitement automatique des langues ELTAL 2016, conférence JEP-TALN-RECITAL 2016, Paris, France, 4 juillet 2016, 12 p. https://hal.archives-ouvertes.fr/hal-01382457

5.2 Lecture active enrichie pour l'intercompréhension

Le principe de la lecture active est d'enrichir un texte avec des informations permettant de mieux en comprendre le sens. Le but est d'augmenter les compétences du lecteur dans une langue donnée plutôt que d'augmenter ses performances comme avec la traduction automatique. Certaines informations rajoutées sont liées à la nature de chaque mot : prononciation, catégorie grammaticale, etc. D'autres peuvent être une traduction dans une langue mieux connue du lecteur. Ces dernières ne sont affichées qu'au survol de la souris pour ne pas gêner la lecture du texte et éviter que le lecteur ne se focalise uniquement sur les traductions.

Le défi sociétal de cette partie est de proposer des outils novateurs pour apprenants de langues étrangères.

Le défi scientifique est utiliser des outils et ressources de traitement automatique des langues pour enrichir les textes et faciliter ainsi leur compréhension par le lecteur.

5.2.1 Concept de lecture active

Un outil de lecture active (Abdellaoui et al., 2018) procède de la manière suivante :

1. Le texte reçu en entrée est envoyé à un lemmatiseur ou analyseur morphologique afin d'obtenir le lemme, les informations grammaticales et éventuellement une transcription de chaque mot.
2. À l'aide des lemmes reçus de l'analyseur, l'outil consulte ensuite un serveur lexical contenant un dictionnaire monolingue ou bilingue afin d'obtenir des informations lexicales sur chaque lemme. Il peut s'agir d'une définition ou d'une traduction.
3. L'outil affiche le texte reçu en entrée en y ajoutant les informations supplémentaires soit au-dessus ou en-dessous du texte, soit en affichant les mots avec des couleurs différentes, soit en ajoutant un code qui affichera une fenêtre surgissante lors du survol du mot par la souris.

Le concept de lecture active n'est pas nouveau. Le premier outil duquel nous avons tiré notre inspiration est *ficus*, élaboré par Mathieu Lafourcade (Lafourcade, Chauché, 1998).

Pour le projet Jibiki.fr (Mangeot-Nagata, 2016), nous avons repris ce concept et mis au point un module de lecture active de textes en français et japonais[45]. Celui-ci propose une transcription en alphabet romain (*romaji*[32]) ou en syllabaire hiragana[33] des textes japonais. Il affiche les traductions françaises des mots japonais au survol de la souris. Celles-ci proviennent du dictionnaire Jibiki.fr consulté via l'API REST (voir Annexe 1).

Ce module a été également adapté au latin dans le cadre du projet Lateo[46]. Pour cela, nous avons utilisé la version latine de l'analyseur TreeTagger38 (Schmid, 1995) et converti en XML le fameux dictionnaire latin-français Gaffiot47.

Dans sa thèse (Shah, 2017), Ritesh Shah a repris le concept de lecture active pour aider à comprendre les tweets spontanés multilingues et à commutation de code en langues étrangères avec une expérimentation sur les tweets indiens et japonais.

5.2.2 Lecture active pour l'intercompréhension

L'objectif du projet Étymolo (Abdellaoui et al., 2018) était de reprendre le concept de lecture active et d'étendre ses fonctionnalités pour faciliter l'intercompréhension dans des langues proches (soit linguistiquement, soit en contact). Le but était de montrer au lecteur les liens possibles entre les mots de différentes langues (ici berbère, français, arabe, anglais) via une étymologie commune ou plus généralement des « cognats ».

Un premier scénario autobiographique a été mis au point par Slimane Abdellaoui pour les berbérophones et arabophones apprenant le français puis l'anglais. Une nouvelle fonctionnalité a été rajoutée : montrer au lecteur les faux amis ou les petites différences orthographiques entre les langues. Par exemple, le français écrit « confortable » et l'anglais « comfortable », etc.

La thèse de Yoann Goudin (Goudin, 2017) traite de l'intercompréhension en langues sinogrammiques. Ce sont des langues qui utilisent ou ont utilisé les sinogrammes (idéogrammes d'origine chinoise) à un moment de leur histoire et qui ont intégré la prononciation qu'avaient ces sinogrammes lors de leur emprunt. Ces 4 langues principales sont le mandarin, le coréen, le japonais et le vietnamien. Malgré le fait que le japonais et le mandarin ne sont pas des langues de la même famille, lorsque les Japonais ont emprunté les sinogrammes, ils ont importé leur prononciation d'origine chinoise. La plupart du temps, il existait déjà déjà un mot d'origine japonaise pour désigner le même concept. En japonais, chaque sinogramme a donc au moins 2 prononciations en fonction de son usage dans un mot : une d'origine japonaise appelée *kunyomi*[47] et une d'origine chinoise appelée *onyomi*[48].

Pour les mots japonais qui se prononcent en onyomi, nous avons donc eu l'idée d'ajouter la prononciation en mandarin de ces mots en utilisant la transcription pinyin49.

45 https://jibiki.fr/lecture
46 http://www.msh-reseau.fr/actualites/le-projet-lateo-incube-la-msh-alpes-ses-enjeux-entre-philologie-numerique-et-histoire-des
47 https://fr.wikipedia.org/wiki/Kun'yomi
48 https://fr.wikipedia.org/wiki/On%27yomi

L'outil inter4sino[49] (voir figure 32) mis en œuvre dans le projet Étymolo fonctionne de la manière suivante :

1. L'utilisateur saisit une phrase en japonais dans la boite texte en haut de l'interface et appuie sur le bouton « Ajouter prononciation et traductions »
2. Le texte japonais est analysé avec l'outil *MeCab*[39]. Celui-ci lemmatise le texte et renvoie également pour chaque mot sa prononciation en *katakana*[50].
3. La prononciation est convertie en *romaji*[32] puis affichée au-dessus des mots japonais
4. Pour chaque lemme, le dictionnaire Jibiki.fr est consulté et la traduction française est stockée dans la page pour affichage lors du survol de la souris.
5. Pour chaque sinogramme, le dictionnaire *Kanjidic*[51] est consulté pour récupérer toutes les prononciations possibles. Ensuite, une comparaison est faite avec la prononciation obtenue à l'étape précédente pour déterminer si le mot est *onyomi* ou *kunyomi*.
6. Les mots *onyomi* sont envoyés à l'analyseur *HanLP*[52] qui renvoie la prononciation en mandarin transcrite en *pinyin* de ces mots.
7. Les mots en *kunyomi* sont surlignés en violet, ceux en *onyomi* sont surlignés en rouge et ceux en *katakana* sont surlignés en jaune.
8. Pour les mots en *onyomi*, le *pinyin* est affiché dessous.

Au bas de la figure 32, le résultat de l'analyse de la phrase saisie en entrée est affiché. La copie d'écran a été prise avec la souris sur le mot *baiku*. C'est pourquoi la traduction française « moto » est affichée en bleu. Ce mot *baiku* est surligné en jaune. C'est un mot d'origine anglaise qui vient de « bike ». Ensuite, on voit que les mots *yokohama*, *kakari* et *osoi* sont en *kunyomi* et que les mots *toukyou*, *densha*, *fun* et *hou* sont en *onyomi*. Ce sont donc des prononciations d'origine chinoise. Le *pinyin* a donc été affiché dessous.

49 https://jibiki.fr/inter4sino
50 Syllabaire japonais : https://fr.wikipedia.org/wiki/Katakana
51 https://www.edrdg.org/wiki/index.php/KANJIDIC_Project
52 http://www.hanlp.com/

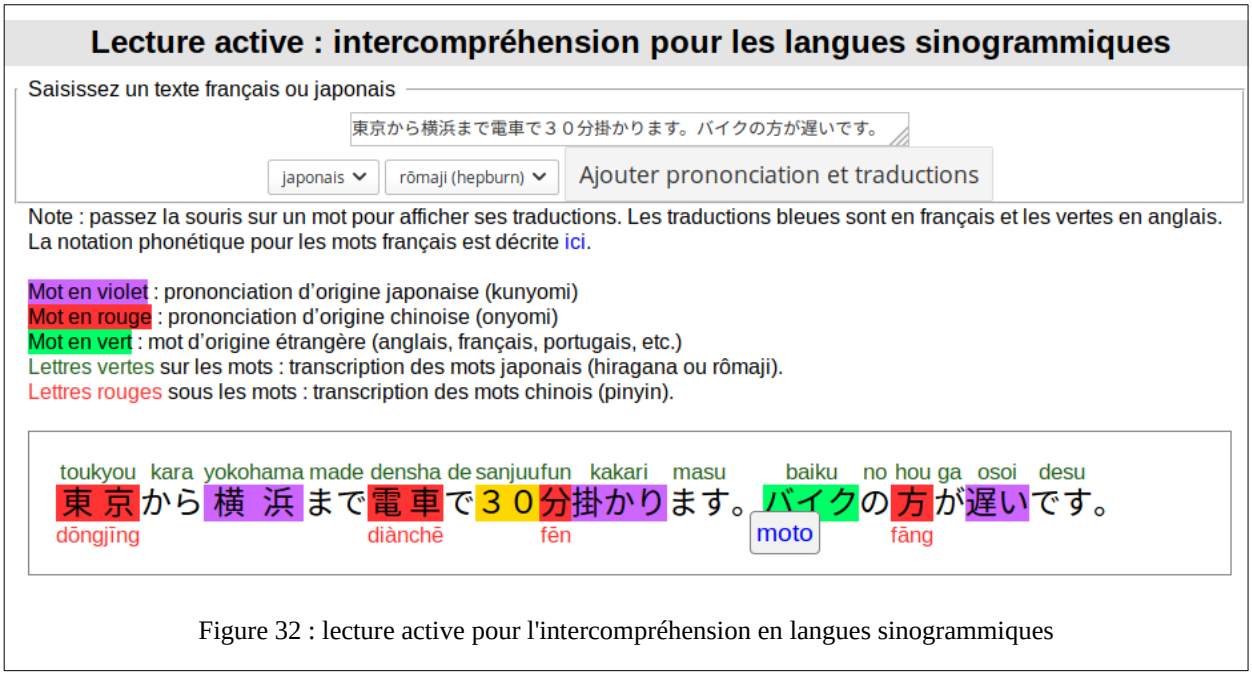


Figure 32 : lecture active pour l'intercompréhension en langues sinogrammiques

L'intérêt de l'outil se situe précisément à cet endroit. De la proximité des transcriptions en *romaji* et en *pinyin*, le lecteur va petit à petit inférer les transformations phonétiques pour passer d'une langue à l'autre. On remarque ici les correspondances suivantes : tou→*dōng*, *kyou*→*jīng*, *den*→*diàn*, *sha*→*chē*, *fun*→*fēn* et *hou*→*fāng*.

5.2.3 Concept de dictionnaire traduisant

Dès les débuts de nos recherches dans le domaine des bases lexicales, la plupart des personnes non spécialistes intéressées par nos projets de dictionnaire avaient les mêmes attentes : que ces dictionnaires traduisent automatiquement des textes. Les concepts de dictionnaire et de système de traduction automatique (TA) étant fort éloignés, nous n'avons pas prêté attention tout de suite à cette attente.

Cependant, avec l'arrivée des systèmes TA statistiques disponibles gratuitement sur Internet, cette attente s'est renforcée. Petit à petit, la frontière entre les dictionnaires et les systèmes de TA s'est estompée en premier lieu dans les usages, puis ensuite dans les réponses mêmes des systèmes. En effet, de plus en plus d'internautes utilisent *Google translate*[53] ou *Deepl*[54] comme dictionnaires en ne tapant qu'un seul mot. Le système répond maintenant à ces attentes en proposant une prononciation, des définitions et des exemples (voir figure 33).

53 https://translate.google.com/
54 https://www.deepl.com/translator

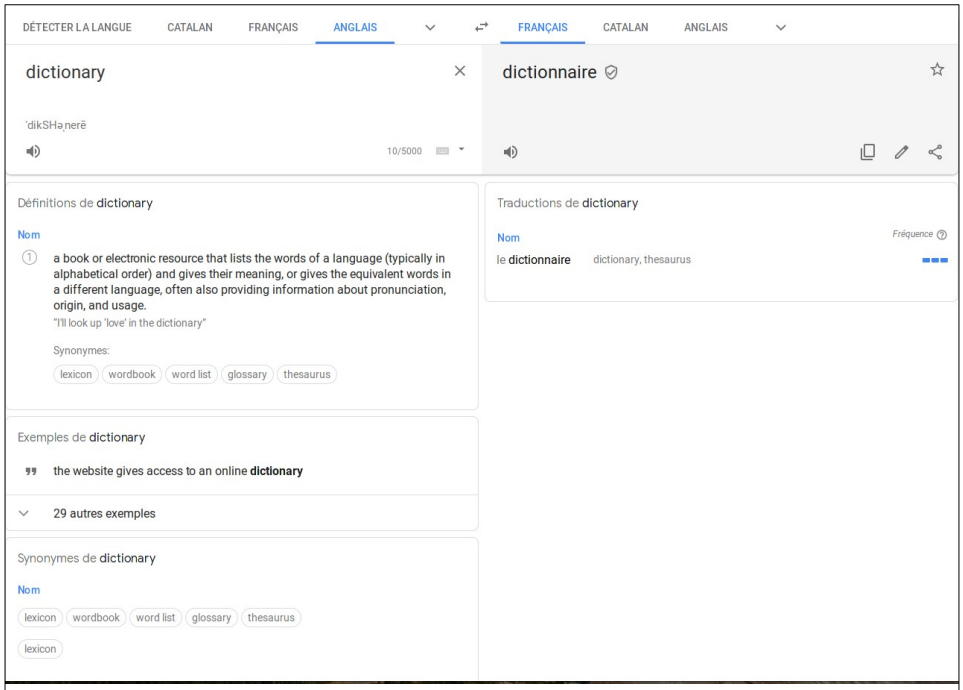


Figure 33 : système de TA *Google translate* avec sa vue dictionnaire

Aucune information n'est donnée quant à la provenance des informations apparaissant dans ces vues dictionnaire. C'est à croire que la confusion entre système de TA et dictionnaire est entretenue sciemment ! Pourtant, nous doutons que ces informations soient calculées avec un système de TA neuronal.

Le projet Papillon faisait apparaître un besoin fort de dictionnaire français-japonais de qualité accessible sur le Web. C'est pourquoi, après la mise en ligne de Jibiki.fr, j'ai été étonné de m'apercevoir en consultant les statistiques de fréquentation du site qu'il n'y avait finalement que très peu d'utilisateurs (8 visites par jour en moyenne dont les nôtres). Nous émettons deux hypothèses pour expliquer ce phénomène. La première rejoint notre constat précédent : les internautes utilisent les systèmes de TA comme dictionnaires. La deuxième est que malgré tous mes efforts, le site Jibiki.fr n'apparaît pas sur la première page des moteurs de recherche lorsque l'on tape les mots-clés « dictionnaire, japonais, français, gratuit ». Sachant que 91 % des clics se font sur la première page, cela explique le manque de visibilité du site.

Pour répondre à la fois aux non-spécialistes qui attendent d'un dictionnaire en ligne qu'il traduise, et pour enrayer la désuétude des dictionnaires traditionnels, je propose de mettre en œuvre le concept de dictionnaire traduisant à l'aide des outils de lecture active enrichie : donner le maximum d'informations au lecteur pour qu'il puisse lui-même traduire son texte.

Article associé au chapitre 5.2

Pierric Mazodier, Mathieu Mangeot, Valérie Bellynck, Yoann Goudin. *Active Reading for Intercomprehension Between Sinogramic Languages*. Natural Language Processing and Information Retrieval 2019, Jun 2019, Tokushima, Japan. https://hal.archives-ouvertes.fr/hal-02165613

6 Perspectives de recherche

Dans les précédentes parties j'ai évoqué des perspectives de recherche ou de développement à court ou moyen terme pour chaque projet abordé. Dans cette partie, je détaille les perspectives théoriques à plus long terme qui me tiennent le plus à cœur, sans oublier que les pistes de recherche empruntées sont souvent dues à des rencontres fructueuses.

6.1 Passage à l'échelle pour les bases lexicales à structure pivot

Des trois premiers objectifs principaux du projet Papillon (voir section 3.1), au moins deux ont été atteints : concevoir un environnement de gestion de bases lexicales en ligne avec la plateforme Jibiki et construire un dictionnaire de qualité et à large couverture japonais-français avec le dictionnaire Jibiki.fr. Le troisième objectif, celui du passage à l'échelle pour les bases lexicales à structure pivot, a été abordé dans le projet LexAlp (Sérasset, 2005) mais la ressource construite était une base terminologique. Le problème reste entier concernant les bases lexicales.

6.1.1 Fusion de deux dictionnaires langue A→langue B et langue B→langue A

Pour le projet Jibiki.fr (voir section 3.5), nous avons construit un dictionnaire japonais→français. J'ai initié le travail inverse de construction d'un dictionnaire français→japonais en utilisant comme ressource principale le dictionnaire Raguet-Martin (Raguet, Martin, 1953). Lorsque le processus de récupération suite à la lecture optique sera terminé, nous souhaitons expérimenter la construction d'une base lexicale à structure pivot en réifiant les deux volumes français→japonais et japonais→français puis en fusionnant les axies dans le dictionnaire pivot. Nous prévoyons d'étudier la bijectivité des liens de traduction dans ce contexte. Les liens de traduction d'un volume se retrouvent-ils dans l'autre et avec quelle proportion ?

Une fois la structure pivot mise en place, il devrait être assez facile de rajouter d'autres langues, comme l'anglais via le dictionnaire JMdict[Error: Reference source not found] (Breen, 2004), et l'allemand via le dictionnaire WaDokuJT[55] (Apel, 2002). Le mandarin pourrait également être ajouté pour les mots constitués uniquement de kanjis et prononcés en *onyomi* (voir section 5.2.2). Il existe également des ressources lexicales libres de droits anglais-mandarin telles que CEDICT[56]. L'ajout de nouvelles ressources sera l'occasion de tester les hypothèses sur les axies décrites dans ma thèse (Mangeot, 2001) pp 190-196.

6.1.2 Création d'un pivot à partir d'une langue commune

Pour le projet iBaatukaay (voir section 3.4), toutes les ressources récupérées sont des dictionnaires bilingues langue nationale→français. Une première étape est de réifier les dictionnaires pour produire un volume pivot. Ensuite, les traductions françaises peuvent être utilisées pour tenter de fusionner les axies. Le problème est que les liens vers les traductions françaises partent du niveau

55 https://wadoku.eu/
56 https://fr.wikipedia.org/wiki/CEDICT

des vocables et non des sens de mot (lexies). Il faut donc trouver des moyens de désambiguïser sémantiquement ces liens pour pouvoir les fusionner.

Une première idée qui vient à l'esprit est d'utiliser le modèle des vecteurs conceptuels défini par Mathieu Lafourcade pour le projet Papillon (Mangeot et al., 2003) mais pas encore mis en œuvre.

Il existe cependant un écueil de taille dans le projet iBaatukaay : les langues nationales sont des langues peu dotées. Il n'existe pas d'analyseur morphologique ni de thésaurus. Il n'est donc pas possible de vectoriser ces dictionnaires avec cette solution.

Le domaine de la désambiguïsation sémantique (ou WSD pour Word-Sense Disambiguation) a toutefois connu de grandes avancées ces dernières années. Il devrait être possible de trouver de nouvelles techniques pour contourner ces limitations.

De par ma (maigre) expérience de ces langues, j'ai constaté que de nombreux termes de leurs lexiques sont similaires soit parce qu'ils proviennent d'emprunts à la langue française, langue de l'ex-colonisateur, soit parce qu'elles sont en contact et proches géographiquement. De plus, leur écriture est basée sur la phonétique. Il serait peut-être possible de calculer des distances phonétiques entre termes de langues différentes reliées à une même traduction française. Si les termes sont phonétiquement proches, on peut faire l'hypothèse qu'ils décrivent le même sens de mot et donc qu'on peut regrouper leurs axies.

6.2 Automatisation de la récupération de données lexicales

Je souhaite aussi avancer pas à pas vers l'automatisation du processus de récupération de données comme je l'ai fait pour la gestion de données lexicales en laissant le contrôle à l'utilisateur. Cette automatisation sera surtout utile pour montrer des exemples concrets et pour aider l'utilisateur lorsqu'il a peu de connaissances linguistiques ou lexicales. Il pourra ensuite raffiner le processus manuellement pour obtenir une meilleure qualité de données en se basant sur les exemples produits.

6.2.1 Récupération de données provenant de lecture optique

Les sorties de lecture optique ont plusieurs formats possibles : un format structuré qui reprend le style et la présentation de la page scannée et un format textuel simple. Lors de mon travail sur le Cesselin (voir section 4.1), les pages de la sortie structurée comprenaient 3 colonnes comme pour la version imprimée[57]. Au départ, j'avais pensé utiliser cette sortie car elle comprend un certain nombre d'informations exploitables pour structurer la microstructure du dictionnaire. Mais je me suis heurté à un certain nombre de problèmes comme le fait que la structuration n'est pas universellement retranscrite. Par exemple, les différents styles et structures ne sont pas systématiquement détectés comme le gras et l'alinéa du mot-vedette. Pour ces raisons mais aussi par manque de temps pour analyser cette sortie, j'ai donc finalement utilisé la sortie textuelle simple et reconstruit la structure des articles à partir des différentes informations détectées.

57 Première page scannée du Cesselin : https://jibiki.fr/data/Cesselin/pages/1_Cesselin_pages.pdf

Pour la récupération du dictionnaire Raguet-Martin, j'ai conservé les 2 types de sortie de lecture optique. J'envisage cette fois de travailler sur la sortie structurée en prenant le temps de bien l'analyser afin de mieux détecter au moins les articles et les mots-vedettes.

Je souhaite également tester des algorithmes d'apprentissage automatique pour reconnaître les structures des articles.

Pour les corrections d'erreurs dans les textes suite à la lecture optique (voir section 4.1.3), il est possible de réutiliser le même ensemble de règles pour chaque langue. Lorsqu'un analyseur morphologique est disponible pour une langue, on peut également automatiser le marquage d'erreurs potentielles.

6.2.2 Récupération de données provenant de traitements de texte

Pour automatiser la récupération de données provenant du format *OpenXML*, je propose d'utiliser le script de signature dictionnairique et le calcul automatique de pointeurs CDM en sortie de la feuille de style XSLT (voir section 4.2.2) qui est appliquée au fichier contenu de ce format. Le résultat des pointeurs CDM est ensuite utilisé pour renommer les balises.

1. Format *OpenXML :*
    - `<office:document-content[...]><office:body><office:text><text:p text:style-name="P1"><text:span text:style-name="T1">abalone</text:span>`
2. Sortie de la feuille XSLT de simplification du balisage :
    - `<volume><P1><T1>abalone</T1>`
3. Calcul des pointeurs CDM :
    - Volume : `/volume`
    - Article : `/volume/P1`
    - Mot-vedette : `/volume/P1`
4. Renommage du balisage avec les pointeurs :
    - `<volume><article><vedette>abalone</vedette>`

Pour obtenir une meilleure efficacité, la feuille de style XSLT et le calcul de pointeurs CDM sont à affiner. Une piste est de tester l'apprentissage automatique des pointeurs CDM par rapport au calcul des heuristiques.

6.3 Pistes de recherche pour la lecture active enrichie

La lecture active enrichie concrétise le concept de dictionnaire traduisant. La recherche dans cet axe est nécessairement pluridisciplinaire. Elle associe informaticiens, linguistes, didacticiens et enseignants de langue. Un certain nombre de didacticiens sont réticents à utiliser des technologies

de TAL. Nous devons faire preuve d'habileté pour leur montrer l'utilité de ces outils. C'est l'intérêt de ce domaine.

6.3.1 Intercompréhension pour d'autres groupes de langues

L'intercompréhension entre langues est possible lorsqu'elles partagent un certain nombre de vocables soit parce que ces langues sont de la même famille, soit parce qu'elles sont en contact et donc partagent un vocabulaire commun via des emprunts réciproques.

Lors du projet Étymolo, le scénario élaboré par Slimane Abdellaoui comprenant le berbère (kabyle), l'arabe, le français et l'anglais n'a pas pu être mis en œuvre faute de temps mais également de ressources en accès libre. En effet, le berbère est une langue encore peu dotée même si des travaux sont en cours notamment à l'Institut Royal de la Culture Amazighe au Maroc. Il est au moins possible d'avancer avec les autres langues du scénario, qui sont toutes bien dotées.

Nous avons également échangé avec Christian Degache et ses collègues, spécialistes de l'intercompréhension en langues romanes. Ils nous ont confirmé l'intérêt d'un tel outil. Nous cherchons actuellement un support pour collaborer de manière effective à sa construction.

À l'avenir, nous souhaitons développer et mettre en œuvre de nouveaux scénarios adaptés aux étudiants qui travailleront sur le sujet. En effet, la motivation de travailler sur sa langue maternelle est un moteur très fort dans notre domaine.

6.3.2 Calcul de similitudes entre langues

Avec l'outil inter4sino (voir section 5.2.2 et figure 32), on peut observer les schémas de correspondance phonétique entre le mandarin et le japonais pour les mots japonais en *onyomi*. XIE Guofang a publié dans son livre « The Secret Rules for Reading Japanese Sinograms » (谢国芳, 2008) une matrice qui référence ces correspondances. Celle-ci a été calculée manuellement par l'auteur.

Nous ambitionnons de calculer automatiquement cette correspondance en utilisant un algorithme d'apprentissage automatique. L'outil devrait être indépendant des langues pour pouvoir calculer des correspondances pour d'autres groupes. Les résultats pourraient être utilisés pour l'enseignement des langues.

6.3.3 Lecture active étymologique

Comme son nom l'indique, le projet Étymolo avait au départ l'ambition d'ajouter des informations étymologiques sur les mots d'une phrase dans un module de lecture active. L'opportunité de travailler sur les langues sinogrammiques m'a détourné momentanément de cet objectif.

Je prévois d'y revenir avec 2 pistes à explorer à court et à moyen terme.

À court terme, dans l'outil *inter4sino* élaboré en collaboration avec Yoann Goudin et Pierric Mazodier, stagiaire de Master, nous souhaitons distinguer la provenance des emprunts lexicaux

japonais pour les mots écrits en *katakana*[50]. Dans le dictionnaire Jibiki.fr (voir section 3.5), les articles provenant du dictionnaire Cesselin contiennent de telles informations pour de nombreux mots. Il faudra compléter le dictionnaire pour les articles issus du dictionnaire JMdict et des liens Wikipedia.

À moyen terme, nous souhaitons avec Valérie Bellynck souhaitons collaborer avec Gilles Sérasset qui a enrichi sa base *dbnary* (Sérasset, 2015) avec des informations étymologiques provenant de Wiktionary (Pantaleo et al., 2017). Pour le moment, seul le Wiktionary anglais a été exploité mais, comme la structure des arbres étymologiques est indépendante des langues, d'autres langues peuvent être exploitées.

Bibliographie

ABDELLAOUI, Slimane, BELLYNCK, Valérie, MANGEOT, Mathieu et BOITET, Christian, 2018. Outillage de l'accès aux textes par la lecture active étymologique multilingue pour apprenants berbérophones et arabophones. In : *Traitement Automatique des Langues Africaines TALAf 2018* [en ligne]. Grenoble, France : Réseau LTT. Septembre 2018. 15 p. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-02054881.

APEL, Ulrich, 2002. WaDokuJT - A Japanese-German Dictionary Database. In : *Papillon 2002 workshop*. Tokyo, Japan : National Institute of Informatic. 18 juillet 2002. 5 p.

BERMENT, Vincent, 2004. *Méthodes pour informatiser des langues et des groupes de langues « peu dotées »*. Thèse de nouveau doctorat, Spécialité Informatique. Grenoble, France : Université Joseph Fourier Grenoble I. 277 p.

BREEN, Jim W., 2004. JMDict: a Japanese-multilingual dictionary. In : *MLR 2004: Coling 2004 workshop on multilingual linguistic resources*. Geneva, Switzerland : Association for Computational Linguistics. 28 août 2004. pp. 71-78.

CESSELIN, Gustave, 1940. *WaFutsu daijiten - Dictionnaire japonais-français*. Tokyo, Japan : Maruzen. 2340 p.

CISSÉ, Mame Thierno, DIAGNE, Abibatou, VAN CAMPENHOUDT, Marc et MURAILLE, Paul, 2007. Mise au point d'une base de données lexicale multifonctionnelle : le dictionnaire unilingue wolof et bilingue wolof-français. In : *LC 2007*. Lorient, France : 2007. pp. 163-70.

COURTIN, Christophe, TALBOT, Stéphane et MANGEOT, Mathieu, 2010. Activity regulation for the participative creation of data on-line. In : *International Conference on Machine and Web Intelligence* [en ligne]. Algiers, Algeria, France : IEEE. octobre 2010. pp. 132-139. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-00968711.

COURTOIS, Blandine, 1990. Un système de dictionnaires électroniques pour les mots simples du français. In : *Langue française, n°87, 1990. Dictionnaires électroniques du français.* 1990. pp. 11-22. DOI 10.3406/lfr.1990.6323.

DESPERRIER, Jean-Marc, 2002. Analyzis of the results of a collaborative project for the creation of a Japanese-French dictionary. In : *Papillon'2002 Seminar*. NII, Tokyo, Japan : juillet 2002.

DOAN-NGUYEN, Hai, 1998. *Techniques génériques d'accumulation d'ensembles lexicaux structurés à partir de ressources dictionnairiques informatisées multilingues hétérogènes*. Thèse de nouveau doctorat, Spécialité Informatique. Grenoble, France : Institut National Polytechnique de Grenoble.

EESTI KEELE INSTITUUT, 1997. *Eesti-vene sõnaraamat [dictionnaire estonien-russe]*. Tallin : Eesti Keele Instituut.

ENGUEHARD, Chantal, KANÉ, Soumana, MANGEOT, Mathieu, MODI, Issouf et SANOGO, Mamadou Lamine, 2011. Vers l'informatisation de quelques langues d'Afrique de l'Ouest. In : CENTRE DES ETUDES INFORMATIQUES, des Systèmes d'Information et de Communication

(éd.), *4ème atelier international sur l'Amazighe et les Nouvelles Technologies* [en ligne]. Rabat, Maroc : Publications de l'Institut Royal de la Culture Amazighe. février 2011. pp. 13-32. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-00843513.

ENGUEHARD, Chantal et MANGEOT, Mathieu, 2013. LMF for a selection of African Languages. In : FRANCOPOULO, Gil (éd.), *LMF: Lexical Markup Framework, theory and practice* [en ligne]. Paris, France : Hermès science. 8 p. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-00959228.

ENGUEHARD, Chantal et MANGEOT, Mathieu, 2014. Computerization of African languages-French dictionaries. In : *CCURL 2014 : Collaboration and Computing for Under Resourced Languages in the Linked Open Data Era* [en ligne]. Reykjavik, Iceland : ELRA. mai 2014. pp. 121-128. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-00994821.

FRANCOPOULO, Gil, BEL, Nuria, GEORGE, Monte, CALZOLARI, Nicoletta, MONACHINI, Monica, PET, Mandy et SORIA, Claudia, 2009. Multilingual resources for NLP in the lexical markup framework (LMF). In : *Language Resources and Evaluation*. 2009. Vol. 43, n° 1, pp. 57-70.

GOUDIN, Yoann, 2017. *L'intercompréhension en langues sinogrammiques : théories, représentations, enjeux, et modalités d'une didactique de la variation* [en ligne]. PhD Thesis. Paris, France : Université Sorbonne Paris Cité. Disponible à l'adresse : http://www.theses.fr/2017USPCF035.

GUT, Yvan, RASHIDA MEGAT RAMLI, Puteri, YUSOFF, Zaharin, CHOY CHUAH, Kim, SAMAT, Salina A., BOITET, Christian, NEDOBEJKINE, Nicolai, LAFOURCADE, Mathieu, GASCHLER, Jean et LEVENBACH, Dorian, 1996. *Kamus Perancis-Melayu Dewan, Dictionnaire francais-malais*. Kuala Lumpur : Dewan Bahasa dan Pustaka.

KAALEP, H-J et MUISCHNEK, K, 2002. *Eesti kirjakeele sagedussõnastik [dictionnaire des fréquences de l'estonien écrit]*. Tartu : Tartu Ülikooli kirjastus.

KANN, Kallista et KAPLINSKI, Nora, 1979. *Eesti-prantsuse sõnaraamat [dictionnaire estonien-français]*. Tallin : Valgus. 604 p.

KHOULE, Mouhamadou, MANGEOT, Mathieu, NGUER, El Hadji Mamadou et CISSÉ, Mame-Thierno, 2016. iBaatukaay : un projet de base lexicale multilingue contributive sur le web à structure pivot pour les langues africaines notamment sénégalaises. In : *Atelier Traitement Automatique des Langues Africaines TALAf 2016, conférence JEP-TALN-RECITAL 2016* [en ligne]. Paris, France : ATALA/AFCP. juillet 2016. 13 p. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-02054921.

KHOULÉ, Mouhamadou, MANGEOT, Mathieu et NGUER, Mamadou, 2018. Manipulation de dictionnaires d'origines diverses pour des langues peu dotées : la méthodologie iBaatukaay. In : *Traitement Automatique des Langues Africaines 2018* [en ligne]. Grenoble, France : septembre 2018. 20 p. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-01992863.

LAFOURCADE, Mathieu et CHAUCHÉ, Jacques, 1998. Ficus - un agent dictionnaire coopératif et extensible. In : *NLP+IA'98*. Moncton, Nouveau Brunswick, Canada : 21 08 1998. 8 p.

LAFOURCADE, Mathieu et JOUBERT, Alain, 2008. JeuxDeMots : un prototype ludique pour l'émergence de relations entre termes. In : *JADT 2008 : 9es Journées internationales d'Analyse statistique des Données Textuelles*. Lyon, France : ENS Lyon. 12 mars 2008. pp. 657-666.

LAPALME, Guy et SÉRASSET, Gilles, 2003. Batch creation of Papillon entries from DiCo. In : *Papillon'2003 Seminar*. Hokkaido University, Sapporo, Japan : http://www.papillon-dictionary.org/ConsultInformations.po. juillet 2003.

LOISEAU, Mathieu, ZAMPA, Virginie et REBOURGEON, Pauline, 2015. Magic Word : premier jeu développé dans le cadre du projet Innovalangues. In : *ALSIC - Apprentissage des Langues et Systèmes d'Information et de Communication* [en ligne]. 2015. Vol. 18, n° 2. DOI 10.4000/alsic.2828. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-01586296.

MANGEOT, Mathieu, 1997. *Outils pour lexicographes naïfs (en informatique).* [en ligne]. Research Report. Grenoble, France. Université Joseph Fourier, Grenoble 1. 64 p. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-00968774.

MANGEOT, Mathieu, 1999. *De l'ordre dans les dictionnaires au GETA* [en ligne]. Grenoble, France. Laboratoire CLIPS. 4 p. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-00968853.

MANGEOT, Mathieu, 2001. *Environnements centralisés et distribués pour lexicographes et lexicologues en contexte multilingue.* [en ligne]. Theses. Grenoble, France : Université Joseph-Fourier - Grenoble I. 280 p. Disponible à l'adresse : https://tel.archives-ouvertes.fr/tel-00006311.

MANGEOT, Mathieu, 2006. Dictionary Building with the Jibiki Platform: Software Demonstration. In : *EURALEX 2006* [en ligne]. Torino, Italy : EURALEX. mai 2006. pp. 121-126. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-00968619.

MANGEOT, Mathieu, 2014. *Construction d'un système lexical multilingue, libre de droits, centré sur le français et le japonais via des méthodes automatiques et contributives* [en ligne]. Research Report. Grenoble, France. Laboratoire d'Informatique de Grenoble. 12 p. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-01294561.

MANGEOT, Mathieu et ENGUEHARD, Chantal, 2011. Informatisation de dictionnaires langues africaines-français. In : *journées LTT 2011* [en ligne]. Villetaneuse, France : Réseau LTT. septembre 2011. 11 p. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-00780181.

MANGEOT, Mathieu et ENGUEHARD, Chantal, 2013. Des dictionnaires éditoriaux aux représentations XML standardisées. In : GALA, NURIA, ZOCK et MICHAEL (éd.), *Ressources Lexicales : contenu, construction, utilisation, évaluation* [en ligne]. Amsterdam/Philadelphia : John Benjamins. pp. 255–290. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-00959229.

MANGEOT, Mathieu et NGUYEN, Hong Thai, 2009. Projet Mot à mot : élaboration d'un système lexical multilingue par le biais de dictionnaires bilingues. In : *journées scientifiques LTT* [en ligne]. Lisbonne, Portugal, France : Réseau LTT. septembre 2009. 12 p. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-00968706.

MANGEOT, Mathieu et NGUYEN, Hong-Thai, 2009. Building lexical resources: towards programmable contributive platforms. In : *IEEE-RIVF 2009, International Conference on Computing and Communication Technologies* [en ligne]. Danang, Vietnam, Vietnam : IEEE. 2009. pp. 84-92. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-00959213.

MANGEOT, Mathieu et SÉRASSET, Gilles, 2001. Papillon Lexical Database Project: Monolingual Dictionaries and Interlingual Links. In : *NLPRS-2001* [en ligne]. Tokyo, France : AFNLP. novembre 2001. pp. 119-125. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-00966294.

MANGEOT, Mathieu, SÉRASSET, Gilles et LAFOURCADE, Mathieu, 2003. Construction collaborative d'une base lexicale multilingue, le projet Papillon. In : *Traitement Automatique des Langues*. 2003. Vol. 44, n° 2, pp. 151-176.

MANGEOT, Mathieu et THEVENIN, David, 2004. Online Generic Editing of Heterogeneous Dictionary Entries in Papillon Project. In : *Proc. of the COLING 2004 conference* [en ligne]. Geneva, Switzerland : COLING. août 2004. pp. 1029-1035. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-00968639.

MANGEOT, Mathieu et TOMOKIYO, Mutsuko, 2018. Contributions et corrections dans le dictionnaire japonais-français Jibiki.fr. In : *11e journées du réseau "Lexicologie, terminologie, traduction" LTT 2018* [en ligne]. Grenoble, France : Réseau LTT. septembre 2018. 10 p. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-02063919.

MANGEOT, Mathieu et TOUCH, Sereysethy, 2010. MotÀMot project: building a multilingual lexical system via bilingual dictionaries. In : *Second International Workshop on Spoken Languages Technologies for Under-Resourced Languages* [en ligne]. Penang, Malaysia : Universiti Sains Malaysia. 2010. 6 p. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-00959194.

MANGEOT-NAGATA, Mathieu, 2016. Collaborative Construction of a Good Quality, Broad Coverage and Copyright Free Japanese-French Dictionary. In : *International Journal of Lexicography*. septembre 2016. Vol. 31, n° 1, pp. 78-112. DOI 10.1093/ijl/ecw035.

MATSUMURA, Akira, 1995. *大辞林 daijirin [grande forêt de mots]*. 2ème édition. Tokyo, Japon : Sanseidō Shoten. ISBN ISBN 4-385-13900-8.

MAX REINHORN et BERMENT, Vincent, 2013. *Dictionnaire français-lao*. Paris, France : Éditions You Feng. 1729 p. ISBN 978-2-84279-523-8.

MEL'ČUK, Igor, 1995. Lexical Functions: A Tool for the description of Lexical Relations in the Lexicon. In : WANNER, Leo (éd.), *Lexical Functions in Lexicography and Natural Language Processing*. Amsterdam/Philadelphia : John Benjamins. pp. 37-102.

MILLER, G. A., BECKWITH, R., FELLBAUM, C ., GROSS, D. et MILLER, K. J., 1990. Introduction to WordNet: an on-line lexical database. In : *International Journal of Lexicography*. 1990. Vol. 3, n° 4, pp. 235-244.

NGUYEN, Hong-Thai, 2009. *Des systèmes de TA homogènes aux systèmes de TAO hétérogènes* [en ligne]. Theses. Grenoble, France : Université Joseph-Fourier - Grenoble I. 242 p. Disponible à l'adresse : https://tel.archives-ouvertes.fr/tel-00447571.

NGUYEN, Hong-Thai, BOITET, Christian et SÉRASSET, Gilles, 2007. PIVAX, an online contributive lexical data base for heterogeneous MT systems using a lexical pivot. In : *SNLP-2007*. Bangkok, Thailand : IEEE + Kasetsart univ. décembre 2007. 6 p.

PANTALEO, Ester, ANELLI, Vito Walter, DI NOIA, Tommaso et SERASSET, Gilles, 2017. Etytree: A Graphical and Interactive Etymology Dictionary Based on Wiktionary. In : *26th International Conference on World Wide Web Companion* [en ligne]. Perth, Australia : International World Wide Web Conferences Steering Committee. 2017. pp. 1635–1640. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-01511911.

POLGUÈRE, Alain, 2000. Towards a theoretically-motivated general public dictionary of semantic derivations and collocations for French. In : *Proceedings of EURALEX'2000*. Stuttgart, Germany : EURALEX. 2000. pp. 517-527.

POLGUÈRE, Alain, 2002. *Notions de base en lexicologie*. Montréal (Quebec), Canada : OLST-Département de linguistique et de traduction, Université de Montréal. 148 p.

RAGUET, Émile et MARTIN, Jean-Marie, 1953. *Dictionnaire français-japonais*. Tokyo, Japon : Hakusuisha. 1467 p.

ROMARY, Laurent, SALMON-ALT, Susanne et FRANCOPOULO, Gil, 2004. Standards going concrete: from LMF to Morphalou. In : *Proceedings of the COLING Workshop on Enhancing and Using Electronic Dictionaries* [en ligne]. Geneva, Swizerland : Association for Computational Linguistics. 2004. pp. 22–28. Disponible à l'adresse : http://portal.acm.org/citation.cfm?id=1610042.1610047.

SCHMID, Helmut, 1995. Improvements in Part-of-Speech Tagging with an Application to German. In : *ACL SIGDAT Workshop* [en ligne]. Dublin, Ireland : ACL. 1995. 9 p. Disponible à l'adresse : http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.52.2255.

SÉRASSET, Gilles, 2005. Multilingual Legal Terminology on the Jibiki Platform: The LexALP Project. In : LAFOURCADE, Mathieu (éd.), *Proc. of Papillon 2005 Workshop*. Chiang Rai, Thailand : Publisher Centre for Research in Speech and Language Processing. 11 décembre 2005. pp. 64-73.

SÉRASSET, Gilles, 2008. La plate-forme "Jibiki" dans le projet LexALP. In : CHIOCCHETTI, ELENA, VOLTMER et LEONHARD (éd.), *Normazione, armonizzazione e pianificazione linguistica; Normierung, Harmonisierung und Sprachplanung; Normalisation, harmonisation et planification linguistique* [en ligne]. Bolzano, Italy : EURAC Research. février 2008. pp. 31-47. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-00965438.

SÉRASSET, Gilles, 2015. DBnary: Wiktionary as a Lemon-Based Multilingual Lexical Resource in RDF. In : *Semantic Web – Interoperability, Usability, Applicability*. 2015. Vol. 6, n° 4, pp. 355-361. DOI 10.3233/SW-140147.

SHAH, Ritesh, 2017. *SUFT-1, a system for helping understand spontaneous multilingual and code-switching tweets in foreign languages : experimentation and evaluation on Indian and Japanese tweets* [en ligne]. Theses. Grenoble, France : Université Grenoble Alpes. 126 p. Disponible à l'adresse : https://tel.archives-ouvertes.fr/tel-01865400.

TOMOKIYO, Mutsuko, MANGEOT, Mathieu et BOITET, Christian, 2017. Development of a classifiers/quantifiers dictionary towards French-Japanese MT. In : *MT Summit 2017* [en ligne]. Nagoya, Japan : AAMT. septembre 2017. 11 p. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-02025447.

TOMOKIYO, Mutsuko, MANGEOT-NAGATA, Mathieu et BOITET, Christian, 2018. Analyse et classification d'exemples illustratifs dans le dictionnaire "Cesselin" en utilisant Google Traduction et un dictionnaire UNL-UWs. In : *11es journées du réseau Lexicologie Terminologie Traduction 2018* [en ligne]. Grenoble, France : Réseau LTT. septembre 2018. 10 p. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-01992882.

TRAN, Michaël, 2006. *Prolexbase : Un dictionnaire relationnel multilingue de noms propres : conception, implémentation et gestion en ligne.* [en ligne]. Thèse de doctorat. Tours, France : Université de Tours. 171 p. Disponible à l'adresse : https://tel.archives-ouvertes.fr/tel-01726999.

WAGNER, Robert A. et FISCHER, Michael J., 1974. The String-to-String Correction Problem. In : *J. ACM*. janvier 1974. Vol. 21, n° 1, pp. 168–173. DOI 10.1145/321796.321811.

ZHANG, Ying, 2016. *Modèles et outils pour des bases lexicales « métier » multilingues et contributives de grande taille, utilisables tant en traduction automatique et automatisée que pour des services dictionnairiques variés* [en ligne]. Theses. Grenoble, France : Université Grenoble Alpes. 202 p. Disponible à l'adresse : https://tel.archives-ouvertes.fr/tel-01469722.

ZHANG, Ying et MANGEOT, Mathieu, 2013. Gestion des terminologies riches : L'exemple des acronymes. In : *TALN-RECITAL 2013* [en ligne]. Sables-d'Olonne, France : ATALA. 2013. 6 p. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-00953764.

ZHANG, Ying, MANGEOT, Mathieu, BELLYNCK, Valérie et BOITET, Christian, 2014. Jibiki-LINKS: a Tool between Traditional Dictionaries and Lexical Networks for Modelling Lexical Resources. In : ZOCK, Michael, RAPP, Reinhard et HUANG, Chu-Ren (éd.), *Cognitive Aspects of the Lexicon (CogALex) 2014* [en ligne]. Dublin, Ireland : ACL. août 2014. pp. 87–98. Disponible à l'adresse : https://hal.archives-ouvertes.fr/hal-01107544.

谢国芳 XIÈ Guófāng, 2008. *《日语汉字读音规律揭秘》[rìyǔ hànzì dúyīn guīlǜ jiēmì] (The Secret Rules for Reading Japanese Sinograms)*. 广州 Guangzhou : 广州世界图书出版社 [shìjiè túshū chūbǎn shè] (Guangzhou World Publishing).

Annexe 1 : API REST pour gérer les ressources dans Jibiki

jibiki REST Application Programming Interface

Summary

List of available dictionaries

URL	api/
Method	GET
ContentType	application/xml
Returns	200 OK & XML (list of dictionaries metadata)
	401 Unauthorized (wrong credentials)

Note: available dictionaries are:

- dictionaries with public access
- dictionaries with restricted access and the user is logged
- dictionaries with private access and the user is logged and
  - in the admin group or
  - in the dictionary admin group or
  - in the dictionary validator group or
  - in the dictionary specialist group or
  - in the dictionary reader group.

Example of query:

```
curl "api/"
```

Answer:

```
<?xml version="1.0" encoding="UTF-8"?>
<d:dictionary-metadata-list xmlns:d="http://www.
```

```
clips.imag.fr/geta/services/dml">
        <dictionary-metadata xmlns="http://www-
clips.imag.fr/geta/services/dml"
            xmlns:d="http://www-clips.imag.fr/geta/services/dml"
            xmlns:xlink="http://www.w3.org/1999/xlink"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            category="bilingual"
            creation-date="13/07/2002 00:00:00"
            fullname="Lexique franco-japonais sur l'armement"
            installation-date="13/07/2002 15:04:00"
            name="Armement"
            owner="GETALP"
            type="monovolume"
            xsi:schemaLocation="http://www-clips.imag.fr/geta/services/dml
http://www-clips.imag.fr/geta/services/dml/dml.xsd">
        [...]
        </dictionary-metadata>
    </d:dictionary-metadata-files>
    <d:dictionary-metadata-files xmlns:d="http://www-
clips.imag.fr/geta/services/dml">
    <dictionary-metadata
       xmlns="http://www-clips.imag.fr/geta/services/dml"
       xmlns:d="http://www-clips.imag.fr/geta/services/dml"
       xmlns:xlink="http://www.w3.org/1999/xlink"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       category="bilingual"
       creation-date="13/07/2002 00:00:00"
       fullname="Lexique franco-japonais sur l'armement"
       installation-date="13/07/2002 15:04:00"
       name="Armement"
       owner="GETALP"
       type="monovolume"
       xsi:schemaLocation="http://www-clips.imag.fr/geta/services/dml
http://www-clips.imag.fr/geta/services/dml/dml.xsd">
    [...]
    </dictionary-metadata>
    </d:dictionary-metadata-files>
</d:dictionary-metadata-list>
```

Description of a dictionary

URL	api/[dictionary]/
Method	GET
ContentType	application/xml
Returns	200 OK & XML (dictionary metadata)
	401 Unauthorized (wrong credentials)
	404 Not Found

See the previous note for available dictionaries
Example of query:

```
    curl -u user:password "api/MyDict/"
```

Answer:

```
<?xml version="1.0" encoding="UTF-8"?>
    <d:dictionary-metadata-files xmlns:d="http://www-
clips.imag.fr/geta/services/dml">
        <dictionary-metadata
```

```
            xmlns="http://www-clips.imag.fr/geta/services/dml"
            xmlns:d="http://www-clips.imag.fr/geta/services/dml"
            xmlns:xlink="http://www.w3.org/1999/xlink"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
            category="bilingual"
            creation-date="13/07/2002 00:00:00"
            fullname="My dictionary"
            installation-date="13/07/2002 15:04:00"
            name="MyDict"
            owner="GETALP"
            type="monovolume"
            xsi:schemaLocation="http://www-clips.imag.fr/geta/services/dml
http://www-clips.imag.fr/geta/services/dml/dml.xsd">
            <languages>
                <source-language d:lang="fra"/>
                <source-language d:lang="jpn"/>
                <target-language d:lang="fra"/>
                <target-language d:lang="jpn"/>
            </languages>
            <contents>general vocabulary</contents>
            <domain>general</domain>
            <source>Mathieu Mangeot - NII</source>
            <authors>MML</authors>
            <legal>all rights belong to NII</legal>
            <access>private</access>
            <comments/>
            <administrators>
                <user-ref name="toto"/>
            </administrators>
            <volumes>
                <volume-metadata-ref source-language="fra"
xlink:href="Armement_fra-metadata.xml"/>
                <volume-metadata-ref source-language="jpn"
xlink:href="Armement_jpn-metadata.xml"/>
            </volumes>
            <xsl-stylesheet name="Armement" xlink:href="Armement-
view.xsl"/>
        </dictionary-metadata>
    </d:dictionary-metadata-files>
```

Creating a dictionary

URL	api/[dictionary]
Method	POST
ContentType	application/xml
Returns	201 Created & XML (dict metadata)
	400 Bad Request (XML not valid)
	401 Unauthorized (wrong credentials)
	409 Conflict (dictionary already existing)
	415 Unsupported Media Type (content-type not XML)
	422 Unprocessable Entity (XML not semantically correct)

The data to be sent with the POST command is the XML of the dictionary metadata and optionally the XSL stylesheet associated to the dictionary.
Note: the user has to be logged and in the admin group.
Example of query:

```
    curl -X POST \
    -u user:password \
    -H "Accept: application/xml" \
```

```
    -H "Content-Type: application/xml;charset=UTF-8" \
    -d '<?xml version="1.0" encoding="UTF-8"?>
    <d:dictionary-metadata-files xmlns:d="http://www-
clips.imag.fr/geta/services/dml">
    <dictionary-metadata
       xmlns="http://www-clips.imag.fr/geta/services/dml"
       xmlns:d="http://www-clips.imag.fr/geta/services/dml"
       xmlns:xlink="http://www.w3.org/1999/xlink"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       category="bilingual"
       creation-date="2015-05-14T07:44:55+02:00"
       fullname="Dictionnaire japonais-français par Gustave Cesselin"
       installation-date="2015-05-14T07:44:55+02:00"
       last-modification-date="2015-05-14T07:45:53+02:00"
       name="Cesselin"
       owner="mangeot"
       type="monodirectional"
       xsi:schemaLocation="http://www-clips.imag.fr/geta/services/dml
http://www-clips.imag.fr/geta/services/dml/dml.xsd">
    [...]
    </dictionary-metadata>
    <xsl:stylesheet>
    [...]
    </xsl:stylesheet>
    <xsl:stylesheet>
    [...]
    </xsl:stylesheet>
    </d:dictionary-metadata-files>
    '\
    "api/Cesselin"
```

Answer is the data sent

Modifying a dictionary

URL	api/[dictionary]
Method	PUT
ContentType	application/xml
Returns	201 Created & XML (dict metadata)
	400 Bad Request (XML not valid)
	401 Unauthorized
	404 Not Found (dictionary not found)
	415 Unsupported Media Type (content-type not XML)
	422 Unprocessable Entity (XML not semantically correct)

The data to be sent with the PUT command is the XML of the dictionary metadata and optionally the XSL stylesheet associated to the dictionary.
Note: the user has to be logged and in the admin group.
Example of query:

```
    curl -X PUT \
    -u user:password \
    -H "Accept: application/xml" \
    -H "Content-Type: application/xml;charset=UTF-8" \
    -d '<?xml version="1.0" encoding="UTF-8"?>
    <d:dictionary-metadata-files xmlns:d="http://www-
clips.imag.fr/geta/services/dml">
    <dictionary-metadata
        xmlns="http://www-clips.imag.fr/geta/services/dml"
        xmlns:d="http://www-clips.imag.fr/geta/services/dml"
        xmlns:xlink="http://www.w3.org/1999/xlink"
```

```
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        category="bilingual"
        creation-date="2015-05-14T07:44:55+02:00"
        fullname="Dictionnaire japonais-français par Gustave Cesselin"
        installation-date="2015-05-14T07:44:55+02:00"
        last-modification-date="2015-05-14T07:45:53+02:00"
        name="Cesselin"
        owner="mangeot"
        type="monodirectional"
        xsi:schemaLocation="http://www-clips.imag.fr/geta/services/dml
http://www-clips.imag.fr/geta/services/dml/dml.xsd">
    [...]
    </dictionary-metadata>
    <xsl:stylesheet>
    [...]
    </xsl:stylesheet>
    </d:dictionary-metadata-files>
    '\
    "api/Cesselin"
```

Answer is the data sent

Deleting a dictionary

URL	api/[dictionary]
Method	DELETE
Returns	204 No Content & XML (contribution)
	401 Unauthorized
	404 Not Found

Note: the user has to be logged and in the admin group.
Example of query:

```
    curl -X DELETE \
    -u user:password \
    "api/Cesselin"
```

Answer is not significant

Description of a volume

URL	api/[dictionary]/[lang]/
Method	GET
ContentType	application/xml
Returns	200 OK & XML (volume metadata)
	401 Unauthorized (wrong credentials)
	404 Not Found

See the previous note for available dictionaries
Example of query:

```
    curl -u user:password "api/MyDict/fra/"
```

Answer:

```
    curl -X POST \
    -u user:password \
    -H "Accept: application/xml" \
    -H "Content-Type: application/xml;charset=UTF-8" \
```

```
    -d '<?xml version="1.0" encoding="UTF-8"?>
    <d:volume-metadata-files xmlns:d="http://www-
clips.imag.fr/geta/services/dml">
    <volume-metadata xmlns="http://www-clips.imag.fr/geta/services/dml"
        xmlns:d="http://www-clips.imag.fr/geta/services/dml"
        xmlns:xlink="http://www.w3.org/1999/xlink"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        category="bilingual"
        creation-date="2015-05-14T07:44:55+02:00"
        fullname="Dictionnaire japonais-français par Gustave Cesselin"
        installation-date="2015-05-14T07:44:55+02:00"
        last-modification-date="2015-05-14T07:45:53+02:00"
        name="Cesselin"
        owner="mangeot"
        type="monodirectional"
        xsi:schemaLocation="http://www-clips.imag.fr/geta/services/dml
http://www-clips.imag.fr/geta/services/dml/dml.xsd">
    [...]
    </volume-metadata>
    <d:template-entry>
    [...]
    </d:template-entry>
    <d:template-interface>
    [...]
    </d:template-interface>
    <xs:schema elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
    [...]
    </xs:schema>
    <xsl:stylesheet>
    [...]
    </xsl:stylesheet>
    </d:volume-metadata-files>
    '\
    "api/Cesselin/jpn"
```

Creating a volume

URL	api/[dictionary]/[lang]
Method	POST
ContentType	application/xml
Returns	201 Created & XML (volume metadata)
	400 Bad Request (XML not valid)
	401 Unauthorized
	404 Not Found (dictionary not found)
	409 Conflict (volume already existing)
	415 Unsupported Media Type (content-type not XML)
	422 Unprocessable Entity (XML not semantically correct)

The data to be sent with the POST command is the XML of the volume metadata, the template entry, and optionally, the XML schema, the template interface and the XSL stylesheet associated to the dictionary.
Note: the user has to be logged and in the admin group.
Example of query:

```
    curl -X POST \
    -u user:password \
    -H "Accept: application/xml" \
    -H "Content-Type: application/xml;charset=UTF-8" \
    -d '<?xml version="1.0" encoding="UTF-8"?>
    <d:volume-metadata-files xmlns:d="http://www-
```

```
clips.imag.fr/geta/services/dml">
    <volume-metadata xmlns="http://www-clips.imag.fr/geta/services/dml"
xmlns:d="http://www-clips.imag.fr/geta/services/dml"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" category="bilingual"
creation-date="2015-05-14T07:44:55+02:00" fullname="Dictionnaire japonais-
français par Gustave Cesselin" installation-date="2015-05-
14T07:44:55+02:00" last-modification-date="2015-05-14T07:45:53+02:00"
name="Cesselin" owner="mangeot" type="monodirectional"
xsi:schemaLocation="http://www-clips.imag.fr/geta/services/dml http://www-
clips.imag.fr/geta/services/dml/dml.xsd">
    [...]
    </volume-metadata>
    <d:template-entry>
    [...]
    </d:template-entry>
    <d:template-interface>
    [...]
    </d:template-interface>
    <xs:schema elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
    [...]
    </xs:schema>
    <xsl:stylesheet>
    [...]
    </xsl:stylesheet>
    </d:volume-metadata-files>
    '\
    "api/Cesselin/jpn"
```

Answer is the data sent

Modifying a volume

URL	api/[dictionary]/[lang]
Method	PUT
ContentType	application/xml
Returns	201 Created & XML (volume metadata)
	400 Bad Request (XML not valid)
	401 Unauthorized
	404 Not Found (dictionary or volume not found)
	415 Unsupported Media Type (content-type not XML)
	422 Unprocessable Entity (XML not semantically correct)

The data to be sent with the PUT command is the XML of the volume metadata, the template entry, and optionally, the XML schema, the template interface and the XSL stylesheet associated to the dictionary.
Note: the user has to be logged and in the admin group.
Example of query:

```
    curl -X PUT \
    -u user:password \
    -H "Accept: application/xml" \
    -H "Content-Type: application/xml;charset=UTF-8" \
    -d '<?xml version="1.0" encoding="UTF-8"?>
    <d:volume-metadata-files xmlns:d="http://www-
clips.imag.fr/geta/services/dml">
    <volume-metadata xmlns="http://www-clips.imag.fr/geta/services/dml"
xmlns:d="http://www-clips.imag.fr/geta/services/dml"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" category="bilingual"
creation-date="2015-05-14T07:44:55+02:00" fullname="Dictionnaire japonais-
```

```
français par Gustave Cesselin" installation-date="2015-05-
14T07:44:55+02:00" last-modification-date="2015-05-14T07:45:53+02:00"
name="Cesselin" owner="mangeot" type="monodirectional"
xsi:schemaLocation="http://www-clips.imag.fr/geta/services/dml http://www-
clips.imag.fr/geta/services/dml/dml.xsd">
    [...]
    </volume-metadata>
    <d:template-entry>
    [...]
    </d:template-entry>
    <d:template-interface>
    [...]
    </d:template-interface>
    <xs:schema elementFormDefault="qualified"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
    [...]
    </xs:schema>
    <xsl:stylesheet>
    [...]
    </xsl:stylesheet>
    </d:volume-metadata-files>
    '\
    "api/Cesselin/jpn"
```

Answer is the data sent

Deleting a volume

URL	api/[dictionary]/[lang]
Method	DELETE
Returns	204 No Content & XML (contribution)
	401 Unauthorized
	404 Not Found

Note: the user has to be logged and in the admin group.
Example of query:

```
    curl -X DELETE \
    -u user:password \
    "api/Cesselin/jpn"
```

Answer is not significant

Requesting a contribution

Note: a contribution is an entry with its metadata (status, author, dates, etc.). A contribution is unique, wether an entry can have multiple values depending of its status.

URL	api/[dictionary]/[lang]/[contributionId]
Method	GET
ContentType	application/xml
	application/json
Returns	200 OK & contribution (XML or json)
	401 Unauthorized
	404 Not Found

Example of query:

```
curl -X GET \
    -H "Accept: application/xml" \
    "api/FeM/fra/fra.abandonner.8814938.c"
```

Answer:

```
<?xml version="1.0" encoding="UTF-8"?>
<volume name="FeM_fra" source-language="fra" target-languages="eng msa"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="http://www-
clips.imag.fr/geta/services/dml/fem.xsd">
<d:contribution
        d:contribid="fra.abandonner.8814938.c"
        d:originalcontribid=""
        xmlns:d="http://www-clips.imag.fr/geta/services/dml"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://www-clips.imag.fr/geta/services/dml
        http://www-clips.imag.fr/geta/services/dml/dml.xsd">
        <d:metadata>
                <d:author>automatic</d:author><
                <d:creation-date>2009/05/07 13:59:57</d:creation-date>
        ...
        <d:data>
                <entry id="fra.abandonner.8814938.e">
                        <headword>abandonner</headword><hom/>
                        <prnc>aban-done-</prnc>
                        <aux/>
                        <body>
                                <sense-list>
                                        <sense>
                                        <pos-list>v.tr.</pos-list>
                                        ...
        </d:data>
</d:contribution>
</volume>
```

Modifying an entire contribution

URL	api/[dictionary]/[lang]/[entryId]	
Method	PUT	
ContentType	application/xml	
Querystring	mode=	standard
		replace
Returns	201 Created & XML (contribution)	
	400 Bad Request (XML not valid)	
	401 Unauthorized	
	404 Not Found	
	415 Unsupported Media Type (content-type not XML)	

The data to be sent with the PUT command is the XML of the contribution (volume + metadata + entry) or the entry only (see the example). The server will create a new contribution with the data sent and will change the status of the existing contribution as "classified".
Note 1: the entry XML code must be complete with volume and contribution tags.
Note 2: the user has to be logged and in the specialist group.
Note 3: A Mode parameter can be used to avoid replace the XML code of the existing contribution instead of creating a new one. Proceed with care because it can be very dangerous. The user has to be logged and in the admin group.

standard	creates a new contribution (default mode)
replace	replace the xml code of the existing contribution

Example of query:

```
curl -X PUT \
    -u user:password \
    -H "Accept: application/xml" \
    -H "Content-Type: application/xml;charset=UTF-8" \
    -d '<?xml version="1.0" encoding="UTF-8"?>
<article id="jpn.ケース.14723001.e">
  <forme>
    <vedette>
      <vedette-romaji>kēsu</vedette-romaji>
      <vedette-hiragana>けえす</vedette-hiragana>
      <vedette-jpn>ケース</vedette-jpn>
    </vedette>
    <cat-gram>名</cat-gram>
  </forme>
  <sémantique>
    <sens>
      <texte-sens>boite, cas, ciasse</texte-sens>
    </sens>
  </sémantique>
</article>
'\
    "api/Cesselin/jpn/jpn.ケース.14723001.e"
```

Answer:

```
<?xml version="1.0" encoding="UTF-8"?>
<volume langue-cible="fra" langue-source="jpn" nom="Cesselin_jpn_fra">
<d:contribution
    xmlns:d="http://www-clips.imag.fr/geta/services/dml"
    d:contribid="jpn.ケース.14723001.c"
    d:originalcontribid="jpn.ケース.14723001.c">
  <d:metadata>
    <d:author>automatic</d:author>
    <d:groups/>
    <d:creation-date>2015/07/15 23:16:14</d:creation-date>
    <d:status>finished</d:status>
    <d:previous-classified-finished-contribution/>
  </d:metadata>
  <d:data>
    <article id="jpn.ケース.14723001.e">
      <forme>
        <vedette>
          <vedette-romaji>kēsu</vedette-romaji>
          <vedette-hiragana>けえす</vedette-hiragana>
          <vedette-jpn>ケース</vedette-jpn>
        </vedette>
        <cat-gram>名</cat-gram>
      </forme>
      <sémantique>
         <sens>
          <texte-sens>boite, cas, ciasse</texte-sens>
        </sens>
      </sémantique>
    </article>
  </d:data>
</d:contribution>
</volume>
```

Modifying part of a contribution

URL	api/[dictionary]/[lang]/[contributionId]/[string]

Method	PUT
ContentType	application/xml
Returns	201 Created & XML (contribution)
	400 Bad Request (XPath not valid)
	401 Unauthorized
	404 Not Found

The data to be sent with the PUT command is the xpath of the element or attribute to be modified.
[string] is the new value.
Note: the user has to be logged.
Example of query:

```
curl -X PUT \
    -u user:password \
    -H "Accept: application/xml" \
    -H "Content-Type: application/xml;charset=UTF-8" \
    -d '/volume/d:contribution/d:data/article/vedette-jpn/text()'\
    "api/Cesselin/jpn/jpn.ケース.14723001.c/ケース"
```

Answer:

```
<?xml version="1.0" encoding="UTF-8"?>
<volume>
<d:contribution
    xmlns:d="http://www-clips.imag.fr/geta/services/dml"
    d:contribid="jpn.ケース.14723001.c"
    d:originalcontribid="jpn.ケース.14723001.c">
  <d:metadata>
    <d:author>automatic</d:author>
    <d:groups/>
    <d:creation-date>2015/07/15 23:16:14</d:creation-date>
    <d:status>finished</d:status>
    <d:previous-classified-finished-contribution/>
  </d:metadata>
  <d:data>
    <article id="jpn.ケース.14723001.e">
      <forme>
        <vedette>
          <vedette-romaji>kēsu</vedette-romaji>
          <vedette-hiragana>けえす</vedette-hiragana>
          <vedette-jpn>ケース</vedette-jpn>
        </vedette>
        <cat-gram>名</cat-gram>
      </forme>
      <sémantique>
         <sens>
          <texte-sens>boite, cas, ciasse</texte-sens>
        </sens>
      </sémantique>
    </article>
  </d:data>
</d:contribution>
</volume>
```

Creating entries or contributions

URL	api/[dictionary]/[lang]/[entryId]
Method	POST
ContentType	application/xml

Returns	201 Created & XML (contribution)
	400 Bad Request (XML not valid)
	401 Unauthorized
	409 Conflict (entry already existing)
	415 Unsupported Media Type (content-type not XML)
422 Unprocessable entity (XML not semantically correct)	

The data to be sent with the POST command is the XML of the entries or contributions (entry + metadata).
Note 1: the XML code of the entries or contributions must be complete with the volume tag.
Note 2: the user has to be logged and in the validator group.
Examples of query:

```
curl -X POST \
    -u user:password \
    -H "Accept: application/xml" \
    -H "Content-Type: application/xml;charset=UTF-8" \
    -d '<?xml version="1.0" encoding="UTF-8"?>
<volume>
<article id="jpn.ケース.14723001.e">
  <forme>
    <vedette>
      <vedette-romaji>kēsu</vedette-romaji>
      <vedette-hiragana>けえす</vedette-hiragana>
      <vedette-jpn>ケース</vedette-jpn>
    </vedette>
    <cat-gram>名</cat-gram>
  </forme>
  <sémantique>
    <sens>
      <texte-sens>boite, cas, ciasse</texte-sens>
    </sens>
  </sémantique>
</article>
</volume>
'\
    "api/Cesselin/jpn/jpn.ケース.14723001.e"
```

```
    curl -X POST \
    -u user:password \
    -H "Accept: application/xml" \
    -H "Content-Type: application/xml;charset=UTF-8" \
    -d '@myfilename.xml'\
    "api/Cesselin/jpn/jpn.ケース.14723001.e"
```

Note: filename.xml is an XML file containing the entry.
Answer:

```
<?xml version="1.0" encoding="UTF-8"?>
<volume>
<d:contribution
    xmlns:d="http://www-clips.imag.fr/geta/services/dml"
    d:contribid="jpn.ケース.14723001.c"
    d:originalcontribid="jpn.ケース.14723001.c">
  <d:metadata>
    <d:author>automatic</d:author>
    <d:groups/>
    <d:creation-date>2015/07/15 23:16:14</d:creation-date>
    <d:status>finished</d:status>
```

```
    <d:previous-classified-finished-contribution/>
  </d:metadata>
  <d:data>
    <article id="jpn.ケース.14723001.e">
      <forme>
        <vedette>
          <vedette-romaji>kēsu</vedette-romaji>
          <vedette-hiragana>けえす</vedette-hiragana>
          <vedette-jpn>ケース</vedette-jpn>
        </vedette>
        <cat-gram>名</cat-gram>
      </forme>
      <sémantique>
         <sens>
          <texte-sens>boite, cas, ciasse</texte-sens>
        </sens>
      </sémantique>
    </article>
  </d:data>
</d:contribution>
</volume>
```

Deleting a contribution

URL	api/[dictionary]/[lang]/[contributionId]
Method	DELETE
Returns	204 No Content & XML (contribution)
	401 Unauthorized
	404 Not Found

Note: the user has to be logged and in the admin group.
Example of query:

```
curl -X DELETE \
    -u user:password \
    "api/Cesselin/jpn/jpn.ケース.14723001.c"
```

Answer is not significant

Querying entries

URL	api/[dictionary]/[lang]/[criteria]/[string]	
URL	api/[dictionary]/[lang]/[criteria]/[string]/[key]	
ContentType	application/xml	
	application/json	
Method	GET	
Querystring	strategy=	strategy
	count=	count
	startIndex=	startIndex
	orderBy=	asc / desc
Returns	200 OK & XML or JSON (entry handle, key value or full XML)	
	401 Unauthorized	
	404 Not Found	

The searches with the [key] part give the value of the key for the corresponding entries. For example, searching an entry with *api/*/eng/cdm-headword/trial/cdm-pos* will give a list of parts-of-speech for the "trial" entry.
Several keys can be used at once. They have to be separated with the "|" pipe symbol. For example, searching an entry with *api/*/jpn/cdm-headword|cdm-reading/ぜんとう/* will give a list of handles for entries with headwords or

readings matching "ぜんとう".
It is possible to query all the dictionaries by putting a ***** instead of the name of the dictionary. It is also possible to do the same for the [lang] and [key] part of the URL.
Several strings can be queried at once. They have to be separated with the "|" pipe symbol. For example, searching an entry with *api/*/eng/cdm-headword/trial|essay/* will give a list of handles for entries with headwords matching "trial" or "essay".
The **handle** criteria is special as it does not give a list of entry handles but the full entry XML content.
Example of query:

```
curl -X GET \
    -H "Accept: application/xml" \
    "api/FeM/fra/cdm-headword/abandon?strategy=EQUAL"
```

Answer:

```
<?xml version="1.0" encoding="UTF-8"?>
<entry-list xmlns="http://www-clips.imag.fr/geta/services/dml">
        <entry dictionary="FeM" lang="fra">
                <criteria value='cdm-headword'>abandon</criteria>
                <handle>8814938</handle>
        </entry>
</entry-list>
```

Example of query:

```
curl -X GET \
    -H "Accept: application/xml" \
    "api/FeM/fra/cdm-headword/abandonner/cdm-pos"
```

Answer:

```
<?xml version="1.0" encoding="UTF-8"?>
<entry-list xmlns="http://www-clips.imag.fr/geta/services/dml">
        <entry dictionary="FeM" lang="fra">
                <criteria value='cdm-headword'>abandonner</criteria>
                <handle>8814938</handle>
                <key value='cdm-pos'>v.t.</key>
        </entry>
</entry-list>
```

Example of query:

```
curl -X GET \
    -H "Accept: application/xml" \
    "api/FeM/fra/handle/8814938"
```

Answer:

```
<?xml version="1.0" encoding="UTF-8"?>
<volume name="FeM_fra" source-language="fra" target-languages="eng msa"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:noNamespaceSchemaLocation="http://www-
clips.imag.fr/geta/services/dml/fem.xsd">
<d:contribution
        d:contribid="fra.abandonner.8814938.c"
```

```
        d:originalcontribid=""
        xmlns:d="http://www-clips.imag.fr/geta/services/dml"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xsi:schemaLocation="http://www-clips.imag.fr/geta/services/dml
        http://www-clips.imag.fr/geta/services/dml/dml.xsd">
        <d:metadata>
                <d:author>automatic</d:author><
                <d:creation-date>2009/05/07 13:59:57</d:creation-date>
        ...
        <d:data>
                <entry id="fra.abandonner.8814938.e">
                        <headword>abandonner</headword><hom/>
                        <prnc>aban-done-</prnc>
                        <aux/>
                        <body>
                                <sense-list>
                                        <sense>
                                        <pos-list>v.tr.</pos-list>
                                        ...
        </d:data>
</d:contribution>
</volume>
```

Example of query:

```
curl -X GET \
    -H "Accept: application/json" \
    "api/Cesselin/jpn/cdm-headword/テーブル/?strategy=EQUAL"
```

Answer:

```
{"entry-list": {
  "entry": {
    "handle": "19734730",
    "criteria": {
      "content": "テーブル",
      "name": "cdm-headword",
      "value": "テーブル",
      "strategy": "EQUAL"
    },
    "dictionary": "Cesselin",
    "lang": "jpn"
  },
  "xmlns": "http://www-clips.imag.fr/geta/services/dml"
}}
```

Criteria and keys
Criteria only are one of the following:

handle	unique entry handle specific to the server database
previous	previous entry following the volume lexicographic order
next	previous entry following the volume lexicographic order

For the above criteria, the string must be an entry handle.
Criteria and keys are one of the following:

cdm-entry-id	entry id
cdm-headword	entry headword
cdm-headword-variant	entry headword variant

cdm-reading	entry headword reading (ex: japanese kana)
cdm-writing	entry headword transcription (ex: pinyin for Chinese)
cdm-pronunciation	entry headword pronunciation (Usually in IPA but it depends on the resource)
cdm-pos	entry part-of-speech (can be "n", "noun", etc. It depends on the resource)
cdm-definition	entry definition
cdm-domain	entry domains
cdm-translation	entry headword translations in another language
cdm-example	entry examples
cdm-idiom	entry idioms
cdm-contribution-id	contribution id
cdm-previous-classified-finished-contribution	previous contribution id
cdm-contribution-author	contribution author
cdm-contribution-creation-date	contribution creation date (`yyyy/mm/dd HH:mm:ss`)
cdm-contribution-finition-date	contribution finition date (`yyyy/mm/dd HH:mm:ss`)
cdm-modification-author	modification author
cdm-modification-date	modification date (`yyyy/mm/dd HH:mm:ss`)
cdm-contribution-status	contribution status

Note: criteria and keys depend on the resource. A criteria or a key might not be available for one specific resource. Other criteria and keys not in the list may exist only for one specific resource.
cdm stands for Common Dictionary Markup
Keys only are one of the following:

entries	XML code of the 100 first entries satisfying the criteria
pivax	UNL translations of a word via axemes and axies

Strategy
Strategy is one of the following:

EQUAL	indexed search
CASE_SENSITIVE_STARTS_WITH	indexed search
CASE_SENSITIVE_ENDS_WITH	sequential search
CASE_SENSITIVE_CONTAINS	sequential search
CASE_INSENSITIVE_EQUAL	indexed search
CASE_INSENSITIVE_STARTS_WITH	indexed search
CASE_INSENSITIVE_ENDS_WITH	sequential search
CASE_INSENSITIVE_CONTAINS	sequential search
NOT_EQUAL	indexed search
GREATER_THAN	indexed search
GREATER_THAN_OR_EQUAL	indexed search
LESS_THAN	indexed search
LESS_THAN_OR_EQUAL	indexed search

Default strategy is `CASE_INSENSITIVE_STARTS_WITH`

- All the answers are in XML and encoded in UTF-8.
- *lang* is the ISO 639-2/T 3 letter code of the language (eg: 'eng' for English,'esp' for Spanish,'fra' for French, etc.)
- All parameters are case sensitive (dictionary, contributionId, string, etc.)
- Dates are in the following format: `yyyy/mm/dd HH:mm:ss`
- It is possible to use the SQL '%' metacharacter in queries.

Querying entries using link (for pivax data)

URL	api/[dictionary]/[lang]/link/[headword]

Method	GET
ContentType	application/xml
	application/json
Returns	200 OK & XML (entryId, headword)
	401 Unauthorized
	404 Not Found

Example of query:

```
curl -X GET \
-H "Accept: application/xml" \
"api/CommonUNLDict/fra/link/manger"
```

Answer:

```
<?xml version="1.0" encoding="UTF-8"?>
<entry-list>
        <entry dictionary="CommonUNLDict" lang="eng">
                <entryId>eng.eat.v</entryId>
                <headword>eat</headword>
        </entry>
        <entry dictionary="CommonUNLDict" lang="fra">
                <entryId>fra.manger.v</entryId>
                <headword>manger</headword>
        </entry>
                <entry dictionary="CommonUNLDict" lang="unl">

<entryId>eat(icl>consume>do,agt>living_thing,obj>concrete_thing,ins>thing)
</entryId>
                <headword>eat</headword>
        </entry>
</entry-list>
```

Authentication methods

Authentication method is one of the following:

BasicAuth	login and password are encoded in base 64 and sent to the server in the request header. This method is not secure because the encryption method is easy to decrypt.
Authentication cookie	cookies are sent to the server in the request header. In order to ask for an authentication cookie, please visit the login page.
Querystring	login=login
	password=password
	This method is not secure at all : login and password are sent in cleartext. They will even be registered on the web server logs!

Misc

If you need more help, send an email to Mathieu Mangeot.

Annexe 2 : API REST pour gérer les utilisateurs dans Jibiki

Le projet **Jibiki.fr**

Dictionnaire japonais-français

Consultation

→

Consultation avancée

jibiki users REST Application Programming Interface

Summary

List of available users

URL	apiusers/users
Method	GET
ContentType	application/xml
	application/json
Returns	200 OK & list of users (XML or json)
	401 Unauthorized (wrong credentials)

Note: if the user is admin, the user email and groups are displayed
Example of query:

```
curl -H 'Accept: application/json' "apiusers/users"
```

Description of a user

URL	apiusers/users/[login]/
Method	GET
ContentType	application/xml
	application/json
Returns	200 OK & user (XML or json)
	401 Unauthorized (wrong credentials)
	404 Not Found

Note: if the user is admin, or herself, the user email and groups are displayed
Example of query:

```
curl -H 'Accept:application/json' \
"apiusers/users/login"
```

Answer:

```
{"user": {
"xmlns": "http://www-clips.imag.fr/geta/services/dml",
"name": "Mathieu Mangeot",
"login": "mangeot"
}}
```

Creating a new user

URL	apiusers/users/[login]
Method	POST
ContentType	application/xml
	application/json
Returns	201 Created & user (XML or json)
	400 Bad Request (XML or json not valid)
	401 Unauthorized (wrong credentials)
	409 Conflict (user already existing)
	422 Unprocessable Entity (XML not semantically correct)

The data to be sent with the POST command is the description of the user in XML or json.
Note 1: the user has to identify herself.
Example of query:

```
curl -X POST \
-u mangeot:password \
-H "Content-Type: application/json;charset=UTF-8" \
-d '{"user": {
"name": "Mathieu Mangeot",
"lang": "fra",
"login": "mangeot",
```

```
    "email": "Mathieu.Mangeot_@_myemailaddress.fr"
    }}'\
    "apiusers/users/mangeot"
```

Modifying an existing user

URL	apiusers/users/[login]
Method	PUT
ContentType	application/xml
	application/json
Returns	200 OK & XML or json (user)
	400 Bad Request (XML or json not valid)
	401 Unauthorized
	404 Not Found (user not found)
	422 Unprocessable Entity (XML not semantically correct)

The data to be sent with the PUT command is the XML or json of the user.
Note: Only the user herself or an admin user in the admin group can modify a user.
Example of query:

```
    curl -X PUT \
    -u mangeot:password \
    -H "Content-Type: application/json;charset=UTF-8" \
    -d '{"user": {
    "name": "Mathieu Mangeot",
    "lang": "eng",
    "login": "mangeot",
    "email": "Mathieu.Mangeot_@_myemailaddress.fr"
    }}'\
    "apiusers/users/mangeot"
```

Deleting an existing user

URL	apiusers/users/[login]
Method	DELETE
Returns	204 No Content & XML or json (user)
	401 Unauthorized
	404 Not Found

Note: Only the user herself or an admin user in the admin group can modify an existing user.
Example of query:

```
    curl -X DELETE \
    -u login:password \
    "apiusers/users/login"
```

Answer is not significant

List of groups that include the user

URL	apiusers/users/[login]/groups
Method	GET
ContentType	application/xml
	application/json
Returns	200 OK & user (XML or json)
	401 Unauthorized
	404 Not Found

Note: the user has to be herself or in the admin group.
Example of query:

```
curl \
-u login:password \
-H 'Accept:application/json' \
"apiusers/users/login/groups"
```

Answer:

```
{"d:group-list": {
"d:group": [
{"name": "admin"},
{
"role": "admin",
"dictionary": "Cesselin",
"name": "admind_Cesselin"
},
{
"role": "admin",
"dictionary": "Kanjidic",
"name": "admind_Kanjidic"
}
],
"xmlns:d": "http://www-clips.imag.fr/geta/services/dml"
}}
```

Adding an existing user into an existing group

URL	apiusers/users/[login]/groups/[groupname]
Method	PUT
ContentType	application/xml
	application/json
Returns	200 OK & XML or json (user)
	401 Unauthorized
	404 Not Found (user not found)

Note: Only the user herself or an admin user in the admin group can modify a user.

Example of query:

```
curl -X PUT \
-u login:password \
"apiusers/users/login/groups/readerd_Cesselin"
```

Removing an existing user from an existing group

URL	apiusers/users/[login]/groups/[groupname]
Method	DELETE
ContentType	application/xml
	application/json
Returns	204 No Content & XML or json (user)
	401 Unauthorized
	404 Not Found (user or group)

Note: Only the user herself or an admin user in the admin group can delete a user.
Example of query:

```
curl -X DELETE \
-u login:password \
"apiusers/users/login/groups/readerd_Cesselin"
```

List of available groups

URL	apiusers/groups
Method	GET
ContentType	application/xml
	application/json
Returns	200 OK & list of groups (XML or json)
	401 Unauthorized

Example of query:

```
curl -H 'Accept: application/json' "apiusers/groups"
```

Answer:

```
{"d:group-list": {
  "d:group": [
    {"name": "admind_Armement"},
    {"name": "admind_Cesselin"},
    {"name": "admind_DiLAF"},
    {"name": "admind_Kanjidic"},
    {"name": "admind_Maniette"},
    {"name": "readerd_Armement"},
```

```
    {"name": "readerd_Cesselin"},
    {"name": "readerd_Maniette"},
    {"name": "specialist"},
    {"name": "validator"},
    {"name": "admin"}
  ],
  "xmlns:d": "http://www-clips.imag.fr/geta/services/dml"
}}
```

List of users members of a groups

URL	apiusers/groups/[groupname]
Method	GET
ContentType	application/xml
	application/json
Returns	200 OK & list of groups (XML or json)

Example of query:

```
curl
-u login:password \
-H 'Accept: application/json' \
"apiusers/groups/admind_Cesselin"
```

```
{"d:group": {
  "xmlns:d": "http://www-
clips.imag.fr/geta/services/dml",
  "members": {"user-ref": "mangeot"},
  "name": "admind_Cesselin",
  "admins": {"user-ref": "mangeot"}
}}
```

Adding several users in a groups

URL	apiusers/groups/[groupname]
Method	POST
ContentType	application/xml
	application/json
Returns	201 Created & user (XML or json)
	400 Bad Request (XML or json not valid)
	401 Unauthorized
	409 Conflict (user already existing)
	422 Unprocessable Entity (XML not semantically correct)

The data to be sent with the POST command is the description of the user in XML or json.
Note 1: the user has to identify herself.
Example of query:

```
curl -X POST \
-u mangeot:password \
-H "Content-Type: application/json;charset=UTF-8" \
-d '{"user-list": {
"user": [
{
"login": "titi",
},
"login": "toto",
}
]
}}'\
"apiusers/groups/readerd_Cesselin/"
```

Removing a list of users from an existing group

URL	apiusers/groups/[groupname]
Method	DELETE
ContentType	application/xml
	application/json
Returns	204 No Content & XML or json (user)
	401 Unauthorized
	404 Not Found (user or group)

Note: Only the user herself or an admin user in the admin group can delete a user.
Example of query:

```
curl -X DELETE \
-u mangeot:password \
-H "Content-Type: application/json;charset=UTF-8" \
-d '{"user-list": {
"user": [
{
"login": "titi",
},
"login": "toto",
}
]
}}'\
"apiusers/groups/readerd_Cesselin/"
```

Adding an existing user into an existing group

URL	apiusers/groups/[groupname]/users/[login]
Method	PUT
ContentType	application/xml
	application/json
Returns	200 OK & XML or json (user)

	401 Unauthorized
	404 Not Found (user not found)

Note: Only the user herself or an admin user in the admin group can modify a user.
Example of query:

```
curl -X PUT \
-u login:password \
"apiusers/groups/readerd_Cesselin/users/toto"
```

Removing an existing user from an existing group

URL	apiusers/groups/[groupname]/users/[login]
Method	DELETE
ContentType	application/xml
	application/json
Returns	204 No Content & XML or json (user)
	401 Unauthorized
	404 Not Found (user or group)

Note: Only the user herself or an admin user in the admin group can delete a user.
Example of query:

```
curl -X DELETE \
-u login:password \
"apiusers/users/login/groups/readerd_Cesselin"
```

List of roles with access to a dictionary

URL	apiusers/dictionary/[dictname]
Method	GET
ContentType	application/xml
	application/json
Returns	200 OK & list of groups (XML or json)
	401 Unauthorized
	404 Not Found (dictionary)

Example of query:

```
curl -H 'Accept: application/json'
"apiusers/dictionary/Cesselin"
```

List of users with a role with access to a dictionary

URL	apiusers/dictionary/[dictname]/[role]
Method	GET

ContentType	application/xml
	application/json
Returns	200 OK & list of groups (XML or json)
	401 Unauthorized
	404 Not Found (dictionary or role)

Example of query:

```
    curl -H 'Accept: application/json'
 "apiusers/dictionary/Cesselin/reader"
```

Gives a role related to a dictionary to a user

URL	apiusers/dictionary/[dictname]/[role]/[login]
Method	PUT
ContentType	application/xml
	application/json
Returns	200 OK & XML or json (user)
	401 Unauthorized
	404 Not Found (dictionary or user not found)

Note: Only an admin or dictionary admin can give a role to a user.
Example of query:

```
    curl -X PUT \
    -u login:password \
    "apiusers/dictionary/Cesselin/reader/toto"
```

Removing a role from a user in a dictionary

URL	apiusers/dictionary/[dictname]/[role]/[login]
Method	DELETE
ContentType	application/xml
	application/json
Returns	204 No Content & XML or json (user)
	401 Unauthorized
	404 Not Found (dictionary, role or user)

Note: Only an admin or dictionary admin can remove a role to a user.
Example of query:

```
    curl -X DELETE \
    -u login:password \
    "apiusers/users/login/groups/readerd_Cesselin"
```

Authentication methods

Authentication method is one of the following:

BasicAuth	login and password are encoded in base 64 and sent to the server in the request header. This method is not secure because the encryption method is easy to decrypt.
Authentication cookie	cookies are sent to the server in the request header. In order to ask for an authentication cookie, please visit the login page.
Querystring	login=login
	password=password
	This method is not secure at all : login and password are sent in cleartext. They will even be registered on the web server logs!

Misc

If you need more help, send an email to Mathieu Mangeot.

Annexe 3 : Articles associés au mémoire

P 113 : article associé au chapitre 1 : Mathieu Mangeot & Chantal Enguehard (2013) *Des dictionnaires éditoriaux aux représentations XML standardisées.* Chapitre 8, livre "Ressources Lexicales : contenu, construction, utilisation, évaluation"

P 144 : article associé au chapitre 2.1 : Mathieu Mangeot & Valérie Bellynck (2018) *Un devin de microstructures pour importer ou normaliser des ressources lexicales.*

P 155 : article associé au chapitre 2.2 : Ying Zhang, Mathieu Mangeot, Valérie Bellynck & Christian Boitet (2014) *Jibiki-LINKS: a tool between traditional dictionaries and lexical networks for modelling lexical resources.*

P 167 : article associé au chapitre 2.3 : Mathieu Mangeot & David Thevenin (2004) *Online Generic Editing of Heterogeneous Dictionary Entries in Papillon Project.*

P 174 : article associé au chapitre 3.1 : Mathieu Mangeot, Gilles Sérasset & Mathieu Lafourcade (2003) *Construction collaborative de données lexicales multilingues, le projet Papillon.*

P 202 : article associé au chapitre 3.2 : Antoine Chalvin & Mathieu Mangeot (2006) *Méthodes et outils pour la lexicographie bilingue en ligne : le cas du Grand Dictionnaire Estonien-Français.*

P 210 : Article associé au chapitre 3.3 : Mathieu Mangeot (2014) *MotàMot project: conversion of a French-Khmer published dictionary for building a multilingual lexical system.*

P 218 : article associé au chapitre 3.4 : Chantal Enguehard, Mathieu Mangeot (2014) *Computerization of African languages-French dictionaries Proc. of Collaboration and Computing for Under Resourced Languages.*

P 226 : article associé aux chapitres 3.5 et 4.1 : Mathieu Mangeot (2016) *Collaborative Construction of a Good Quality, Broad Coverage and Copyright Free Japanese-French Dictionary.*

P 257 : article associé au chapitre 4.2 : Mathieu Mangeot & Chantal Enguehard (2018) *Dictionaries for Under-Resourced Languages: from Published Files to Standardized Resources Available on the Web.*

P 295 : article associé au chapitre 5.1 : Mathieu Mangeot, Valérie Bellynck, Emmanuelle Eggers, Mathieu Loiseau & Yoann Goudin (2016) *Exploitation d'une base lexicale dans le cadre de la conception de l'ENPA Innovalangues.*

P 312 : article associé au chapitre 5.2 : Pierric Mazodier, Mathieu Mangeot, Valérie Bellynck, Yoann Goudin. *Active Reading for Intercomprehension Between Sinogramic Languages.*

Chapitre 10

Des dictionnaires éditoriaux aux représentations XML standardisées

Mathieu Mangeot
GETALP-LIG, 41 rue des Mathématiques, BP 53
F-38041 GRENOBLE CEDEX 9, Université de Savoie
Chantal Enguehard
LINA CNRS UMR 6241, Université de Nantes

Introduction

Créer un dictionnaire électronique *ex nihilo* est un travail coûteux car cette tâche mobilise, sur une longue période, le travail de personnes qualifiées, si ce n'est en lexicographie, au moins en linguistique. Lorsque l'environnement socio-économique ne permet pas de rassembler les ressources nécessaires à la confection d'un dictionnaire électronique spécialement dédié au Traitement Automatique des Langues Naturelles (TALN) et que des dictionnaires éditoriaux existent, ces dictionnaires représentent une ressource importante qu'il s'agit d'utiliser pour initialiser la création de ressources lexicales électroniques.

Cet article présente des aspects théoriques et pratiques concernant la conversion de dictionnaires éditoriaux en dictionnaires électroniques. Il prend en compte la question de la limitation des moyens économiques et techniques et de la faible disponibilité des personnes qualifiées. Il est particulièrement destiné aux lexicographes linguistes ou formateurs qui réaliseront la conversion de dictionnaires éditoriaux. C'est pourquoi il détaille des points de méthodologie qui pourront sembler triviaux à des informaticiens spécialistes du traitement automatique des langues mais qui sont pourtant susceptibles d'engendrer des blocages lors d'un processus de conversion menés par des personnes peu familières de la technologie informatique, des expressions régulières ou des formats.

Nos expériences de terrain concernent les langues peu dotées en logiciels et ressources informatiques[1] et qui sont parlées principalement en Asie du sud-est (khmer, malais, vietnamien) et au Sahel (bambara, haoussa, kanouri, tamajaq, zarma), aussi la majeure partie des exemples cités et des situations socio-linguistiques décrites concerne-elle ces zones.

Après un rapide historique consacré aux formats des dictionnaires électroniques, nous présentons deux normes qui leurs sont dédiées. La question des langues peu dotées est exposée et est suivie de quelques exemples de dictionnaires éditoriaux les concernant. Les principales difficultés techniques sont détaillées. Les grandes lignes de la méthodologie de conversion sont énoncées dans la sixième partie et ensuite détaillées : conversion vers un format passerelle à l'aide d'expressions régulières (partie 7) ou d'outils spécialisés (partie 8) ; la partie 9 présente la conversion vers le format cible. La dernière partie est dédiée à la consultation des ressources à travers une plate-forme de gestion de ressources en ligne.

1. Historique des formats électroniques de dictionnaire

Dans cette partie, nous abordons uniquement les formats électroniques de dictionnaire. Pour un historique plus complet, se reporter au chapitre 2.

[1]que nous désignerons simplement par “langues peu dotées”.

1.1. Bandes de photocomposition et première standardisation : SGML

Dès les débuts de l'informatique, il est apparu intéressant d'utiliser un ordinateur pour élaborer des dictionnaires. Les premiers formats électroniques sont d'une part les dictionnaires compilés spécialement pour une application informatique et d'autre part les bandes de photocomposition, représentant un ensemble de commandes envoyées aux imprimantes afin d'imprimer des dictionnaires au format papier. Dans ce chapitre, nous laisserons de côté les dictionnaires compilés pour nous concentrer sur les bandes de photocomposition, représentant les dictionnaires éditoriaux.

À l'époque (au début des années 80), chaque maison d'édition avait mis au point son propre langage de commande pour ses imprimantes. Les bandes n'étaient pas standardisées et ne pouvaient donc pas être échangées entre maisons d'édition ou servir sur des imprimantes de marque différente.

Charles Goldfarb, employé par IBM, a alors mis au point le premier langage de balisage, SGML[2], qui deviendra une norme ISO[3] en 1986. Immédiatement, cette innovation a permis l'échange de documents électroniques de taille importante et peut être considérée comme un premier pas vers la standardisation des formats. Ce langage de balisage, directement inspiré par les bandes d'impression, reflètait principalement des indications de style et de mises en forme plutôt qu'une structuration logique de document. Ainsi, les exemples d'usage exprimés en italique sont-ils marqués à l'aide de l'élément <i>. De même, les articles sont repérés avec un élément de paragraphe <p> Par conséquent la structure des articles est alors majoritairement implicite.

1.2. Simplification du balisage : XML[4]

Il apparaît rapidement un problème avec SGML : la définition de sa structure assez lâche permet d'omettre certaines balises fermantes, ce qui nécessite de décrire précisément la structure de chaque document dans un document annexe appelé Document Type Definition (DTD). Si la DTD n'est pas disponible lors de la lecture, des ambiguïtés d'interprétation peuvent apparaître.

Article au format SGML tiré du Trésor de la Langue Française

```
<a><b>Lexicologie<i>subst. f&ea;m. <br><p>&Ea;tude scientifique du lexique. <i>L'objet de la lexicologie est une th&ea;orie compr&ea;hensive du fait lexical, tant au niveau des structures (lexique, vocabulaires) que des unit&ea;s (mot, idiome)</i> (REY, <i>Le Lexique : images et mod&eg;les</i>, Paris, Colin, 1977, p. 159).</a>
```

Interprétation

Les balises <b> et <i> indiquent que les textes qui les suivent doit être mis, respectivement, en gras ou en italique. Comme la fin de certaines portions de textes concernées par ces balises n'est pas spécifiée, plusieurs interprétations sont possibles, en voici deux :

> **Lexicologie** *subst. fém.*
> Étude scientifique du lexique. *L'objet de la lexicologie est une théorie compréhensive du fait lexical, tant au niveau des structures (lexique, vocabulaires) que des unités (mot, idiome).* (REY, *Le Lexique : images et modèles*, Paris, Colin, 1977, p. 159).
>
> **Lexicologie *subst. fém.***
> **Étude scientifique du lexique. *L'objet de la lexicologie est une théorie compréhensive du fait lexical, tant au niveau des structures (lexique, vocabulaires) que des unités (mot, idiome).* (REY, *Le Lexique : images et modèles*, Paris, Colin, 1977, p. 159).**

Figure 1. Exemple d'article au format SGML avec ambiguïté d'interprétation

[2]SGML : Standard Generalized Markup Language.
[3]ISO : International Organization for Standardization.
[4]XML : Extensible Markup Language.

Pour analyser de tels documents SGML, il fallait développer des analyseurs contextuels, capables de tenir compte du contexte dans lequel apparaissent les éléments afin de les interpréter correctement. Le contexte étant décrit à l'aide d'une DTD. Ces analyseurs sont difficiles à programmer. Il est apparu souhaitable de mettre au point un nouveau format de balisage que des analyseurs hors contexte, bien plus simples à programmer, puissent traiter.

C'est pourquoi, en 1997, sous la houlette du W3C[5], s'est formé un groupe de travail ayant comme objectif de définir un nouveau format de balisage. Les travaux de ce groupe ont débouché en 1998 sur le standard XML. XML est un sous-ensemble de SGML qui oblige à associer une balise fermante à chaque balise ouvrante. Il supprime de fait les ambiguïtés d'interprétation et simplifie la programmation des analyseurs. La référence aux DTD pour analyser les documents est alors optionnelle.

En parallèle, les codages de caractères évoluent. Le jeu de caractères codés ASCII[6], premier codage standardisé a été élaboré dans les années 60 par des américains. Il définit seulement 128 caractères, la plupart des diacritiques du français n'y sont pas inclus. SGML utilisant le codage ASCII utilise des entités caractères pour tous les caractères qui ne sont pas inclus dans la table ASCII. Ces entités commencent par le caractère « & », suivi du nom de l'entité et terminé par un autre caractère particulier (habituellement un « ; »). SGML n'a pas standardisé le nom des entités. C'est pourquoi on trouve des noms différents pour les mêmes caractères. Par exemple *&ea.* et *é* représentent un « é ».

La mise au point du langage de balisage HTML, sous-ensemble de SGML utilisé pour le codage des documents Web en 1991 a été l'occasion de standardiser la définition des entités. Le e avec accent aigü sera représenté par l'entité *é*, le a avec accent grave par *à*, etc.

À la même époque est mis au point le standard Unicode ambitionnant de définir une seule table de codage pour tous les alphabets utilisés à travers le monde. La version 2.0, sortie en 1996 à la suite des travaux du comité ISO 10646 (Haralambous, 2004) permet de représenter des caractères sur quatre octets, ce qui dépasse le million de possibilités.

XML a donc choisi un encodage Unicode, l'UTF-8, comme encodage par défaut. Il n'est alors plus nécessaire d'utiliser des entités caractères pour les caractères qui ne sont pas inclus dans la table ASCII. Ils peuvent être notés directement dans le document.

La norme XML est immédiatement utilisée pour représenter des dictionnaires. Tim Bray, co-rédacteur de son cahier des charges, s'est directement inspiré de son travail sur l'informatisation de l'Oxford English Dictionary entre 1987 et 1989.

Article au format SGML	Article au format XML
`<a><b>`**`Lexicologie`**`<i>`*`subst. f&ea.m.`* `<p>&Ea.tude scientifique du lexique. <i>`*`L'objet de la lexicologie est une th&ea.orie compr&ea.hensive du fait lexical, tant au niveau des structures (lexique, vocabulaires) que des unit&ea.s (mot, idiome).`*`</i> (REY, <i>`*`Le Lexique : images et mod&eg.les`*`</i>, Paris, Colin, 1977, p. 159).</a>`	`<a><b>`**`Lexicologie`**`</b><i>`*`subst. fém.`*`</i> <p>Étude scientifique du lexique. <i>`*`L'objet de la lexicologie est une théorie compréhensive du fait lexical, tant au niveau des structures (lexique, vocabulaires) que des unités (mot, idiome).`*`</i> (REY, <i>`*`Le Lexique : images et modèles`*`</i>, Paris, Colin, 1977, p. 159).</p></a>`

Figure 2. Exemple d'article au format SGML et au format XML[7]

[5]W3C : World Wide Web Consortium.
[6]ASCII : American Standard Code for Information Interchange.
[7]Le texte balisé n'est pas stylé ; les stylages en gras et italique ont été ajoutés pour améliorer la lisibilité du texte.

1.3. Séparation du fond et de la forme : sémantique et feuilles de style

L'arrivée de XML s'accompagne d'un changement de rôle des documents électroniques. Destinés jusqu'ici exclusivement à l'impression, ils vont pouvoir également servir d'autres buts comme l'affichage direct à l'écran. Il est alors possible et souhaitable de séparer la forme (représentation graphique du texte) du fond (texte et structuration des articles).

Les dictionnaires, qui étaient jusqu'alors représentés de manière implicite avec principalement des informations de style (ex : texte en gras pour la vedette), vont être représentés avec une structure explicite associée à des informations sémantiques. Les informations de mise en page et de style seront stockées de manière séparée.

Convertir un dictionnaire éditorial représenté avec des informations de style vers une représentation standardisée nécessite donc de rendre explicite la structure implicite des articles en indiquant la sémantique de chaque information (figure 2). Ainsi, lorsque la structure est explicite, la vedette peut-elle être repérée par l'élément <*mot-vedette*> et, dans les indications de style stockées à part, il est indiqué que les vedettes doivent être affichés en gras.

Article avec structure implicite	Article avec structure explicite
`<a>` `  <b>`**`Lexicologie`**`</b>` `  <i>subst. fém.</i> ` `  <p>Étude scientifique du lexique.<i>L'objet de la lexicologie est une théorie compréhensive du fait lexical, tant au niveau des structures (lexique, vocabulaires) que des unités (mot, idiome).</i>` `  (REY, <i>Le Lexique : images et modèles</i>, Paris, Colin, 1977, p. 159).` `  </p>` `</a>`	`<article id="a23301">` `  <bloc-vedette>` `  <mot-vedette>`**`Lexicologie`**`</mot-vedette>` `  <grammaire>subst. fém.</grammaire>` `  </bloc-vedette>` `  <bloc-sens>` `  <définition>Étude scientifique du lexique.` `  </définition>` `  <exemple>L'objet de la lexicologie est une théorie compréhensive du fait lexical, tant au niveau des structures (lexique, vocabulaires) que des unités (mot, idiome).</exemple>` `  <ref-exemple>` `  <auteur>REY</auteur>` `  <œuvre>Le Lexique : images et modèles<œuvre>` `  <réference>Paris, Colin, 1977, p. 159).</réference>` `  </ref-exemple>` `  </bloc-sens>` `</article>`

Figure 3. Marquage explicite des informations d'un article

Cette séparation du fond et de la forme apporte de grandes améliorations.

En premier lieu, les informations étant repérées explicitement, il est aisé d'extraire certaines d'entre elles ou de leur appliquer des traitement spécifiques. Ainsi, l'exemple d'usage de l'article balisé explicitement de la figure 3 est le texte encadré par les balises <*exemple*>. Cette extraction est bien plus complexe lorsque l'article est balisé implicitement puisque l'information recherchée est repérée par des balises qui ne lui sont pas spécifiques. Ainsi, dans l'article balisé implicitement de la figure 2, la balise indiquant l'italique étant présente pour plusieurs types d'informations (la catégorie lexicale d'une part et l'exemple d'usage d'autre part) il est difficile pour une machine de distinguer ces deux types d'information.

En second lieu, comme les traitements sont devenus plus simples à réaliser, il devient possible de créer facilement plusieurs traitements. Cette capacité est surtout exploitée en ce qui concerne l'affichage : plusieurs mises en pages et plusieurs styles différents peuvent être associés au même fichier contenant les informations à afficher. Le langage informatique CSS[8] a été développé spécialement

[8]CSS : Cascading Style Sheets (feuilles de style en cascade).

pour gérer l'affichage tandis que XSLT[9] vise la fonctionnalité plus large de conversion d'un document XML vers un autre format (y compris XML, ou encore des formats spécifiquement dédiés à l'affichage tels que HTML) à l'aide de transformations structurelles.

L'exemple de la figure 4 est issu d'une feuille CSS. Il permet de spécifier des informations de style destinées à l'affichage. La première règle affiche le mot-vedette en gras, la deuxième affiche la classe grammaticale, les exemples et le nom de l'œuvre en italique.

```
mot-vedette {
  font-weight: bold;
}
grammaire, exemple, œuvre {
  font-style: italic;
}
```

Figure 4. Feuille CSS permettant la mise en page d'un article

L'exemple de la figure 5, issu d'une feuille XSLT, permet d'afficher le numéro des blocs de sens et d'ajouter des parenthèses autour des références bibliographiques d'un l'exemple d'usage.

```
<xsl:template match="bloc-sens-num">
       <br/><xsl:apply-templates select="@num"/>
       <xsl:apply-templates/>
</xsl:template>

<xsl:template match="ref-exemple"><xsl:text> (</xsl:text><xsl:apply-templates/>
<xsl:text>) </xsl:text></xsl:template>
```

Figure 5. Feuille XSLT permettant la mise en page d'un article

2. Normes de représentation de dictionnaires

Dans cette partie, nous discutons des normes ayant pour but de standardiser les structures des articles de dictionnaires.

2.1. La Text Encoding Initiative (TEI)

2.1.1. Présentation

La TEI[10] est pilotée par un consortium regroupant des organismes principalement étatsuniens (ACH[11], ACL[12], ALLC[13]) et européens (ATILF[14], LORIA[15], INIST[16]) financés par des fonds de recherche publics.

L'objectif de la TEI est de définir un format d'échange, de création et de stockage de textes annotés à l'aide d'un jeu d'éléments standardisés. L'organisation est modulaire. Il existe un ensemble commun d'éléments "core tag set" complété par huit ensembles de base pour les différents genres de textes : prose, poésie, drame, parole, dictionnaires, terminologie, base générale et mixé. Des modules additionnels viennent enrichir les descriptions : corpus, alignement, etc.

Le standard TEI est un ensemble de directives (guidelines) pour l'encodage d'un texte. Les débuts des travaux en 1986 sont fondés sur SGML. Plusieurs versions se succèdent : la première version complète est P1 (1990) ; P3 (1994) constitue un standard *de facto* pour les corpus ; P4 (2002) est

[9]XSLT : Extensible Stylesheet Language Transformations. XSLT est un langage de transformation d'arbres XML.
[10]`www.tei-c.org`
[11]ACH : The Association for Computers and the Humanities.
[12]ACL : The Association for Computational Linguistics.
[13]ALLC : The Association for Literary and Linguistic Computing.
[14]ATILF : Laboratoire Analyse et Traitement Informatique de la Langue Française.
[15]LORIA : Laboratoire lorrain de recherche en informatique et ses applications.
[16]INIST : Institut de l'Information Scientifique et Technique.

compatible avec XML ; P5 (2007) introduit la notion de version intermédiaire tous les six mois. Nous nous focalisons ci-dessous sur cette dernière version dont les directives sont structurées en vingt-trois chapitres.

2.1.2. En-tête

Chaque document encodé selon le standard TEI doit comporter un en-tête ayant pour rôles l'identification du document, la caractérisation globale de son contenu, la mémoire électronique de ses variations, et la documentation fine de la stratégie d'encodage qui a été appliquée au texte. L'en-tête est donc nécessaire à la réutilisation et à la maintenance du texte. Le chapitre 2 de la version P5 y est entièrement consacré (Text Encoding and Interchange, 2004).[17]

2.1.3. Dictionnaire

Le dictionnaire peut apparaître comme un genre sur mesure pour être décrit par un langage structuré et pour être consulté en ligne du fait de sa structuration en articles théoriquement construits selon un même modèle. Dans les faits, des régularités structurelles existent mais il y a aussi beaucoup de variations avec des régularités locales comme le fait qu'un composant d'identification des items traités précèdent le(s) composant(s) de traitement. Si nous considérons un dictionnaire donné, la régularité structurelle prévaut le plus souvent, même s'il peut y avoir des séquences de composants d'articles optionnels et répétables, et non ordonnés (parfois codés en tant que contenus mixtes XML). Il s'ensuit que n'importe quel élément peut apparaître presque n'importe où dans un article de dictionnaire.

Le chapitre 9 du standard TEI est centré sur les dictionnaires. Il est composé de constituants génériques (<*front*> = texte préliminaire, <*body*> = corps du document, <*back*> = texte postliminaire <*div*>, <*div0*>, <*div1*> = divisions du texte, <*entry*> = entrée structurée, <*entryFree*> = entrée libre, <*superEntry*> = super-entrée), de données structurantes (<*form*> = informations sur une forme, <*hom*> = homographe, <*sense*> = sens de mot, etc.) et de données informatives (<*def*> = définition, <*pos*> = catégorie grammaticale, <*usg*> = usage).

La figure 6 montre un exemple de structure d'article décrite à l'aide d'éléments de la TEI. Les pointillés indiquent des éléments optionnels comme <number>. En bas de certains éléments, sont indiquées les cardinalités : l'élément <cit> doit être présent au moins une fois et peut être répété à l'infini.

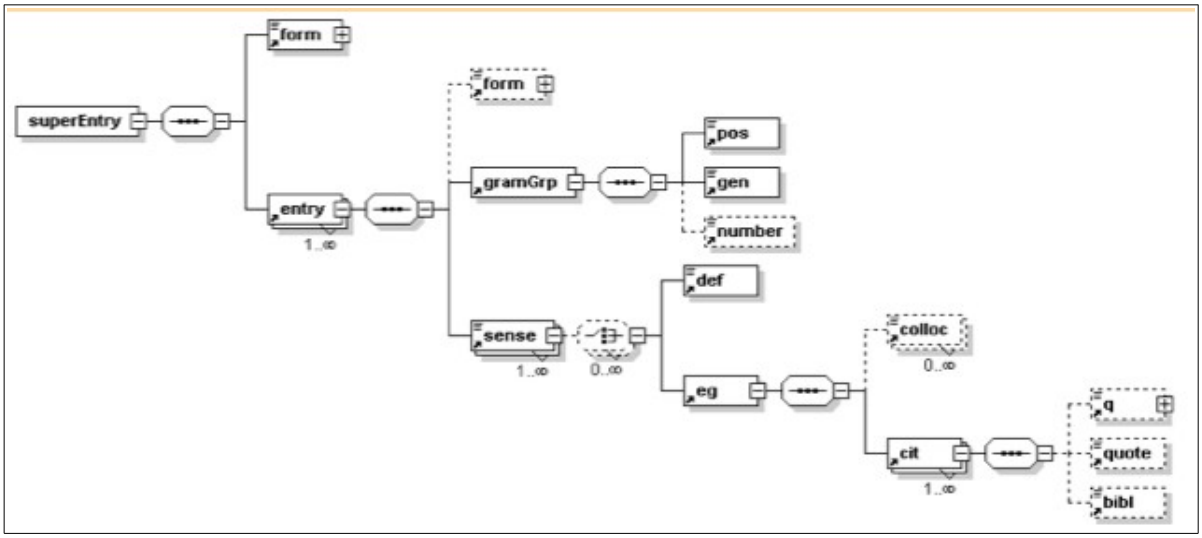


Figure 6. *Article décrit à l'aide d'éléments de la TEI*

La figure 7 reprend l'exemple d'entrée précédent et l'encode selon la TEI.

[17] Voir par exemple l'en-tête du lexique Morphalou de formes fléchies du français : http://www.cnrtl.fr/lexiques/morphalou/

```
<entry>
  <form>
    <orth>lexicologie</orth>
  </form>
  <gramGrp>
    <pos>subst.</pos>
    <gen>fém.</gen>
  </gramGrp>
  <sense n="1">
    <def>Étude scientifique du lexique.</def>
    <eg>
      <cit type="example">
        <quote>L'objet de la lexicologie est une théorie compréhensive du fait
          lexical, tant au niveau des structures (lexique, vocabulaires) que des
          unités (mot, idiome).</quote>
      </cit>
      <bibl>
        <author>REY</author>, <title>Le Lexique : images et modèles</title>.
        <pubPlace>Paris</pubPlace>, <publisher>Colin</publisher>,
        <date>1977</date>, <biblScope type="pp">p.159</biblScope>.
      </bibl>
    </eg>
  </sense>
</entry>
```

Figure 7. Article *lexicologie* encodé selon la TEI

Pour faire face au problème de structuration des articles, la TEI propose une solution binaire : un article peut être représenté par un élément *<entry>* dont la structure est rigide et très codifiée ; la représentation par un élément *<entryFree>* peut lui être préférée car elle admet d'insérer dans l'article n'importe quels éléments et dans n'importe quel ordre.

Dans la pratique, l'élément *<entry>* se révèle trop contraignant à utiliser. Par conséquent, les lexicographes préfèrent utiliser l'élément *<entryFree>*, mais celui-ci est trop lâche pour permettre une réelle standardisation et des échanges de données encodées avec la TEI.

2.1.4. Utilisation de la TEI

La TEI a rencontré un réel succès pour le codage des corpus. Il n'en est malheureusement pas de même pour les dictionnaires. La solution proposée (dichotomie entre les éléments *<entry>* et *<entryFree>*) n'est pas satisfaisante. De plus, les données des corpus ne sont pas la propriété des éditeurs au même titre que les articles de dictionnaires que leurs lexicographes ont rédigés. Les collaborations et adoptions de standards peuvent être plus aisées les concernant (même si les éditeurs britanniques se sont montrés plus collaboratifs en la matière que les éditeurs français). Il s'ensuit que dans la pratique, très peu de dictionnaires sont encodés avec la TEI. Il s'agit plutôt de lexiques non commerciaux (CJKV English dictionary, dictionary of buddhism[18]).

Un autre facteur limitant est le manque d'outils de description de macrostructures[19] pour les bases lexicales plurilingues. De telles bases comprennent plusieurs volumes logiques, chaque volume regroupant des articles d'une même langue. Il est alors nécessaire de définir précisément les volumes et les liens entre ceux-ci qui peuvent être parfois très complexes, par exemple dans le cas de structures pivot à plusieurs étages, de réseaux lexicaux, etc.).

2.2. Lexical Markup Framework

Dans le domaine du TALN, les dictionnaires électroniques sont vus comme des ressources au même titre que les corpus. Pour les désigner, nous utiliserons par la suite, le terme de *ressource lexicale* qui regroupe les lexiques, les dictionnaires et les bases lexicales. La nécessité de partager

[18] http://www.buddhism-dict.net/

[19] Nous définissons la macrostructure comme l'organisation des volumes d'une base lexicale et liens inter-volumes.

ces ressources a abouti en 2003 au projet de standardisation Lexical Markup Framework (Romary, Salmon-Alt & Francopoulo, 2004) qui consiste en un méta-modèle séparant les parties lexicale, grammaticale et sémantique (voir figure 8). La classe principale est une ressource lexicale *Lexical Resource*. Elle contient une classe décrivant la méta-information sur le lexique *Global Information* et un ou plusieurs lexiques *Lexicon*. Le lexique contient une ou plusieurs entrées lexicales *Lexical Entry*. Une entrée lexicale contient une ou plusieurs formes de l'entrée *Form* ainsi qu'un ou plusieurs sens *Sense*. Les formes contiennent des variantes orthographiques *Form representation*. Les sens peuvent à leur tour contenir des sens de manière récursive. Les sens peuvent contenir des définitions *Definition* qui contiennent des contenus textuels *Text Representation* et des descriptions narratives *Statement*. La classe *representation* permet de faire un lien entre les variantes orthographiques *Form Representation* et leurs occurrences dans un texte *Text Representation*. Ce méta-modèle est devenu une norme ISO sous le numéro 24613:2008 en novembre 2008 (Francopoulo, Bel, George, Calzolari, Monachini & Soria, 2009).

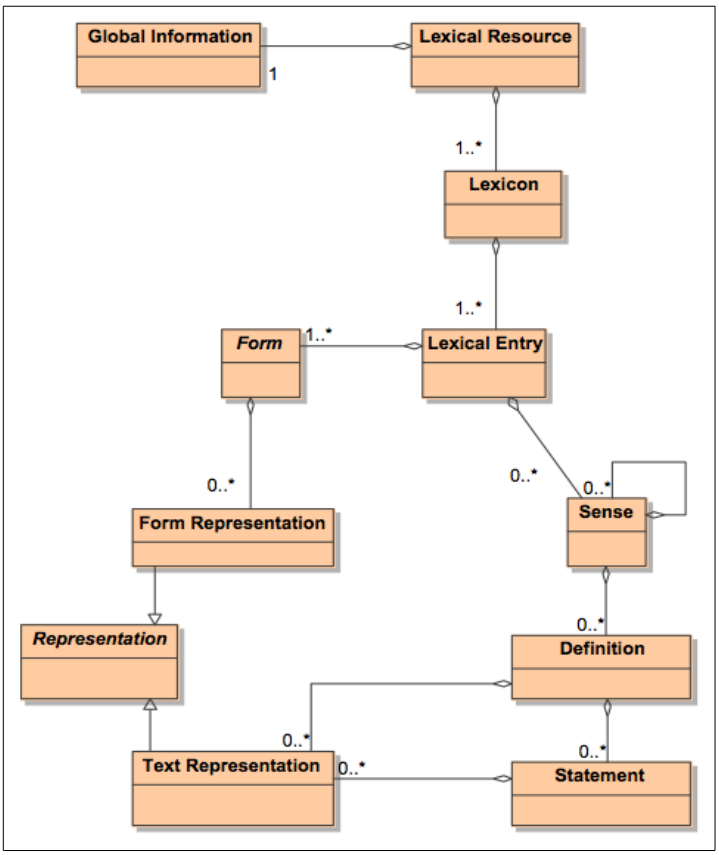


Figure 8. Méta-modèle de base de LMF

Ce méta-modèle, véritable partie centrale de la norme, peut être enrichi par des extensions qui doivent être des sous-classes des classes existantes dans le modèle de base. Il existe des extensions pour la morphologie, la syntaxe, la sémantique, etc. De nouvelles extensions pourront être développées par la suite pour répondre à des besoins spécifiques.

Les données peuvent être de deux types : soit une chaîne de caractères Unicode, soit une valeur choisie dans une liste de catégories provenant de la spécification MARTIF, (norme ISO 12620:2009).

La partie normative s'arrête à ce stade, en particulier elle ne définit aucun nom d'élément à utiliser. Il existe bien sûr des exemples de documents XML dans la partie informative de la norme mais il n'est en aucun cas obligatoire d'utiliser ces noms. Selon nous, c'est précisément l'intérêt de la norme LMF de permettre la conception d'un dictionnaire respectant la norme LMF tout en utilisant ses propres éléments. Il se peut même que, tout comme Monsieur Jourdain faisait de la prose sans le sa-

voir, un certain nombre de ressources lexicales respectent déjà le format LMF même si elles ont été conçues avant l'établissement de cette norme.

Afin de faciliter l'échange de données, il conviendra toutefois de proposer systématiquement, en accompagnement d'une ressource, un programme permettant de convertir la ressource avec ses éléments propres vers les éléments utilisés dans la partie informative de LMF et vice-versa. Un programme XSLT est par exemple tout à fait approprié pour ce genre de manipulation.

En conclusion, nous conseillons au lecteur d'utiliser la TEI uniquement pour l'en-tête de son dictionnaire car celui-ci est très détaillé et son usage est plus répandu que le chapitre concernant les dictionnaires, notamment pour les corpus. Pour le contenu, il est préférable d'utiliser une structure respectant le standard LMF. L'en-tête de la TEI correspondra alors au contenu de la classe *Global Information* de la norme LMF..

3. Des ressources lexicales pour les langues peu dotées

3.1. Problématique des langues peu dotées

Quoiqu'un inventaire précis soit difficile à réaliser, il existerait actuellement environ 6 000 langues parlées par les êtres humains. Parmi celles-ci 200 à 300 seulement sont écrites. Le passage de l'oral à l'écrit est complexe et ne peut se limiter à une simple transcription des sons entendus. Il est nécessaire de mener des études linguistiques afin de réaliser une description de la langue. Il s'agit de régler de multiples questions : déterminer le système de transcription qui sera utilisé[20] et, à l'intérieur de celui-ci, choisir les signes les plus adéquats, puis les règles orthographiques, syntaxiques, etc. Enfin, les langues sont plus ou moins bien dotées en ce qui concerne leur support par des outils informatiques : clavier adapté, correcteur orthographique, synthèse de la parole, traduction automatique, etc. Une classification fondée sur l'estimation de l'équipement d'une langue en outils et ressources informatiques détermine trois classes : les langues très bien dotées ou langues-τ (par exemple, l'anglais ou le français), les langues moyennement dotées ou langues-μ (par exemple le portugais ou le suédois), et les langues peu dotées ou langues-π (par exemple le bambara ou le kanouri) (Berment, 2004).

L'appellation langues peu dotées recouvre des situations contrastées. Nous citons ici trois conjonctures :

- il s'agit de la langue officielle d'un État, comme l'est l'irlandais (ou gaélique d'Irlande) en Irlande.

- il s'agit d'une langue sans statut officiel, devenue langue régionale : en France, par exemple, le basque ou le breton ; le ladin en Italie, le cornish (langue de Cornouailles) au Royaume-Uni.

- il s'agit d'une langue nationale d'un État dont la langue officielle (celle des actes officiels, de l'enseignement, des lois) est différente et souvent issue d'un autre État anciennement colonisateur (Calvet, 1996). C'est le cas par exemple, du kanouri au Niger, où il y a dix autres langues nationales et où la langue officielle est le français.

Nous focalisons notre attention sur les langues peu dotées dont le contexte socio-économique est caractérisé par des ressources réduites : d'une part il y a peu de de linguistes ayant comme langue maternelle une langue peu dotée et exerçant leur activité professionnelle sur cette langue, d'autre part le budget consacré au développement de ressources linguistiques est faible. C'est le cas des pays du Sahel où le budget de l'État est insuffisant, notamment en ce qui concerne l'éducation, et où le taux d'analphabétisme est très élevé. Les investissements de l'État consacrés à la planification linguistique et, en particulier, au développement de ressources linguistiques électroniques, sont par conséquent très limités. Les quelques travaux qui sont menés sont caractérisés par leur discontinuité

[20]La situation politique ou religieuse influence parfois le choix du système de transcription, par exemple quand il s'agit de trancher entre le système alphabétique occidental ou arabe, ou un alphabet local (Calvet 1987).

dans le temps et leur dissémination spatiale qui nuisent à leur pérennisation (Streiter, Scannell & Stuflesser, 2006).

Développer *ex nihilo* des ressources lexicales nécessite des budgets importants, des personnes qualifiées et disponibles, et la capacité à mener un projet durant plusieurs années, conditions qui ne peuvent être réunies dans de nombreux pays. Cependant, il existe parfois des dictionnaires éditoriaux (souvent bilingues) qui peuvent être exploités[21] pour réaliser, en quelques semaines et à faible coût, une première version d'une ressource électronique. La collecte des fichiers informatiques les constituant constitue un préalable important. Toutefois, en leur absence, le dictionnaire peut être saisi à nouveau quand seul un exemplaire imprimé a pu être collecté.

Quel que soit son format (électronique ou imprimé) un dictionnaire représente une somme de connaissances importantes et qui peut être récupérée. Dans tous les cas les auteurs ou l'éditeur du dictionnaire initial doivent être associés au projet afin d'obtenir leur accord pour que la ressource lexicale qui sera produite puisse être largement diffusé sous forme électronique et visible sur internet.

La méthodologie de conversion que nous présentons fait intervenir des linguistes spécialistes des langues concernées et des informaticiens connaissant le traitement automatique des langues. Elle prend en compte la limitation des ressources économiques en limitant le temps de travail des personnes et en privilégiant l'utilisation de logiciels gratuits.

3.2. Dictionnaires rédigés par un seul auteur

Certains dictionnaires sont l'œuvre d'un seul auteur. Celui-ci y a en général consacré part importante de sa vie.

Nombre de ces dictionnaires sont bilingues car leur auteur vise à faire connaître une langue qu'il a appris à maîtriser alors qu'il a pour langue maternelle une autre langue. C'est le cas par exemple du dictionnaire français-khmer de Denis Richer (Association Pays perdu). Beaucoup ont été rédigés par des religieux qui à l'origine s'étaient installés pour évangéliser les peuples de territoires colonisés (pères blancs en Afrique, jésuites portugais en Asie). Nous avons par exemple travaillé sur le dictionnaire bambara-français du Père Charles Bailleul (Bailleul, 1996).

Il existe aussi des dictionnaires élaborés par des personnes lettrées, souvent linguistes, désireuses de mettre leur savoir au service de leur langue maternelle comme le dictionnaire élémentaire hausa-français d'Abdou Minjinguini (2003) ou encore le dictionnaire zarma de Issoufi Alzouma Umarou (1997) (qui, fait rare, est monolingue).

Les plus récents de ces dictionnaires sont généralement rédigés entièrement avec un logiciel de traitement de texte (Word, WordPerfect). Leur structure évolue au fil de leur élaboration, sans nécessairement respecter une standardisation. En ce qui concerne les contenus, des listes de valeurs *a priori* fermées comme les catégories grammaticales peuvent fluctuer ; les abréviations connaissent également des variations, certaines peuvent même être absentes de la liste des abréviations située en début d'ouvrage. Ces dictionnaires connaissent également beaucoup d'instabilité dans la structuration des entrées surtout si celles-ci sont complexes.

3.3. Dictionnaires construits dans le cadre de projets

Les dictionnaires construits dans le cadre de projets rassemblant un groupe de plusieurs personnes sont généralement le fruit d'un travail de réflexion en amont sur la structure utilisée et la définition des listes fermées de valeurs comme les classes grammaticales.

[21]Dans la mesure où les auteurs ont donné leur accord.

Les dictionnaires de ce type les plus récents sont construits avec des outils de lexicographie comme Linguist Shoebox/Toolbox[22] et FieldWorks Language Explorer (FLEx)[23] de la Société Internationale de Linguistique (qui construit des dictionnaires là aussi dans un but évangélique) ou TshwaneLex (TLex)[24].

Bien souvent, même si ces outils sont capables d'exporter leurs contenus vers une structure XML, cette opération n'a pas été effectuée et les fichiers produits par les outils lexicographiques ne sont pas disponibles. Seuls sont utilisables les fichiers Word destinés à l'impression.

Nous avons par exemple travaillé sur :

- les dictionnaires en langues nationales du Niger du projet DiLAF de conversion de dictionnaires éditoriaux bilingues langue africaine-français (Enguehard, Kane, Mangeot, Modi & Sanogo, 2012) ;

- Le dictionnaire FeM français-anglais-malais (Gut et al., 1996) et ses dérivés (FeV, pour le vietnamien, FeT pour le thaï).

4. Difficultés techniques

4.1. Des langues écrites mais peu standardisées

Bien que le caractère peu doté d'une langue ait été défini uniquement en référence à son équipement en outils et ressources informatiques, il s'inscrit souvent dans un contexte de rareté des connaissances linguistiques : les études sur la langue sont peu nombreuses, elles sont peu accessibles car elles ne sont pas publiées dans des revues ou des actes de conférence, et elles ne sont pas disponibles sur internet. De plus, ces langues sont également peu présentes à l'école, que ce soit comme matière d'étude ou comme langue d'enseignement. Quelques exceptions méritent cependant d'être citées :

Au Niger, des écoles expérimentales ont été créées dans les années 1980. L'enseignement y est entièrement dispensé dans une langue nationale pendant la première moitié du cycle primaire puis, pendant la seconde moitié de ce cycle, le français fait son apparition et la langue nationale est étudiée en tant que matière. La dernière année, l'enseignement de déroule uniquement en français. Il en sera de même pendant le reste de la scolarité : au collège, au lycée et dans l'enseignement supérieur (Programme Décennal du Développement de l'Éducation du Niger, 2003).

En Équateur, le peuple shuar (que nous appelons improprement jivaro) s'est structuré en une Fédération des Centres Shuars en 1964. Dans les années 1970 cette fédération a organisé, entre autres, la fondation d'écoles primaires dans les villages avec un soutien par des programmes radiophoniques. Ces écoles sont bilingues : l'enseignement y est dispensé quasiment intégralement en shuar les deux premières années pour évoluer vers un enseignement à parité en shuar et en espagnol en dernière année (Calvet, 1987).

Au-delà de la réussite de ces approches quant à l'alphabétisation des enfants, la création de cursus scolaires favorise la rédaction et l'édition de quelques manuels pédagogiques qui constituent des corpus de textes écrits par des spécialistes des langues, parfois linguistes, ayant une bonne connaissance de la langue de rédaction[25]. De tels corpus constituent des ressources électroniques susceptibles d'être exploitées pour constituer des ressources lexicales. Leur taille reste cependant réduite.

D'autres textes en langues nationales voient le jour. Ils sont écrits par des journalistes, des auteurs de contes, de romans, etc., n'ayant la plupart du temps aucun accès à des ressources linguistiques. Par conséquent, ces textes sont écrits dans une langue peu standardisée présentant en particulier de

[22]www.sil.org/computing/toolbox/
[23]fieldworks.sil.org/flex/
[24]tshwanedje.com/tshwanelex/
[25]Au Niger, par exemple, les manuels (en cinq langues nationales) sont rédigés et édités par l'Institut National de Documentation, de Recherche et d'Animation Pédagogiques (INDRAP).

nombreuses variations orthographiques. Ces corpus ne peuvent donc pas être utilisés comme source de données pour construire automatiquement des ressources lexicales.

Dans ce contexte de dénuement, la récupération d'un dictionnaire éditorial constitue une première étape susceptible d'accélérer la constitution de ressources utilisables pour des applications de traitement automatique des langues naturelles. Dans certains dictionnaires, l'orthographe des mots est standardisée et est conforme aux études linguistiques ; dans d'autres, comme (Bailleul, 1996), les variantes sont explicitement signalées et situées géographiquement tandis que l'orthographe officielle est signalée en sus des graphies usuelles. De plus, les définitions, et les exemples d'usage constituent un corpus de phrases exploitable et de nombreuses entrées sont accompagnées d'informations morphologiques permettant de calculer les différentes formes d'une même entrée.

4.2. Des caractères spéciaux

Le développement d'outils de traitements automatiques d'une langue sous sa forme écrite nécessite (même si ce n'est pas suffisant) que tous les caractères soient codés de manière identique dans tous les textes, ce codage devant être partagé par tous les ordinateurs. Les codages élaborés depuis les années 1960 pour la table ASCII et les années 1970 pour les tables adaptées aux langues non anglophones ont constitué un compromis entre cet objectif et la taille mémoire réduite des ordinateurs de cette époque. Au tournant des années 1990, les progrès technologiques quant au stockage d'informations de plus en plus volumineuses ont permis d'envisager d'associer un code unique à chacun des caractères de toutes les écritures de langues : c'est ce que vise le standard Unicode (Haralambous, 2004).

Définis dans les années 1960, donc bien avant Unicode, les alphabets de la plupart des langues nationales africaines utilisent des caractères spéciaux qui étaient absents des tables de caractères de l'époque. Par exemple, le caractère b crossé (ɓ, Ɓ) apparaît dans l'alphabet haoussa tandis que le caractère n vélaire voisé (ŋ, Ŋ) figure dans les alphabets bambara, tamajaq et soŋay-zarma.

Constituées avant tout dans un but d'édition et donc d'impression sur support papier, des polices permettant d'afficher ces caractères spéciaux ont alors été créées en redessinant le glyphe de certains caractères (Chanard & Popescu-Belis, 2001). Ces polices ont permis pendant des décennies l'édition de textes en langues nationales mais interdisent tout traitement automatique des langues portant sur les textes édités (Enguehard, 2009). L'habitude de les utiliser s'étant installée, les fichiers sources des dictionnaires au format initial utilisent de telles polices. Il est donc nécessaire de les convertir vers Unicode afin que le codage des caractères respecte les standards internationaux.

Identifier les caractères à convertir est une tâche difficile car, d'une part, les dictionnaires, de taille importante (plusieurs milliers d'entrées), ne peuvent donner lieu à une relecture exhaustive et, d'autre part, il est parfois fait usage de plusieurs polices de caractères au sein du même document.

Établir les caractères de remplacement peut s'avérer délicat si les polices originales ne sont pas disponibles, ce qui est la situation la plus courante. Dans ce cas, il est préférable de disposer d'une version imprimée afin d'être certain des conversions à établir. Cette étape peut devenir plus subtile quand un même caractère est utilisé pour afficher des glyphes différents. Par exemple l'esperluette & est redessinée comme le t avec point suscrit ṭ de l'alphabet tamajaq dans la police 'Albasa Tamjq', comme le d crossé ɗ de l'alphabet haoussa dans les polices 'AlbasaRockwellhau' et 'Hausa' et comme le e ouvert ɛ de l'alphabet bambara dans les polices 'Times New bambara' et 'Arial Bambara'. Il peut arriver aussi que des polices différentes portent le même noms. Enfin, un même caractère peut être utilisé, au sein du même document, pour afficher des glyphes différents. Par exemple dans le dictionnaire bilingue tamajaq-français (Programme de soutien à l'éducation de base, 2007), le caractère p minuscule (U+0070) a été utilisé comme tel dans les parties des entrées rédigées en français, et redessiné comme un schwa ə dans la police 'Tamajaq Literacy2 TT20.4 SILSop' pour les parties en tamajaq .

Les caractères avec signe(s) diacritique(s), souvent utilisés pour noter la tonalisation, doivent parfois être convertis car leurs différents codages ne sont pas identifiés comme identiques par tous les logiciels. Par exemple, la lettre u minuscule avec accent aigu existe en deux codages, "U+0075 U+0301" et "U+00FA"[26]. Comme nous souhaitons que les ressources électroniques produites soient utilisables par des outils de TALN que nous ne connaissons pas *a priori*, il est nécessaire d'uniformiser les codages.

4.3. Caractères, alphabets et Unicode

Le standard Unicode vise à représenter tous les caractères de toutes les écritures des langues (Desgraupes, 2005). Bien qu'il évolue à chaque nouvelle version (la version 6.1 code 110 116 caractères) (Unicode, 2012), certains alphabets restent encore incomplets, en particulier ceux des langues peu dotées, car introduire de nouveaux caractères nécessite des moyens humains et financiers qui leur font défaut. Voici quelques exemples de lacunes que nous avons relevés.

4.3.1. Digraphes

Les digraphes notent un seul son mais sont graphiquement composés de deux caractères. Leur usage modifie l'ordre du tri lexicographique. Ainsi, en haoussa et en kanouri, le digraphe sh est situé après la lettre s[27]. Donc le verbe *sha* « boire » est situé après le mot *suya* « frite » dans un dictionnaire haoussa, et le nom *shadda* « basin » est situé après le verbe *suwuttu* « dénouer » dans un dictionnaire kanouri.

Ces subtilités peuvent être difficile à traiter au niveau logiciel. Deux solutions peuvent être envisagées :

1 – Les digraphes manquants peuvent être introduits en tant que signe dans le répertoire Unicode. Certains, utilisés par d'autres langues, y figurent déjà, parfois avec leur différentes casses. Par exemple, DZ (U+01F1), Dz (U+01F2), dz (U+01F3) sont utilisés en slovaque ; NJ (U+01CA), Nj (U+01CB), nj (U+01CC) en croate et pour transcrire la lettre Њ de l'alphabet cyrillique en serbe ; etc.

2 – Les alphabets sont modifiés afin de ne plus faire apparaître les digraphes, ce qui a pour conséquence de modifier l'ordre lexicographique . C'est le cas, par exemple, de l'alphabet français dans lequel le son /ʃ/ est noté par la suite des deux caractères c et h ; le même son est noté par la suite des deux caractères s et h en anglais. Cet aménagement présente l'inconvénient de rompre avec le principe de représenter chaque son par un caractère[28]. De plus, des difficultés politiques peuvent surgir puisqu'il s'agit de modifier la définition d'alphabets de langues nationales qui ont été fixés par décrets.

4.3.2. Caractères avec signes diacritiques

Certains des caractères portant des signes diacritiques figurent dans Unicode comme un unique signe, d'autres ne peuvent être obtenus que par composition. Ainsi, la lettre j avec caron de l'alpha-

[26]En ce qui concerne les éditeurs de texte que nous avons utilisés, les fonctions 'Recherche' d'Open Office et Notepad++ considèrent les caractères correspondants aux deux codages énoncés comme différents alors que celle de gedit les assimile à un même caractère.

[27] Ordre alphabétique de l'alphabet haoussa du Niger : a b ɓ c d ɗ e f fy g gw gy h i j k kw ky ƙ ƙw ƙy l m n o p r s sh t ts u w y y' z (République du Niger, 1999a).
Ordre lexicographique de l'alphabet kanouri du Niger : a b c d e ə f g h i j k l m n ny o p r ɾ s sh t u w y z (République du Niger, 1999b).
Neuf digraphes absents d'Unicode avec leurs trois casses apparaissent dans ces alphabets : fy, gw, gy, ky, kw, ƙy, ƙw, sh, ts.

[28]Ce principe a été adopté lors de réunions ayant pour but de fixer des alphabets communs à plusieurs langues transfrontalières d'Afrique. La première, en septembre 1978, organisée par l'UNESCO au CELTHO (Centre d'études linguistiques et historiques par tradition orale) à Niamey, crée l'« Alphabet africain de référence » fondé sur les conventions de l'IPA (International Phonetic Association) et de l'IAI (International African Institute).

bet tamajaq existe dans Unicode en tant que signe ǰ (U+1F0), mais sa forme majuscule doit être composée avec la lettre J et le signe caron (U+30C).

5. Processus de conversion des ressources lexicales dans un format standardisé

5.1. Différents niveaux de ressources

Les ressources lexicales peuvent être classées selon trois niveaux (Mangeot, 2001) :

— Le premier niveau (format original) est constitué de ressources exprimées dans un format original, généralement peu standardisé provenant par exemple d'un logiciel de traitement de texte au format .doc, ou de n'importe quel autre format différent de XML. Bien souvent, la structure des articles est implicite et les informations de mise en forme ne sont pas séparées de celles de la structuration. Toutefois, lorsque le document texte est issu d'un logiciel de lexicographie, comme Shoebox (Buseman, Buseman, Jordan & Coward, 2000) par exemple, les noms de style reprennent souvent les noms des champs qui avaient été définis dans l'outil (Lexeme, Section1, Part_of_speech, etc.). Il est alors relativement facile de reconstruire la structure d'origine.

— Le deuxième niveau comprend les mêmes ressources converties en un format XML conservant la structure originale (format passerelle). Les informations sont cette fois toutes marquées explicitement et la structure des articles est apparente.

— Le troisième niveau (format cible) est celui d'une nouvelle ressource dont la structure est définie en fonction des besoins et des objectifs du projet. Bien souvent, le choix du format cible est guidé par l'objectif de produire des ressources lexicales réutilisables pour des applications de TALN.

La conversion de chaque dictionnaire éditorial produira donc deux nouvelles ressources distinctes : une au format passerelle et une au format cible (voir Figure 9).

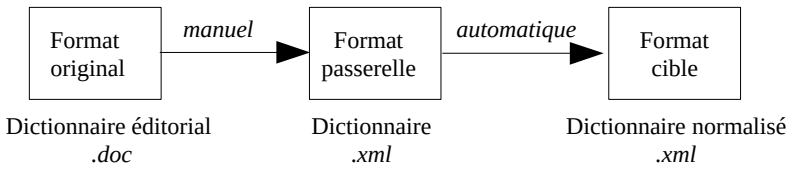


Figure 9. Processus de conversion

La ressource produite au format passerelle est la conversion la plus fidèle possible de la version initiale du dictionnaire dans un format XML. Il est préférable que cette conversion soit réalisée par des linguistes afin de corriger des éventuelles erreurs dans les données. Il s'agit de garder la structure originale des articles en explicitant les parties implicites. Lors de cette étape, il faut garder à l'esprit le format cible pour essayer de s'en approcher lorsque des choix doivent être effectués mais l'objectif n'est pas de respecter des standards à la lettre.

Concernant la microstructure, si, par exemple, dans le dictionnaire original, le choix a été fait de ne faire qu'un seul article pour toutes les classes grammaticales d'un mot, cette structuration n'est pas modifiée, même si elle ne correspond pas au standard du format cible.

Concernant la macrostructure, celle-ci ne devra pas être modifiée. Si le dictionnaire original est constitué d'un seul volume, le dictionnaire au format cible contiendra également un seul volume.

La ressource au format passerelle reste lisible et facile à appréhender par les linguistes chargés de corriger et d'enrichir le dictionnaire (par exemple, traduire les exemples dans une autre langue). Le format étant XML, il est également possible de la visionner directement dans un navigateur Web en y associant une feuille de style ou de l'importer sur une plate-forme de manipulation de ressources

lexicales telle que Jibiki (Mangeot, 2003) pour consultation et édition en ligne. Cette ressource servira de référence dans le futur si de nouvelles versions sont produites.

La ressource au format cible est produite à partir de celle au format passerelle. Les choix gouvernants la structuration peuvent s'éloigner du format initial pour aller vers un format cible défini par les acteurs du projet.

Pour le format cible, nous conseillons de se conformer au standard LMF afin de faciliter la réutilisation des données dans des projets de traitement automatique des langues. Cependant, ce format devra être, si nécessaire, adapté afin de prendre en compte la spécificité des langues.

Concernant la macrostructure du format cible, il est alors possible de modifier celle du format original pour en définir une nouvelle adaptée aux objectifs du projet. Par exemple, dans le cas d'un dictionnaire bilingue langue A → langue B ne contenant qu'un seul volume dans le format original, la macrostructure du format cible pourra être composée d'un volume langue A → langue B et du volume inverse langue B → langue A ou même d'une macrostructure pivot : un volume langue A, un volume pivot contenant les liens entre les deux langues et un volume langue B (Mangeot et al., 2003). Le volume inverse ainsi constitué sert dans un premier temps de « squelette » et doit être enrichi par la suite afin de constituer une ressource de qualité.

La ressource au format passerelle peut être ensuite convertie automatiquement dans le format cible en utilisant par exemple un programme XSLT car il s'agit dans les deux cas de fichiers XML.

5.2. Établissement des jeux d'éléments du format passerelle

Pour repérer les types d'information des articles, il faut choisir un jeu d'éléments. Se pose alors la question du choix de la langue utilisée pour les éléments. Le choix de l'anglais, langue internationale de la recherche, peut être privilégié. Mais, dans bien des cas, d'une part l'anglais n'est pas une langue présente dans les dictionnaires manipulés, et d'autre part, elle n'est pas maîtrisée par tous les linguistes travaillant sur le projet. Dans le cas de projets d'informatisation de langues peu dotées, il est important d'inciter les partenaires à utiliser les termes de leur langue pour définir le nom des éléments en langue source (ou langues nationales). Cette démarche peut éventuellement donner lieu à la création de nouveaux termes qui n'existaient pas dans ces langues et ainsi contribuer au transfert de connaissances et d'idées et, par conséquent, au développement scientifique et technologique (Diki-Kidiri, 2004). Dans le cas des langues peu dotées, d'un point de vue politique, il s'agit de s'éloigner d'une vision post-coloniale des statuts sociaux de ces langues et de contribuer à leur valorisation. Les partenaires linguistes peuvent donc définir des jeux d'éléments XML avec des noms qui, dans leur majorité, sont exprimés dans leur propre langue.

À titre d'exemple, voici les noms d'éléments choisies pour le dictionnaire kanouri-français (Programme de soutien à l'éducation de base, 2004) dans le projet DiLAF :

Nom d'élément en kanouri	**Équivalent français**
kalma	mot
bowodu	prononciation
naptu_curo_nahauyen	catégorie grammaticale
maana	signification
misal	exemple
kalakta	traduction en langue
maana_tiloa	synonyme
fərəm	antonyme
bowodu_gade	variante
mane	voir

Figure 10. *Jeu d'éléments choisis pour le dictionnaire kanouri-français du projet DiLAF*

6. Conversion vers le format passerelle à l'aide d'expressions régulières

L'ensemble des opérations décrites dans cette étape a été conçu afin de pouvoir être principalement mis en œuvre par des linguistes ou des lexicographes avec l'assistance d'informaticiens spécialistes en traitement automatique des langues.

6.1. Conversion du format texte vers XML

La figure 11 montre un extrait du dictionnaire kanouri-français (Programme de soutien à l'éducation de base, 2004) au format original .odt (ouvert avec OpenOffice) après conversion des caractères spéciaux . Les exemples suivants seront également extraits de ce dictionnaire.

abərwa[1]	[àbàrwà] cu. Kəska təngəri, kalu ngəwua dawulan tada cakkidə. *Kəryende kannua nangaro, abərwa cakkiwawo.* Fa.: ananas
abərwa[2]	[àbàrwà] cu. Tada abərwaye. *Abərwa bafiya, jauro lelea..* Fa.: fruit d'ananas

Figure 11. *Extrait du dictionnaire kanouri-français au format original*

Le format Open Document Format (ODF) ~~est un standard OASIS. La version 1.0 est également un standard ISO 26300:2006. La version 1.1 a été approuvée par OASIS le 2 février 2007. La version 1.2 est en cours de rédaction. Ce format est utilisé de manière native par les suites bureautiques de la famille de OpenOffice (StarOffice, NeoOffice, LibreOffice).~~ ~~Il~~ a le précieux avantage d'être fondé sur un format XML. Le titre de cette partie est donc un peu trompeur. Au lieu d'une conversion d'un format vers un autre, il s'agit de récupérer le contenu XML du document puis de le transformer pour obtenir un document XML dans le format souhaité.

Un document .odt au format ODF est en fait une archive zippée de plusieurs fichiers, dont le contenu textuel est balisé en XML. Ce contenu est stocké dans le fichier content.xml situé à la racine de cette archive. Pour récupérer ce fichier, il suffit de suivre quelques manipulations astucieuses. Sur MacOs, il est nécessaire de créer un dossier vide puis d'y copier le fichier .odt. Ensuite, il faut ouvrir un terminal puis exécuter la commande unzip sur le fichier .odt. Sur Windows, il faut changer l'extension du fichier .odt en .zip puis ouvrir l'archive .zip.

Le fichier content.xml peut alors être extrait de l'archive et renommé puis placé dans un autre répertoire. Il devient le fichier de base sur lequel se poursuivent les traitements.

L'étape suivante consiste à éditer ce fichier avec un éditeur de texte "brut". Cet éditeur devra *a minima* comporter des fonctionnalités de recherche et de remplacement supportant un langage d'expressions régulières et une coloration de syntaxe (pour faciliter le travail). Les outils gratuits en source ouverte comme Notepad++ font très bien l'affaire.

Le fichier étant constitué d'une seule ligne, la première manipulation est l'ajout des sauts de ligne. Si le document initial est bien rédigé, un article de dictionnaire correspond, dans la grande majorité des cas, à un paragraphe. Insérer un saut de ligne devant ou à la fin de chaque paragraphe permet donc de visualiser chaque article sur une seule ligne. Ce traitement est réalisé par l'expression régulière suivante[29].

```
s/<text:p/\r<text:p/g
```

L'en-tête contenant les informations spécifiques à OpenOffice peut maintenant être supprimé.

[29]Dans cet article, nous utilisons la syntaxe d'expressions régulières du langage Perl.

6.2. Marquage explicite des informations

Cette étape consiste à marquer explicitement toutes les parties d'information constituant les articles.

Dans le fichier d'origine, chaque type d'information (par exemple, la définition, les exemples d'usage, etc.) se distingue souvent des autres par l'usage d'un style différent. Si le fichier d'origine a été bien conçu, il est donc possible d'identifier un style correspondant à l'information. La figure 12 montre une partie de l'article *abəɾwa* « ananas » du dictionnaire kanouri-français au format ODF. Le style utilisé pour marquer la prononciation est "Phonetic_20_form". Ce fichier, conçu à l'origine avec le logiciel Shoebox, garde dans le nom du style, l'étiquette utilisée pour marquer l'information avec Shoebox. Pour d'autres dictionnaires, le nom du style est plus cryptique. Par exemple, pour le dictionnaire bambara-français du père Charles Bailleul, qui a été rédigé directement avec WordPerfect, les traductions françaises sont notées avec le style "T21". Il est cependant possible de se repérer en comparant avec le fichier d'origine.

```
<text:span text:style-name="Phonetic_20_form"><text:span text:style-
name="T7">[àbàdárò]</text:span> </text:span>
```

Figure 12. Extrait d'article (phonétique) au format ODF XML

Le marquage explicite des informations peut donc être réalisé par le remplacement judicieux du balisage ODF par un balisage conforme au jeu d'éléments provisoire précédemment définis. Le balisage ODF reste cependant complexe, et il serait délicat de mettre au point, pour chaque dictionnaire, une transformation XSLT. Nous avons choisi de le réaliser par une succession d'opérations de recherche et de remplacement utilisant des expressions régulières car ces opérations peuvent être réalisées par les partenaires linguistes[30]. Pour l'exemple de la figure 12, une première expression régulière supprime l'élément "T7" :

```
s/<text:span text:style-name="T7">([^<]+)<\/text:span>//g
```

La deuxième expression remplace l'élément de style "Phonetic_20_form" par "bowodu" :

```
s/<text:span text:style-name="Phonetic_20_form">([^<]+)  <\/
text:span>/<bowodu>$1<\/bowodu>/g
```

Le remplacement de tous les éléments aboutit au résultat de la figure 13. L'étape de marquage des informations est maintenant terminée.

```
<kalma>dole</kalma><bowodu>[dólè]</bowodu><naptu_curo_na-
hauyen>cu.</naptu_curo_nahauyen><maana>Karwu cəragəna bi wa-
jənama    tədidə.</maana><misal><version    tələm="ka">Kambi
nokkənidə, dole maararo ngaakke ingi falləkke kokki.</ver-
sion></version></misal><kalakta   tələm="fa">obligation</ka-
lakta>
```

Figure 13. Article converti avec des éléments de la version passerelle

Cette méthodologie simple à mettre en place convient bien aux dictionnaires construits par des projets (voir 1.4.3) qui utilisent généralement des logiciels spécifiques. Nous l'avons utilisée pour de nombreux dictionnaires tels que ceux du projet DiLAF ou le dictionnaire français-khmer de Denis Richer (Richer, 2007).

[30]Il s'agit aussi de favoriser le transfert de connaissances : les personnes ne connaissant pas les expressions régulières apprennet à les manipuler à l'occasion de ce projet.

6.3. Validation XML

Pour pouvoir utiliser des outils manipulant des fichiers XML, il faut que ceux-ci soient bien formés d'un point de vue de la syntaxe XML. Lors de l'étape précédente, les remplacements successifs ont été effectués sur un fichier au format ODF bien formé, aussi le résultat devrait-il être théoriquement bien formé. Il est apparu que, dans la pratique, ce n'est jamais le cas car les expressions régulières qui ont été successivement appliquées sont écrites par des linguistes récemment formés et donc peu aguerris à leur pratique. Il s'avère donc indispensable de vérifier que le fichier est bien formé. Cette étape permet de détecter certaines erreurs de balisage dues à l'étape précédente et de les corriger.

Un analyseur (parseur) XML est nécessaire pour vérifier la syntaxe du document. L'usage de logiciels gratuits étant privilégié, il est préférable d'utiliser un simple navigateur Web tel que FireFox. Celui-ci inclut un analyseur XML capable d'indiquer la localisation de la première erreur rencontrée dans le fichier. La figure 14 montre le résultat de l'analyse d'un fichier qui n'est pas bien formé (il manque le guillemet fermant de la valeur de l'attribut *@lamba*).

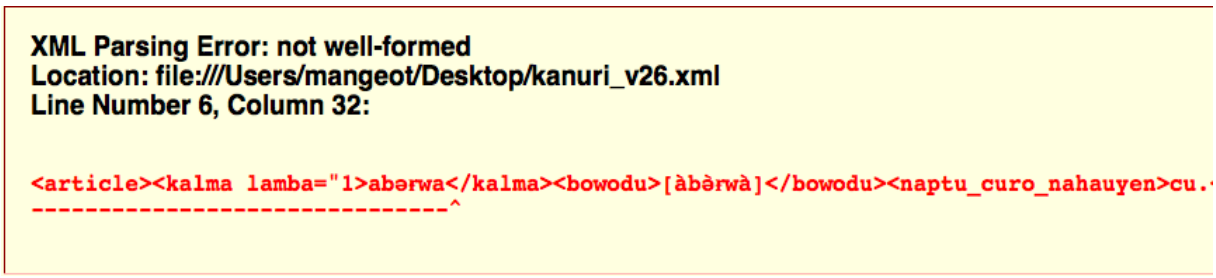


Figure 14. Affichage d'une erreur de syntaxe XML par Firefox

Une fois l'erreur localisée, il faut vérifier si elle se répète dans le fichier (ce qui est probable dans le cas d'une manipulation erronée lors d'une étape précédente). Si c'est le cas, il faut élaborer une expression régulière qui corrige l'erreur de manière systématique plutôt que d'effectuer des corrections au fur et à mesure du texte.

Le fichier XML est maintenant bien formé. Il est alors possible de le manipuler avec des outils XML.

6.4. Vérification des listes de valeurs

Pour un dictionnaire, la classe grammaticale prend sa valeur dans une liste fermée. L'étape de la vérification des informations prenant théoriquement leur valeur dans une liste fermée est importante car elle permet de détecter certaines erreurs introduites par des manipulations erronées lors des étapes précédentes ou déjà présentes dans le fichier d'origine avant conversion. Il s'agit aussi d'établir ou de corriger cette liste fermée qui, généralement, n'est pas connue ou comporte des erreurs.

Cette vérification systématique procède de manière destructrice et est donc opérée sur une copie du fichier à traiter. Il s'agit de ne garder que les valeurs à vérifier.

Dans l'exemple de la figure 13, l'expression suivante extrait la classe grammaticale balisée par <naptu_curo_nahauyen> :

```
s/^.*<naptu_curo_nahauyen>([^<]+)<\/.*$/$1/
```

La liste obtenue est ensuite triée par ordre alphabétique. Elle peut alors être scrutée pour détecter rapidement les irrégularités : si une valeur n'apparaît qu'une seule fois, il est fort probable que ce soit une erreur. Dans le dictionnaire kanouri-français, nous avons corrigé "kəld." en "kəl." (conjonction), "n." en "cu." (nom), "ono" en "cok." (idéophone), etc.

6.5. Corrections structurelles et sémantiques

Le fichier de travail étant bien formé, une feuille de style CSS peut être définie pour visualiser les données directement dans un navigateur. Un affichage compact et un style différent pour chaque type d'information permettent au linguiste de déceler les erreurs de structuration d'un article. Dans l'exemple de la figure 15, nous voyons qu'il manque l'exemple (en italique) pour l'entrée "acca ambe".

Le langage XSLT permet de modifier les données avant l'affichage. Il est alors possible, par exemple, d'ajouter comme identifiant unique d'un article, son mot-vedette puis, pour chaque renvoi ou synonyme de définir un lien hypertexte pointant vers l'article correspondant. Lorsque le linguiste parcourt le fichier, il peut alors cliquer sur les liens hypertextes (signalés par un soulignement) pour vérifier que les renvois et synonymes font également l'objet d'articles dans le dictionnaire. Dans l'exemple de la figure 15, l'entrée "acca" contient un renvoi vers l'entrée "hâjjà". Le mot est souligné, ce qui indique la présence d'un lien hypertexte.

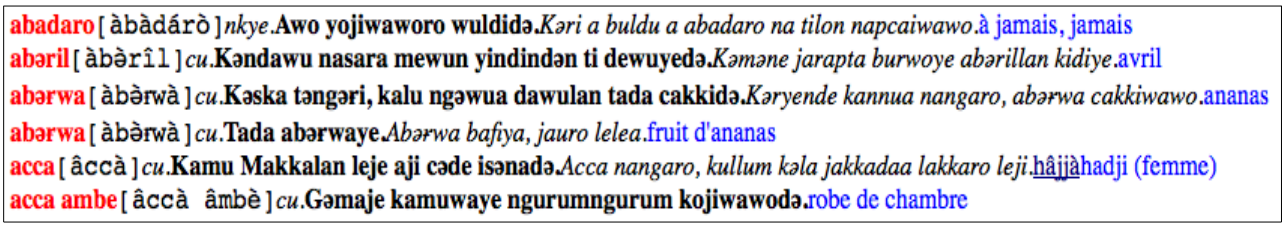

abadaro [àbàdárò] *nkye*.**Awo yojiwaworo wuldidə.***Kəri a buldu a abadaro na tilon napcaiwawo*.à jamais, jamais
abəril [àbərîl] *cu*.**Kəndawu nasara mewun yindindən ti dewuyedə.***Kəmənе jarapta burwoye abərillan kidiye*.avril
abərwa [àbərwà] *cu*.**Kəska təngəri, kalu ngəwua dawulan tada cakkidə.***Kəryende kannua nangaro, abərwa cakkiwawo*.ananas
abərwa [àbərwà] *cu*.**Tada abərwaye.***Abərwa bafiya, jauro lelea*.fruit d'ananas
acca [âccà] *cu*.**Kamu Makkalan leje aji cədе isənadə.***Acca nangaro, kullum kəla jakkadaa lakkaro leji*.hâjjàhadji (femme)
acca ambe [âccà âmbè] *cu*.**Gəmaje kamuwaye ngurumngurum kojiwawodə.**robe de chambre

Figure 15. Vue compacte dans un navigateur

Lorsque des programmes XSLT s'avèrent nécessaires il est indispensable que des informaticiens participent au projet de conversion.

Cette étape de visualisation des données est essentielle, non seulement pour détecter des erreurs, mais aussi parce qu'elle permet aux linguistes de comprendre les avantages de l'encodage des données en XML, en particulier le fait que différentes mises en formes (styles) peuvent être associées aux articles balisés. En apprenant des rudiments du langage CSS, ils peuvent alors modifier eux-mêmes les feuilles de style.

7. Conversion vers le format passerelle à l'aide d'outils spécialisés

Lorsque les dictionnaires sont susceptibles de contenir beaucoup d'erreurs de structuration ou que celle-ci est implicite et relâchée, ce qui est souvent le cas avec des dictionnaires destinés à l'impression et rédigés par un seul auteur, la conversion fondée sur des expressions régulières devient difficile. Il est alors possible d'utiliser des outils de récupération beaucoup plus puissants basés sur une grammaire de description de la structure explicite que l'on vise à récupérer. Des générateurs d'analyseurs lexicaux et syntaxiques robustes comme Lex et Yacc (ou Flex et Bison) peuvent être utilisés. Il existe également des outils plus évolués programmés sur mesure pour la récupération de dictionnaires tels que RECUPDIC écrit par Hai Doan-Nguyen dans le cadre de sa thèse (Doan-Nguyen, 1998). Toutefois, l'usage de ces outils nécessite des compétences informatiques élevées excluant, de fait, les linguistes et lexicographes de cette étape.

La traduction par analyse utilise un formalisme nommé H-grammar. L'utilisateur décrit la grammaire du dictionnaire à récupérer en H-grammar. Il ajoute ensuite les actions de construction d'objets et de détection d'erreurs. La détection d'erreurs permet de corriger automatiquement les erreurs les plus fréquentes. Si un détail est faux dans un article, il n'est pas rejeté en bloc. Un compilateur utilise ensuite la description pour construire l'ensemble d'objets constituant une représentation structurée du dictionnaire.

7.1. Exemple d'article avant récupération

La figure 16 représente un article du dictionnaire BABEL, un dictionnaire monolingue anglais d'abréviations dans le domaine informatique (Kind, 1997) au format d'origine avant la récupération.

```
.COM    Command (file name extension) +
        Commercial Business (Domain Name) [Internet]
```

Figure 16. Article de BABEL avant récupération

Il arrive fréquemment qu'un article ne vérifie pas la syntaxe indiquée par ses auteurs. Dans BABEL, par exemple, on peut trouver des parenthèses en trop, des "+" oubliés, etc. Il faut alors normaliser. L'article de la figure 16 donné en exemple est correct.

Cet article a une structure implicite : c'est sa présentation qui reflète sa structure. Les différentes informations sont distinguées par leur mise en forme et des caractères spéciaux (les parenthèses "()", le "+", les crochets "[]", etc.). Il faut donc modifier cet article pour le réutiliser. Pour récupérer l'article avec l'outil H-grammar, il faut écrire une grammaire de récupération dans ce formalisme. Voici quelques éléments concernant l'écriture d'une grammaire H-grammar.

7.2. Grammaire de récupération

Une grammaire de récupération H-grammar se compose de six mots-clefs suivis de leurs instructions :

`#grammar` : nom de la grammaire ;

`#syntax-rules` : règles d'analyse syntaxique pour la récupération ;

`#start-symbol` : symbole de départ de la grammaire ;

`#lexical-rules` : règles d'analyse lexicale pour construire les items lexicaux ;

`#lexical-order` : ordre de préférence entre les items lexicaux ;

`#working-code` : fonctions Common Lisp intégrables dans les règles syntaxiques.

La figure 17 montre une grammaire H-grammar permettant la récupération des articles de BABEL. Dans les règles d'analyse syntaxique, le caractère "§" correspond au symbole nul, le caractère ">" devant un nom de symbole comme ">hwd" indique que ce symbole est terminal. Dans les règles d'analyse lexicale, la notation "_¨10" signifie que le symbole est composé de 10 caractères, le symbole `to-cparen` correspond à une chaîne de caractères se terminant par une parenthèse fermante, le symbole `to-cbrak` correspond à une chaîne de caractères se terminant par un crochet fermant.

```
#grammar babel-glossary /* Acquisition du glossaire BABEL */
#syntax-rules
:1: babel-entry(;entry) -> >hwd(;hwd) body(;body) -- (#!babel((trim-whites hwd) body); entry).
:2: body(;body) -> sense(;S1) sense*(;S*) -- cons(S1 S*; body).
:3: sense(;S)-> >exps(;exps) expl?(;expl) subj?(;subj) -- (#!sense((trim-whites exps) (if expl (trim-
whites expl)) (if subj (trim-whites subj))); S).
:4: expl?(;expl) -> "(" >to-cparen(;expl) ")".
:5: expl?(;expl) -> §(nil;expl).
:6: subj?(;subj) -> "[" >to-cbrak(;subj) "]".
:7: subj?(;subj) -> §(nil; subj).
:8: sense*(;S*) -> "+" sense(;S1) sense*(;S*1) -- cons(S1 S*1; S*).
:9: sense*(;S*) -> §(nil; S*).
#start-symbol babel-entry /* départ de la grammaire*/
#lexical-rules hwd -> _¨10. /* Headword prend 10 cars */
exps -> ![+([]*. to-cparen -> >[)]. to-cbrak -> >[\]].
#lexical-order ("+" "(" ")" "[" "]" hwd exps expl subj)
#working-code (sia-defclass babel () (hwd body))
```

```
(sia-defclass sense () (exps expl subj))
(defun trim-whites (string) (string-trim '(#\Space #\Tab #\Newline) string))
```

Figure 17. Grammaire H-grammar de récupération de BABEL

Expliquons maintenant les règles d'analyse syntaxique **#syntax-rules** :

`sense*` est un élément non-terminal. Le signe **`*`** n'est pas l'opérateur de Kleene mais signale une liste de `sense`.

`expl?` est un élément non terminal. Le signe **`?`** Note la présence ou l'absence de **`expl`**.

La règle 1 produit un article babel **`babel-entry`** à partir du mot-vedette **`hwd`** et d'un corps **`body`**.

La règle 2 produit un corps **`body`** à partir d'une liste de sens **`sense*`**.

La règle 3 produit un sens à partir d'une définition **`exps`**, d'une explication **`expl`** et d'un domaine **`subj`**.

Les règles 4 et 5 produisent une explication **`expl`** à partir d'un texte entre parenthèses "**()**".

Les règles 6 et 7 produisent un domaine **`subj`** à partir d'un texte entre crochets "**[]**".

Les règles 8 et 9 produisent une liste de sens **`sense*`** à partir de deux sens **`sense`** séparés par un "+".

Cette grammaire est interprétée ensuite par un compilateur Macintosh Common Lisp qui produit des objets LISP correspondant aux articles récupérés.

7.3. Exemple d'article après récupération

La figure 18 montre le résultat de la récupération de l'article BABEL original après compilation avec H-grammar :

```
(BABEL
  (HWD . ".COM")
  (BODY LIST
    (SENSE
      (EXPS . "Command")
      (EXPL . "file name extension")
      (SUBJ . NIL))
    (SENSE
      (EXPS . "Commercial Business")
      (EXPL . "Domain Name")
      (SUBJ . "Internet"))))
```

Figure 18. Article de BABEL après récupération (objet LISP)

L'article obtenu est un objet LISP dans lequel les informations sont marquées explicitement. Il est alors facile de les réutiliser automatiquement pour produire de nouveaux ensembles lexicaux, ou encore de transformer l'article en un autre format exprimant une structure explicite, comme XML par exemple, et obtenir ainsi un article dans un format passerelle.

8. Du format passerelle vers le format cible

Dans cette partie, nous montrons comment représenter des ressources dans un format cible en conformité avec la partie informative de la version 16 de la DTD de la norme LMF 24613:2008. Les exemples sont tirés des dictionnaires du projet DiLAF.

Chaque information est présentée sous forme d'une valeur de l'attribut *@val* de l'élément *"feat"* (pour "feature") et associée à l'attribut *@att* qui renseigne quant à la nature de l'information.

8.1. Ressource lexicale

Un dictionnaire LMF est une ressource lexicale pour laquelle est indiquée la référence ISO 639-3 du codage des noms de langues. Elle englobe le lexique dans lequel figure le code de la langue source du dictionnaire et l'ensemble des articles.

Nom francophone de la langue	**Nom anglophone de la langue**	**Code ISO 639-3**
bambara	Bambara	bam
français	French	fra
haoussa	Hausa	hau
kanouri	Kanuri	kau
soŋay zarma	Zarma	dje
tamajaq	Tamajaq	ttq

Figure 19. Codes ISO 639-3 des noms des langues des dictionnaires du projet DiLAF

```
<LexicalResource>
 <GlobalInformation>
  <feat att="languageCoding" val="ISO 639-3"/>
 </GlobalInformation>
 <Lexicon>
  <feat att="language" val="kau"/>
  <LexicalEntry id="a">
  ...
```

Figure 20. Début du dictionnaire kanouri-français au format cible LMF

Les articles figurent dans le lexique (élément <*Lexicon*>), chacun est encadré par l'élément <*LexicalEntry*> associé à un identifiant unique.

Nous détaillons maintenant la représentation des parties composant une notice.

8.2. Catégorie lexicale

La catégorie lexicale d'une entrée est directement signalée dans l'entrée lexicale comme valeur associée à *"partOfSpeech"*. Par conséquent, une entrée de dictionnaire possédant plusieurs catégories lexicales doit être traduite en plusieurs entrées lexicales.

Exemple : en kanouri, *dole* « obligation » est un nom commun.

```
     <feat att="partOfSpeech" val="commonNoun"/>
```

Figure 21. Notation de la catégorie lexicale dans LMF

~~Le catalogue des catégories de parties du discours de l'ISO figure en Annexe B.~~

Le travail sur des langues peu dotées peut amener à découvrir de nouvelles catégories de parties du discours telle que “idéophone” pour le kanouri. Elles seront alors communiquées à l'ISO pour être incluses au catalogue existant.

8.3. Lemme

Les formes orthographique et phonétique d'une entrée sont associées, respectivement aux attributs *@writtenForm* et *@phoneticForm* et enchassées dans le bloc <*Lemma*>.

Voici la partie lexicale de l'entrée `dole` du dictionnaire kanouri-français :

```
<LexicalEntry id="dole">
  <feat att="partOfSpeech" val="commonNoun"/>
  <Lemma>
    <feat att="writtenForm" val="dole"/>
    <feat att="phoneticForm" val="dólè"/>
  </Lemma>
```

Figure 22. *Partie lexicale de l'entrée "dole" au format LMF*

8.4. Variations morphologiques

L'élément <*Wordform*> entoure les informations portant sur les variations morphologiques exprimées en extension car elles ne sont pas régulières, comme les pluriels irréguliers ou encore l'état d'annexion, fréquents dans le dictionnaire tamajaq-français. Les formes au singulier et au pluriel figurent explicitement en combinant deux éléments <feat>, l'un indiquant la forme ("*writtenForm*") et l'autre le nombre (*"grammaticalNumber")*.

Voici un exemple avec l'entrée *ălawwa* « lance-pierres » :

```
<WordForm>
  <feat att="writtenForm" val="ălawwa"/>
  <feat att="grammaticalNumber" val="singular"/>
</WordForm>
<WordForm>
  <feat att="writtenForm" val="ilawwan"/>
  <feat att="grammaticalNumber" val="plural"/>
</WordForm>
<WordForm>
  <feat att="writtenForm" val="ălawwa"/>
  <feat att="grammaticalNumber" val="singular"/>
  <feat att="contextualVariation" val="annexion"/>
</WordForm>
<WordForm>
  <feat att="writtenForm" val="ălawwan"/>
  <feat att="grammaticalNumber" val="plural"/>
  <feat att="contextualVariation" val="annexion"/>
</WordForm>
```

Figure 23. Variations morphologiques de l'entrée *ălawwa* au format LMF

8.5. Bloc sémantique

Le bloc sémantique est délimité par l'élément <Sense>. Il est possible d'avoir plusieurs blocs sémantiques. Chaque bloc sémantique regroupe la définition et les éventuelles relations sémantiques de synonymie ou antonymie.

La définition est délimitée par l'élément <Definition>. L'équivalent français est délimité par l'élément <Equivalent>.

Les relations sémantiques sont traitées différemment selon la nature de la cible mise en relation. Si la cible est une entrée lexicale monosémique, elle ne possède qu'un seul bloc sémantique, l'identifiant de l'entrée lexicale visée peut donc être directement indiqué comme valeur de l'attribut *@RelatedForm*. En revanche, si la cible est une entrée lexicale possédant plusieurs blocs sémantiques, il est nécessaire de préciser celui qui est visé. Dans ce cas il est nécessaire d'utiliser l'élément <SenseRelation> et d'indiquer l'identifiant du bloc sémantique visé comme valeur de l'attribut *@targets*.

Ce distingo apparaît peu adapté à une ressource lexicale en évolution car, en cas d'enrichissement du dictionnaire par l'ajout d'un sens à une entrée par ailleurs en relation sémantique avec une autre entrée, il faudrait modifier les représentations de ces entrées. Dans cette situation, il est préférable d'identifier tous les blocs sémantiques par un identifiant unique susceptible de servir de référence.

Exemple : en kanouri, *alau* « créature » est synonyme d'*alitta*. Ces deux entrées sont monosémiques. Chaque bloc sémantique est identifié par l'entrée concaténée au chiffre 1.

```
<LexicalEntry id="alau">
  <Lemma>
    <feat att="writtenForm" val="alau"/>
    <feat att="phoneticForm" val="álâu"/>
  </Lemma>
  <Sense id="alau1">
    <Definition>
      <feat att="text" val="Awo duwon dunia adəlan Kəmandeye alakcənadə."/>
    </Definition>
    <SenseRelation targets="alitta1">
      <feat att="type" val="synonym"/>
    </SenseRelation>
  </Sense>
</LexicalEntry>
```

Figure 24. Entrée *alau* comportant un renvoi vers l'entrée *alitta* au format LMF

La relation d'antonymie se signale par la valeur l'usage de *"antonym"* comme valeur de type sémantique tandis qu'une relation sémantique non précisée sera caractérisée par la valeur *"seeAlso"*.

Les exemples d'usage présentent de potentielles difficultés que révèle l'examen attentif du dictionnaire bambara-français. Ce dictionnaire présente l'originalité de traduire tous les exemples en français et de les présenter sous une forme tonalisée. De plus, certains proverbes, dont la traduction ne permet pas de comprendre le sens, sont munis d'une explication supplémentaire. Un exemple d'usage peut donc présenter jusqu'à quatre caractéristiques auxquelles pourraient s'ajouter d'autres informations telles la traduction dans d'autres langues.

exemple	bamusokɔrɔ kɔ tɛ a ŋɛɲɛ ni a ma dɛnɛ ye.
forme tonalisée	bàmùsòkɔrɔ kɔ tɛ à ŋɛɲɛ ni à ma dɛ̀nɛ ye.
traduction	le dos de la chèvre ne lui démange pas, tant qu'elle n'a pas vu un mur.
explication	créer des besoins, rôle de la publicité.

Figure 25. Exemple d'usage pour l'entrée "bamusokɔrɔ" ("chèvre adulte") issu du dictionnaire bambara-français

Un exemple d'usage est entouré par l'élément <*Context*>. Chacune des caractéristiques est entouré par l'élément <*TextRepresentation*>. L'exemple précédent devient :

```
<Context>
 <TextRepresentation>
  <feat att="language" val="bam"/>
  <feat att="writtenForm" val="bamusokɔrɔ kɔ tɛ a
```

```
      ŋɛɲɛ ni a ma dɛnɛ ye."/>
  </TextRepresentation>
  <TextRepresentation>
    <feat att="language" val="bam"/>
    <feat att="phoneticForm" val="bàmùsòkɔrɔ kɔ tɛ à
      ŋɛɲɛ ni à ma dɛ̀nɛ ye."/>
  </TextRepresentation>
  <TextRepresentation>
    <feat att="language" val="fra"/>
    <feat att="writtenForm" val="le dos de la chèvre
      ne lui démange pas, tant qu'elle n'a pas vu
      un mur."/>
  </TextRepresentation>
  <TextRepresentation>
    <feat att="language" val="fra"/>
    <feat att="explanation" val="créer des besoins,
      rôle de la publicité."/>
  </TextRepresentation>
</Context>
```

Figure 26. Exemple d'usage pour l'entrée "bamusokɔrɔ" ("chèvre adulte") au format LMF

9. Mise en ligne sur une plate-forme de gestion de ressources en ligne

Il existe de nombreux logiciels de gestion de ressources en ligne conçus, pour la plupart, spécifiquement pour une ressource particulière. Très peu de logiciels permettent de manipuler des ressources XML sans modifier leur structure. Il existe néanmoins quelques outils commerciaux tels que TshwaneLex[31] ou le Dictionary Publishing System[32] de IDM. Dans un contexte de travail sur les langues peu dotées, nous préférons cependant utiliser un outil gratuit et en source ouverte (open-source). C'est pourquoi nous avons choisi la plate-forme générique de gestion de ressources lexicales en ligne Jibiki.

9.1. Description de la plate-forme utilisée

Jibiki est une plate-forme générique en ligne dédiée à la manipulation de ressources lexicales avec gestion d'utilisateurs et groupes, consultation de ressources hétérogènes et édition générique d'articles de dictionnaires. La plate-forme est programmée entièrement en Java, elle est fondée sur l'environnement "Enhydra". Toutes les données sont stockées au format XML dans une base de données (Postgres).

Ce site Web communautaire propose principalement deux services : une interface unifiée permettant d'accéder simultanément à de nombreuses ressources hétérogènes (dictionnaires monolingues, dictionnaires bilingues, bases multilingues, etc.) et une interface d'édition spécifique pour contribuer directement aux dictionnaires disponibles sur la plate-forme.

Plusieurs projets de construction de ressources lexicales ont utilisé ou utilisent cette plate-forme avec succès. C'est le cas par exemple du projet GDEF de dictionnaire bilingue estonien-français (Chalvin & Mangeot, 2006) ou plus récemment du projet MotÀMot (Mangeot, 2009). Le code de cette plate-forme est disponible gratuitement en source ouverte en téléchargement depuis la forge du laboratoire LIG[33].

9.2. Import d'une ressource existante sur la plate-forme

L'unique condition à respecter pour importer une ressource existante sur la plate-forme Jibiki est que celle-ci soit au format XML. En effet, un système de pointeurs communs dans des structures hétérogènes permet de manipuler les ressources sans modifier leur structure. Chaque pointeur est

[31] tshwanedje.com/tshwanelex/
[32] www.idm.fr/products/dictionary_writing_system_dps/27/
[33] ligforge.imag.fr/projects/jibiki/

indexé dans une base de données et permet d'effectuer une recherche rapide. Ce système est appelé Common Dictionary Markup (CDM). Il existe des pointeurs communs prédéfinis pour les objets lexicaux que l'on trouve fréquemment dans la plupart des dictionnaires (voir Figure 29). Il est possible également de définir des pointeurs spécifiques à une ressource.

L'importation d'un dictionnaire peut donc se faire aussi bien au format passerelle qu'au format cible ; il peut s'agir d'un dictionnaire structuré suivant les recommandations de la TEI comme de la norme LMF, etc.

Pour préparer l'import, il est nécessaire de décrire la structure du dictionnaire et des volumes dans des fichiers de méta-données. Ceux-ci seront utilisés par la plate-forme lors de l'import.

Le dictionnaire et sa macrostructure sont décrits à l'aide d'un formulaire HTML dans un premier fichier de méta-données (voir figure 27).

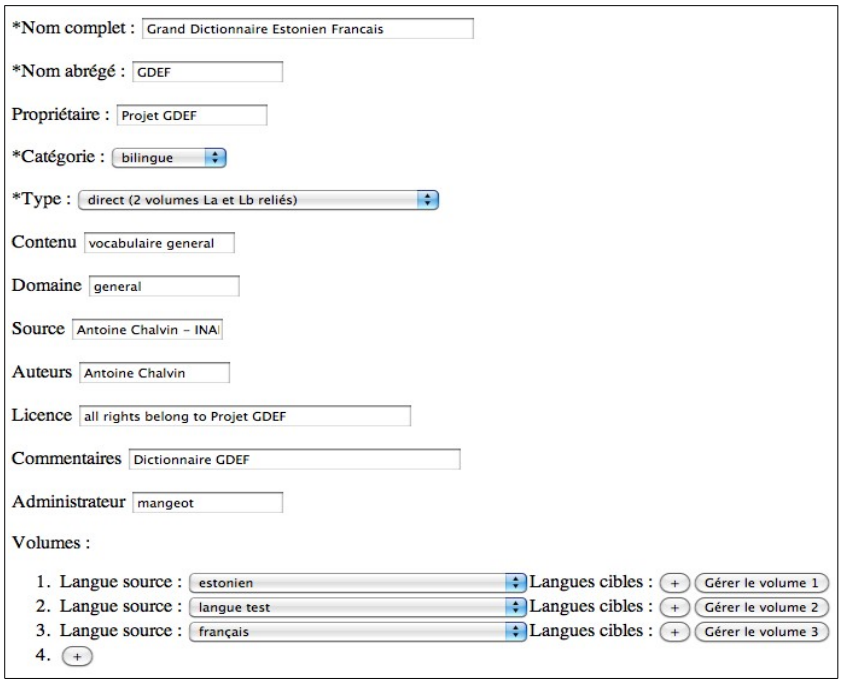


Figure 27. Méta-données du dictionnaire GDEF estonien-français

Chaque volume et sa microstructure sont décrits dans un fichier de méta-données distinct (voir figure 28).

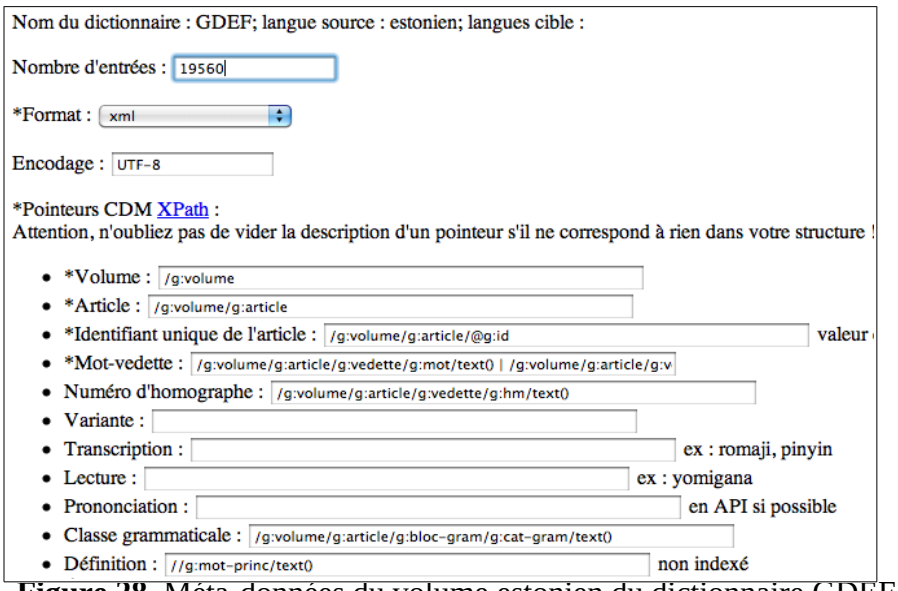


Figure 28. Méta-données du volume estonien du dictionnaire GDEF

La microstructure est décrite à l'aide de pointeurs communs identifiant les mêmes éléments d'information, les pointeurs CDM utilisant le standard XPath. La figure 29 montre la valeur des pointeurs CDM pour les dictionnaires FeM (français-anglais-malais), Oxford-Hachette (français-anglais) et JMdict (japonais-multilingue).

Pointeur CDM	FeM	OHD	JMdict
Volume	/volume	/volume	/JMdict
Entry (article)	/volume/entry	/volume/se	/Jmdict/entry
Entry ID (identifiant de l'article)	/volume/entry/@id		/Jmdict/entry/ent_seq /text()
Headword (mot-vedette)	/volume/entry/headword/ text()	/volume/se/hw/text()	/Jmdict/entry/k_ele/ keb/text()
Pronunciation (prononciation)	/volume/entry/prnc/text()	/volume/se/pr/ph/text()	
PoS (catégorie grammaticale)	//sense-list/sense/pos-list/ text()	/volume/se/hg/ps/text()	/Jmdict/entry/sense/ pos/text()
Domain (domaine)		//u/text()	
Example (exemple)	//sense1/expl-list/expl/fra	//le/text()	/Jmdict/entry/sense/ gloss/text()

Figure 29. Pointeurs CDM pour les dictionnaires FeM, OHD et JMdict

9.3. Consultation et édition des données en ligne

Les dictionnaires mis en ligne sur une instance de la plate-forme Jibiki sont consultables en ligne par le grand public. L'interface de consultation est multi-critères et multi-ressources. Elle s'appuie sur les pointeurs CDM décrits précédemment.

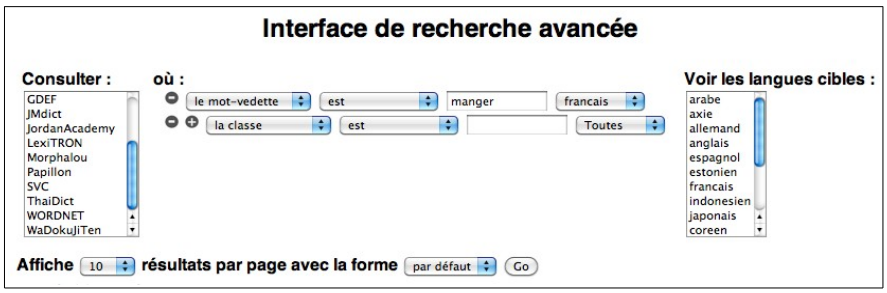


Figure 30. Interface de recherche avancée de la plate-forme Jibiki

Il est possible de contribuer directement en ligne. Le système de gestion de permissions intégré à la plate-forme permet de demander la révision et la validation des contributions par un responsable avant leur intégration finale. La plate-forme autorise également, grâce à une interface de programmation (API), l'utilisation à distance par d'autres logiciels ou services.

L'éditeur est fondé sur un modèle d'interface HTML instancié par l'article à éditer. Le modèle peut être généré automatiquement depuis une description de la structure de l'entrée à l'aide d'un schéma XML. Ce modèle peut être modifié ensuite pour améliorer le rendu à l'écran. La seule information nécessaire à l'édition d'un article de dictionnaire est donc le schéma XML représentant la structure de cette entrée.

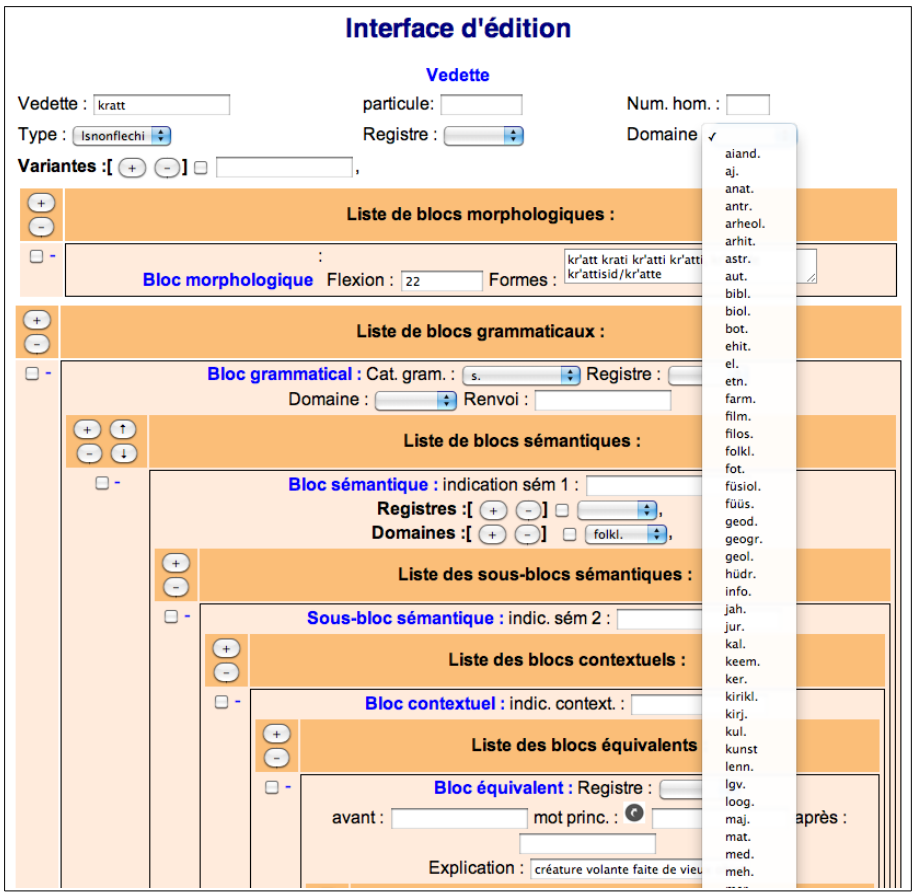


Figure 31. Interface d'édition d'un article estonien du dictionnaire GDEF

Conclusion

Récupérer des dictionnaires éditoriaux pour les convertir en des dictionnaires électroniques s'est révélé être un processus délicat, semé d'embûches, mais pourtant réalisable. La démarche est économiquement valide car le temps de travail et le coût sont minimisés : un dictionnaire peut être converti en quelques semaines. La part de travail humain reste importante et l'horizon d'une conversion entièrement automatisée reste donc lointain. Toutefois le gain est bien réel par rapport aux années d'investissements qu'exigerait la création d'un dictionnaire électronique. Il apparaît que l'indispensable collaboration entre linguistes et informaticiens constitue une aubaine pour effectuer des transferts de connaissances. Non seulement, les connaissances de chacun sont indispensables, mais il faut aussi que chaque partenaire accroisse ses compétences dans le savoir de l'autre discipline.

Ces ressources électroniques ainsi élaborées sont développées a priori pour servir le traitement automatique des langues. Au-delà de cet usage spécialisé, les locuteurs des langues peu dotées peuvent aussi y avoir accès car elles sont conçues pour être visibles sur la Toile. Dans des contextes de pauvreté où les locuteurs n'ont souvent jamais vu un dictionnaire de leur langue mais où l'accès au web s'améliore, l'enjeu quant aux retombées pour les populations est majeur. De plus, la visibilité des résultats constitue une motivation supplémentaire pour les personnes participant à la conversion.

Les dictionnaires ainsi convertis sont incomplets et loin d'être parfaits, et des corrections devront être apportées. Toutefois, ils constituent un marche-pied vers d'autres développements : construction de corpus bilingues quand les entrées présentent des exemples traduits ; mise au point d'analyseurs morphologiques portant sur les flexions quand ces informations sont disponibles ; etc. La mise

à disposition de ces ressources peut susciter la vocation de chercheurs ne travaillant pas sur ces langues, et donc accroître le potentiel de recherches.

Financement

Le projet DiLAF est financé par le Fonds Francophone des Inforoutes de l'Organisation Internationale de la Francophonie.

Œuvres citées

Andries, P. (2004). Proposition d'ajout de l'écriture tifinaghe. Organisation internationale de normalisation. Jeu universel des caractères codés sur octets (JUC). ISO/IEC JTC 1/SC 2 WG 2 N2739.

Charles, B. (1996). Dictionnaire bambara-français.

Berment, V. (2004). Méthodes pour informatiser des langues et des groupes de langues "peu dotées". Thèse de nouveau doctorat, spécialité informatique, Université Joseph Fourier Grenoble I, Grenoble, France.

Buseman, A., Buseman, K., Jordan, D., Coward, D. (2000). The linguist's shoebox: tutorial and user's guide: integrated data management and analysis for the field linguist. volume viii. Waxhaw, North Carolina : SIL International.

Calvet, L.-J. (1987). La guerre des langues. Paris, Payot.

Calvet, L.-J. (1996). Les politiques linguistiques. Paris, PUF.

Chalvin, A. Mangeot, M. (2006). Méthodes et outils pour la lexicographie bilingue en ligne : le cas du Grand Dictionnaire Estonien-Français. Actes d'EURALEX 2006, Turin, Italie, 6-9 septembre.

Chanard, C. Popescu-Belis, A. (2001). Encodage informatique multilingue : application au contexte du Niger. Cahiers du Rifal (cont. Terminologies Nouvelles), n. 22, pages 33-45.

Desgraupes, B. (2005). Passeport pour Unicode. Vuibert France. (ISBN 2-7117-4827-8)

Diki-Kidiri M. (2004). Multilinguisme et politiques linguistiques en Afrique. Colloque Développement durable, leçons et perspectives, Ouagadougou.

Doan-Nguyen H. (1998). Techniques génériques d'accumulation d'ensembles lexicaux structurés à partir de ressources dictionnairiques informatisées multilingues hétérogènes. Thèse de nouveau doctorat, Spécialité Informatique, Institut National Polytechnique de Grenoble, 168 pages.

Enguehard, C. (2009). Les langues d'Afrique de l'Ouest : de l'imprimante au traitement automatique des langues. Sciences et Techniques du Langage, 6, pages 29-50. (ISSN 0850-3923)

Enguehard C., Kane S., Mangeot M., Issouf M., Sanogo M-L. (2012). Vers l'informatisation de quelques langues d'Afrique de l'Ouest. JEP-TALN-RECITAL 2012, Workshop TALAf 2012: NLP for African Languages, pages 27-40, Grenoble, France, juin.

Francopoulo G., Bel N., George M., Calzolari N., Monachini M., Pet M., and Soria C. (2009). Multilingual resources for NLP in the lexical markup framework (LMF). Language Resources and Evaluation, 43(1), pages 57–70, March.

Gut Y. Ramli P. R. M., Yusoff Z., Kim Ch. Ch., Samat S. A., Boitet Ch., Nédobejkine N., Lafourcade M. et al. (1996). Kamus Perancis-Melayu Dewan, dictionnaire français-malais. Dewan Bahasa Dan Pustaka, Kuala Lumpur, 667 p.

Haralambous Y. (2004). Fontes & codages. O'Reilly France.

Kind, I. (1997). A Glossary of Computer Oriented Abbreviations and Acronyms, Version 97B.

Mangeot M. (2001). Environnements centralisés et distribués pour lexicographes et lexicologues en contexte multilingue. Thèse de nouveau doctorat, Spécialité Informatique, Université Joseph Fourier Grenoble I, 27 septembre, 280 pages.

Mangeot M., Sérasset G., Lafourcade M. (2003). Construction collaborative de données lexicales multilingues : le projet Papillon. Revue TAL, Vol. 44:2/2003, pages 151-176.

Mangeot M. (2009). Projet Mot à mot : élaboration d'un système lexical multilingue par le biais de dictionnaires bilingues. Actes des journées scientifiques LTT 2009, Lisbonne, Portugal, 15-17 octobre, 12 pages.

Mijinguini, A. (2003). Dictionnaire élémentaire hausa-français.

Modi, I. (2007). Les caractères tifinagh dans Unicode. Actes du colloque international "le libyco-berbère ou le tifinagh : de l'authenticité à l'usage pratique", pages 241-254, ed. Haut Commissariat à l'amazighité (HCA), pages 21-22, mars, Alger.

Arrêté 212-99 de la République du Niger. (1999). Alphabet haoussa.

Arrêté 213-99 de la République du Niger. (1999). Alphabet kanouri.

Arrêté 214-99 de la République du Niger. (1999). Alphabet tamajaq.

Programme Décennal du Développement de l'Éducation (PDDE), Niger. (2003). Enseignement bilingue, pages 121-132.

Richer, D. (2007) Dictionnaire français-khmer (en phonétique), Cambodge : Édition DR, 696 p.

Romary L., Salmon-Alt S., Francopoulo G. (2004). Standards going concrete: from LMF to Morphalou. Proceedings of the COLING Workshop on Enhancing and Using Electronic Dictionaries. ElectricDict'04, pages 22–28, Stroudsburg, PA, USA: Association for Computational Linguistics.

Programme de soutien à l'éducation de base (Soutéba). (2004). Dictionnaire kanouri-français destiné pour le cycle de base 1.

Programme de soutien à l'éducation de base (Soutéba). (2007). Dictionnaire tamajaq-français destiné à l'enseignement du cycle de base 1.

Streiter O., Scannell K., Stuflesser M. (2006). Implementing NLP projects for non-central languages : Instructions for funding bodies, strategies for developers. Machine Translation, volume 20.

Text Encoding and Interchange P4. Guidelines for Electronic Text Encoding and Interchange. 2004. www.tei-c.org/release/doc/tei-p4-doc/html/

Umaru, I. A. (1997). Zarma ciine - kaamuusu kayna. Editions Alpha.

Unicode. (2005).The Unicode Standard 4.1. Tifinagh, range 2D30-2D7F.

Unicode. (2012). About Versions of the Unicode Standard, 2012. www.unicode.org/versions/, consulté le 5 juillet 2012.

Adresse des auteurs :

Mathieu Mangeot
GETALP-LIG
41 rue des mathématiques
BP 53
38041 Grenoble Cedex 9
France

Chantal Enguehard
LINA
2, rue de la Houssinière
BP 92208
44322 Nantes Cedex 03
France

Un devin de microstructures pour importer ou normaliser des ressources lexicales

Mathieu Mangeot, Valérie Bellynck

Laboratoire LIG, équipe GETALP
Bâtiment IMAG CS 40700
38058 GRENOBLE CEDEX 9, FRANCE
{mathieu.mangeot,valerie.bellynck}@imag.fr

Résumé

Dans cet article, nous présentation un outil pour annoter une ressource non structurée ou semi-structurée, de façon à automatiser l'import de ses données dans un environnement générique facilitant et uniformisant son exploitation. Cet outil s'appuie sur le calcul d'une signature, caractérisant la ressource par des informations comptables sur ses éléments, et sur des heuristiques exploitant les statistiques de la signature pour repérer les entrées et la microstructure de chaque entrée dans le cas d'une ressource lexicale à importer dans la plateforme de bases lexicales Jibiki. Le résultat produit est géré dans iPolex, un entrepôt dédié aux traitements de fichiers de ressources lexicales permettant d'y ajouter des méta-données. La plateforme Jibiki accède à iPoLex pour importer directement des ressources lexicales bien formées et suffisamment renseignées. Le processus d'import ainsi constitué est utilisé pour créer et peupler des bases lexicales dans Jibiki. Les ressources ainsi importées sont ensuite consultables et éditables en ligne.

A micro-structure guesser to import or normalize lexical resources

In this article, we present a tool to annotate an unstructured or semi-structured resource, in order to automate the import of its data in a generic environment that facilitates and standardizes its use. This tool is based on the computation of a signature, characterizing the resource with accounting information about its elements, and on heuristics using signature statistics to identify the entries and microstructure of each entry in the case of a lexical resource to be imported into the Jibiki lexical database platform. The product result is managed in iPolex, a lexical data warehouse dedicated to processing lexical resource files to add metadata. The Jibiki platform accesses iPoLex to directly import lexical resources. The import process thus constituted is used to create and populate lexical bases in Jibiki. The imported resources are then available for consultation and editing online.

Mots-clés : iPolex, signature lexicale, jibiki, ressource lexicale

Keywords : iPolex, lexical signature, jibiki, lexical resource

1. Introduction

À l'heure actuelle, la plupart des ressources lexicales que nous manipulons sont au format XML, ce qui est une grande avancée par rapport aux débuts de l'électronisation des ressources où chacune avait son propre format. Par contre, force est de constater que, contrairement aux corpus, les efforts de standardisation ont été vains (Enguehard & Mangeot, 2013). Il en résulte que chaque ressource définit sa propre microstructure et son propre jeu de balises pour la décrire. Cette situation freine l'exploitation de ressources existantes aussi bien pour la consultation et l'édition que pour leur réutilisation dans de nouvelles ressources. Nos travaux se situent précisément dans ce domaine de la manipulation de ressources lexicales hétérogènes.

Même si les structures sont différentes, la plupart des types d'information se retrouvent d'une ressource à l'autre : le mot-vedette, la prononciation, la catégorie grammaticale, les exemples, les traductions, etc. La plupart du temps, il est possible de les repérer manuellement même si l'on ne maîtrise pas la langue représentée. Dans cet article, nous montrons qu'il est possible de repérer ces types d'information automatiquement puis de proposer ensuite à l'utilisateur de confirmer les hypothèses calculées, et ainsi d'améliorer grandement la manipulation de ressources lexicales.

Pour le prouver, nous proposons un outil qui analyse la structure d'une ressource lexicale et en construit une signature lexicale représentant les informations calculées sur cette structure. La signature lexicale est ensuite utilisée pour calculer des heuristiques permettant de formuler des hypothèses sur les différents types d'information présents dans la ressource analysée. Il est possible également d'utiliser cette signature lexicale pour détecter un certain nombre d'erreurs potentielles dans la structure analysée. Les hypothèses précédemment calculées servent ensuite à proposer à l'utilisateur un ensemble de pointeurs permettant d'identifier les informations repérées dans la structure. Celui-ci les validera ou non en fonction de son analyse manuelle. Les pointeurs sont ensuite utilisés pour importer automatiquement la ressource analysée dans la plateforme Jibiki permettant de la consulter et de l'éditer en ligne.

2. Signature dictionnairique

La signature d'une ressource lexicale au format XML est une présentation condensée de sa structure XML et de son contenu. Elle permet de visualiser le document et de repérer en un coup d'œil d'éventuelles erreurs. Ce concept, inventé pour nos besoins propres, peut être utilisé pour tout type de document XML. Le but est de caractériser toute ressource lexicale d'un point de vue structurel. Il s'agit donc d'une description synthétique présentant un ensemble d'indicateurs quantitatifs pour chaque élément de texte repérable par le même mécanisme. Pour chaque mécanisme, les valeurs calculées sont principalement le nombre de tels éléments et le nombre moyen de caractères des éléments de textes ainsi repérés.

Par exemple, dans le cas de documents XML où les éléments de texte sont encadrées par des balises, nous produisons une présentation indentée : chaque ligne présente un élément XML et est décalée autant qu'elle est imbriquée.

La signature est créée par un analyseur XML lors de l'analyse d'un document. Du fait de la grande taille des dictionnaires, nous avons opté pour un parseur événementiel SAX[1] (Simple Api for XML) plutôt que DOM[2] (Document Object Model).

Pour chaque élément XML rencontré, un certain nombre d'informations est stocké en mémoire : son nom, sa position dans l'arbre XML et le nombre de fois qu'il apparaît dans le document à cette position.

Chaque attribut est aussi analysé : on stocke son nom et pour sa valeur, on compte le nombre de valeurs différentes jusqu'à un maximum de 1000[3], le nombre de valeurs classées par ordre alphabétique selon le tri Unicode et le nombre total de valeurs.

Pour le texte contenu dans l'élément, on stocke également le nombre moyen de caractères et de mots (chaînes de caractères séparées par des espaces) ainsi que le nombre de valeurs différentes jusqu'à un maximum de 1000, le nombre de valeurs classées par ordre alphabétique selon le tri Unicode et le nombre total de valeurs.

Les informations ainsi collectées lors de l'analyse du document XML sont ensuite affichées de manière condensée sous forme d'arbre XML où chaque élément n'apparaît qu'une fois et seule la balise ouvrante est affichée, ce qui permet un gain de place. Les éléments fils sont indentés par rapport à leur élément père. Si le nombre de valeurs différentes pour un attribut ou pour du texte est inférieur à 50, toutes les valeurs sont affichées à la suite (voir Figure).

Figure 1 : signature du dictionnaire DiLAF zarma-français

1 https://fr.wikipedia.org/wiki/Simple_API_for_XML

2 https://fr.wikipedia.org/wiki/Document_Object_Model

3 pour éviter de saturer la mémoire et de ralentir les calculs

L'exemple de la figure se lit de la manière suivante :

- l'élément *dilaf* est l'élément racine de la ressource. Il n'apparaît qu'une seule fois dans toute la ressource à cette place dans l'arbre XML (logique pour un élément racine…). Il contient deux attributs. L'attribut *src* n'apparaît qu'une fois (là aussi c'est logique puisque l'élément n'apparaît qu'une fois !) et sa valeur est *dje*. L'attribut *trg* n'apparaît aussi qu'une seule fois (*idem*) et sa valeur est *fra*.

- l'élément *article* est le premier élément fils de l'élément *dilaf* rencontré dans l'arbre XML car il est indenté de deux espaces vers la droite. Il apparaît 6 914 fois dans toute la ressource à cette place dans l'arbre XML. Il n'a pas d'attribut. Son contenu textuel est d'en moyenne 1 caractère. Il contient 2 valeurs différentes qui sont classées par ordre alphabétique. Il contient en tout 3 valeurs. Les valeurs différentes sont une virgule « , » qui apparaît une fois et un point « . » qui apparaît deux fois.

- l'élément *saniize* est le premier élément fils de l'élément *article* rencontré dans l'arbre XML. Il apparaît également 6 914 fois dans toute la ressource à cette place dans l'arbre XML. Il a un attribut *lamba* qui a 10 valeurs différentes et dont 2080 valeurs sont classées par ordre alphabétique sur un total de 3 338. La liste des valeurs différentes est ensuite affichée avec le nombre de valeurs pour chaque : « 1 » apparaît 383 fois, « 10 » apparaît une fois, « 2 » apparaît 337 fois, etc. Son contenu textuel est d'en moyenne 6,2 caractères et 1 mot. Il a 704 valeurs différentes et 6 865 valeurs sont classées par ordre alphabétique sur 6 914. À noter ici que le nombre total de valeurs est supérieur à 1000. Le nombre de valeurs différentes n'est donc pas significatif.

- l'élément *ciiyaŋ* est le deuxième fils de l'élément *article* rencontré dans l'arbre XML. Il apparaît également 6 914 fois dans toute la ressource à cette place dans l'arbre XML. Il n'a pas d'attribut. Son contenu textuel est d'en moyenne 6,8 caractères et 1 mot. Il a 759 valeurs différentes et 6 507 valeurs sont classées par ordre alphabétique sur 6 914. Note : la figure affiche 6 917 au lieu de 6 914 mais cette erreur a été corrigée depuis.

- l'élément *kanandi* est le troisième fils de l'élément *article* rencontré dans l'arbre XML. Il apparaît également 6 914 fois dans toute la ressource à cette place dans l'arbre XML. Il n'a pas d'attribut. Son contenu textuel est d'en moyenne 3,9 caractères et 1 mot. Il a 25 valeurs différentes et 5 070 valeurs sont classées par ordre alphabétique sur 6 884. La liste des valeurs différentes est ensuite affichée avec le nombre de valeurs pour chaque : « alteeb. » apparaît 6 fois, « bsif. » apparaît 1 fois, « ceey » apparaît 5 fois, etc. À noter que cette liste de valeurs est coupée par la copie d'écran.

Les autres éléments se décrivent selon le même principe.

3. Calcul des heuristiques

Dans cette partie, nous montrons comment est utilisée la signature dans le cas des bases lexicales, et donc la signature dictionnairique pour deviner la structure de ces bases.

Pour identifier chaque type d'information (article, mot-vedette, prononciation, catégorie grammaticale, exemples, etc.), nous avons élaboré une fonction de calcul d'heuristiques spécifique qui utilise la signature dictionnairique. Les heuristiques sont fondées sur un travail d'analyse de plus d'une centaine de ressources lexicales mené depuis les débuts de nos recherches dans le domaine (Corréard & Mangeot, 1999). À chauqe heuristique est associé un score. Les valeurs de score sont fixées au départ arbitrairement en fonction de l'importance supposée de l'heuristique. Les valeurs sont éventuellement modifiées en fonction des résultats obtenus sur quelques analyses afin d'obtenir un meilleur résultat. Le résultat de la fonction est un pointeur XPath[4] qui identifie précisément le type d'information dans la structure analysée. Nous avons appelé l'ensemble de ces pointeurs « CDM » pour Common Dictionary Markup (Mangeot, 2002). Nous détaillons ci-dessous les principales fonctions que nous avons élaborées.

3.1. Le volume

Notre définition du volume est l'élément XML parent de celui de l'article. Dans la plupart des cas, c'est également l'élément racine du document mais pas toujours. Nous avons relevé 2 % des ressources où la racine du document n'est pas l'élément père des articles. Il s'agit des ressources respectant la partie informative du standard Lexical Markup Framework[5] où l'on trouvera le résultat suivant :

```
/LexicalResource/Lexicon
```

La fonction de calcul du volume reprend le résultat de celle de calcul de l'article et supprime un niveau dans le pointeur XML. Pour la signature dictionnairique de la figure , la fonction renvoie le pointeur Xpath suivant :

```
/dilaf
```

3.2. L'article

Pour chaque élément dans la signature, la fonction calcule un score. L'élément dont le score est le plus élevé est sélectionné pour le résultat.

Le calcul des scores se fait récursivement en partant des éléments fils du volume. S'il y a plusieurs éléments frères, l'élément dont la fréquence est la plus élevée est sélectionné.

4 https://www.w3.org/TR/xpath/all/

5 https://fr.wikipedia.org/wiki/Lexical_Markup_Framework

Le score de départ est de 0,1. Il est de 1 si le nom de l'élément contient « entry ».

Ensuite, le score final est calculé en divisant le score par le niveau de l'élément dans l'arbre XML. Plus l'élément est profond dans l'arbre, plus son score sera faible.

Pour la signature lexicale de la figure , la fonction renvoie le pointeur Xpath suivant :

```
/dilaf/article
```

3.3. L'identifiant de l'article

Cette fonction calcule un score uniquement pour les attributs de l'élément article.

Le score de départ est de 0. Il est augmenté de 0,3 si le nom de l'attribut contient « id ».

Le score est ensuite augmenté de 0,5 si le nombre de valeurs différentes est égal au nombre de valeurs totales, ce qui démontre une valeur unique pour chaque article.

Pour la signature lexicale de la figure , la fonction ne renvoie rien car l'article n'a pas d'identifiant.

3.4. Le mot-vedette

Les éléments sélectionnés doivent contenir du texte.

Le score de départ de cette fonction est de 0.

Il est augmenté de 0,4 s'il y a autant de mots-vedette que d'articles et de 0,3 si leur nombre est proche à 1 % près.

Il est augmenté de 0,4 si le nom de l'élément contient les mots « headword » ou « vedette ».

Il est augmenté de 0,3 s'il y a en moyenne peu de mots (de 1 à 3).

Il est augmenté de 0,3 s'il y a beaucoup de valeurs différentes (proche du nombre total de valeurs).

Il est augmenté de 0,1 * le rang dans sa fratrie. Le mot-vedette est souvent le premier de sa fratrie.

Ensuite, le score final est calculé en divisant le score par le niveau de l'élément dans l'arbre XML. Plus l'élément est profond dans l'arbre, plus son score sera faible.

Pour la signature lexicale de la figure , la fonction renvoie le pointeur Xpath suivant :

```
/dilaf/article/sanniize/text()
```

Il est à noter que cette fonction et la plupart des suivantes sont optimisées pour une valeur placée dans le contenu textuel d'un élément et non dans un attribut. Il est envisageable de prendre en compte ce dernier cas, mais sa fréquence est assez faible et concerne essentiellement les ressources au format LMF (voir la partie 3.1).

3.5. Le numéro d'homographe

Cette fonction calcule un score uniquement pour les attributs de l'élément du mot-vedette.

Le score de départ est de 0. Il est augmenté de 0,1 si le nom de l'attribut contient « hn » pour « homograph number »

Le score est ensuite augmenté de 5 divisé par le nombre de valeurs différentes. Il y a peu de valeurs différentes.

Le score est ensuite augmenté de 0,3 si la valeur de l'attribut est un entier. Les numéros d'homographes sont la plupart du temps des entiers.

Pour la signature lexicale de la figure , la fonction renvoie le pointeur Xpath suivant :

```
/dilaf/article/sanniize/lamba
```

3.6. La prononciation

Les éléments sélectionnés doivent contenir du texte.

Le score de départ de cette fonction est de 0.

Il est augmenté de 0,2 si le nombre de valeurs est proche de celui du mot-vedette.

Il est augmenté de 0,7 si le nom de l'élément contient le mot «pron».

Il est augmenté de 0,2 s'il y a en moyenne peu de mots (de 1 à 2).

Il est augmenté de 0,3 s'il y a beaucoup de valeurs différentes (proche du nombre total de valeurs).

Ensuite, le score final est calculé en divisant le score par le niveau de l'élément dans l'arbre XML.

Pour la signature lexicale de la figure , la fonction renvoie le pointeur Xpath suivant :

```
/dilaf/article/ciiyaŋ/text()
```

3.7. La catégorie grammaticale

Les éléments sélectionnés doivent contenir du texte et avoir un nombre de valeurs proche ou supérieur à celui du mot-vedette.

Le score de départ de cette fonction est de 0.

Il est augmenté de 0,45 si le nom de l'élément contient les mots «pos», « cat » ou « gram ».

Il est augmenté de 0,2 s'il y a en moyenne peu de mots (de 1 à 3).

Il est augmenté de 0,3 s'il y a peu de valeurs différentes (une liste fermée de valeurs).

Ensuite, le score final est calculé en divisant le score par le niveau de l'élément dans l'arbre XML.

Pour la signature lexicale de la figure , la fonction renvoie le pointeur Xpath suivant :

```
/dilaf/article/kanandi/text()
```

3.8. Le sens de mot

Les éléments sélectionnés ne doivent pas contenir de texte.

Le score de départ de cette fonction est de 0.

Il est augmenté de 0,3 s'il y a au moins 1,5 fois plus de sens que d'articles.

Il est augmenté de 0,75 si le nom de l'élément contient le mot «sens».

Il est augmenté de 0,3 si l'élément contient un attribut avec peu de valeurs différentes, ce qui pourrait correspondre à un numéro de sens.

Ensuite, le score final est calculé en divisant le score par le niveau de l'élément dans l'arbre XML.

Pour la signature lexicale de la figure , la fonction renvoie le pointeur Xpath suivant :

```
/dilaf/article/maana/silmaŋ
```

Le bloc de sens est ensuite calculé comme le volume à partir de l'article en supprimant le dernier élément du pointeur Xpath. Pour la signature lexicale de la figure , la fonction renvoie le pointeur Xpath suivant :

```
/dilaf/article/maana
```

Il est à noter que dans ce cas précis, le calcul du sens de mot est incorrect mais celle du bloc de sens correcte. L'élément représentant le sens est le même que le bloc de sens.

3.9. La définition

Les éléments sélectionnés doivent contenir du texte et le nombre moyen de mots doit être supérieur à 2.

Le score de départ de cette fonction est de 0.

Il est augmenté de 0,3 si le nombre de valeurs est proche de celui du mot-vedette ou supérieur.

Il est augmenté de 0,001 * le nombre moyen de caractères (s'il y a beaucoup de caractères).

Il est augmenté de 0,5 si le nom de l'élément contient les mot «definition» ou « définition » et de 0,3 s'il contient les mots « def » ou « déf ».

Il est augmenté de 0,3 s'il y a beaucoup de valeurs différentes (proche du nombre total de valeurs).

Il est augmenté de 0,05 / le rang dans la fratrie (la définition vient tôt dans la fratrie et souvent avant l'exemple).

Il est augmenté de 0,1 / le niveau de l'élément dans l'arbre XML (il vient tôt dans l'arbre XML)

Pour la signature lexicale de la figure , la fonction renvoie le pointeur Xpath suivant :

```
/dilaf/article/maana/bareyaŋ
```

3.10. L'exemple

Les éléments sélectionnés doivent contenir du texte et le nombre moyen de mots doit être supérieur à 2.

Le score de départ de cette fonction est de 0.

Il est augmenté de 0,3 si le nombre de valeurs est proche de celui du mot-vedette ou supérieur.

Il est augmenté de 0,01 * le nombre moyen de mots (s'il y a beaucoup de mots).

Il est augmenté de 0,001 * le nombre moyen de caractères (s'il y a beaucoup de caractères).

Il est augmenté de 0,5 si le nom de l'élément contient les mot «example» ou «exemple» et de 0,2 s'il contient le mot « ex ».

Il est augmenté de 0,3 s'il y a beaucoup de valeurs différentes (proche du nombre total de valeurs).

Il est augmenté de 0,01 * le rang dans la fratrie (l'exemple vient tard dans la fratrie, après la définition).

Il est augmenté de 0,01 * le niveau de l'élément dans l'arbre XML (il vient tard dans l'arbre XML)

Pour la signature lexicale de la figure , la fonction renvoie le pointeur Xpath suivant :

```
/dilaf/article/maana/silmaŋ/version
```

Le bloc d'exemples est ensuite calculé comme le volume à partir de l'article en supprimant le dernier élément du pointeur Xpath. Pour la signature lexicale de la figure , la fonction renvoie le pointeur Xpath suivant :

```
/dilaf/article/maana/silmaŋ
```

4. Détection d'erreurs

La signature lexicale permet également de détecter un certain nombre d'erreurs potentielles. Nous en détaillons 4 types principaux ci-dessous illustrés à l'aide de l'exemple de la figure .

4.1. Erreurs de structuration

Il arrive que des éléments ou des blocs d'éléments soient mal placés dans la structure d'un article. L'affichage des éléments de manière indentée avec la fréquence permet de repérer ces erreurs plus facilement.

Dans l'exemple de la figure , on trouve un élément *complémént* au même niveau que les éléments *article*. Outre le fait qu'il y a une erreur d'orthographe (il faut écrire *complément*), cet élément devrait se situer à l'intérieur de l'article avec les autres éléments *complément* qui apparaissent 113 fois.

4.2. Listes fermées de valeurs

Lorsqu'il y a un maximum de 50 valeurs différentes, les valeurs sont toutes affichées avec pour chacune sa fréquence. Cela permet de contrôler toutes les listes fermées de valeurs comme par exemple la catégorie grammaticale. On peut supposer qu'une valeur avec une fréquence très faible et qui est très proche d'une autre valeur est une erreur potentielle. Par exemple, une catégorie grammaticale comme le « nom masculin » sera la plupart du temps représentée par « n.m. » et quelquefois par « n. m. » (avec un espace en plus). Il sera possible ensuite de rectifier ces erreurs à l'aide d'un autre programme.

Dans l'exemple de la figure , on trouve une valeur *bsif.* qui apparaît une fois et une valeur *dsif.* qui apparait 4 fois. On peut supposer que *bsif. e*st une erreur et devrait être remplacé par *dsif.* Il s'agira de demander confirmation à un lexicographe spécialiste de la langue en question.

4.3. Compte incorrect

Certains éléments doivent apparaître au moins autant de fois que d'autres. Dans un dictionnaire, on trouve nécessairement au moins autant de mots-vedettes que d'articles. De même, s'il y a une catégorie grammaticale ou une prononciation, il y en a généralement une par article.

Dans l'exemple de la figure , on trouve le même nombre d'éléments *article, sanniize, ciiyaŋ, kanandi, bareyaŋ* (6 914). Par contre, pour l'élément feeriji, on voit qu'il en manque 6 914 - 6 253 = 661.

Pour l'élément *soŋay,* on voit qu'il n'y en a qu'un seul. On peut s'interroger sur son utilité.

Pour les éléments *kanandi* et *bareyaŋ,* on constate également qu'il y a un certain nombre d'éléments vides (sans texte). Pour *kanandi*, il y en a 6 914 – 6 884 = 30 et pour *bareyaŋ*, il y en a 6 914 – 6 864 = 50.

4.4. Texte mal placé

Certains éléments servent uniquement à structurer le document et ne contiennent donc pas de texte. Il arrive que du texte apparaisse malgré tout dans un élément qui ne devrait pas en contenir. La signature nous renseigne sur ce point.

Dans l'exemple de la figure , c'est visiblement le cas de l'élément *article* qui ne devrait pas contenir de texte. On constate qu'il contient une virgule « , » et deux points « . ». Ce texte est donc à supprimer.

5. Utilisation dans les plateformes iPoLex et Jibiki

5.1. Description du script d'analyse

Le script d'analyse produit la signature lexicale et calcule les différentes hypothèses pour chaque type d'information lexicale décrites dans la section 3. Il affiche également le tableau de chaque élément XML rencontré avec sa fréquence, trié par ordre alphabétique. Enfin, il calcule un squelette d'article vide avec tous les éléments rencontrés lors de l'analyse. Ces informations serviront lors de l'import de la ressource dans la plateforme Jibiki (voir section 6).

Programmé en perl, le script utilise pour l'analyse XML le module perl XML::Parser basé sur le parseur XML SAX expat de James Clark. Le script a une taille de 24 Ko et comporte 884 lignes de code.

Ce script a été testé sur une machine équipée d'un processeur Intel Xeon E5 4 cœurs 2,4 Ghz avec 32 Go de RAM avec le système d'exploitation Linux Debian 3.16 Jessie.

L'analyse prend 15 secondes pour le dictionnaire DiLAF bilingue zarma-français de taille 3,3 Mo contenant 6 914 articles. C'est le dictionnaire qui a généré la signature de la figure 1.

L'analyse prend 6 minutes et 35 secondes pour un dictionnaire bilingue anglais-français de taille 17 Mo contenant 39 809 articles.

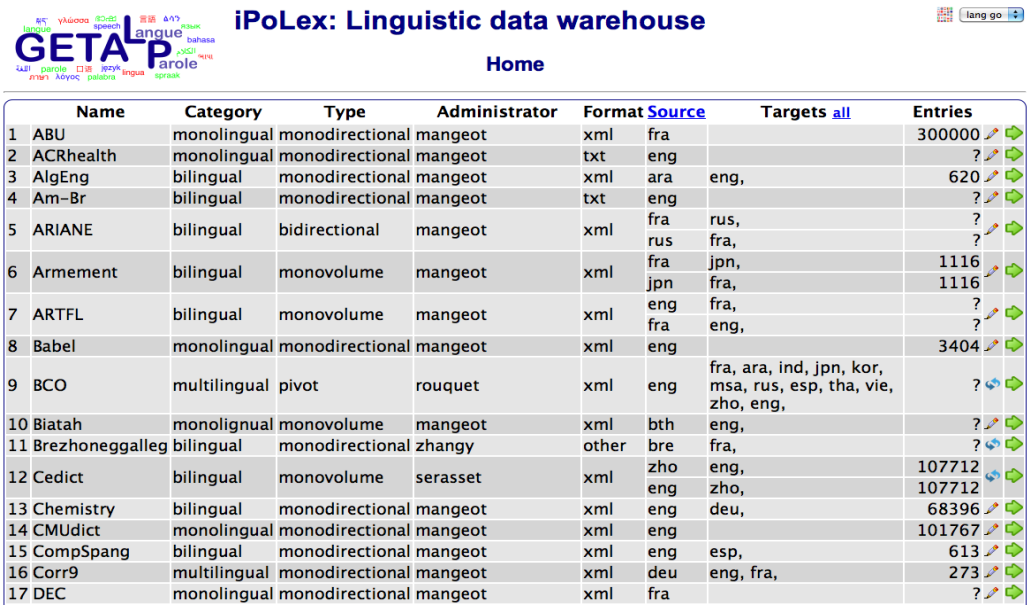
Ce script est intégré au code source de la plateforme iPolex et disponible sur le compte Github de Mathieu Mangeot[6].

5.2. Description de la plateforme iPolex

iPoLex est un « entrepôt lexical à services ». Il est apparu au cours des projets utilisant Jibiki comme une nécessité. Il s'agit en effet de fournir une infrastructure pérenne et visible pour les données lexicales récoltées dans le cadre d'un projet.

Pour clarifier cette idée, on peut utiliser une métaphore : il s'agit de réaliser le pendant, pour les données lexicales, de la fonctionnalité d'une forge qui assure le partage et la construction collaborative de logiciels. L'image de l'entrepôt serait peut-être plus adaptée si l'on réduisait la description de l'outil désiré à cette seule fonctionnalité.

Le service Web iPoLex a pour objectif de stocker toutes les ressources lexicales dans un endroit unique, accessible via le Web, avec accès restreint aux membres du laboratoire. L'accès à l'outil est réservé aux membres du laboratoire. L'authentification se fait automatiquement via l'annuaire LDAP du laboratoire. L'outil, programmé en PHP, se veut simple et rustique. Il n'utilise pas de base de données. Sa page d'accueil présente la liste des ressources disponibles (voir Figure 2). Il est possible de les trier selon leur nom, leur langue source ou leurs langues cibles.



	Name	Category	Type	Administrator	Format	Source	Targets all	Entries
1	ABU	monolingual	monodirectional	mangeot	xml	fra		300000
2	ACRhealth	monolingual	monodirectional	mangeot	txt	eng		?
3	AlgEng	bilingual	monodirectional	mangeot	xml	ara	eng,	620
4	Am-Br	bilingual	monodirectional	mangeot	txt	eng		?
5	ARIANE	bilingual	bidirectional	mangeot	xml	fra	rus,	?
						rus	fra,	?
6	Armement	bilingual	monovolume	mangeot	xml	fra	jpn,	1116
						jpn	fra,	1116
7	ARTFL	bilingual	monovolume	mangeot	xml	eng	fra,	?
						fra	eng,	?
8	Babel	monolingual	monodirectional	mangeot	xml	eng		3404
9	BCO	multilingual	pivot	rouquet	xml	eng	fra, ara, ind, jpn, kor, msa, rus, esp, tha, vie, zho, eng,	?
10	Biatah	monolignual	monovolume	mangeot	xml	bth	eng,	?
11	Brezhoneggalleg	bilingual	monodirectional	zhangy	other	bre	fra,	?
12	Cedict	bilingual	monovolume	serasset	xml	zho	eng,	107712
						eng	zho,	107712
13	Chemistry	bilingual	monodirectional	mangeot	xml	eng	deu,	68396
14	CMUdict	monolingual	monodirectional	mangeot	xml	eng		101767
15	CompSpang	bilingual	monodirectional	mangeot	xml	eng	esp,	613
16	Corr9	multilingual	monodirectional	mangeot	xml	deu	eng, fra,	273
17	DEC	monolingual	monodirectional	mangeot	xml	fra		?

Figure 2 : Page d'accueil du site iPolex : liste des ressources disponibles

Pour chaque ressource, on affiche :

- son nom,
- sa catégorie (monolingue, bilingue, multilingue),
- son type (monovolume, monodirectionnel, bidirectionnel, pivot, etc.),
- le login de l'administrateur de la ressource,
- son format,
- sa langue source,
- ses langues cibles
- et le nombre d'entrées.

On peut ensuite accéder aux métadonnées de la ressource ainsi qu'aux données brutes en cliquant sur les boutons correspondants : crayon ou flèches bleues en cercle pour les métadonnées et flèche verte pour les données brutes. Seuls les administrateurs peuvent modifier les métadonnées d'une ressource existante. Si l'on est administrateur d'une ressource, le bouton affiché pour accéder aux métadonnées sera alors le crayon et sinon les flèches bleues en cercle.

Chaque ressource est décrite par au moins deux fichiers de métadonnées. Le premier fichier décrit la ressource et sa macrostructure. On retrouve les informations affichées sur la page d'accueil de iPolex. La figure 3 montre les métadonnées du dictionnaire DiLAF.

6 https://github.com/mangeot/ipolex/blob/master/pl/dictionary_analysis.pl

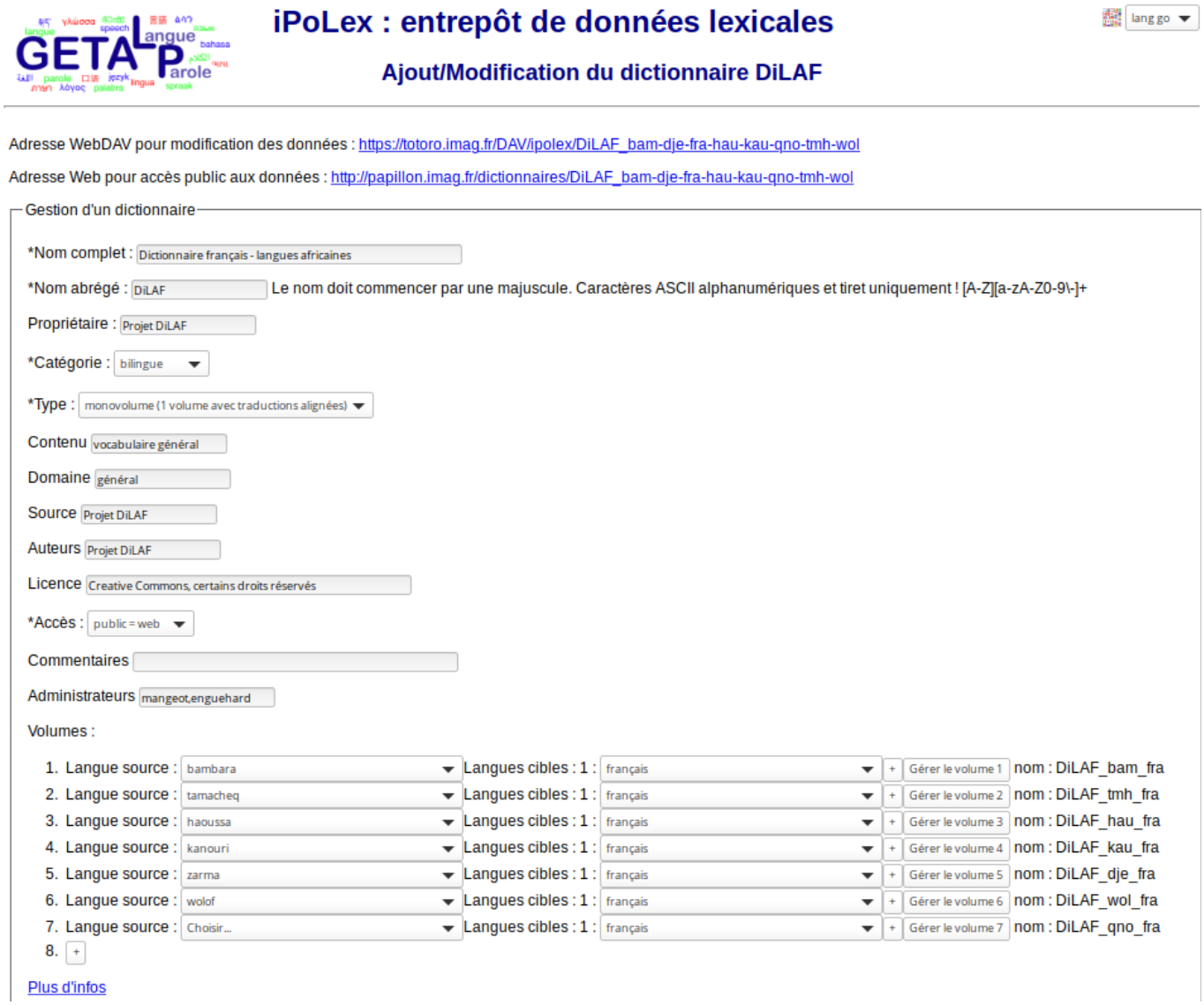


Figure 3 : Description du dictionnaire DiLAF dans la plateforme iPolex

Ensuite, chaque volume qui compose cette ressource est à son tour décrit dans un fichier de métadonnées du volume. Par exemple, un dictionnaire bilingue bidirectionnel sera composé de deux volumes : un volume langue 1 → langue 2 et un volume langue 2 → langue 1 et chaque volume sera décrit par un fichier de métadonnées. Les informations décrites dans ces fichiers sont : le nombre d'entrées, le format du fichier (principalement XML), l'encodage du fichier (principalement UTF-8), les pointeurs CDM XPath décrivant la microstructure et un article squelette vide.

La figure 4 montre les métadonnées du volume DiLAF zarma → français.

Figure 4 : Description du volume DiLAF zarma → français dans la plateforme iPolex

Ces fichiers de métadonnées peuvent être ensuite utilisés directement pour importer la ressource sur la plate-forme Jibiki.

Le code source de la plateforme iPolex est disponible sur le compte Github de Mathieu Mangeot[7]. La plateforme est également disponible prête à l'emploi sous forme d'image Docker sur le compte docker de Mathieu Mangeot[8].

7 https://github.com/mangeot/ipolex/

5.3. Intégration du script d'analyse dans la plateforme iPolex

L'outil iPolex permet également d'importer une nouvelle ressource. C'est lors de ce processus que le script d'analyse d'une ressource lexicale sera utilisé.

Lors de l'import d'une ressource, un répertoire est créé sur le serveur. Ensuite, lors de la saisie des métadonnées, des fichiers correspondants sont générés au format XML et stockés dans le répertoire de la ressource.

La première étape consiste à saisir les métadonnées de la ressource à importer. Ces données sont saisies grâce au formulaire HTML de la figure 3. Lors de l'enregistrement du formulaire, un fichier de métadonnées du dictionnaire au format XML est généré et stocké sur le serveur.

Ensuite pour chaque volume constituant la ressource, il faut d'abord téléverser le fichier de données sur le serveur. Ensuite, le script d'analyse des ressources lexicales décrit précédemment sera utilisé pour calculer le nombre d'articles, les pointeurs CDM décrivant les informations lexicales présentes dans la microstructure ainsi qu'un article squelette vide.

Ces informations sont ensuite affichées dans un formulaire HTML (voir figure 4) pour que l'utilisateur qui importe la ressource puisse rectifier les pointeurs CDM qui sont incorrects (dont les heuristiques de calcul n'ont pas fonctionné). Lors de l'enregistrement du formulaire, un fichier de métadonnées du volume au format XML est généré et stocké sur le serveur.

L'outil trang[9] est également utilisé pour générer un schéma XML décrivant la structure XML du volume. Ce schéma sera ensuite utilisé par la plateforme jibiki pour paramétrer automatiquement l'éditeur d'articles en ligne.

5.4. Intégration dans la plateforme Jibiki

Jibiki (Mangeot & Thévenin, 2004 ; Zhang et al., 2014) est une plate-forme générique en ligne pour manipuler des ressources lexicales avec gestion d'utilisateurs et groupes, consultation de ressources hétérogènes et édition générique d'articles de dictionnaires. C'est un site Web communautaire développé au départ pour le projet Papillon (Sérasset & Mangeot, 2001). La plate-forme est programmée entièrement en Java, basée sur l'environnement Enhydra[10]. Toutes les données sont stockées au format XML dans une base de données (Postgres).

Ce site Web propose principalement deux services : une interface unifiée permettant d'accéder simultanément à de nombreuses ressources hétérogènes (monolingues, dictionnaires bilingues. bases multilingues, etc.) et une interface d'édition spécifique pour contribuer directement aux dictionnaires disponibles sur la plate-forme.

L'import d'une ressource dans la plate-forme se fait automatiquement à partir des fichiers de méta-données générés par l'outil iPolex présenté précédemment. Il suffit d'indiquer à l'interface d'import l'URL des méta-données décrivant le dictionnaire et à partir de là, toutes les autres informations (méta-données des volumes, article squelette, schémas XML, etc.) ainsi que les données sont téléchargées automatiquement puis ajoutées dans la base de données.

L'éditeur est fondé sur un modèle d'interface HTML instancié avec l'article que l'on veut éditer. Le modèle peut être généré automatiquement depuis une description de la structure de l'entrée à l'aide d'un schéma XML. Il peut être modifié ensuite pour améliorer le rendu à l'écran. La seule information nécessaire à l'édition d'un article de dictionnaire est donc le schéma XML représentant la structure de cette entrée.

Plusieurs projets de construction de ressources lexicales ont utilisé ou utilisent toujours cette plate-forme avec succès. C'est le cas par exemple du projet GDEF de dictionnaire bilingue estonien-français (Mangeot & Chalvin, 2006) ou du dictionnaire bilingue japonais-français jibiki.fr[11] (Mangeot, 2016). Le code source de la plateforme Jibiki est disponible sur le compte Github de Mathieu Mangeot[12]. La plateforme est également disponible prête à l'emploi sous forme d'image Docker sur le compte docker de Mathieu Mangeot[13].

6. Conclusion et perspectives

Dans cet article, nous avons montré une série d'outils permettant de mettre en place un processus complètement automatisé d'import d'une ressource lexicale au format XML sur une plateforme de consultation et d'édition en ligne. Le script d'analyse qui génère la signature lexicale et calcule les heuristiques permettant de décrire la microstructure de la ressource à l'aide de pointeurs pour chaque type d'information lexicale est la partie centrale de l'automatisation de ce processus.

Le processus d'import a été testé avec succès sur de nombreuses ressources (environ une centaine à ce jour).

Concernant le script d'analyse d'une ressource lexicale, nous avons envisagé quelques perspectives afin d'améliorer son efficacité.

---

8 https://hub.docker.com/r/mangeot/ipolex

9 http://www.thaiopensource.com/download/

10 http://enhydra.ow2.org/about.html

11 http://jibiki.fr/

12 https://github.com/mangeot/jibiki

13 https://hub.docker.com/r/mangeot/jibiki

L’utilisation d’un devineur de langues permettrait d’automatiser complètement le processus d’import. Toutefois plusieurs difficultés se présentent : les dictionnaires bilingues ou multilingues comportent des segments de texte dans différentes langues, certaines informations textuelles ne sont pas spécifiques à une langue (prononciation, transcriptions, etc.), les segments de texte sont très courts et enfin, seules les langues les mieux dotées sont devinées par ces outils.

Nous souhaitons tester le calcul d’heuristiques avec un réseau de neurones pour comparer les résultats avec nos heuristiques.

Nous aimerions tester également l’utilisation de la signature sur des documents qui ne sont pas des ressources lexicales.

Références

Corréard, Marie-Hélène ; Mangeot, Mathieu (1999) XML- A Solution For LDBs, EDs and MRDs? Proc. of COMPLEX 1999, Pécs, Hungary, 16-19 June 1999, 6 p.

Enguehard, Chantal ; Mangeot, Mathieu (2013) *LMF for a selection of African Languages.* Chapter 7, book "LMF: Lexical Markup Framework, theory and practice", Ed. Gil Francopoulo, Hermès science, Paris, France, 17 p.

Mangeot, Mathieu (2001). *Environnements centralisés et distribués pour lexicographes et lexicologues en contexte multilingue.* Thèse de nouveau doctorat, Spécialité Informatique, Université Joseph Fourier Grenoble I, jeudi 27 septembre 2001, 280 p.

Mangeot, Mathieu (2002) *An XML Markup Language Framework for Lexical Databases Environments: the Dictionary Markup Language.* Proc. of International Standards of Terminology and Language Resources Management, LREC 2002 workshop, Las Palmas, Islas Canarias, Spain, 28 May 2002, pp. 37-44.

Mangeot, Mathieu (2016) *Collaborative Construction of a Good Quality, Broad Coverage and Copyright Free Japanese-French Dictionary.* International Journal of Lexicography, Oxford University Press (OUP), 2016, 31 (1), pp.78-112.

Mangeot, Mathieu ; Enguehard, Chantal (2013) *Des dictionnaires éditoriaux aux représentations XML standardisées. Chapitre 8, livre "Ressources Lexicales : contenu, construction, utilisation, évaluation",* Eds. Nuria Gala & Michael Zock, Linguisticae Investigationes Supplementa, John Benjamins Publishing, Amsterdam, Pays-Bas, 24 p.

Mangeot, Mathieu ; Thevenin, David (2004) *Online Generic Editing of Heterogeneous Dictionary Entries in Papillon Project.* Proc. of COLING 2004, ISSCO, Université de Genève, Switzerland, 23-27 August 2004, vol 2/2, pp 1029-1035.

Zhang, Ying ; Mangeot, Mathieu ; Bellynck, Valérie ; Boitet, Christian (2014) *Jibiki-LINKS: a tool between traditional dictionaries and lexical networks for modelling lexical resources.* Proceedings of the 4th Workshop on Cognitive Aspects of the Lexicon (CogALex) 2014 (Eds. Michael Zock, Reinhard Rapp, Chu-Ren Huang), Dublin, Ireland, 23 August 2014, 12 p.

Jibiki-LINKS: a Tool between Traditional Dictionaries and Lexical Networks for Modelling Lexical Resources

ZHANG Ying[1, 2] Mathieu MANGEOT[1] Valérie BELLYNCK[1] Christian BOITET[1]

1. GETALP-LIG, 41 rue des Mathématiques BP53, 38041 Grenoble Cedex

2. SAS Lingua et Machina, Domaine de Voluceau, Rocquencourt, 78153 Le Chesnay

{ying.zhang, mathieu.mangeot, valerie.bellynck, christian.boitet}@imag.fr

Abstract

Between simple electronic dictionaries such as the TLFi (computerized French Language Treasure)[1] and lexical networks like WordNet[2] (Diller et al., 1990; Vossen, 1998), the lexical databases are growing at high speed. Our work is about the addition of rich links to lexical databases, in the context of the parallel development of lexical networks. Current research on management tools for lexical databases is strongly influenced by the field of massive data ("big data") and by the Web of data ("linked data"). In lexical networks, one can build and use arbitrary links, but possible queries cannot model all the usual interactions with lexicographers-developers and users, that are needed, and derive from the paper world. Our work aims to find a solution that allows for the main advantages of lexical networks, while providing the equivalent of paper dictionaries by doing the lexicographic work in lexical DBs.

1 Introduction

The growing importance of IT in all human activities extends and expands the needs and usages of all key digital resources that include lexical resources. Thus, while applications valuing the linguistic processes rely on increasingly abstract representations, modelled for computer operations, it remains that models coming from the historical construction of resources foster human understanding, and therefore, the building of tools for studies centring on the humanities.

In this this section, we place the emergence of the concept of lexical database between electronic dictionaries and lexical networks. We show that this concept is still valid, that it is still necessary to enrich it, and that our work on improving tools for lexical databases helps solve real problems.

To do this, we analyse in the second section the evolution of lexical resources in 4 main steps (simple electronic dictionaries, simple lexical databases, multilevel and multiversion lexical databases, and lexical networks) and present the associated problems. In the third section, we present Jibiki-LINKS, a platform for building multilingual lexical databases that enriches the Jibiki generic platform by introducing the concept of *rich link* between the components it manages (dictionary entries and dictionary volumes). Finally, we show that it allows the construction of lexical databases such as Pivax-UNL, which support scaling up.

2 From computerized dictionaries to lexical databases with rich links

The first computerized lexical resources are electronic versions of printed dictionaries, mainly monolingual or bilingual. The use of computers has helped to overcome the constraints of the paper form. The impossibility to inverse bilingual dictionaries led to a model having a "pivot" consisting of axies[3]. Lexical pivot-based databases are invertible and transitive, but rooted on the form of the

[1] http://atilf.atilf.fr

[2] http://wordnet.princeton.edu

[3] "Axie" = "interlingual meaning," by analogy with "lexie".

155

Zock/Rapp/Huang (eds.): Proceedings of the 4th Workshop on Cognitive Aspects of the Lexicon, pages 87–98,
Dublin, Ireland, August 23, 2014.

symbols, while the lexical networks allow a move towards the direct manipulation of semantic tokens, regardless of their surface form, and thus of the language.

In this section, we present the evolution of approaches, distinguishing four main types of lexical resources, the limitations that motivated this evolution, and the remaining hard problems.

2.1 Simple electronic dictionaries

A simple electronic dictionary is an electronic version of a printed dictionary, or the computer representation of a new kind of the same type of dictionary, for example, the TLFi[4], the morphological and bilingual dictionaries of Apertium[5], etc. A simple electronic dictionary contains either one volume or two volumes. The electronic version of a monolingual paper dictionary is (usually implicitly) based on its *microstructure*, that is to say, on the organization of its entries in the form of a small tree organizing the information it contains. In a paper dictionary, the presentation of an entry reflects the microstructure, but the microstructure is not always directly retrievable from it (for example, parts in italics can correspond to different types of information units, such as idiom or example of use).

In absolute terms, it is always possible to represent the information specified in each entry of a dictionary according to a common structure. In reality, the structures of paper dictionaries are less rigorous than what would be required for automatic processing, so that manual editing is required.

A bilingual paper dictionary is generally based on a structure in two volumes, one for each language pair, each volume conforming to the same microstructure. There are therefore generally one volume from language A (Lg A) to language B (Lg B) and a mirror volume from Lg B to Lg A. We define the *macrostructure* of a dictionary as the organization of the volumes that make up its structure. These macrostructures constitute the bulk of the printed dictionaries.

2.2 Lexical databases

A lexical database is a tool for unifying any set of dictionaries, where each dictionary can be monolingual, bilingual or multitarget. A multilingual lexical database is composed of volumes that are monolingual, direct multilingual, or indirect multilingual, i.e. connecting the entries of different languages via a pivot structure. It has an overall macrostructure, and a microstructure for each of its volumes. A link between 2 entries is realized by the software tool as a direct link, or as 2 links going through an intermediate language, or as a semantic link, etc.

The lack of symmetry of the correspondence between the entries of bilingual dictionaries (from word senses to words, not word senses) led to the concept of *interlingual pivot*. In the pivot macrostructure developed and used for the Papillon-NADIA multilingual lexical database (Sérasset and Mangeot, 2001), there is only one monolingual volume for each language. *Lexies* are word senses (of a lexeme or an idiom) and make up the entries of these volumes. To group the lexies of different languages together, there is a pivot volume of *axies* (interlingual acceptions). An axie connects synonymous lexies. The links are established only between lexies and axies. This is the simplest macrostructure for a pivot-based multilingual lexical resource that allows for the extraction of usage dictionaries for all pairs in all directions. The concept of axie-based pivot structure has been validated by the Papillon project and then included in the Lexical Markup Framework standard (Francopoulo et al. 2009).

2.3 Multilevel and multiversion databases

In this type of lexical database, several monolingual volumes are allowed for each *lexical space*[6], A volume of *axemes* (monolingual acceptions) is introduced to link synonymous lexies of the considered lexical space. Also, various levels are introduced to tag entries according to different points of view (sublanguage, version, type of link, reliability, preference). The simple links of previous versions are replaced by *rich links* that can be established not only between lexies, axemes and axies, but also between entries and subentries, monolingually (lexicosemantic functions) or bilingually (translations).

---

[4] Trésor de la Langue Française informatisé, http://atilf.atilf.fr/

[5] http://wiki.apertium.org/wiki/User:Alessiojr/Easy_dictionary_-_Application-GSOC2010

[6] A lexical space of a natural language contains various levels (wordform, lemma, lexie, prolexeme, proaxeme); it can also contain the lexical symbols of an artificial semantic representation language (e.g., the UWs of UNL).

For example, there is a 3-level macrostructure (lexie, axeme, axie) in PIVAX (Nguyen & al., 2007) and a 4-level macrostructure (lexie, prolexeme, proaxie, axie) in ProAxie (Zhang & Mangeot, 2013), described in more detail in section 4.1. Both allow us to manage one or more monolingual volumes for each lexical space. That has been quite useful in the ANR Traouiero GBDLex-UW++ subproject, during which we stored the UNL part of many UNL-Li dictionaries (the UW interlingual lexemes, built with slightly different conventions by different UNL groups for their languages), and tried then to unify them in a new monolingual UNL dictionary (using a set of "UW++" built from WordNet and from the previous UNL dictionaries).

2.4 Lexical network

A lexical network brings together the set of words that denote ideas or realities that refer to the same theme, as well as all the words that, because of the context and certain aspects of their meaning, also evoke this theme[7]. The theme may possibly be very broad. It is possible to represent the full vocabulary of a language in a lexical network, such as, for French, the JeuxDeMots network (Lafourcade and Joubert, 2010) or RFL (Lexical Network of French (Lux-Pogodalla, Polguère 2011)).

Lexical networks are traditionally represented as graphs. Nodes represent the lexemes of one or more languages, and links represent the relationships between these lexemes (translation, synonymy, etc.). A lexical network can be monolingual or multilingual. One can create syntactic, morphological and semantic relations between lexemes.

Although lexical networks have many advantages, they are not suitable for all usages. For example, lexical networks like WordNet (Diller & al., 1990; Vossen, 1998), HowNet (Dong et al., 2010) and MindNet (Dolan and Richardson, 1996) (Richardson et al., 1998) are not browsable in alphabetical order. But we need that possibility to have an idea of the content of a lexical repository, whatever its nature, or to play word games, or to find a word one has on the tip of the tongue[8]. On the other hand, in a lexical network, the concept of volume is missing, which prevents to create a resource in a simple way when studying a new language.

For example, the lexical network DBNary (Sérasset, 2012), which is based on the Lemon model (McCrae et al., 2011), contains millions of terms, but does not allow labelling the links. To navigate in this system, one must write SPARQL queries, which is not within the reach of everyone.

2.5 Conclusion: features, limitations and hard problems

Research efforts focus today mainly on lexical networks, but much remains to be done on the preceding types (pivot, multilevel). In particular, the import of lexical databases in lexical networks causes a loss of information, especially information born by the attributes of rich links. For example, what concerns the history, the etymology or the evolution of word senses is not systematically imported into lexical networks. They therefore cannot meet the needs of the humanities, nor allow the transition to "digital humanities."

A lexical network is actually the type of structure that enables the greatest freedom of representation. Indeed, we can create entries and links arbitrarily. But the possible queries cannot model all the usual interactions with lexicographers-developers and users, which come from the world of paper, and are felt necessary. They allow us to represent all categories of lexical resources, but the analogy with the real world is lost. Thus, the practical expertise of linguists-lexicographers is lost.

We must continue to equip lexical databases, because that is the right level to transfer the techniques used by lexicographers-linguists. Also, modelling by a volume-based macrostructure allows keeping a link to the original paper world. Moreover, there are already reusable resources of these types. That is why we focus on the management of resources having multiversion and multilevel macrostructures.

3 Reuse of rich links

In this section, we present an improvement that consists in introducing into lexical databases relational

[7] http://ddata.over-blog.com/xxxyyy/3/12/82/15/GRAMMAIRE/champs-et-reseaux-lexicaux.pdf

[8] For that kind of functionality, multiple sorting on subsets of inflected forms and on arbitray types of information seems to be a necessary first level of computer aid.

information in the form of *rich links* that will bring them closer to lexical networks. An important point is that these links may bear arbitrary labels.

3.1 Presentation of the Jibiki platform

Jibiki is a generic platform that enables the construction of contributive websites dedicated to the construction of multilingual lexical databases. That platform has been developed mainly by Mathieu Mangeot (Mangeot & Chalvin, 2006) and Gilles Sérasset (Sérasset & Mangeot, 2001). It has been used in various projects (EU LexALP project, Papillon project, GDEF project, etc.). The code is available in open source, and freely downloadable by SVN from ligforge.imag.fr. With this platform, one can perform import, export, edit and search operations in lexical databases. One can also manage the contributions. Jibiki allows handling almost all lexical resources of XML type, by using different microstructures and macrostructures.

In the Jibiki approach, resources are organized in *volumes*, which makes it easier to achieve the equivalent of paper dictionaries, keeping the mental image of the representation of the dictionary, while offering new interactions allowed in the digital world. Usages of dictionaries in Jibiki are also similar to those of paper dictionaries. For example, one can consult a database in alphabetical order, indicate a source and/or target language, group lexies in vocables, navigate in a volume, etc.

3.2 Classical Common Dictionary Markup

Version 1 of Jibiki uses "CDM pointers" (Common Dictionary Markup (Mangeot, 2002)) to import, view and edit any type of microstructure without modifying it. CDM pointers are also used to index specific parts of the information, and then allow a multi-criteria search.

Each CDM pointer indicates the path (XPath) to the corresponding element in the XML microstructure of the described resource (see Figure 1). Its description is stored in a XML metadata file. When the resource is imported in the Jibiki platform, the pointers are computed, and the result is stored in a table of the (postgresql) database, for each volume. This table is considered as an indexing table.

```
<cdm-elements>
 <cdm-volume xpath="/g:volume"/>
 <cdm-entry xpath="/g:volume/g:article"/>
 <cdm-entry-id xpath="/g:volume/g:article/@g:id" />
 <cdm-headword xpath="/g:volume/g:article/g:vedette/g:mot/text()" d:lang="fra" />
 <cdm-pronunciation xpath="//g:prononciation/text()" d:lang="fra" />
 <cdm-pos xpath="//g:cat-gram/text()" d:lang="fra" />
 <cdm-definition xpath="/g:volume/g:article/g:vedette/g:mot/text()"/>
</cdm-elements>
```

Figure 1: CDM pointers for the French volume of the GDEF[9] resource (Mangeot and Chalvin, 2006)

CDM tags	FeM[10] (Gut et al., 1996)	OHD[11]	JMdict[12] (Breen, 2004)
Volume	/volume	/volume	/JMdict
Entry	/volume/entry	/volume/se	/JMdict/entry
Entry ID	/volume/entry/@id		/JMdict/entry/ent_seq/text()
Headword	/volume/entry/headword/text()	/volume/se/hw/text()	/JMdict/entry/k_ele/keb/text()
Pron	/volume/entry/prnc/text()	/volume/se/pr/ph/text()	
PoS	//sense-list/sense/pos-list/text()	/volume/se/hg/ps/text()	/JMdict/entry/sense/pos/text()
Domain		//u/text()	
Example	//sense1/expl-list/expl/fra	//le/text()	/JMdict/entry/sense/gloss/text()

Table 1: Examples of Common Dictionary Markup

[9] GDEF is a large Estonian-French dictionary that is being created by the Franco-Estonian lexicography association (see http://estfra.ee/GDEF.po).

[10] FeM is a French-English-Malay dictionary (30000 entries, 50000 lexies, 8000 idioms, 10000 examples of use).

[11] OHD is abbreviation of Oxford-Hachette Dictionary, which is a French-English dictionary.

[12] JMdict is a Japanese-multilingual dictionary.

The translation links are treated at this stage with conventional CDM pointers, as classical information elements. It is not possible to index other information carried by the links, such as weights or labels.

Hence, multilevel macrostructures cannot be modelled in a generic manner with Jibiki-v1 and traditional CDM pointers. For example, it is not possible to link the same volume to several volumes at different levels. This has forced us initially to use palliatives that did not scale up. It became necessary to modify the conceptual model. We addressed these shortcomings in a new version, Jibiki-LINKS.

Table 1 above is an example of CDM for the different resources.

3.3 New version of Jibiki with CDM LINKS

To manage multilevel macrostructures, we enriched the CDM with a richer description of the links (see Figure 2). For each link, more information can be indexed:

- the identifier of the source entry.
- the identifier of the target entry.
- the identifier of the XML element of the source entry containing the link. For example, the sense number in a polysemous entry having a translation link for each translation direction. That allows us to precisely retrieve the origin of the link.
- the link name. It is used to distinguish between different types of links in a single entry, such as a translation link and a synonymy link.
- the target language (three-letter code ISO 639-2 / T).
- the target volume.
- the type of link. Some types are predefined, because they are used by the algorithms that compute the rich links (translation, axeme, axie), but it is possible to use other types of links.
- a label whose text is arbitrary.
- a weight whose value must be a real number.

These links can be established between two entries of the same volume or between two different volumes. The same volume may group entries connected to several volumes.

To realize the implementation of rich links, we separated the module processing the links from the module processing other CDM pointers. It means we have two CDM tables in the database associated to each volume. The first stores CDM traditional pointers, and the second CDM LINKS. All information of LINKS can be found in this table.

```
<cdm-elements>
  <cdm-volume    xpath="/p:volume"/>
  <cdm-entry    xpath="/p:volume/p:vocable"/>
  <cdm-entry-id    xpath="/p:volume/p:vocable/@p:id"/>
  <cdm-headword    xpath="/p:volume/p:vocable/p:lemma/text()"/>
  <cdm-headword-variant    xpath="/p:volume/p:vocable/p:altspelling/text()"    d:lang="eng"/>
  <cdm-pos xpath="/p:volume/p:vocable/p:pos/text()" />
  <cdm-sense-id    xpath="/p:volume/p:vocable/p:lexie/@p:id"/>
  <links>
     <link name="axeme" xpath="/p:volume/p:vocable/p:lexie/p:entryref">
          <type xpath="@type" />
          <volume xpath="@volume" />
          <value xpath="@p:idref" />
          <lang xpath="@lang" />
          <label xpath="@p:relation-mono" />
     </link>
  </links>
</cdm-elements>
```

Figure 2: CDM-LINKS for the English volume of the CommonUNLDict resource

3.4 Approach by rich links in searching in a complex lexical network

To explain how we create arbitrary links, let us give an example. A free label is available for each link. For example, in a lexical resource including SMS, in French "A+" has a link to "Over" with a "SMS" label, in English "L8R" corresponds to "later" with a "SMS" label, and the label of the link between "Over" and "later" is "translation."

A ProAxie macrostructure (Zhang & Mangeot, 2013) has been implemented on the Jibiki-Links platform. We present another example of rich links for semantic search in section 4.1.

3.5 Algorithms for computing rich links

The computer implementation is based on two algorithms. The first collects the links, and the second builds the result. More precisely, the first looks for all possible links in the set of all rich links of all volumes, for a desired entry. The second recursively performs the following steps: (1) selection of the start entry; (2) search of the links to other entries; (3) treatment of labels; (4) recursive call of the algorithm on the connected entry; (5) integration of the XML code of the entry connected to the start entry; (6) display.

4 Experimentation

4.1 Examples of multilevel macrostructures

We have already installed several multilevel macrostructures on Jibiki-LINKS. Here are 3 examples.

MotÀMot: trilingual lexical database with a pivot structure (Mangeot & Touche, 2010)

This project (2009-2012) has computerized a French-Khmer classical dictionary, initially in Word, into a Jibiki database (see http://jibiki.univ-savoie.fr/motamot/).

The macrostructure is composed of a monolingual volume for each language and a central pivot volume. However, in order not to confuse users, the contributing interface shows a classic view of a bilingual dictionary. Each bilingual link language A → language B added via this interface is actually translated into the background by creating two interlingual links as well as an axie link representing the original translation, to finally get: language A → pivot axie → language B (see Figure 3).

If a contributor wants to add a translation link between a vocable Va of language A and a vocable Vb of language B, s/he can establish this link at different levels. The ideal solution is to connect a word meaning (lexie) La of the vocable Va to another word meaning Lb of the vocable Vb. In this case, the link is bijective and Lb is also connected to La.

If the contributor cannot choose between word meanings, s/he can connect directly the word meaning La to the vocable Vb and the link is tagged for refinement.

With the pivot macrostructure, if two links language A → language B and language B → language C exist, then it will automatically create a link language A → language C tagged for refinement.

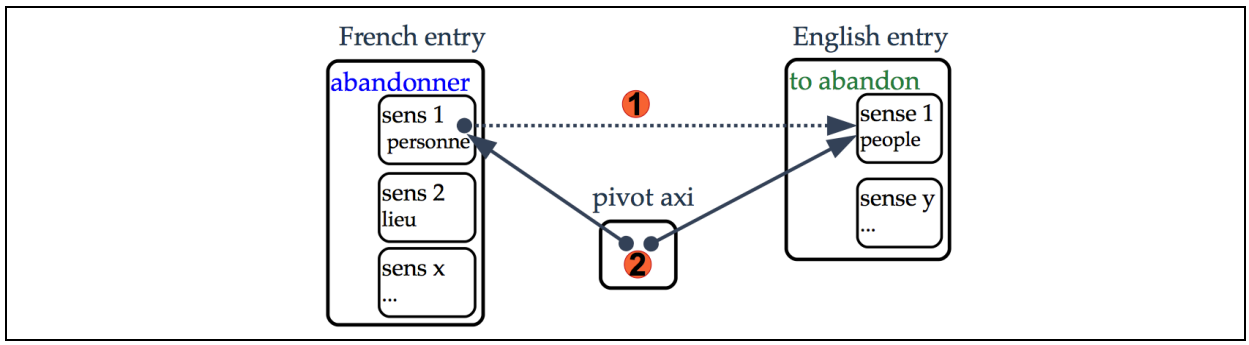


Figure 3: Example of MotÀMot

ProAxie: multilingual extension of ProxlexBase (Tran, 2006)

The ProAxie macrostructure aims at solving the problem of linking several terms that refer to one and the same referent, in particular for the management of acronyms (Zhang et Mangeot, 2013). In this macrostructure, there are two different layers. The base layer consists of two types of volume: volumes of lexies and volumes of axies. The axies are used to connect the lexies that match each other exactly. For example, one translates "ONU" by "UN" (see Figure 4) from French into English.

The "Pro" layer allows us to propose to users translations having the same referential meanings. This layer includes the volumes of *prolexemes* (Tran, 2006) and one volume of *proaxies*. A prolexeme entry links lexies having the same meaning with a label (aka, acronym, definition, etc.). A proaxie entry connects prolexemes of different languages. If one cannot find the translations directly using the lower layer, one will get the translations proposed by the "Pro" layer.

For example, for "Nations-Unies", translations by "United Nations" and "UN" will be proposed, with the "alias" label.

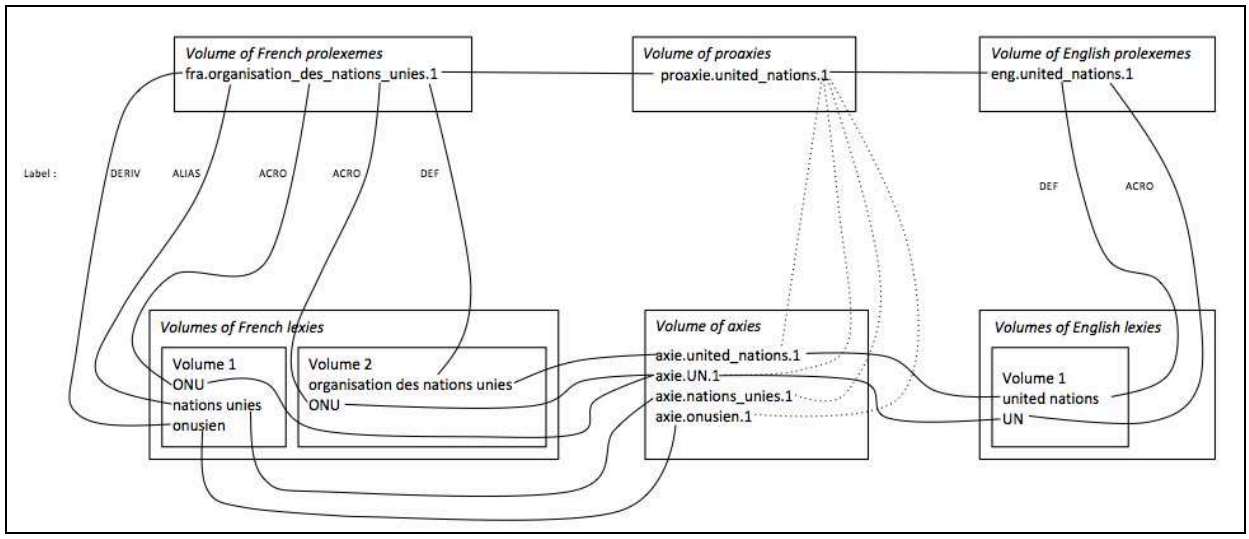


Figure 4: Example of ProAxie

For each natural language, there are one or more volumes of lexies, and a single volume of prolexemes. For each dictionary, there is a volume of axies and a volume of proaxies.

This gives three levels of translation, classified according to the precision obtained.

(1) The system finds a lexie directly, using the volume of axies. That is the first and most accurate level of translation.

(2) The system searches a link to the prolexemes volume of the source language with a certain label. When it finds the link in the proaxies volume, it follows the prolexeme link of the target language, and finally arrives at the volume of lexies in the target language, and finds a lexie that has the same label. That is the second, intermediate level.

(3) The system finds the lexies going through prolexemes and proaxies, without a corresponding label. These proposed lexies constitute the third and least accurate level.

Pivax: lexical multilingual multiversion database with 3 levels

The Pivax macrostructure has three levels: *lexie, axeme* and *axie* (Nguyen & al., 2007). Axemes are monolingual acceptions, and group monolingual lexies having the same meaning. Axies group synonymous axemes of different languages in a central "hub". In some situations, a lexical database has several volumes for a single language. For example, when there are several editions, or when the lexical resource is created for a machine translation system: one may have one volume coming from Systran, one from Ariane/Héloïse, one from IATE[13], etc. This macrostructure allows us to manage multiple volumes in the same language. Given a language, there are one or more volumes of lexies and a single volume of axemes. For any Pivax database, there is only one volume of axies. The links between the lexies and the axemes and between the axemes and the axies are rich links with attributes such as type, target volume, target language, free label, weight, etc.

4.2 CommonUNLDict: toward scaling up with a resource of Pivax type

In this section, we present the CommonUNLDict resource that uses the Pivax macrostructure. We have implemented this resource on the Pivax-UNL platform, which is an instance of Jibiki-Links. Users can easily use this resource via the link http://getalp.imag.fr/pivax/Home.po.

Resource created by linguists

Thanks to CDM-LINKS, all types of XML formats can be used in an instance of Jibiki-LINKS without modification. One needs only simple knowledge about XML to create a resource for Jibiki-LINKS. In addition, very useful available tools can be used to create an XML file, such as oXygen[14] that allows the creation of a DTD using a graphical interface.

[13] "A single database for all EU-related terminology (InterActive Terminology for Europe) in 23 languages opens to the public", 2007)

[14] http://www.oxygenxml.com

The CommonUNLDict resource has been created by the Russian lexicographer and linguist Viacheslav Dikonov (Dikonov & Boguslavsky, 2009). Figure 5 shows the graph of a monolingual volume structure using oXygen. In this example, each volume contains a large quantity of vocables, and each vocable includes one or more lexie. We will explain this structure in section 3.2.3.

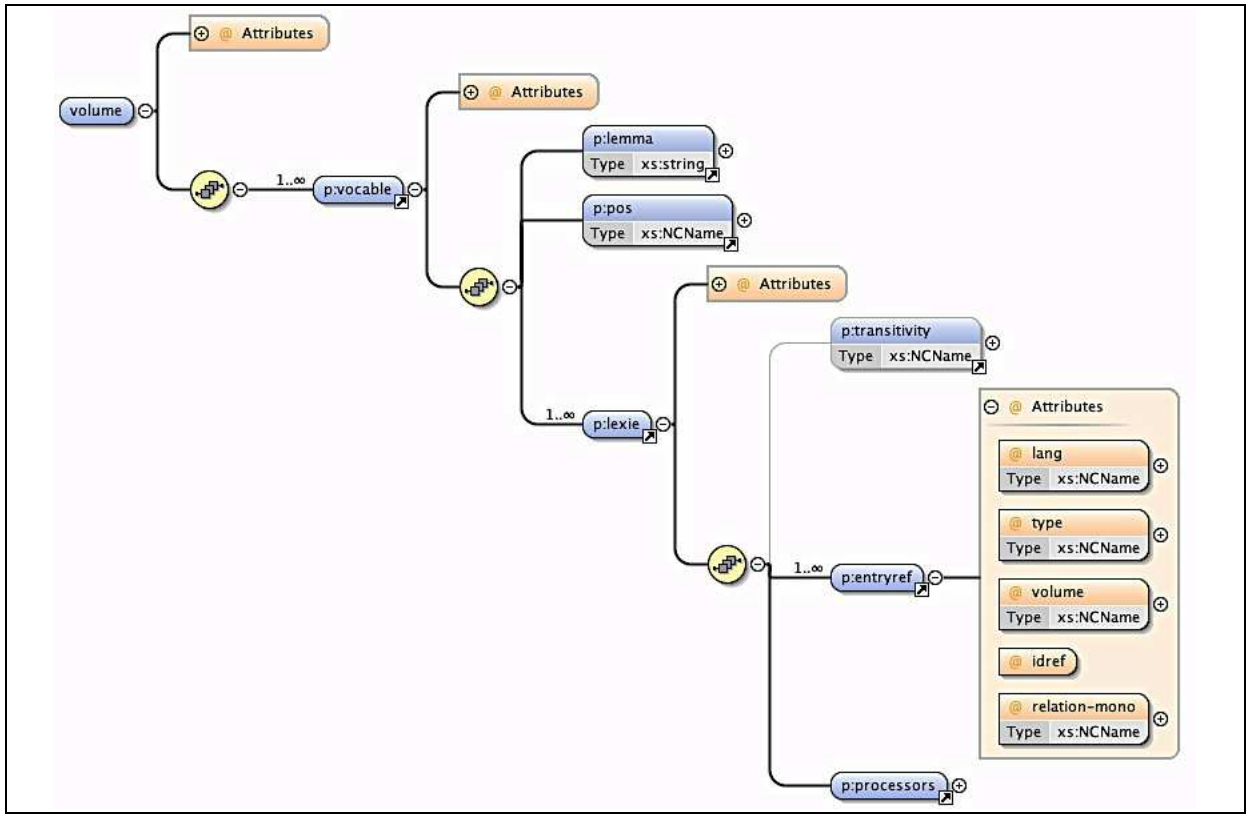


Figure 5: Structure of a monolingual volume

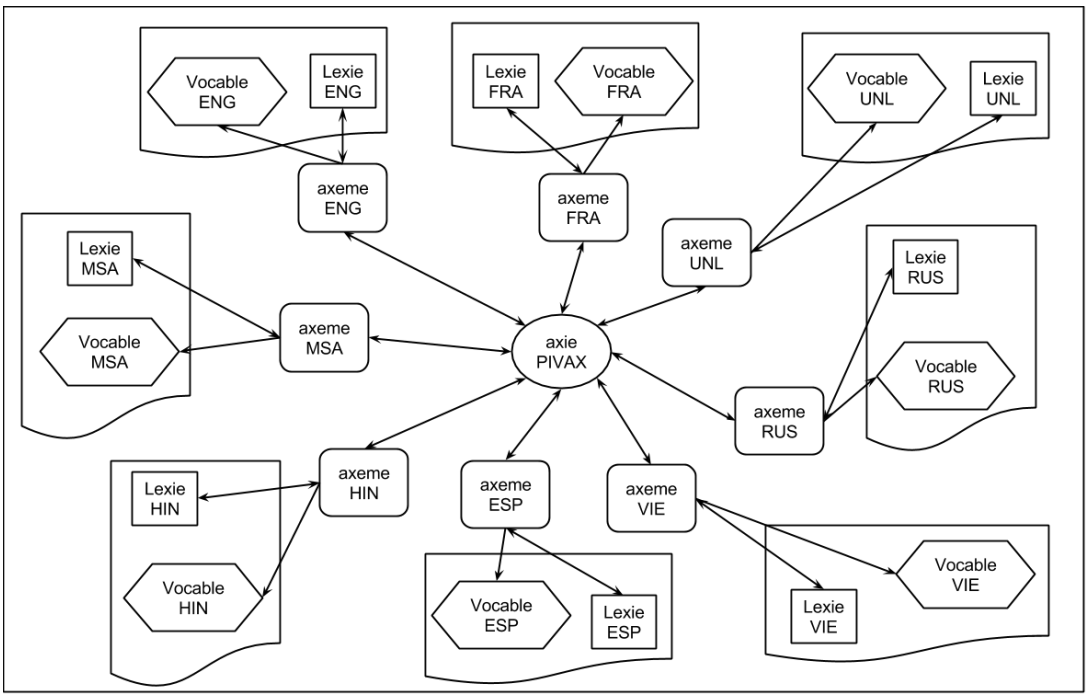


Figure 6: Macrostructure of CommonUNLDict

Macrostructure of CommonUNLDict

CommonUNLDict contains 8 languages (7 natural languages, French, English, Hindi, Malay, Russian, Spanish, Vietnamese, and the UNL language) and 17 volumes (8 volumes of monolingual data, 8 volumes of monolingual axemes, and 1 volume of axies ("interlingual meanings"). The macrostructure of CommonUNLDict is diagrammed in Figure 6. For each language, there is only one volume of monolingual data (vocables and lexical items) and a single volume of axemes. For the whole CommonUNLDict, there is only one volume of axies.

Microstructure of CommonUNLDict

The microstructure is the structure of the entries (Mangeot, 2001). In the CommonUNLDict resource, there are three types of entries (vocables, axemes and axies) and 720 K entries in total.

See Table 2.

Volume	*Language*	*Entries*
CommonUNLDict_axi	axi	82804
CommonUNLDict_eng	English	45471
CommonUNLDict_eng-axemes	English	82069
CommonUNLDict_esp	Spanish	7080
CommonUNLDict_esp-axemes	Spanish	22254
CommonUNLDict_fra	French	27537
CommonUNLDict_fra-axemes	French	48312
CommonUNLDict_hin	Hindi	31255
CommonUNLDict_hin-axemes	Hindi	50380
CommonUNLDict_msa	Malay	37342
CommonUNLDict_msa-axemes	Malay	31699
CommonUNLDict_rus	Russian	28475
CommonUNLDict_rus-axemes	Russian	45020
CommonUNLDict_unl	unl	82804
CommonUNLDict_unl-axemes	unl	82804
CommonUNLDict_vie	Vietnamese	6585
CommonUNLDict_vie-axemes	Vietnamese	8819

Table 2: Number of entries of CommonUNLDict

All volumes of the same type have the same microstructure. The example below (see Figure 7) shows the microstructure of a volume of *vocables*. Each entry of vocable type allows us to describe all detailed information, such as part of speech (POS), pronunciation, etc. Each vocable includes one or more lexies (word senses). Figure 2 shows an example. Therefore the number of axemes is greater than or equal to the number of vocables. In this microstructure, the "entryref" attribute allows us to manage the links between lexies and the entries of axeme type.

```
<p:vocable p:id="fra.ADN.n">
                <p:lemma>ADN</p:lemma>
                <p:pos>n</p:pos>
                <p:gender>m</p:gender>
                <p:lexie p:id="CommonUNLDict.lexie.fra.ADN.1">
                        <p:entryref type="axeme" volume="CommonUNLDict_fra-axemes" p:idref="CommonUNLDict
(icl&gt;polymer&gt;thing,equ&gt;deoxyribonucleic_acid)" lang="FRA" p:relation-mono="OTHER"/>
                        <p:processors>
                                <p:processor p:name="Ariane" p:access="Public">
                                <p:procref type="entry" id="ADN" var="CAT(CATN),GNR(MAS)" lang="FRA"/>
                                </p:processor>
                        </p:processors>
                </p:lexie>
</p:vocable>
```

Figure 7: Microstructure of a volume of lexies

- In this example, the value of "type" is the type of link, the value of "volume" is the target volume, the value of "idref" is the identifier of the axeme entry, the value of "lang" is the target language, and the value of "relationship-mono" is the label.

- The microstructure of the entries of axeme type allows us to describe the links with entries of lexie type and the links with entries of axie type. The microstructure of the axies allows us to describe the links with the entries of axeme type.

Response time and use case

The tests were performed with an instance of Jibiki-LINKS installed on a machine with an Intel Core i3 processor at 3.3 GHz with 8 GB of RAM.

The tool used to perform queries is wget. The command is run directly on the server to avoid the latency due to the network. We give three examples in Table 3, which show the number of links computed by the system, of entries displayed, of queries, of different languages, and the average response time. The response time, less than 1 second in these cases, is generally satisfactory. For better understanding, there is some details about the example "manger" (see Figure 8). We search "manger" in French, and find one entry with id "fra.manger.v" in the French vocable volume. The search direction is "up". This entry links to another entry of the volume of French axemes, whose id is CommonUNLDict.axeme.fra.eat(icl>consume>do, agt>living_thing, obj>concrete_thing, ins>thing)[15]. This axeme entry links with one axie entry and the vocable entry fra.manger.v. Because the search direction is "up", we just go to the axie entry. When we arrive in the volume of axies, the search direction is changed to "down". The axie entry links to 6 different axeme entries. We search each axeme entry and its links. Because the search direction is "down", we only take into account vocable entries links. For each axeme entry, we find at least one vocable entry. In other cases, one vocable entry has more than one lexie, so it links to one or several axeme entries, and there are more links.

Search argument	Links	Displayed entries	Number of requests	Different languages	Average time (ms)
French vocable "manger"	14	6	10	6	19.7
French vocable "recherche"	66	27	10	6	73.5
UNL "search(icl>action)"	51	20	10	6	56

Table 3: Response time on three examples

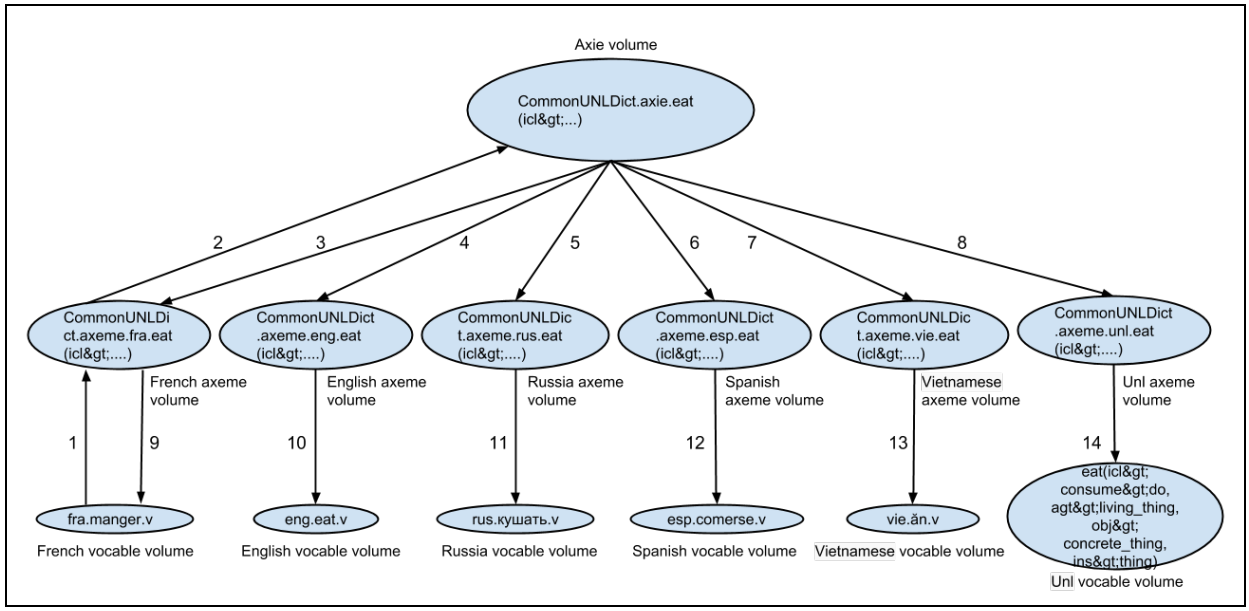


Figure 8: Links in the case of "manger"

Figure 9 shows the display of the interface for a classical search in a Web browser.

[15] In order to better display figure, we have simplified the id in figure 8.

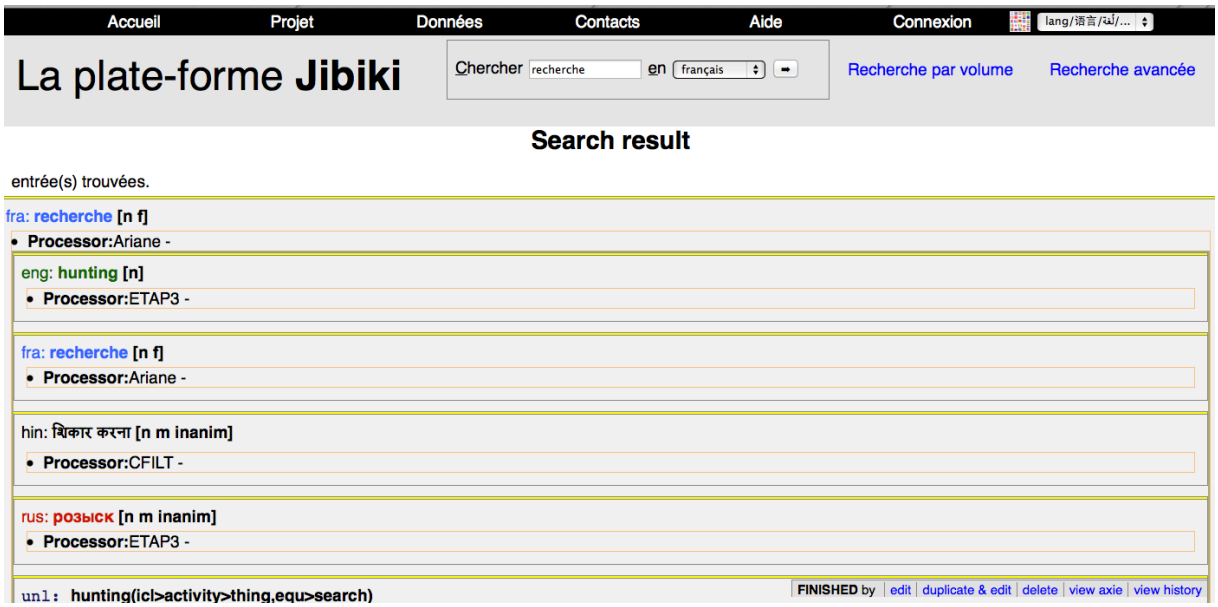


Figure 9: Display of the interface for a classical search

5 Conclusion and perspectives

In this article, we analysed the different types of lexical resource and presented a method of modelling lexical resources using *volumes*. This method allows us to manage complex resources while providing facilities for manipulation and treatment equivalent to those of a paper dictionary.

Jibiki-LINKS is a new version of the Jibiki platform, which can manage resources based on multilevel macrostructures using rich links, bearing attributes such as target volume, weight, type, language, open label, etc. To realize the implementation of rich links, we separated the module processing the links from the module processing other CDM pointers. Jibiki-LINKS has been used to implement the MotÀMot, ProAxie and Pivax macrostructures.

On the Pivax-UNL platform, another instance of the Jibiki-LINKS-based Pivax macrostructure, we have installed the volumes corresponding to the CommonUNLDict resource of V. Dikonov, and tested our platform with that resource.

There is also a UW (UNL interlingual lexemes) resource of 8G entries that was created from DBpedia by David Rouquet. In that resource, there are several volumes for the same language. As links were poorly structured, we are currently working on this resource in order to recompute them. We hope to be able to import this resource, and to make tests at that very large scale in the near future.

To sum up, lexical databases equipped with rich links allow for importing XML-based electronic dictionaries without loss of information, whether they have been elaborated from source or printable forms (such as Word, rtf, ps, pdf) or directly produced in XML from a relational database, or using a dedicated editor knowing their microstructures. They also allow us to automatically produce from them a pivot-based macrostructure organised in *volumes*, and after that to edit and improve them, using a mixed textual and graphical interface to merge or split lexies, axemes or axies, or to enrich the links with appropriate labels. The introduction of rich links to multilevel lexical databases enhances them with a very interesting aspect of the lexical networks while keeping the classical ways of using dictionaries and of performing lexicographic work.

References

EU-IATE (2007) A single database for all EU-related terminology (InterActiveTerminology for Europe) in 23 languages opens to the public. *Press release*. Brussels. 2007-06-28.

Breen, J. W., (2004) JMdict : a Japanese-Multilingual Dictionary. In Gilles Sérasset, Susan Armstrong, Christian Boitet, Andrei Pospescu-Belis, and Dan Tufis, editors, post COLING Workshop on Multilingual Linguistic Resources, Geneva, Switzerland, 28th August. International Committee on Computational Linguistics.

Dikonov V., Boguslavsky I., (2009) Semantic Network of the UNL Dictionary of Concepts. *Proceedings of the SENSE Workshop on conceptual Structures for Extracting Natural language SEmantics Moscow*, Russia, July 2009, 7 p.

Diller, G.A., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K.J. (1990) Introduction to WordNet: an on-line lexical database, *International Journal of Lexicography 3(4)*, pp. 235-244.

Dolan, W.B. & Richardson, S.D., (1996) Interactive Lexical Priming for Disambiguation. Proc. *MIDDIM'96, Post-COLING seminar on Interactive Disambiguation*, C. Boitet ed. Le Col de Porte, Isère, France. 12-14 août 1996. vol. 1/1 : pp. 54-56.

Dong, Z.D., Dong, Q., Hao, C.L., (2010). HowNet and Its Computation of Meaning. In Actes de *COLING-2010*, Beijing, 4 p.

Francopoulo, G., Bel, N., George, M., Calzolari, N., Monachini, M., Pet, M. and Soria, C. (2009). Multilingual resources for NLP in the lexical markup framework (LMF). *In journal de Language Resources and Evaluation, March 2009, Volume 43,* pp. 55-57.

Gut, Y., Ramli, P. R. M., Yusoff, Z., Kim, Ch. Ch., Samat, S. A., Boitet, Ch., Nédobejkine, N., Lafourcade, M., Gaschler, J. and Levenbach, D. (1996). *Kamus Perancis-Melayu Dewan, dictionnaire français-malais.* Dewan Bahasa Dan Pustaka, Kuala Lumpur, 667 p.

Lafourcade, M., Joubert, A. (2010). Computing trees of named word usages from a crowdsourced lexical network. *Investigationes Linguisticae*, vol. XXI, pp. 39-56

Lux-Pogodalla, V., Polguère, A. (2011) Construction of a French Lexical Network: Methodological Issues. Proceedings of *the First International Workshop on Lexical Resources, WoLeR 2011. An ESSLLI-2011 Workshop*. Ljubljana, 2011, pp. 54-61.

Mangeot, M. (2002). An XML Markup Language Framework for Lexical Databases Environments: the Dictionary Markup Language. In Actes de *LREC-2002*, pp. 37-44.

Mangeot, M & Chalvin, A. (2006). Dictionary Building with the Jibiki Platform: the GDEF case. In Actes de *LREC-2006*, Genoa, pp. 1666-1669.

Mangeot, M. & Touch, S., (2010) MotÀMot project: building a multilingual lexical system via bilingual dictionaries. Proc. *SLTU 2010: Second International Workshop on Spoken Languages Technologies for Under-Resourced Languages*, Penang, Malaysia, 2010, 6 p.

McCrae, J., Spohr, D. and Cimiano, P., (2011) Linking lexical resources and ontologies on the semantic web with lemon. Proc. *ESWC'11*, Berlin, pp. 245-259.

Nguyen, H.T., Boitet, C. and Sérasset, G. (2007). PIVAX, an online contributive lexical data base for heterogeneous MT systems using a lexical pivot. In Actes de *SNLP-2007*, Bangkok, 6 p.

Richardson, S.D., Dolan, W.B. and Vanderwende, L. (1998) MindNet: acquiring and structuring semantic information from text, no. MSR-TR-98-23.

Sérasset, G. (2012) Dbnary: Wiktionary as a LMF-based Multilingual RDF network. In Actes de *LREC-2012*, Istanbul, 7 p.

Sérasset, G. & Mangeot, M. (2001). Papillon Lexical Database Project: Monolingual Dictionaries and Interlingual Links. In Proc. *NLPRS-2011*, Tokyo, pp. 119-125.

Tran, M. (2006). Prolexbase : Un dictionnaire relationnel multilingue de noms propres : conception, implémentation et gestion en ligne. *Thèse de doctorat*, Tours, pp. 54-57.

Vossen, P., (1998) EuroWordNet: A Multilingual Database with Lexical Semantic Networks, *Computers and the Humanities,* 32(2-3).

Zhang, Y. & Mangeot, M., (2013). Gestion des terminologies riches : l'exemple des acronymes. In Actes de *TALN-2013*, Les Sables d'Olonne, 8 p.

Online Generic Editing of Heterogeneous Dictionary Entries in Papillon Project

Mathieu MANGEOT
Unit Terjemahan Melalui Komputer
Universiti Sains Malaysia,
11800, Pulau Pinang
Malaysia
mathieu@mangeot.org

David THEVENIN
National Institute of Informatics
Hitotsubashi 2-1-2-1913 Chiyoda-ku
JP-101-8430 Tokyo
Japan
thevenin@nii.ac.jp

Abstract

The Papillon project is a collaborative project to establish a multilingual dictionary on the Web. This project started 4 years ago with French and Japanese. The partners are now also working on English, Chinese, Lao, Malay, Thai and Vietnamese. It aims to apply the LINUX cooperative construction paradigm to establish a broad-coverage multilingual dictionary. Users can contribute directly on the server by adding new data or correcting existing errors. Their contributions are stored in the user space until checked by a specialist before being fully integrated into the database. The resulting data is then publicly available and freely distributable. An essential condition for the success of the project is to find a handy solution for all the participants to be able to contribute online by editing dictionary entries.In this paper, we describe our solution for an online generic editor of dictionary entries based on the description of their structure.

1 Introduction

The Papillon Project (Sérasset and Mangeot, 2001) is a cooperative project for a multilingual dictionary on the Web with the following languages: English, Chinese, French, Japanese, Lao, Malay, Thai and Vietnamese. The dictionary structure makes it very simple to add a new language at any time. It aims to apply the LINUX construction paradigm to establish a multilingual usage dictionary with broad-coverage.

This project is based on the participation of voluntary contributors. In order to be really attractive, this project must imperatively find a convenient solution so that contributors can easily edit the dictionary entries. Since Papillon dictionary is available on a web server and the contributors are located all around the world, the obvious solution is to implement an editor available online. Unfortunately, the existing solutions (HTML forms, java applets) have important limitations. Thus, we propose an entirely generic solution that can adapt very easily not only the interfaces to the various entry structures needing to be edited but also to the user needs and competences.

Firstly, we outline the issue addressed in this paper; and draw up an overview of the existing methods for dictionary entry edition. A presentation of the chosen method follows detailing its integration in the Papillon server. Finally, we show an example of the online edition of a dictionary entry.

2 Addressed Issue and Requirements

In this paper, the addressed issue is how to edit online dictionary entries with heterogeneous structures.

2.1 Online Edition

In order to build a multilingual dictionary that covers a lot of languages, we need large competences in those languages. It may be possible to find an expert with enough knowledge of 3 or 4 languages but when that number reaches 10 languages (like now), it is almost impossible. Thus, we need contributors from all over the world.

Furthermore, in order to avoid pollution of the database, we plan a two-step integration of the contributions in the database. When a contributor finishes a new contribution, it is stored into his/her private user space until it is revised by a specialist and integrated into the database. Then, each data needs to be revised although the revisers may not work in the same place of the initial contributors.

Thus, the first requirement for the editor is to work online on the Web.

2.2 Heterogeneous Entry Structures

The Papillon platform is built for generic purposes. Thus, it can manipulate not only the

Papillon dictionary but also any kind of dictionary encoded in XML (Mangeot, 2002). The lexical data is organized in 3 layers:

- Limbo contains dictionaries in their original format and structure;
- Purgatory contains dictionaries in their original format but encoded in XML;
- Paradise contains the target dictionary, in our case Papillon dictionary.

The Purgatory data can be reused for building the Paradise dictionary.

We would like then to be able to edit different dictionaries structures from Paradise but also from Purgatory. Furthermore, being Papillon a research project, entry structures may evolve during the life of the project, since they are not fixed from the beginning.

Hence, the second requirement is that the editor must deal with heterogeneous and evolving entry structures.

2.3 Extra Requirements

Previous requirements must be fulfilled, whilst the following ones are optional.

The contributors will have various competences and use the editor for different purposes (a specialist in speech may add the pronunciation, a linguist may enter grammatical information, a translator would like to add interlingual links, and a reviewer will check the existing contributions, etc.).

The second optional requirement concerns the adaptation to the user platform. The increasing number of smart mobile phones and PDAs makes real the following scenarios: adding an interlingual link with a mobile phone, adding small parts of information with a PDA and revising the whole entry with a workstation.

It would then be very convenient if the editor could adapt itself both to the user and to the platform.

2.4 Final Aim

Guided by these requirements, our final aim is to generate, as much automatically as possible, online interfaces for editing dictionary entries. It has to be taken into account the fact that entry structures are heterogeneous and may vary and to try to adapt as much as possible these interfaces to the different kinds of users and platforms.

3 Overview of Existing Editing Methods

3.1 Local and Ad Hoc

The best way to implement a most comfortable editor for the users is to implement an ad-hoc application like the one developed for the NADIA-DEC project: DECID (Sérasset, 1997). It was conceived to edit entries for the ECD (Mel'čuk et al., 1984889296). The Papillon microstructure is based on a simplification of this structure. We were indeed very interested by such software. It is very convenient - for example - for editing complex lexical functions.

But several drawbacks made it impossible to use in our project. First, the editor was developed ad hoc for a particular entry structure. If we want to change that structure, we must reimplement changes in the editor.

Second, the editor is platform-dependent (here written and compiled for MacOs). The users have to work locally and cannot contribute online.

3.2 Distributed and Democratic

This solution implemented for the construction of the French-UNL dictionary (Sérasset and Mangeot, 1998) project is called "democratic" because it uses common and widespread applications (works on Windows and MacOs) such as Microsoft Word.

The first step is to prepare pre-existing data on the server (implemented here in Macintosh Common Lisp). Then, the data is converted into rtf by using a different Word style for each part of information (the style "headword" for the headword, the style "pos" for the part-of-speech, etc.) and exported. The clients can open the resulting rtf files locally with their Word and edit the entries. Finally, the Word rtf files are reintegrated into the database via a reverse conversion program.

This solution leads to the construction of 20,000 entries with 50,000 word senses. It was considered as a very convenient method, nevertheless, two important drawbacks prevented us to reuse this solution. The first is that in order to convert easily from the database to rtf and vice-versa, the dictionary entry structure cannot be too complex. Furthermore, when the user edits the entry with Word, it is very difficult to control the syntax of the entry, even if some Word macros can partially remedy this problem.

The second is the communication between the

users and the database. The Word files have to be sent to the users, for example via email. It introduces inevitably some delay. Furthermore, during the time when the file is stored on the user machine, no other user can edit the contents of the file. It was also observed that sometimes, users abandon their job and forget to send their files back to the server.

3.3 Online and HTML Forms

In order to work online, we should then use either HTML forms, or a Java applet. The use of HTML forms is interesting at a first glance, because the implementation is fast and all HTML browsers can use HTML forms.

On the other hand, the simplicity of the forms leads to important limitations. The only existing interactors are: buttons, textboxes, pop-up menus, and checkboxes.

JavaScripts offer the possibility to enrich the interactors by verifying for example the content of a textbox, etc. However, very often they raise compatibility problems and only some browsers can interpret them correctly. Thus, we will avoid them as much as possible.

One of the major drawbacks of this solution is our need to modify the source code of the HTML form each time we want to modify the entry structure. We also need to write as many HTML forms as there are different entry structures.

3.4 Online and Java Applets

In order to remedy the limitations of the HTML forms and to continue to work online, there is the possibility to use a java applet that will be executed on the client side. Theoretically, it is possible to develop an ad hoc editor for any complicated structure, like the 3.1 solution.

Nevertheless, the problems linked to the use of a java applet are numerous: the client machine must have java installed, and it must be the same java version of the applet. Furthermore, the execution is made on the client machine, which can be problematic for not very powerful machines. Moreover, nowadays there is a strong decrease of java applets usage on the Web mainly due to the previous compatibility problems.

3.5 Conclusion

As a result, none of these existing solutions can fully fulfil our requirements: online edition and heterogeneous entry structures. We might then use other approaches that are more generic like the ones used in interface conception in order to build our editor. In the remainder of this paper, we will detail how we used an interface generation module in Papillon server in order to generate semi-automatically editing interfaces.

4 Using an Interface Generation Module

This Papillon module has to generate graphic user interfaces for consulting and editing dictionary entries. We base our approach on the work done on Plasticity of User interfaces (Thevenin and Coutaz, 1999) and the tool ART-Studio (Calvary et al., 2001). They propose frameworks and mechanisms to generate semi-automatically graphic user interfaces for different targets. Below we present the design framework and models used.

4.1 Framework for the UI generation

Our approach (Calvary et al., 2002) is based on four-generation steps (Figure 1). The first is a manual design for producing initial models. It includes the application description with the data, tasks and instances models, and the description of the context of use. This latter generally includes the platform where the interaction is done, the user who interacts and the environment where the user is. In our case we do not describe the environment, since it is too difficult and not really pertinent for Papillon. From there, we are able to generate the Abstract User Interface (AUI). This is a platform independent UI. It represents the basic structure of the dialogue between a user and a computer. In the third step, we generate the Concrete User Interface (CUI) based on the Abstract User Interface (AUI). It is an instantiation of the AUI for a given platform. Once the interactor (widget) and the navigation in UI have been chosen, it is a prototype of the executable UI. The last stage is the generation of Final User Interface (FUI). This is the same as concrete user interface (CUI) but it can be executed.

We will now focus on some models that describe the application.

4.2 Application Models: Data & Task

The Data model describes the concepts that the user manipulates in any context of use. When considering plasticity issues, the data model should cover all usage contexts, envisioned for the interactive system. By doing so, designers obtain a global reusable reference model that can be specialized according to user needs or

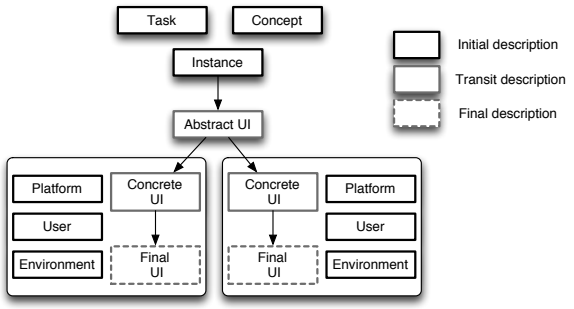


Figure 1: Multitarget Generation Framework

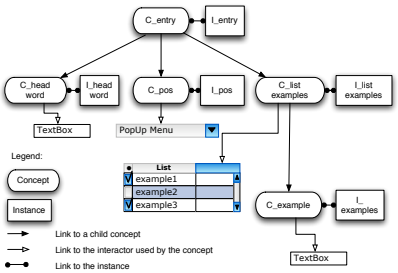


Figure 2: Data Model Structure

more generally to context of use. A similar design rationale holds for tasks modeling. For the Papillon project, the description of data model corresponds to the XML Schema description of dictionary and request manipulation. The tasks' model is the set of all tasks that will be implemented independently of the type of user. It includes modification of the lexical database and visualization of dictionaries.

As showed on Figure 2, the model of concepts will drive the choice of interactors and the structure of the interface.

4.3 Instance Model

It describes instances of the concepts manipulated by the user interface and the dependence graph between them. For example there is the concept "Entry" and one of its instances "scientifique". (cf. Figure 3).

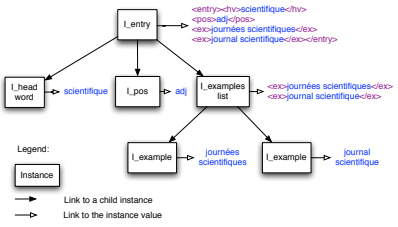


Figure 3: Relation Between the XML Entry and its Corresponding Instance

This model is described at design time, before generation, and linked with the task model (a task uses a set of instances). Each instance will be effectively created at run-time with data coming from the Papillon database.

4.4 Platform and Interactors Models

A platform is described by interaction capacity (for example, screen size, mouse or pen, keyboard, speech recognition, etc.). These capacities will influence the choice of interactors, presentation layouts or the navigation in the user interface.

Associated to the platform there are the interactors (widgets) proposed by the graphic toolsbox of the targeted language (for example Swing or AWT for Java). In this project interactors are coming from HMTL Forms (textBox, comboBox, popup menu, button, checkBox, radioButton) and HTML tags. We also had to build more complex interactors by a combination of HTML Forms and HTML Tags.

4.5 User Model

Previous research has shown the difficulty to describe the cognitive aspects of user behavior. Therefore, we will simplify by defining different user classes (tourist, student, business man, etc.). Each class will be consisting of a set of design preferences. Depending on the target class, the generator will use appropriate design rules.

The model is not yet implemented; it is implicitly used in the data & task models. We defined different views of data according to the target:

- all data is rendered for the workstation editing interface for lexicographers,
- only headword and grammatical class are rendered and examples are browsable on the mobile phone interface for a "normal" dictionary user.

4.6 Concrete User Interface Model

This model, based on an independent user interface language, describes the graphic user interface, as the final device will render it. It is target-dependent.

4.7 Final User Interface

From the CUI model, the generator produces a final interface that will be executed by the targeted device, and links it with the Papillon database. In our case we produce:

- HTML code for the workstation,

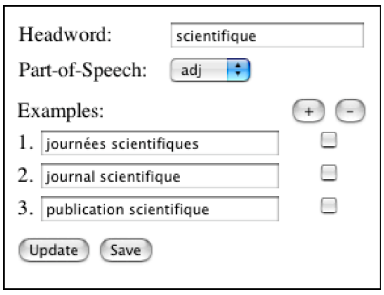


Figure 4: Generated GUI

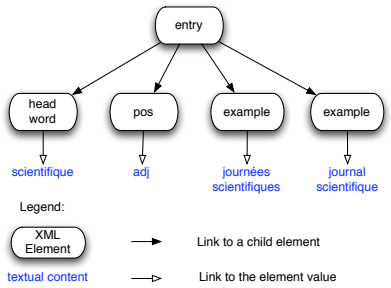


Figure 5: Abstract View of an Entry

- Tiny XHTML code for AU mobile phones,
- and CGI links for the communication with the database.

Figure 4 shows a simple example of a final generated UI.

5 Integrating the Module in Papillon Server

5.1 Implementation

The Papillon server is based on Enhydra, a web server of Java dynamic objects. The data is stored as XML objects into an SQL database: PostgresQL.

ARTStudio tool is entirely written in Java. For its integration into the Papillon/Enhydra server, we created a java archive for the codes to stay independent.

The Papillon/Enhydra server can store java objects during a user session. When the user connects to the Papillon server with a browser, a session is created and the user is identified thanks to a cookie. When the user opens the dictionary entry editor, the java objects needed for the editor will be kept until the end of the session.

5.2 A Working Session

When the editor is launched, the models corresponding to the entry structure are loaded. Then, if an entry is given as a parameter (editing an existing entry), the entry template is instantiated with the data contained in that entry. If no entry is given, the template is instantiated with an empty entry. Finally, the instantiated models and entry templates are stored into the session data and the result is displayed embedded in an HTML form, through a Web page (Figure 4).

Then, after a user modification (e.g. adding an item to the examples list), the HTML form sends the data to the server via a CGI mechanism. The server updates the models and template stored in the session data and sends back the modified result in the HTML page.

At the end of the session, the modified entry is extracted from the session data and then stored as a contribution in the database.

6 An Editing Example

6.1 A Dictionary Entry

Figure 5 shows an abstract view of a simple dictionary entry. It is the entry "scientifique" (scientific) of a French monolingual dictionary. The entry has been simplified on purpose. The entries are stored as XML text into the database.

6.2 Entry Structure

The generation of the graphic interface is mostly based on the dictionary microstructure. In the Papillon project, we describe them with XML schemata. We chose XML schemata instead of DTDs because they allow for a more precise description of the data structure and handled types. For example, it is possible to describe the textual content of an XML element as a closed value list. In this example, the French part-of-speech type is a closed list of "nom", "verb", and "adj".

Figure 6 is an abstract view of the structure corresponding to the previous French monolingual dictionary entry.

6.3 Entry Displayed in the Editor

The dictionary entry of Figure 5 is displayed in the HTML editor as in Figure 4. In the following one (Figure 7), an example has been added in the list by pushing the + button.

6.4 A More Complex Entry

In the following figure (Figure 8), we show the entry 食べる (taberu, to eat) of the Papillon Japanese monolingual volume. The entry structure comes from the DiCo structure (Polguère,

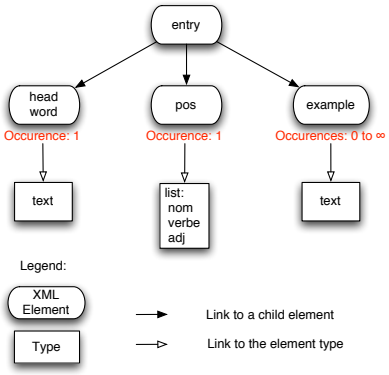


Figure 6: Structure of an Entry

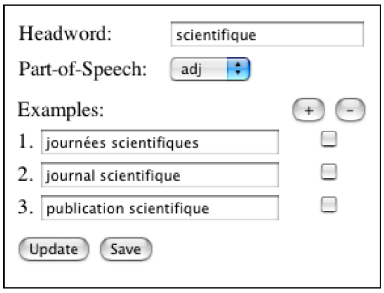


Figure 7: Entry Displayed in the Editor

2000), a light simplification of the ECD by Mel'čuk & al.

Two interesting points may be highlighted. You can note that not only the content of the entry is in Japanese, but also the text labels of the information. For example, the first one, 見出し語 (midashigo) means headword. The interface generator is multitarget: it generates the whole HTML content. It is then possible to redefine the labels for each language.

The second point is the complexity of the entry structure. There is a list of lexical functions. Each lexical function consists of a name and a list of valgroups (group of values), and in turn, each valgroup consists of a list of values. Finally, each value is a textbox. The lists are nested the one in the other one and it is possible to use the lists + and - operators at any level.

7 Evaluation

7.1 Preamble

This paper focuses on one particular functionality of the Papillon platform: the generic editor. Its purpose is not to present the building of Papillon dictionary or the progress of Papillon Project as a whole. The evaluation will then focus on the editor.

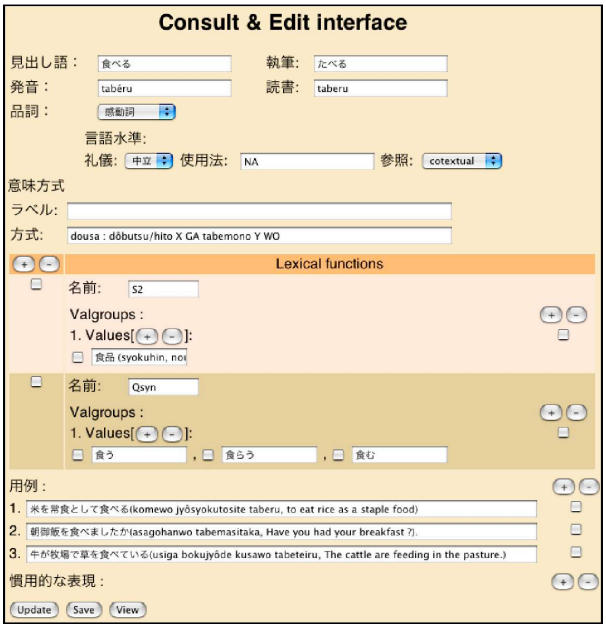


Figure 8: Papillon Entry Displayed in the Editor

The contribution phase on Papillon project has not begun yet. Thus, for the moment, very few users tested the editor. We have not yet enough data to evaluate seriously the usability of the interface. Then, the evaluation will be driven on the technical aspects of the editor.

In order to evaluate the genericity and the usability of the editor, we generated interfaces for two other dictionary structures: the GDEF Estonian-French dictionary and the WaDokuJiTen Japanese-German dictionary.

7.2 Edition of the GDEF dictionary

The GDEF project (Big Estonian-French Dictionary) is managed by Antoine Chalvin from INALCO, Paris. The dictionary microstructure is radically different from the Papillon dictionary as you will see in Figure 9 compared to figure 8. You may notice the 6 levels of recursion embedded in the entry structure.

It took about one week to write the interface description files for the new dictionary structure in order to generate properly a complete interface for the GDEF dictionary.

7.3 Edition of the WaDokuJiTen

The WaDokuJiTen project is managed by Ulrich Apel, now invited researcher at NII, Tokyo. The dictionary is originally stored in a FileMaker database. It has more than 200,000 entries. It took four days to export integrate the dictionary in Papillon platform and to write the

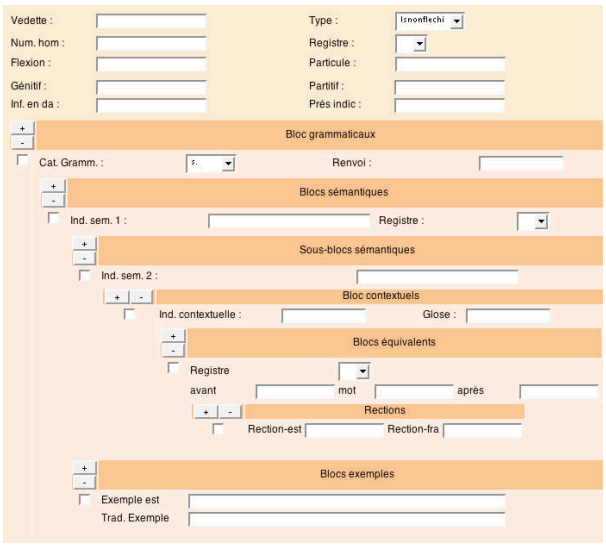


Figure 9: GDEF Entry Displayed in the Editor

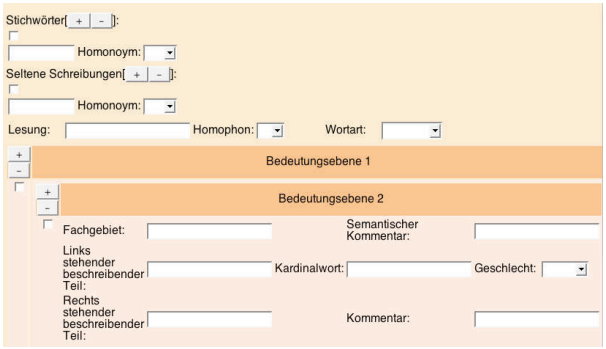


Figure 10: WaDokuJiTen Entry Displayed in the Editor

files needed for the generation of the editor interface. The integration was done in 4 steps: export the dictionary from FileMaker into an XML file, tag the implicit structure with a perl script, write the metadata files and upload the dictionary on the Papillon server.

The dictionary microstructure is simpler than the previous one (see figure 10). It took only two days to write the files needed for the generation of the editor interface.

8 Conclusion

The implementation of ARTStudio and Papillon plateform started separately four years ago. The development of the HTML generation module in ARTStudio and its integration into Papillon platform took about a year from the first specifications and the installation of a complete and functional version on the Papillon server. The collaboration between a specialist of computational lexicography and a specialist of the adaptability of interfaces has produced very original and interesting work. Furthermore, the evaluation and the feedback received from the users is very positive. Now, we want to further pursue this work following several paths.

First of all, only a specialist can use the existing interface for Papillon entry since it is too complex for a beginner. We plan to generate different interface types adapted to the varied user needs and competences. Thanks to the modularity of the editor, we need only to describe the tasks and instance models corresponding to the desired interface.

For the moment, the interface generation is not fully automatic; some of the model descriptions used by the editor have to be written "by hand". This is why we are working now on automating the whole generation process and the implementation of graphical editors allowing users to post-edit or modify a generated interface description.

References

Gaëlle Calvary, Joëlle Coutaz, and David Thevenin. 2001. Unifying reference framework for the development of plastic user interfaces. In *EHCI'01, IFIP WG2.7 (13.2) Working Conference*, pages 173–192, Toronto, Canada.

Gaëlle Calvary, Joëlle Coutaz, David Thevenin, Quentin Limbourg, Nathalie Souchon, Laurent Bouillon, and Jean Vanderdonckt. 2002. Plasticity of user interfaces: A revised reference framework. In *Proc. TAMODIA 2002*, pages 127–134, Bucharest, Romania. INFOREC Publishing House.

Mathieu Mangeot. 2002. An xml markup language framework for lexical databases environments: the dictionary markup language. In *International Standards of Terminology and Language Resources Management*, pages 37–44, Las Palmas, Spain, May.

Igor Mel'čuk, Nadia Arbatchewsky-Jumarie, Léo Eltnisky, Lidija Iordanskaja, Adèle Lessard, Suzanne Mantha, and Alain Polguère. 1984,88,92,96. *DEC : Dictionnaire explicatif et combinatoire du français contemporain, recherches lexico-sémantiques I,II,III et IV*. Presses de l'université de Montréal, Canada.

Alain Polguère. 2000. Towards a theoretically-motivated general public dictionary of semantic derivations and collocations for french. In *Proceeding of EURALEX'2000, Stuttgart*, pages 517–527.

Gilles Sérasset and Mathieu Mangeot. 1998. L'édition lexicographique dans un système générique de gestion de bases lexicales multilingues. In *NLP-IA*, volume 1, pages 110–116, Moncton, Canada.

Gilles Sérasset and Mathieu Mangeot. 2001. Papillon lexical database project: Monolingual dictionaries and interlingual links. In *NLPRS-2001*, pages 119–125, Tokyo, 27-30 November.

Gilles Sérasset. 1997. Le projet nadia-dec : vers un dictionnaire explicatif et combinatoire informatisé ? In *LTT'97*, volume 1, pages 149–160, Tunis, Septembre. Actualité scientifique, AUPELF-UREF.

David Thevenin and Joëlle Coutaz. 1999. Plasticity of user interfaces: Framework and research agenda. In *Interact'99 Seventh IFIP Conference on Human-Computer Interaction*, volume 1, pages 110–117, Edinburgh, Scotland.

Construction collaborative d'une base lexicale multilingue

Le projet Papillon

Mathieu Mangeot-Lerebours[*] — Gilles Sérasset[**] — Mathieu Lafourcade[***]

[*] *National Institute of Informatics*
Hitotsubashi 2-1-2-1913 Chiyoda-ku Tokyo 101-8430 Japan
mangeot@nii.ac.jp

[**] *GETA-CLIPS, IMAG, Université Joseph Fourier*
BP 53, 38041 Grenoble cedex 9
Gilles.Serasset@imag.fr

[***] *TAL-LIRMM, Université de Montpellier II*
161, rue Ada, 34392 Montpellier cedex 5
lafourcade@lirmm.fr

RÉSUMÉ. *Nous présentons le projet Papillon dédié la construction d'une base lexicale multilingue linguistiquement riche. Ce projet s'appuie sur le principe de construction collaborative, qui permet à chacun, professionel ou amateur, institution ou individu, de contribuer, dans la mesure de ses moyens, à ce grand chantier. Pour qu'un tel travail collaboratif puisse s'amorcer, il est nécessaire de fournir un ensemble conséquent d'informations lexicales multilingues, sur lesquels les contributeurs pourront s'appuyer. Après avoir présenté l'architectures linguistique, lexicale et informatique du projet Papillon, nous détaillons la méthode utilisée pour créer les informations initiales mises à disposition des contributeurs.*

ABSTRACT. *This paper presents the Papillon project dedicated to the building of a linguistically rich multilingual lexical database. This project is based on collaborative construction principle, which allows each one, professional or amateur, institution or individual, to contribute, with its own means, to this building task. For such a collaboratif work to be effective, it is necessary to provide a important set of multilingual lexical information, that will be the base of the contributors' work. After a presentation of the linguistic, lexical and software architectures of the Papillon project, we detail the method used to create the initial lexical information.*

MOTS-CLÉS : *Base lexicale multilingue, Dictionnaire, travail collaboratif.*

KEYWORDS: *Multilingual lexical database, dictionary, collaborative work.*

2[e] soumission à *Traitement Automatique des Langues*, le 30 mars 2003.

1. Introduction

Qu'elle soit implicite ou explicite, la connaissance linguistique reste un constituant fondamental des systèmes de traitement des langues. Le coût généralement constaté de création d'une connaissance lexicale explicite (un dictionnaire) est l'un des freins majeurs dans le développement d'un système de traitement des langues (TAL).

De la même manière, malgré le nombre et la diversité des dictionnaires à usage humain, il reste de nombreux trous à combler. Ainsi, un francophone ne peut actuellement trouver de dictionnaire bilingue français-japonais lui donnant une transcription utilisable des traductions en kanji (idéogrammes japonais) et lui fournissant des informations qui lui sont nécessaires (les spécificateurs numériques du japonais par exemple). Ces besoins sont encore plus flagrants pour des locuteurs de langues moins représentées au niveau lexical.

Dans cet article, nous présentons tout d'abord les motivations du projet Papillon dont l'objectif est de combler ce manque en construisant une base lexicale fortement multilingue offrant des informations linguistiquement riches. Les coûts de construction d'une telle base sont réduits par l'adoption d'une stratégie (présentée en 2.2) basée sur le modèle « open source » où les données disponibles se voient constamment enrichies par des contributions d'utilisateurs aux compétences diverses. Enfin, les coûts restants sont rendus acceptables par l'adoption d'une structure linguistique et lexicale (détaillées en 2.3) favorisant la réutilisation des données construites.

Nous décrivons ensuite l'implémentation du serveur de communauté au travers duquel se fait le travail de construction de cette base. Après avoir donné une vue d'ensemble du serveur Papillon (en 3.1), et présenté les principes de représentation des différentes structures de données manipulées (en 3.2), nous détaillons les méthodes utilisées pour offrir un service d'accès unifié aux diverses données disponibles sur le site (en 3.3).

La stratégie adoptée implique un travail initial de construction d'une amorce de base lexicale contenant un ensemble d'entrées initiales non détaillées, qui servira de base aux contributions des utilisateurs. L'architecture interlingue de la base rend cette construction relativement difficile. Nous présentons donc les outils (en 4.1) et méthodes (en 4.2) mises en œuvre pour cette étape d'amorçage.

2. Le projet Papillon

2.1. *Motivations du projet*

Le projet Papillon a été initié suite à différents constats :

– Il n'existe pas à l'heure actuelle de dictionnaire français-japonais électroniques et gratuits. De plus, les dictionnaires existants sont en général conçus pour les Japonais. La transcription des kanjis (idéogrammes japonais) est, dans la plupart des cas, omise. Les francophones ne peuvent donc pas se servir de ces dictionnaires à moins de

savoir lire le japonais. De plus, d'autres informations nécessaires pour s'exprimer en japonais font aussi défaut. Il existe par exemple, une grande variété de spécificateurs numériques en japonais. Certains échappent à toute logique. Il est donc indispensable que cette information soit accessible.

– Pour un francophone, il est beaucoup plus difficile d'obtenir des informations lexicales sur le malais ou le thaï que sur l'anglais.

Les besoins en données lexicales restent donc importants, non seulement pour un utilisateur humain, mais aussi pour les systèmes de traitements des langues, non seulement pour un francophone, mais pour tout utilisateur humain quelle que soit sa langue.

La principale difficulté réside dans les coûts prohibitifs de construction de grandes quantités de données. Par exemple, le projet Electronic Dictionary Research (EDR) de construction d'un dictionnaire japonais-anglais a nécessité plus de 1200 hommes années de travail. Son prix de vente, 14 000 €environ, est très inférieur aux coûts réels de construction qui ne seront probablement jamais rentabilisés. Il est cependant encore trop élevé pour un particulier. De ce fait, seules des institutions peuvent l'acquérir. De plus, les données fournies à ce prix sont utilisables principalement par certains systèmes de traduction automatique fondés sur des techniques particulières.

Le projet Papillon met en œuvre plusieurs stratégies pour réduire ces coûts et les rendre acceptables :

– En utilisant une structure lexicale suffisamment générale et complète pour que la plupart des applications du TAL y trouvent (de manière directe ou indirecte) les données dont elles ont besoin.

– En offrant des outils simples permettant à de non-spécialistes de partager leur connaissance naturelle de leur langue maternelle. La compétence des spécialistes étant utilisée afin de nettoyer et valider les informations ainsi obtenues.

– En construisant une base multilingue fondée sur une approche interlingue par acceptions, qui permet, en factorisant l'ensemble des connaissances bilingues disponibles, de s'appuyer sur les langues bien dotées pour avancer sur les langues moins représentées.

– Enfin, en appliquant le paradigme de construction « open-source » à la construction de données lexicales : chaque utilisateur contribue bénévolement à la base lexicale et les ressources sont ensuite disponibles gratuitement pour tous.

L'utilisation du paradigme « open-source » a déjà été utilisée dans des projets similaires de construction collaboratives de données lexicales sur le Web, parfois depuis plusieurs années. Le projet EDICT de construction de dictionnaire japonais-anglais dirigé par Jim Breen, professeur à l'université Monash en Australie, a démarré, il y a plus de 10 ans. De plus, des projets parallèles d'adaptation de ce dictionnaire à d'autres langues, comme le français conduit par Jean-Marc Desperrier ([DES 02]), ont démarré avec succès. D'autres projets de construction bilingue de dictionnaires incluant le japonais ont été lancés plus récemment comme SAIKAM, japonais-thaï et WaDoKuJiten, allemand-japonais.

C'est l'utilisation conjointe de l'ensemble des stratégies énoncées qui est novatrice. Nous pensons en effet que chacune des trois premières stratégies renforce l'impact du paradigme « open-source ». La première, en couvrant de nombreux besoins, permet d'impliquer des spécialistes du TAL qui apporteront leur pierre à l'édifice. La seconde, en permettant à des utilisateurs non-spécialistes de s'impliquer dans le projet, élargie le nombre de contributeurs potentiels. La troisième, en proposant une approche multilingue dès le début du projet, nous permet d'impliquer des partenaires de nombreux pays.

Lancé en 2000 par Emmanuel Planas, François Brown de Colstoun et Mutsuko Tomokiyo, le projet Papillon a été lancé en partenariat avec le National Institute of Informatics à Tokyo (Frédéric Andrès). Après trois séminaires (dont 2 à Tokyo et 1 à Grenoble), de nombreux partenaires se sont manifestés et ont souhaité rejoindre le projet : Jim Breen, auteur du dictionnaire EDICT (Université Monash, Australie), Francis Bond (NTT, Keihanna), Yves Lepage (ATR, Keihanna), Ulrich Appel, auteur du dictionnaire allemand-japonais WaDoKuJiten, Jean-Marc Desperrier, responsable de l'adaptation au français du dictionnaire EDICT, l'université Kasetsart et le NECTEC (Bangkok, Thaïlande), l'Universiti Sains Malaysia (Penang, Malaisie), les universités de Da Nang et de Hanoi (Vietnam), etc. Actuellement, les langues couvertes sont l'allemand, l'anglais, le français, le japonais, le lao, le malais, le thaï, le vietnamien et, très récemment, le chinois. Des contacts sont en cours concernant les langues indiennes.

2.2. *Stratégie de construction de la base lexicale multilingue*

Le succès du projet Papillon dépend de sa capacité à intégrer des informations fragmentaires de toutes natures dans un modèle unique. Ces informations peuvent provenir de dictionnaires existants ou d'utilisateurs contributeurs. Dans le premier cas, il s'agit d'informations cohérentes, disponibles dans un modèle propre, duquel nous extrayons les informations que nous souhaitons représenter dans le modèle de la base lexicale multilingue Papillon. Dans le second cas, il s'agit d'informations parcellaires exprimées dans le modèle de la base Papillon, sous forme de modification de données existantes.

Les contributions ne peuvent donc exister que s'il existe un ensemble minimal d'informations lexicales sur lesquelles les contributeurs apporterons des modifications (ajout, correction ou suppression). Il est donc primordial d'adopter une stratégie « en largeur » qui commence par une étape d'amorçage dont le but est d'obtenir automatiquement, à partir de dictionnaires existants, une première base lexicale contenant de nombreuses entrées associées à des informations minimales. Cette étape d'amorçage est complexe du fait de l'utilisation d'une architecture lexicale interlingue. Nous la détaillons dans la partie 4.

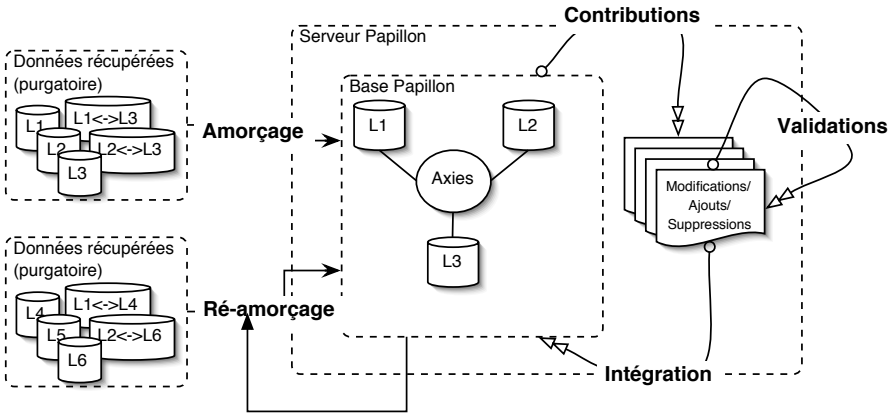


Figure 1. *Stratégie de construction de la base Papillon*

Une fois ce premier travail effectué, la base obtenue est installée sur le serveur Papillon. Les contributions sont effectuées sur cette base qui entre dans une phase de constante évolution.

Nous distinguons trois tâches dans l'évolution de cette base :

– **La contribution** où chaque utilisateur, spécialiste ou non, peut proposer des modifications (ajout d'informations dans des lexies existantes, ajout de lexies, suppression de lexies). Cette tâche pourra également être effectuée par des agents automatiques (acquisition de données à partir de corpus, etc.).

– **La validation** où les contributions seront soumises, directement ou indirectement à l'accord des utilisateurs, spécialistes ou non (des outils spécifiques seront développés afin de recueillir ces validations auprès d'utilisatrices non-spécialistes). Cette tâche pourra également être effectuée par des agents automatiques (confrontation à des corpus, ou à des ressources existantes, etc.).

– **L'intégration** où des utilisateurs de confiance acceptent ou rejètent les contributions (validées ou non) pour qu'elles soient effectivement appliquées à la base.

Cette méthode permet un certain contrôle sur la qualité de la base, mais pose un problème aux contributeurs qui ne souhaitent pas de délai entre le moment où ils contribuent et le moment où la contribution est prise en compte. Pour éviter ce délai, les modifications sont stockées dans l'espace personnel du contributeur et sont appliquées automatiquement à chaque fois qu'il consulte les entrées concernées. Ainsi les contributions sont instantanément visibles par le contributeur qui pourra aussi les partager avec d'autres utilisateurs.

2.3. *Architectures linguistique et lexicale*

2.3.1. *Macrostructure de la base lexicale*

La macrostructure de la base lexicale Papillon a été définie dans [SÉR 94b] et expérimentée à petite échelle pour la construction d'une petite base lexicale multilingue par [BLA 95].

Cette macrostructure est fondée sur la notion d'acceptions. Chaque dictionnaire monolingue est vu comme un ensemble d'acceptions d'une langue. Les liens entre les langues sont établis grâce à un dictionnaire pivot qui contient un ensemble d'acceptions interlingues (que nous appelons axies).

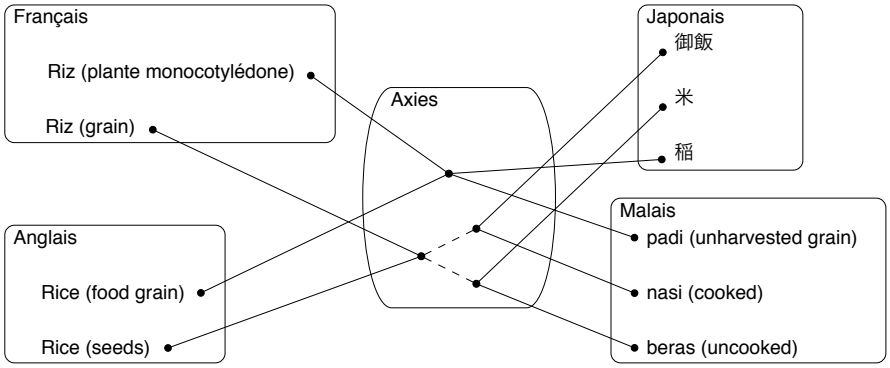


Figure 2. *Utilisation d'un lien inter axies pour représenter les phénomènes contrastifs de l'équivalence lexicale.*

Les vrais problèmes contrastifs de l'équivalence lexicale (qu'il ne faut pas confondre avec la polysémie monolingue, l'homonymie ou la synonymie comme [MEL 01] l'explique clairement) sont représentés grâce à un lien entre axies. Ce phénomène se retrouve dans l'exemple de la traduction du mot « Riz » dans 4 langues. Pour cet exemple, nous avons utilisé les acceptions définies dans des dictionnaires monolingues du commerce.

Chaque langue définit une acception correspondant au sens français désignant le riz comme une plante céréalière. Il est donc aisé de relier chacune de ces acceptions monolingues à une axie unique.

Par contre, ni le français, ni l'anglais ne définissent d'acceptions différentes suivant que le « riz » (considéré comme aliment) est cuit ou non alors que le japonais et le malais font cette distinction. Une axie ne peut être reliée à la fois aux acceptions malaises de « nasi » et « berak », à moins qu'on ne souhaite les considérer comme des synonymes (ce qui serait une erreur ici). Il faut donc créer 3 axies différentes pour relier ces acceptions monolingues : la première correspond aux acceptions de « nasi » et de « 御飯 » (gohan) ; la seconde correspond aux acceptions de « beras » et de

« 米 » (kome) et la dernière correspond aux acceptions de « riz » et de « rice ». Les traductions sont ensuite établies par l'ajout de deux liens entre la dernière axie et les deux premières.

Il faut noter que ce lien inter-axie ne représente que le fait que les deux acceptions reliées peuvent être utilisées comme traduction l'une de l'autre. Il ne porte pas de sémantique particulière et ne doit pas être confondu avec un lien ontologique.

2.3.2. *Microstructure des articles*

La structure de chacune des unités du lexique est issue de la théorie sens-texte ([MEL 84, MEL 88, MEL 92, MEL 96]). En effet, cette théorie étant indépendante des langues, elle nous permet de manipuler une structure unique pour toutes les langues de la base (à certaines nuances prêt, exposées plus bas).

Dans la théorie sens-texte, et dans le dictionnaire explicatif et combinatoire (DEC) qui en est sa composante lexicale, les articles sont nommés des lexies. Nous reprenons ici la définition d'une lexie de [POL 02] :

> Une lexie, aussi appelée unité lexicale, est un regroupement 1) de mots-formes ou 2) de constructions linguistiques qui ne se distinguent que par la flexion.
>
> Dans le premier cas, il s'agit de lexèmes et dans le second cas, de locutions.
>
> Chaque lexie (lexème ou locution) est associée à un sens donné. Que l'on retrouve dans le signifié de chacun des signes (mots-formes ou constructions linguistiques) auxquels elle correspond.

Les lexies sont ensuite regroupées en vocables. Nous reprenons ici la définition d'un vocable de [POL 02] :

> Un vocable est un regroupement de lexies qui sont associées aux mêmes signifiants et qui ont un lien sémantique évident.

Dans les dictionnaires monolingues de la base Papillon, nous reprenons la notion de lexie, qui constitue l'unité du lexique. Ces lexies correspondent à la notion d'acceptions monolingues évoquée dans le paragraphe précédent. Par contre, la notion de vocable n'est pas explicitement exprimée dans la base lexicale Papillon, cette notion se retrouvera au niveau de l'interface entre la base et ses utilisateurs.

Le DEC est actuellement constitué de 4 volumes regroupant 558 vocables en tout. C'est un dictionnaire expérimental avec une structure assez complexe et qui ne peut (encore) servir à un usage général. C'est pourquoi un projet de simplification du DEC (le projet DiCo, [POL 00]) a été lancé récemment par Alain Polguère et Igor Mel'čuk avec l'aide des étudiants de l'Observatoire de Linguistique Sens-Texte de l'université de Montréal au Canada.

Néanmoins, nous avons souhaité formaliser de manière plus détaillée certaines parties présentes implicitement dans la structure Dico.

Ainsi, dans l'exemple de la figure 3, nous représentons explicitement le fait que la fonction « Qsyn » (quasi synonymes) a trois valeurs, réparties dans deux groupes distincts : « assassinat » et « homicide » d'une part, « crime » d'autre part. Cette

Nom de l'unité lexicale	Meurtre.1
Propriétés grammaticales	nom, masc
Formule sémantique	action de tuer : PAR L'individu X DE L'individu Y
Régime	X = I = de N, A-poss Y = II = de N, A-poss
Fonctions lexicales	QSyn assassinat, homicide#1 ; crime V0 tuer A0 meurtrier-adj S1 auteur [de ART]//meurtrier-n /* Nom pour X*/
Exemples	La mésentente pourrait être le mobile du meurtre.
Idiomes	_appel au meurtre_, _crier au meurtre_

Figure 3. *Exemple d'une entrée du DiCo.*

répartition permet de noter qu'il existe une plus grande distance sémantique entre « meurtre » et « crime » qu'entre « meurtre » et « homicide ». Enfin, nous explicitons le lien qui est établi entre le sens 1 de « homicide » et cette lexie.

De plus, nous avons adopté un mécanisme qui nous permet d'adapter légèrement cette structure aux particularités des différentes langues de la base. Ainsi, les valeurs possibles comme propriétés grammaticales du thaï et du français sont différentes. Enfin, nous avons rajouté au dictionnaire du japonais les notions particulières de niveau de politesse, niveau d'usage et niveau de référence. Cette structure lexicale, exprimée en XML, est détaillée § 3.2.2.

3. Le serveur contributif Papillon

3.1. *Vue d'ensemble*

La construction, la gestion, la maintenance et une partie de l'exploitation de la base lexicale multilingue Papillon se fait par l'intermédiaire d'un site de communauté entièrement dynamique. Ce site a été construit entièrement en java, avec le serveur d'application « Enhydra » et la base de donnée « PostgreSQL ». L'ensemble des données utilisées dans ce site sont au format XML et sont exprimées en Unicode (UTF-8). Tous les outils utilisés sont des outils « open source ». Ce site est accessible à l'URL : `http ://www.papillon-dictionary.org/`.

3.1.1. *Services disponibles*

Outre les services inhérents à la création collaborative d'une base lexicale multilingue (interface de consultation/modification de la base Papillon, gestion des contributions, des utilisateurs, de l'historique, etc.), nous avons souhaité fournir 3 autres services pour rendre le site à la fois plus pratique et plus attractif :

– **Un service d'archivage de la liste de discussion des utilisateurs**. Ce service,

quasi systématique dans les sites de communauté, présente ici une particularité. En effet, les discussions se font dans différentes langues et les messages échangés arrivent dans des encodages divers (ISO-LATIN1, SJIS, EUC, UTF8, etc.). Ces encodages doivent être correctement reconnus et transcris en UTF8 afin d'être archives. Nous avons dû adapter les outils standard pour cela.

– **Un service de partage d'informations entre les utilisateurs**. Ce service permet à un utilisateur du site Papillon d'y ajouter des informations qu'il trouve pertinentes pour les autres utilisateurs. L'interface de ce service a été particulièrement soignée, afin que tout utilisateur, quel que soit son niveau en informatique, puisse obtenir un résultat très satisfaisant. L'utilisateur se contente de rédiger un document en HTML avec n'importe quel éditeur du commerce. Ce document peut contenir divers fichier HTML (avec des liens internes ou externes), des images ou d'autres données. Il se contente ensuite de transférer ce document sur le site (en utilisant son client http préféré). Ce document sera analysé, le code html sera corrigé automatiquement, et il sera intégré au site Papillon. Il prendra la même forme que les autres pages (mêmes entêtes, mêmes fonctionnalités, même comportement général). Ce service permet de plus de gérer des documents multilingues : le document est disponible dans plusieurs langues et les lecteurs verront automatiquement la version qui leur convient.

– **Un service d'accès unifié à de nombreux dictionnaires**. Ce service permet à tout utilisateur d'accéder, par une interface unique, à de nombreux dictionnaires monolingues et bilingues. Un utilisateur cherchant un mot dans une langue pourra obtenir les entrées correspondant à ce mot dans l'ensemble des dictionnaires disponibles. Avec ce service, nous espérons attirer des utilisateurs qui ne sont, dans un premier temps que « consommateurs » de données lexicales. Avec le temps, nous pensons que certains d'entre eux deviendront, à leur tour, « producteurs » de données.

3.1.2. *Organisation des données*

Les données lexicales disponibles sur le site Papillon sont diverses. Il peut s'agir de données de la base lexicale Papillon en cours de construction ou de données provenant de dictionnaires existants libres de droits ou qui nous ont été transmis par leurs auteurs. Ces données lexicales sont réparties dans 3 « zones » :

– *Les limbes* contiennent les données lexicales dans leur format propriétaire d'origine. Lorsqu'un dictionnaire nous est fourni, il est disponible dans cette zone en attendant d'être « récupéré ». Chaque dictionnaire est associé à un fichier de méta données contenant toutes les informations disponibles à son propos (son nom, les languages qu'il contient, la date de création, sa taille, ses auteurs, son domaine éventuel, etc.). Les dictionnaires présents dans cette zone peuvent être téléchargés tels-quels, mais les entrées qu'ils contiennent ne peuvent être obtenues individuellement.

– Après récupération, les données lexicales des limbes sont disponibles dans *le purgatoire*. Cette récupération consiste à définir en XML la structure du dictionnaire original (qui n'est pas modifiée). Les données sont ensuite transformées en XML en encodées en UTF8.
Les éléments de la structure XML obtenue (ou de la structure originale si le diction-

naire était déjà disponible en XML) sont ensuite identifiés grâce au mécanisme de pointeurs CDM détaillé § 3.3.2. Cette identification est stockée dans le fichier de méta données du dictionnaire. Elle permet d'accéder individuellement aux entrées de ce dictionnaire par des requêtes émises via l'interface unifiée disponible sur le site Papillon.

– *Le paradis* contient les entrées de la base lexicale multilingue Papillon. Ce dictionnaire est lui aussi accessible par l'interface unifiée disponible sur le site. Par contre, il est le seul dictionnaire qui puisse être modifié par l'interface d'édition.

3.1.3. *Implémentation*

L'application Papillon est organisée en trois couches (figure 4). La première couche prend en charge l'interface vers les utilisateurs. La seconde couche contient l'ensemble des services fournis aux utilisateurs. La dernière couche gère la persistance des données XML manipulées. Cette architecture rend facile l'ajout et la modification des interfaces de l'application vers ses clients (une interface WML est en cours de développement pour permettre l'accès via des téléphones portables). Elle permet aussi de s'abstraire de la manière dont les données sont stockées. Dans l'implémentation actuelle, les données XML sont stockées dans une base relationnelle « open source », via l'API java standard JDBC. Cette application peut être dupliquée sur de nombreux serveurs afin de répartir la charge.

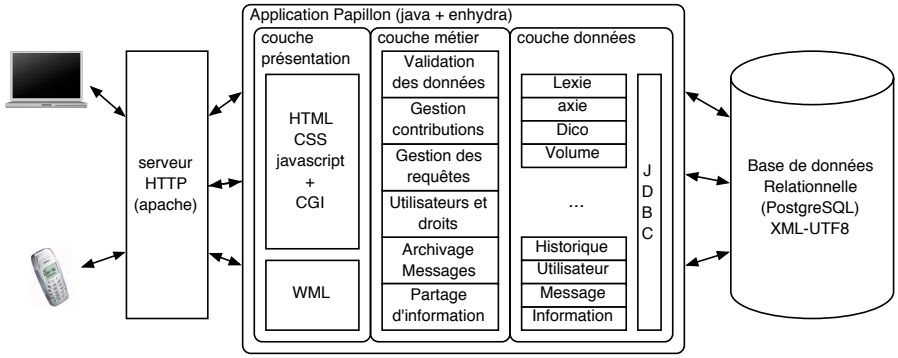


Figure 4. *Architecture du serveur de communauté Papillon.*

3.2. *Représentation des données de la base lexicale multilingue Papillon*

Dans le cadre de travaux antérieurs ([SÉR 94a, SÉR 94b] puis [MAN 01] et [MAN 02]), nous avons défini un cadre de représentation de données lexicales hétérogène. Pour le projet Papillon, nous utilisons ce cadre implémenté en XML (figure 5).

L'implémentation de notre microstructure s'appuie sur 3 niveaux de représentations : un cadre générique de représentation de dictionnaires (Dictionary Markup Lan-

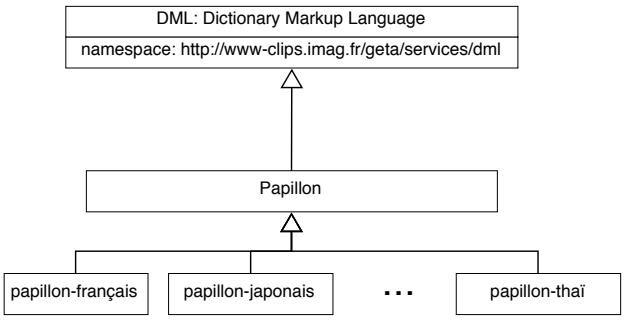


Figure 5. *Définition XML des structures de dictionnaires monolingues de la base Papillon.*

guage), un niveau de représentation de la structure commune à tous les dictionnaires de la base Papillon et un niveau de représentations représentant les spécificités de chaque langue.

L'ensemble des structures est implémenté en XML. Ces trois niveaux sont implémentés à l'aide de schémas XML qui incluent un mécanisme d'héritage simple.

3.2.1. *DML : description de dictionnaires en XML*

Le premier niveau de représentation est un cadre général, qui défini les concepts généraux, qui peuvent être utilisés pour représenter n'importe quel dictionnaire. Ce cadre est générique et permet la représentation aisée de dictionnaires hétérogènes.

Ce cadre est un espace de nom (namespace) XML nommé DML (Dictionary Markup Language, figure 6). Toute donnée d'une base lexicale peut être décrite en utilisant des éléments DML. Ce cadre défini non seulement les objets de base permettant de représenter des structures lexicales complexes (arbre, graphe, automate, etc.), mais aussi des types généraux utiles (booléen, date, langue, etc.) ainsi que les APIs permettant à des clients d'utiliser les données décrites ou à des fournisseurs de rajouter des services.

3.2.2. *Structure commune des dictionnaires monolingues Papillon*

La structure commune des dictionnaires monolingues de la base Papillon est définie par un schéma XML qui utilise l'espace de nom défini dans le DML. Ce schéma XML décrit la structure générale des lexies.

```
<element name="lexie">
  <complexType>
    <sequence>
      <element ref="d:headword" minOccurs="1" maxOccurs="1" />
      <element ref="d:writing" minOccurs="0" maxOccurs="1" />
      <element ref="d:reading" minOccurs="0" maxOccurs="1" />
```

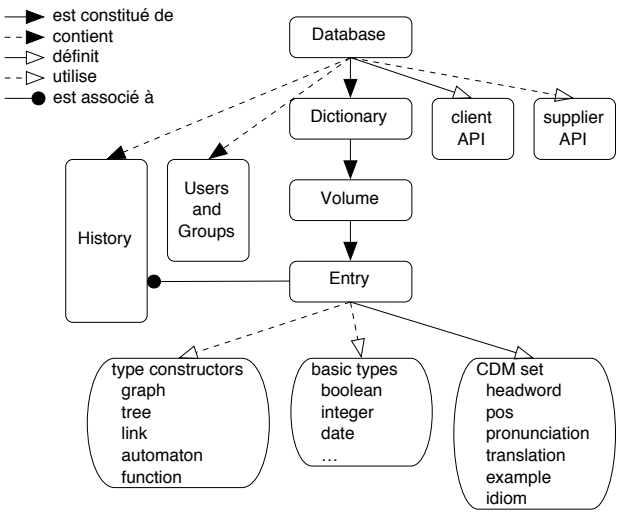


Figure 6. *Les concepts du DML.*

```
      <element ref="d:pronunciation" minOccurs="0" maxOccurs="1" />
      <element ref="d:pos" minOccurs="1" maxOccurs="1" />
      <element ref="d:language-levels" minOccurs="0" maxOccurs="1" />
      <element ref="d:semantic-formula" minOccurs="1" maxOccurs="1" />
      <element ref="d:government-pattern" minOccurs="0" maxOccurs="1" />
      <element ref="d:lexical-functions" minOccurs="0" maxOccurs="1" />
      <element ref="d:examples" minOccurs="0" maxOccurs="1" />
      <element ref="d:full-idioms" minOccurs="0" maxOccurs="1" />
      <element ref="d:more-info" minOccurs="0" maxOccurs="1" />
    </sequence>
    <attribute ref="d:id" use="required" />
  </complexType>
</element>
```

Chaque élément de cette structure est lui aussi défini à ce niveau. Ainsi, à ce niveau, l'élément « pos » (catégorie) est défini comme une chaîne de caractères, tandis que l'élément « lexical-functions » (les fonctions lexicales) est défini comme une liste de « function », qui est un type de base du DML.

```
<element name="pos" type="d:posType" />
<simpleType name="posType">
  <restriction base="string" />
</simpleType>
<element name="lexical-functions">
  <complexType>
    <sequence maxOccurs="unbounded">
```

```
      <element ref="d:function" />
    </sequence>
  </complexType>
</element>
```

3.2.3. *Adaptation de la structure à chaque langue de la base*

Le troisième niveau permet d'adapter légèrement la structure commune définie ci-dessus aux spécificités de chacune des langues de la base lexicale Papillon. Ainsi, chaque langue de la base possède un schéma qui lui est propre et qui redéfinit certains des éléments XML évoqués plus haut. Ainsi, l'élément « pos » (catégorie) du dictionnaire français est redéfini comme suit :

```
<simpleType name="posType">
  <restriction base="d:posType">
    <enumeration value="n.m." />
    <enumeration value="n.m. inv." />
    <enumeration value="n.m. pl." />
    <enumeration value="n.m., f." />
    <enumeration value="n.f." />
    <enumeration value="n.f. pl." />
    ...
  </restriction>
</simpleType>
```

De la même manière, une langue peut redéfinir les valeurs possibles pour les niveaux d'usage et de politesse. Il est souhaitable que chaque langue définisse de manière explicite une liste fermée de valeurs possibles pour ces éléments plutôt que de se fier à la structure générale qui, par défaut, accepte n'importe quelle chaîne de caractère. C'est en effet à partir de ces schémas XML que sont générées les interfaces de saisie.

3.3. *Accès unifié à des dictionnaires existants*

3.3.1. *Interface de consultation*

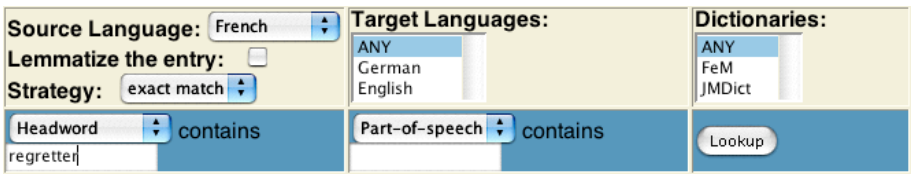


Figure 7. *Interface d'accès unifié aux dictionnaires du purgatoire et du paradis.*

L'interface unifiée permet de faire une recherche sur le lemme, sur la catégorie, sur la prononciation ou sur une traduction d'une entrée de dictionnaire (figure 7). Pour certaine langue, une lemmatisation est disponible pour permettre à l'utilisateur débutant d'entrer une occurrence quelconque. On peut de plus choisir les langues cibles et les dictionnaires dans lesquels se fera la recherche. Le service de lemmatisation est un service externe que nous accédons via Internet.

3.3.2. *Un mécanisme de pointeurs communs : CDM*

Cette interface de requête ne peut fonctionner que s'il existe un moyen d'identifier les éléments sur lesquels portent la recherche (entrée, catégorie, prononciation, traduction) dans des dictionnaires ayant des structures différentes. Cette identification est faite en établissant une correspondance entre un élément particulier d'un dictionnaire et un éléments définis par l'espace de nom DML (cf. § 3.2.1).

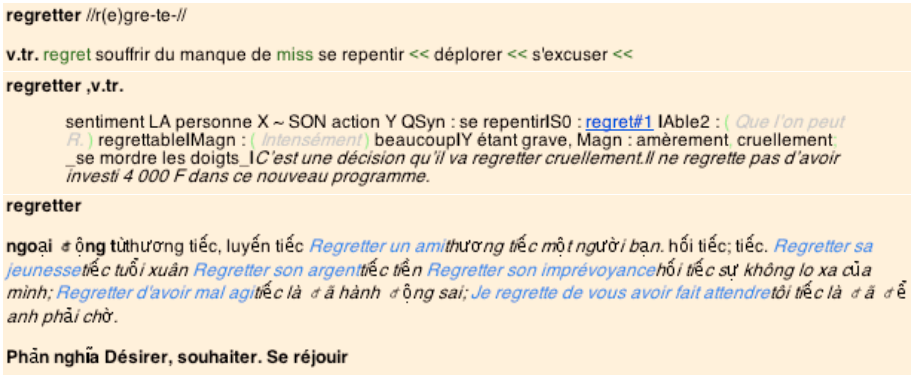
Le sous-ensemble du DML avec lesquels une telle correspondance peut être faite est nommé CDM (Common Dictionary Markup). Il est en constante évolution. La figure 8 en donne un exemple d'utilisation.

Element CDM	**équivalent TEI**	**FeM**	**OHD**	**NODE**
<entry>	(entry)	<fem-entry>	<se>	<se>
<headword>	(hom)(orth)	<entry>	<hw>	<hw>
<pronunciation>	(pron)	<french_pron>	<pr><ph>	<pr><ph>
<etymology>	(etym)			<etym>
<syntactic-sense>	(sense level="1")		<sense n=1>	<s1>
<pos>	(pos)(subc)	<french_cat>	<pos>	<ps>
<lexie>	(sense level="2")		<sense n=2>	<s2>
<indicator>	(usg)	<gloss>	<id>	
<label>	(lbl)	<label>		<la>
<example>	(def)	<french_sentence>	<ex>	<ex>
<definition>	(eg)			<df>
<translation>	(trans)(tr)	<english_equ> <malay_equ>		<tr>
<collocate>	(colloc)		<co>	
<link>	(xr)	<cross_ref_entry>	<xr>	<xg> <vg>
<note>	(note)		<ann>	

Figure 8. *Correspondance entre les éléments CDM et des éléments des dictionnaires TEI, FeM (français-anglais-malais), Oxford-Hachette (français-anglais) et Oxford (anglais)*

3.3.3. *Présentation des résultats*

Le résultat d'une requête est un ensemble d'unités codées en XML. Leurs structures sont variées car elles proviennent de dictionnaires différents.



regretter //r(e)gre-te-//

v.tr. regret souffrir du manque de miss se repentir << déplorer << s'excuser <<

regretter ,v.tr.

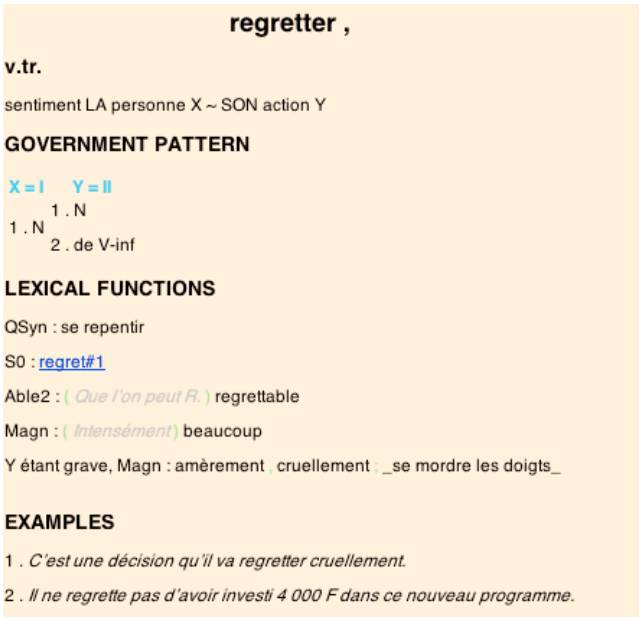
sentiment LA personne X ~ SON action Y QSyn : se repentirIS0 : regret#1 IAble2 : (*Que l'on peut R.*) regrettableIMagn : (*Intensément*) beaucoupIY étant grave, Magn : amèrement cruellement _se mordre les doigts_I*C'est une décision qu'il va regretter cruellement.Il ne regrette pas d'avoir investi 4 000 F dans ce nouveau programme.*

regretter

ngoại động từthương tiếc, luyến tiếc *Regretter un ami*thương tiếc một người bạn. hối tiếc; tiếc. *Regretter sa jeunesse*tiếc tuổi xuân *Regretter son argent*tiếc tiền *Regretter son imprévoyance*hối tiếc sự không lo xa của mình; *Regretter d'avoir mal agi*tiếc là đã hành động sai; *Je regrette de vous avoir fait attendre*tôi tiếc là đã để anh phải chờ.

Phản nghĩa Désirer, souhaiter. Se réjouir

Figure 9. *Trois résultats de la requête « regretter ». Le premier résultat provient du dictionnaire français-anglais-malais (FeM), le suivant de la base Papillon (entrée récupérée du DiCo), le dernier provient du dictionnaire français-vietnamien (vietDict).*



regretter ,

v.tr.

sentiment LA personne X ~ SON action Y

GOVERNMENT PATTERN

X = I Y = II

1 . N

1 . N

2 . de V-inf

LEXICAL FUNCTIONS

QSyn : se repentir

S0 : regret#1

Able2 : (*Que l'on peut R.*) regrettable

Magn : (*Intensément*) beaucoup

Y étant grave, Magn : amèrement , cruellement , _se mordre les doigts_

EXAMPLES

1 . *C'est une décision qu'il va regretter cruellement.*

2 . *Il ne regrette pas d'avoir investi 4 000 F dans ce nouveau programme.*

Figure 10. *Forme inspirée du DEC pour la lexie « regretter.1 » du dictionnaire Papillon.*

Ces structures sont transformées en des « formes » qui dépendent des possibilités de l'interface utilisateur. Ces formes sont obtenues en appliquant des transformations XSL aux résultats de la requête. Ces transformations sont elles-mêmes des données du

serveur et plusieurs transformations peuvent être disponibles pour une même structure. Un utilisateur peut ajouter sa propre forme au serveur.

L'utilisateur peut facilement choisir la forme qu'il souhaite. Celle-ci peut mettre en valeur certaines informations lexicales ou en cacher d'autres. La figure 10 présente la lexie « regretter » (sentiment : personne X regrette son action Y) présentée selon une forme inspirée du DEC.

4. Construction d'une base lexicale initiale

Notre stratégie de construction commence la construction automatique d'une première base lexicale multilingue qui sert de base au travail contributif (l'amorçage). Cette construction se fait en deux étapes. Dans un premier temps, nous construisons les dictionnaires monolingues. Dans un second temps, nous construisons la base d'acceptions qui fera le lien entre les différents dictionnaires monolingues.

Ces deux étapes se font en combinant un certain nombre de ressources lexicales monolingues et bilingues. Un exemple de tels croisements est largement illustré par [QUA 01]. Cependant, l'architecture lexicale choisie nous impose d'établir les liens de traduction au niveau des lexies (acceptions monolingues), ce qui pose le problème de la sélection correcte des différents sens de termes polysémiques.

Pour pouvoir faire cette sélection de manière automatique, nous avons choisi d'utiliser le modèle des vecteurs conceptuels ([CHA 96], [LAF 99]) qui défini une notion de distance que nous interprétons comme une distance sémantique.

4.1. *Le modèle des vecteurs conceptuels*

4.1.1. *Définition et notion de distance*

Le modèle des vecteurs conceptuels a été présenté dans [LAF 02]. On associe à tout segment textuel (mot, syntagme, texte), une association thématique qui prend la forme d'une vecteur de concepts. Le jeux de concepts est prédéfini et constitue un espace générateur sur lequel les sens peuvent se projeter (la figure 11 donne une représentation graphique de deux de ces vecteurs). Par exemple, les sens de ‹*barrage*› peuvent être projetés sur les concepts suivant (les CONCEPT[*intensité*] étant ordonnés par intensité décroissante) : $V_{barrage} =$ (BARRIÈRE[*0.84*], OBSTACLE[*0.83*], ÉLECTRICITÉ[*0.82*], SPORT[*0.77*], FLOT[*0.76*], GUERRE[*0.76*], ...).

Dans ce modèle vectoriel, nous disposons des notions de *similarité* (utilisée habituellement en recherche d'information) et de *distance angulaire* $D_A(V_1, V_2)$. Cette dernière est une vrai mesure de distance (contrairement à la notion de similarité) et elle vérifie les propriétés de réflexivité, symétrie et inégalité triangulaire. Nous l'interprétons comme une évaluation de la *proximité thématique* entre sens de mots.

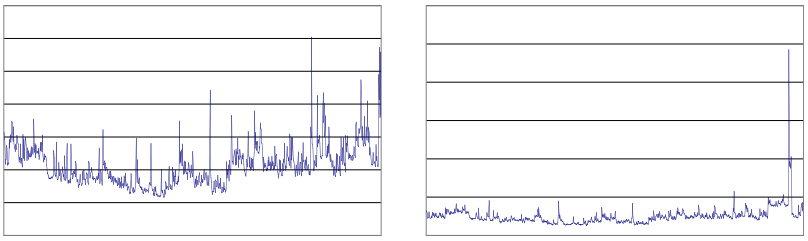


Figure 11. *Représentation graphique des vecteurs des termes* échange *(très polysémique) et* cession.

4.1.2. *Notion de contextualisation faible*

L'opération de contextualisation faible, notée $\Gamma(V_1, V_2)$ (elle aussi définie dans [LAF 02]), nous est aussi très utile dans le travail d'amorçage. En effet, les composantes communes à V_1 et V_2 seront fortement présentes dans $\Gamma(V_1, V_2)$. Cette notion permet d'amplifier les propriétés saillantes d'un vecteur dans un contexte donné.

Pour exploiter cette notion, nous associons à chaque vocable un vecteur égal à la somme normée des vecteurs de ses acceptions. En appliquant l'opération de contextualisation faible à un vecteur de vocable polysémique et à un vecteur contexte (de vocable polysémique ou non), nous obtenons un vecteur qui se rapprochera de l'un des sens du vocable considéré.

Par exemple, le terme *bank* est ambigu et son vecteur est globalement la moyenne entre les vecteurs des sens *river bank* et *money institution*. Si le vecteur de *bank* est contextualisé par celui de *river*, alors les concepts du champs sémantique lié à la finance seront considérablement inhibés.

4.2. *Construction de la base*

4.2.1. *Construction des dictionnaires monolingues*

En compilant les informations extraites de dictionnaires variés, (Hachette, Thésaurus Larousse, dictionnaire de synonymes de l'université de Caen, Wordnet, dictionnaire Oxford...) nous avons construit les lexies des dictionnaires monolingues français et anglais. Ces lexies, sont codées selon le schéma XML Papillon, mais contiennent très peu d'information (mot forme, catégorie et définition en langue naturelle).

Notre premier travail consiste à calculer le vecteur conceptuel associé à chacune de ces lexies. Le jeu de concept (les dimensions de l'espace) est prédéfini à l'aide des concepts présents dans le thésaurus Larousse.

Un indexage manuel de 5000 termes dans chaque langue, nous permet de connaître un premier ensemble de vecteurs. Ensuite, la définition de chaque lexie est analysée avec l'analyseur morphosyntaxique SYGMART. À partir des vecteurs des mots

connus de la définition, et de l'arbre d'analyse produit, nous calculons les vecteurs associés à chaque lexie et à chaque mot forme. Ce processus est itéré jusqu'à stabilité.

Nous disposons ainsi de dictionnaires monolingues vectorisés que nous définissons comme suit :

$$D_a = \{Lex_i\} \quad \text{et} \quad Lex_i = (w_i, \text{cat}_i, \text{def}_i, V_i) \tag{1}$$

Dans un dictionnaire monolingue vectorisé D_a, chaque vocable v correspond à n lexies $Lex_i = (v, \text{cat}_i, \text{def}_i, V_i)(n \geq 1)$. Si $n = 1$ le vocable v est strictement monosémique.

Nous donnons ci dessous un exemple des lexies du vocable *exiger* (dans cet exemple, les définitions sont issues du dictionnaire Hachette).

> **exiger.1** V, #s=1# Réclamer, en vertu d'un droit réel ou que l'on s'arroge. (Exiger le paiement de réparations) - (Exiger que (+subj)) (Il exige qu'on vienne), V_1
> **exiger.2** V, #s=2# Imposer comme obligation. (Allez-y, le devoir l'exige) (Les circonstances exigent que vous refusiez), V_2
> **exiger.3** V, Nécessiter. (Construction qui exige une main-d'oeuvre abondante), V_3

4.2.2. *Construction du dictionnaire interlingue d'acception*

Initialement, nous créons une acception interlingue (axie) pour chaque lexie des dictionnaires monolingues. Chaque acception est associée à un vecteur conceptuel. Initialement, ce vecteur est égal au vecteur de la lexie correspondante. La figure 12 présente l'état de la base lexicale après cette étape.

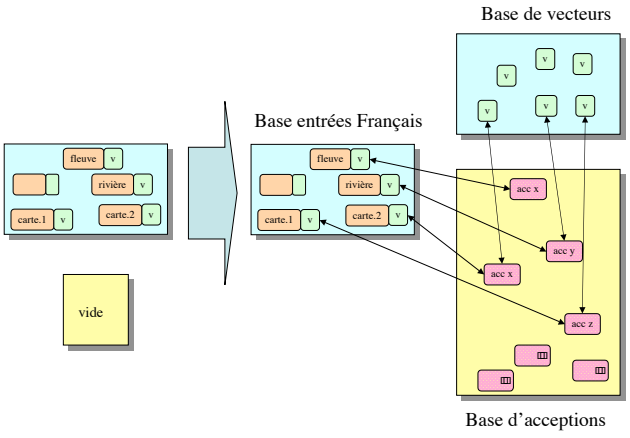


Figure 12. *Chargement initial du dictionnaire interlingue d'acceptions.*

Le travail qu'il reste à faire consiste à réduire cet ensemble d'acceptions à l'aide d'associations bilingues extraites de dictionnaire D_{a-b} (dictionnaires bilingues d'une langue source A vers une langue cible B) que nous définissons comme suit :

$$A(D_{a-b}, w) = \{Sa_i\} \quad \text{et} \quad Sa_i = (w, \text{cat}, \text{glose}^*, \text{equiv}^+)$$

Dans le dictionnaire D_{a-b}, le terme w est associée à n sous-entrée. Chaque sous-entrée contient : une information morphologique (au moins la catégorie morphosyntaxique, Nom, Verbe, Adjectif, Adverbe), zéro ou plus gloses, et au moins un équivalent dans la langue cible. Les gloses sont des termes optionnels qui permettent à l'utilisateur de sélectionner le sens dont il est question si le terme est polysémique. Ce sont ces même gloses qui permettent d'associer via les vecteurs conceptuels une sous-entrée du dictionnaire bilingue à une lexie du dictionnaire monolingue (en cas d'ambiguïté). Un exemple typique d'association bilingue (anglais-français) est :

demand ==
demand.1 VT, g{money, explanation, help}, e{exiger, réclamer})
demand.2 VT, g{higher pay}, e{revendiquer, réclamer})
demand.3 N, g{person}, e{demande})
demand.4 N, g{duty, problem, situation}, e{revendication, réclamation})
demand.5 N, g{for help, for money}, e{demande})

Dans le cas d'associations bilingues entre deux équivalents monosémiques (par exemple *babouin* $\rightarrow$ *baboon*) nous fusionnons les acceptions correspondantes. Un avertissement est émis pour le lexicographe si la distance entre les vecteurs des 2 acceptions est trop importante. Dans ce cas, il est vraisemblable, qu'au moins un des vecteurs conceptuels ne dispose pas d'activation pertinente.

Pour les autres associations bilingues, il nous faut identifier la lexie correspondant à chaque sous-entrée et la lexie correspondant à chaque équivalent.

Appariement lexie-association Pour chaque sous-association Sa_i, nous calculons un vecteur contexte V_C qui est la somme des vecteurs de ses gloses :

Le vecteur de la glose est le vecteur global, toutefois on sélectionne les sens dont les catégories morphosyntaxiques sont compatibles. Le vecteur associé à Sa_i est le calcul de la contextualisation faible (fonction Γ) entre le vecteur issu du dictionnaire monolingue pour w et le vecteur contexte. Le vecteur estimé $V_\approx$ de Sa_i est :

$$V_\approx(Sa_i) = \Gamma(V(w), V_C(Sa_i)) \tag{2}$$

À chaque sous-association Sa_i il est maintenant possible d'apparier un vecteur issu du dictionnaire monolingue. Il s'agit du vecteur (et donc du sens) qui est le plus proche du vecteur estimé (figure 13).

$$V(Sa_i) = Min(D_A(V(Se_j), V_\approx(Sa_i)) \tag{3}$$

Ainsi, nous avons apparié une partie des lexies et des associations bilingues. C'est-à-dire qu'en pratique nous avons pu fournir un vecteur aux associations bilingues ce

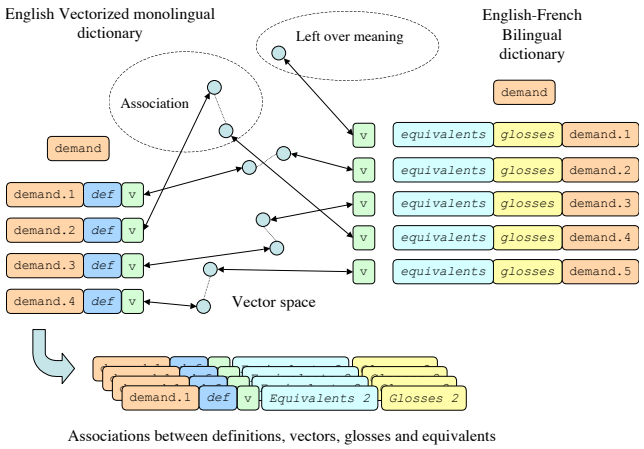


Figure 13. *Les associations sont appariées aux lexies dont le vecteur sémantique est le plus proche.*

qui est la condition pour les relier à des acceptions (dans les cas de polysémie). À la fin de ce processus, certaines des lexies du dictionnaire monolingue vectorisé disposent d'un lien unique vers une entrée du dictionnaire bilingue.

Liaison lexie-acceptions Il s'agit ici d'associer une lexie S_b du langage cible à une acception interlingue Ax_a issue d'une lexie du langage source.

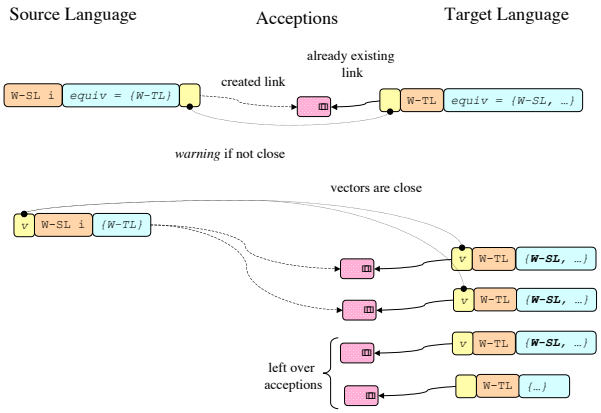


Figure 14. *Liaison Lexie-Acception dans le cas d'un équivalent monosémique (en haut) et d'un équivalent polysémique (en bas).*

On considèrera deux vecteurs conceptuels comme *suffisamment proche* si leur distance thématique est inférieur à un seuil t. Plus ce seuil est faible, plus le niveau de

confiance du lien vers l'acception est fort. En retour, il risque d'être difficile d'automatiquement réaliser l'association. Une valeur de seuil acceptable s'avère être $\pi/4$. Les différentes situations sont les suivantes :

1) **Un sens S vers un seul équivalent monosémique.** Ce cas consiste à sélectionner directement les termes. Les vecteurs conceptuels ne sont pas utilisés ici, si ce n'est pour effectuer une vérification. Si les deux vecteurs conceptuels ne sont pas raisonnablement proches, un message d'alerte est envoyé au lexicographe. Le problème peut aussi bien venir d'une erreur des dictionnaires bilingues, ou qu'un des vecteurs (ou les deux) à des activations inadéquates.

2) **Un sens S vers un équivalent polysémique.** Il faut alors sélectionner le sens équivalent S_b qui pourrait être acceptable (figure 14) Un filtre consiste à sélectionner les équivalents inverses, puis parmi les sens restant (s'il y en a plusieurs) choisir celui dont le vecteur est le plus proche.

3) **Un sens S vers plusieurs équivalent polysémique.** Ce cas est une généralisation des cas précédents.

4) **Cas d'erreur.** L'erreur principale provient de la constitution d'un ensemble vide. Cela peut arriver si les informations dans le dictionnaire bilingue sont inconsistantes. On remarquera que cela arrive relativement souvent en pratique.

4.2.3. *Nettoyage des liens*

L'architecture lexicale choisie impose des contraintes de bonne formation qu'il nous faut prendre en compte dans le peuplement du dictionnaire interlingue d'acceptions.

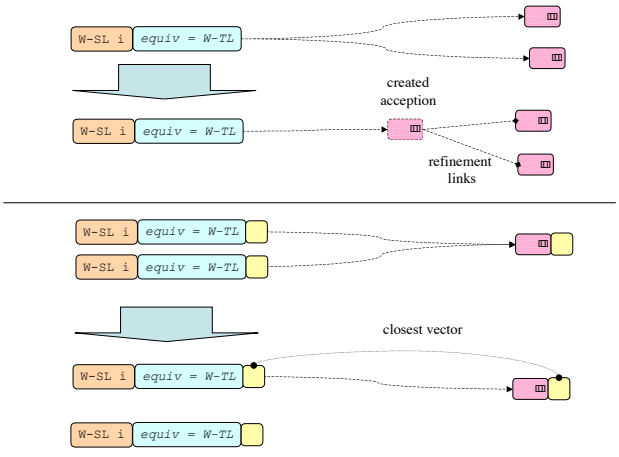


Figure 15. *Nettoyage de liens.* Partie supérieure *: Liens multiples. Une acception intermédiaire est créée ainsi que des liens inter-acceptions.* Partie inférieure *: Sens multiples. On conserve le lien minimisant la distance entre acception interlingue et lexie.*

1) Il y a, au plus, un lien possible d'une lexie vers une acception.

2) Deux lexies ne peuvent pas être liée à la même acception. Si ces sens sont synonymes, cette relation sera explicitée à l'aide d'une fonction lexicale (voir [SCH 01, SCH 02] pour une généralisation de l'approche à plusieurs fonctions lexicales caractéristiques).

Le nettoyage des liens consiste à détruire des liens qui auraient été créés abusivement (au vue des contraintes de bonne formation de la base interlingue) et d'en réévaluer certains. Deux cas peuvent se présenter : une lexie particulière est liée à plusieurs acceptions interlingues, ou une acception interlingue particulière est liée à plus d'une lexie dans une même langue.

1) **Liens multiples.** Pour résoudre ce problème, il est nécessaire de créer une acception intermédiaire. Le sens est alors lié seulement à la nouvelle acception (les liens précédents sont détruits). Des liens de raffinement de sens sont créés entre la nouvelle acception et les précédentes (figure 15).

2) **Sens multiples.** Nous avons à choisir parmi les sens lequel doit être retenu comme liable à l'acception, les autres liens étant détruits. La sélection se fait sur la base du plus proche vecteur.

En toute généralité, les deux situations peuvent survenir en même temps. Le processus de nettoyage est appliqué itérativement avec une priorité en faveur de la création d'acceptions intermédiaires.

4.3. *Premiers résultats et discussion*

Un certain nombre d'aspects doivent être mentionnés leur développement dépassant l'objet de cet article.

1) Vecteurs d'acceptions. Pour chaque acception, nous calculons son propre vecteur conceptuel. Ce vecteur est la moyenne des vecteurs des sens monolingues (dans l'ensemble des langues de la base) associés à l'acception. Ces vecteurs sont stockés comme s'ils constituaient un nouveau dictionnaire monolingue (cf. figure 14). Ils servent essentiellement à confirmer ou infirmer une proposition de liens lors de l'ajout de nouvelles entrées monolingues (à des acceptions déjà existantes).

2) Pondération des liens. Chaque lien créé automatiquement se voit attribuer une valeur de confiance (entre 0 et 1). Cette valeur correspond à la similarité entre le vecteur de l'acception et celui de l'entrée monolingue. Si un lien est confirmé par le lexicographe, le niveau de confiance vaut 1. L'exploitation des fonctions lexicales permet de moduler la valeur de confiance attribuée par le système ([SCH 01]).

3) Le processus de peuplement est effectué itérativement par des agents autonomes. Un agent explore la base d'acceptions et essaye d'évaluer les liens ou d'en créer. Par exemple, une acception pendante (avec un seul lien) doit être reliée à une entrée monolingue pour chacune des autres langues. Dans le cas d'entrée polysémique ou d'équivalent multiple, seule la source monolingue vectorisée nous concerne dans

la mesure où seuls les vecteurs conceptuels sont à la base du processus de décision. Les entrées orphelines doivent également être traité par la recherche d'une acception adéquate. Le processus est globalement convergent surtout dans la mesure où des liens sont fortement confirmés par les contributeurs humains.

Nos expériences de croisement de dictionnaires et de peuplement et liage automatique de la base d'acceptions nous ont permis dans le cas du français-anglais de générer environ 20000 acceptions dont environ 15000 était correctement liés. Le reste consistait en acceptions pendantes (soit du côté anglais soit du côté français). La plus grande difficulté concerne les entrées qui ne sont pas directement lexicalisées dans une langue. Dans ce cas, l'équivalent se réduit à une phrase explicative ou à une paraphrase. Ces traductions ne se retrouvent pas dans les dictionnaires monolingues de la langue cible. Par exemple le terme abêtir se traduit par *to make stupid, to turn into a moron* qui ne constitue pas des entrées du dictionnaire monolingue anglais. Afin de régler ce problème nous avons décidé de générer de telles entrées monolingues qui seront complétées par la suite (en particulier au niveau de leurs fonctions lexicales). Une petite frange d'acceptions (moins de 4%) et de sens (monolingue français ou anglais) sont incorrectement liés ou disposent de liens dont le seuil de confiance est inférieur à $1/2$. Il s'agit en général de termes très polysémiques (verbes support par exemple) qui génèrent beaucoup de forme lexicale voire de locutions dont la forme exacte peut être sujette à des variations. Cela engendre en particulier des amas d'acceptions lié par des raffinements de sens. En toute objectivité, l'approche fournit des résultats dans le rappel est important mais dont la précision est parfois médiocre. En l'occurrence, c'est très exactement la situation souhaitée où le lexicographe humain peut intervenir. Les termes polysémiques dont les champs sémantiques sont relativement distincts (par exemple un terme comme botte) sont correctement traités par les vecteurs conceptuels. Notre approche permet également d'améliorer la qualité des vecteurs conceptuels. Il s'agit ici d'une exploitation du graphe que représente les liens et les acceptions (indépendamment de sa construction). En particulier l'apport des lexicographes sur des informations lexicales permet d'augmenter la pertinence de certains vecteurs ce qui en retour améliore les performances du processus de peuplement.

5. Conclusion

Cet article présente un projet de construction d'une base lexicale multilingue linguistiquement riche. Par l'utilisation d'une stratégie basées sur le modèle « open source », nous souhaitons réduire les coût d'une telle construction en utilisant les compétences naturelles d'internautes volontaires. L'adoption d'une architecture lexicale interlingue qui sépare clairement les informations monolingues des informations interlingues présente de nombreux avantages dans ce cadre. D'une part, elle permet de distinguer les contributions interlingues des contributions monolingues qui requièrent des compétences différentes. D'autre part, elle permet de s'appuyer sur des langues bien dotées pour construire des données interlingues brutes impliquant des langues plus pauvres.

Le choix d'une architecture linguistique monolingue basée sur la composante lexicale de la théorie sens-texte d'Igor Mel'čuk, favorisera une réutilisation future de ces données dans de nombreuses et diverses applications de traitement des langues. De plus, la richesse des données construites rend plus intéressant le travail de contribution, les utilisateurs apportant de nombreuses informations originale, que l'on ne trouve actuellement dans aucun dictionnaire existant.

Le serveur de communauté sur lequel s'appuie le dictionnaire Papillon est en cours de construction, néanmoins, il est déjà fonctionnel et permet notamment un accès unifié à des dictionnaire existants. Ce service permet de rendre le site attractifs à des utilisateurs qui seront peut-être nos futurs contributeurs. Il nous a permis de plus de valider notre choix d'utiliser des outils standards (serveurs d'applications Java, XML pour la représentation des données, XSL pour leur manipulation, etc.). Ce serveur facilite aussi le travail des différents partenaires en proposant des services de partage de documents et d'archivage de liste de diffusion. Notre prochain objectif est d'ouvrir un service qui permettra des contributions en ligne par l'intermédiaire d'interfaces adaptables à l'utilisateur et à son environnement. À terme, nous souhaitons permettre des contributions en ligne (à partir d'un navigateur internet standard) et hors ligne (à partir d'application autonomes spécialisées). Le défi qui suivra consistera à animer une communauté de contributeurs et à trouver différentes motivations à même d'encourager la participation d'utilisateurs aux profils divers.

Une première réponse à ce défi réside dans la stratégie de construction employée, qui impose une phase d'amorçage assez complexe, mais néanmoins nécessaire pour disposer d'un ensemble de données suffisant qui sert à la fois de base de travail (les contributions sont vues comme des modifications de ces données) et de motivation (les données ainsi construites sont accessibles en ligne). Nos travaux préliminaires nous ont permis de produire une base d'acception sur le français et l'anglais par une méthode adaptable à d'autres langues avec un coût raisonnable. Il nous ont permis aussi d'associer un vecteur conceptuel à chaque acception créée. Ainsi, lors de la prise en compte de contributions, nous disposons de critères nous permettant de construire un agent automatique servant à la validation.

6. Bibliographie

[BAR 01] BARRIÈRE C., COPECK T., « Building Domain Knowledge from Specialized Texts », *TIA 2001*, Nancy, 2001.

[BLA 95] BLANC E., « Une maquette de base lexicale multilingue à pivot lexical : PARAX », *Lexicomatique et Dictionnairique, Actes du colloque LTT*, Universités Francophones, Actualités scientifiques, AUPELF-UREF, 1995, p. 43-58.

[BOU 93] BOURRIGAULT D., « Analyse locale pour le repèrage des termes complexes dans les textes », *TAL*, vol. 34, n° 2, 1993, p. 105-118.

[CHA 90] CHAUCHÉ J., « Détermination sémantique en analyse structurelle : une expérience basée sur une définition de distance », *TA Informations*, vol. 31, n° 1, 1990, p. 17-24.

[CHA 96] CHAUCHÉ J., SANDFORD E., « Détermination sémantique en analyse structurelle : une expérience basée sur une définition de distance », *Actes de MIDDIM-96*, Le Col de Porte, France, Août 1996, p. 56-66.

[CRU 95] CRUSE D. A., TOGIA P., « Towards a cognitive model of antonymy », *Lexicology*, vol. 1, 1995, p. 113-141.

[DEE 90] DEERWESTER S., DUMAIS S., LANDAUER T., FURNAS G., HARSHMAN R., « Indexing by latent semantic analysis », *Journal of the American Society of Information Science*, vol. 416, n° 6, 1990, p. 391-407.

[DES 02] DESPERRIER J.-M., « Analyzis of the results of a collaborative project for the creation of a Japanese-French dictionary », *Papillon'2002 Seminar*, NII, Tokyo, Japan, July 2002, http ://www.papillon-dictionary.org/ConsultInformations.po.

[FEL 95] FELLBAUM C., « Co-occurrence and antonymy », *International Journal of Lexicography*, vol. 8, 1995, p. 281-303.

[FIS 73] FISCHER W. L., *Äquivalenz und Toleranz Strukturen in der Linguistik zur Theory der Synonyma*, Max Hüber Verlag, München, 1973.

[GWE 87] GWEI G., FOXLEY E., « A Flexible Synonym Interface with application examples in CAL and Help Environments », *The Computer Journal*, vol. 30, n° 6, 1987, p. 551-557.

[HAM 01] HAMON T., NAZARENKO A., « La structuration de terminologie : une nécessaire coopération », *TIA 2001*, Nancy, 2001.

[HAT 01] HATHOUT N., « Analogies morpho-synonymiques. Une méthode d'acquisition automatique de liens morphologiques á partir d'un dictionnaire de synonymes », *TALN 2001*, Tours, July 2001, p. 223-232.

[HEA 98] HEARST M., « Automated discovery of Wordnet relations », FELLBAUM C., Ed., *Wordnet An Electronic Lexical Database*, MIT Press, Cambridge, MA, 1998, p. 131-151.

[JUS 91] JUSTESON J., KATZ S., « Co-occurrences of antonymous adjectives and their contexts », *Computational Linguistics*, vol. 17, 1991, p. 1-19.

[LAF 99] LAFOURCADE M., SANDFORD E., « Analyse et désambiguïsation lexicale par vecteurs sémantiques », *TALN'99*, Cargèse, juillet 1999, p. 351-356.

[LAF 01a] LAFOURCADE M., « Lexical sorting and lexical transfer by conceptual vectors », *First International Workshop on MultiMedia Annotation (MMA'2001)*, Tokyo, January 2001, page 6.

[LAF 01b] LAFOURCADE M., PRINCE V., « Synonymies et vecteurs conceptuels », *TALN 2001*, Tours, Juillet 2001, p. 233-242.

[LAF 02] LAFOURCADE M., PRINCE V., SCHWAB D., « Vecteurs conceptuels et structuration émergente de terminologies », *TAL*, vol. 43, n° 1, 2002, p. 43-72.

[MAN 01] MANGEOT-LEREBOURS M., « Environnements centralisés et distribués pour lexicographes et lexicologues en contexte multilingue », Thèse de nouveau doctorat, Spécialité Informatique, Université Joseph Fourier Grenoble I, Septembre 2001.

[MAN 02] MANGEOT-LEREBOURS M., « An XML Markup Language Framework for Lexical Databases Environments : the Dictionary Markup Language », *LREC Workshop on International Standards of Terminology and Language Resources Management*, Las Palmas, Islas Canarias, Spain, May 2002, p. 37-44.

[MEL 84] MEL'ČUK I., ARBATCHEWSKY-JUMARIE N., ELTNISKY L., IORDANSKAJA L., LESSARD A., *DEC : Dictionnaire explicatif et combinatoire du français contemporain*,

recherches lexico-sémantiques I, Presses de l'université de Montréal, Montréal(Quebec), Canada, 1984.

[MEL 88] MEL'ČUK I., ARBATCHEWSKY-JUMARIE N., DAGENAIS L., ELNITSKY L., IORDANSKAJA L., LEFEBVRE M.-N., MANTHA S., *DEC : Dictionnaire explicatif et combinatoire du français contemporain, recherches lexico-sémantiques II,* Presses de l'université de Montréal, Montréal(Quebec), Canada, 1988.

[MEL 92] MEL'ČUK I., ARBATCHEWSKY-JUMARIE N., IORDANSKAJA L., MANTHA S., *DEC : Dictionnaire explicatif et combinatoire du français contemporain, recherches lexico-sémantiques III,* Presses de l'université de Montréal, Montréal(Quebec), Canada, 1992.

[MEL 96] MEL'ČUK I., ARBATCHEWSKY-JUMARIE N., IORDANSKAJA L., MANTHA S., POLGUÈRE A., *DEC : Dictionnaire explicatif et combinatoire du français contemporain, recherches lexico-sémantiques IV,* Presses de l'université de Montréal, Montréal(Quebec), Canada, 1996.

[MEL 01] MEL'ČUK I., WANNER L., « Towards a Lexicographic Approach to Lexical Transfer in Machine Translation (Illustrated by the GermanRussian Language Pair) », *Machine Translation*, vol. 16, n° 1, 2001, p. 21-87, Kluwer Academic Publishers.

[PLO 98] PLOUX S., VICTORRI B., « Construction d'espaces sémantiques á l'aide de dictionnaires de synonymes », *TAL*, vol. 39, n° 1, 1998, p. 161-182.

[POL 00] POLGUÈRE A., « Towards a theoretically-motivated general public dictionary of semantic derivations and collocations for French », *Proceeding of EURALEX'2000, Stuttgart*, 2000, p. 517-527.

[POL 02] POLGUÈRE A., « Notions de base en lexicologie », OLST-Département de linguistique et de traduction, Université de Montréal, 2002.

[QUA 01] QUAH C. K., BOND F., YAMAZAKI T., « Design and Construction of a machine-tractable Malay-English Lexicon », *Proceedings of AsiaLex*, Seoul, 2001, p. 200-205.

[RES 95] RESNIK P., « Using Information contents to evaluate semantic similarity in a taxonomy », *IJCAI-95*, 1995.

[RIL 95] RILOFF E., SHEPHERD J., « A corpus-based bootstrapping algorithm for Semi-Automated semantic lexicon construction », *Natural Language Engineering*, vol. 5, n° 2, 1995, p. 147-156.

[SAL 68] SALTON G., *Automatic Information Organisation and Retrieval*, McGraw-Hill, New York, 1968.

[SCH 01] SCHWAB D., « Vecteurs conceptuels et fonctions lexicales : application à l'antonymie », Mémoire de DEA Informatique, LIRMM, Montpellier, 2001.

[SCH 02] SCHWAB D., LAFOURCADE M., PRINCE V., « Amélioration de la représentation sémantique lexicale par les vecteurs conceptuels : le rôle de l'antonymie », *JADT 2002, vol. 2*, 2002, p. 701-712.

[SÉR 94a] SÉRASSET G., « Approche oecuménique au problème du codage des structures linguistiques », BLACHE P., Ed., *TALN-94 : Le traitement automatique du langage naturel en France aujourd'hui*, vol. 1, 7 to 8 April 1994, p. 109-118.

[SÉR 94b] SÉRASSET G., « Sublim : un système universel de bases lexicales multilingues et Nadia : sa spécialisation aux bases lexicales interlingues par acceptions », Thèse nouveau doctorat, Université Joseph Fourier-Grenoble 1, Décembre 1994.

[SPA 86] SPARCK JONES K., *Synonymy and Semantic Classification*, Edinburgh Information Technology Series, Edinburgh University Press, 1986.

[VER 01] VERLINDE S., SELVA T., « DAFA - Dictionnaire d'Apprentissage du Français des Affaires », http ://www.projetdafa.net, 2001.

[YAR 92] YAROWSKY D., « Word-Sense Disambiguation Using Statistical Models of Roget's Categories Trained on Large Corpora », *COLING'92*, Nantes, 1992, p. 454-460.

Table des matières

MÉTHODES ET OUTILS POUR LA LEXICOGRAPHIE BILINGUE EN LIGNE : LE CAS DU GRAND DICTIONNAIRE ESTONIEN-FRANÇAIS

Antoine CHALVIN

INALCO
2, rue de Lille
F-75005 PARIS

Mathieu MANGEOT

Condillac - LISTIC - Université de Savoie
Campus Scientifique
F-73376 LE BOURGET DU LAC
CEDEX

RÉSUMÉ

Le projet de construction du Grand dictionnaire estonien-français (GDEF), du fait de sa spécificité — une équipe rédactionnelle dispersée —, a immédiatement ressenti la nécessité d'utiliser des méthodes informatiques innovantes permettant le travail à distance et en réseau. Les initiateurs de ce projet ont donc tout naturellement décidé d'utiliser une plate-forme générique de construction de dictionnaires en ligne : la plate-forme Jibiki, fruit de recherches en lexicographie computationnelle.
Après avoir exposé les conditions générales dans lesquelles s'inscrit ce projet de lexicographie bilingue en ligne (nécessité d'un tel dictionnaire, travail à distance, structure complexe, bases de données lexicales utilisées), l'article explique les méthodes de travail mises en œuvre dans ce cadre (protocole de rédaction en trois étapes) et les solutions informatiques qui les rendent possibles (interface de rédaction en ligne, gestion des contributions, import-export de données, outils annexes).

Méthodes et outils pour la lexicographie bilingue en ligne :
le cas du Grand dictionnaire estonien-français

L'objet de la présente communication est de décrire les méthodes de travail mises en œuvre dans le cadre d'un projet de lexicographie bilingue en ligne, le Grand dictionnaire estonien-français (GDEF)[1], ainsi que les solutions informatiques qui les rendent possibles.

1. Exposé du problème

1.1. Nécessité d'un nouveau dictionnaire estonien-français

L'adhésion de l'Estonie à l'Union européenne et le développement de l'enseignement du français dans ce pays font apparaître comme une urgente nécessité la réalisation d'un nouveau dictionnaire estonien-français. Il n'existe en effet aucun grand dictionnaire estonien-français répondant aux exigence de la lexicographie moderne et reflétant l'état actuel de la langue estonienne, dont le lexique a considérablement évolué au cours des quinze dernières années. Le dernier dictionnaire d'une certaine ampleur (Kann et Kaplinski 1979), réalisé à l'époque soviétique, est aujourd'hui en grande partie périmé. Il est en outre entaché de graves défauts de conception, auxquels il ne semble pas possible de remédier par une simple mise à jour.

1.2. Une équipe rédactionnelle bilingue et dispersée

Une équipe bilingue de linguistes et de traducteurs a donc décidé en 2002 de se lancer dans la rédaction d'un Grand dictionnaire estonien-français d'environ 80 000 entrées. L'un des problèmes principaux auquel le projet s'est trouvé confronté, du fait de son caractère binational, a été celui de la dispersion géographique des rédacteurs entre trois villes (Paris, Tallinn, Tartu) et deux pays (la France et l'Estonie).

1.3. Travail à distance

Cette dispersion imposait de définir des méthodes et d'élaborer des outils permettant de travailler entièrement à distance. Il était donc nécessaire, avant le début de la rédaction proprement dite, de mettre en place un environnement de travail virtuel permettant : 1) de saisir les articles au format

[1] Dictionnaire réalisé par l'Association franco-estonienne de lexicographie, sous la responsabilité d'Antoine Chalvin (directeur scientifique), Madis Jürviste, Indrek Koff et Jean Pascal Ollivry, avec le soutien de l'Agence intergouvernementale de la Francophonie, de la Fondation Robert Schuman et du Centre culturel français de Tallinn.

XML, 2) de consulter la base de données au fur et à mesure de sa constitution, 3) d'organiser les différentes étapes et les différents aspects du travail rédactionnel (attribution des articles, suivi de l'avancement du travail, révision, validation, discussions entre les rédacteurs, etc.)

1.4. Une structure d'article complexe

La réalisation d'une interface d'édition était rendue plus difficile par la relative complexité de la structure d'article (en raison du degré de détail visé par le dictionnaire) : celle-ci comporte en effet jusqu'à cinq niveaux de blocs emboîtés (blocs grammaticaux, sémantiques, sous-blocs sémantiques, blocs contextuels, équivalents) et plusieurs éléments susceptibles d'avoir un nombre d'occurrences infini, ce qui rendait impossible l'utilisation de formulaires HTML classiques.

1.5. Plate-forme de construction

Une solution informatique satisfaisante à l'ensemble de ces problèmes a pu être trouvée grâce à une instance de la plate-forme Jibiki, complétée et adaptée progressivement aux besoins spécifiques du projet entre décembre 2003 et juillet 2005. Cette plate-forme, conçue au départ pour le projet Papillon (Mangeot et al. 2003), est un environnement générique en ligne permettant la rédaction et la consultation de tous types de dictionnaires : glossaires terminologiques, dictionnaires bilingues, bases lexicales multilingues, etc. Elle a été développée principalement à partir de 2001 par Mathieu Mangeot (Université de Savoie) et Gilles Sérasset (Université de Grenoble 1), notamment grâce à des recherches (Mangeot 2001) menées au sein du laboratoire GETA-CLIPS de Grenoble. La plate-forme est implantée en Java à l'aide d'outils en source libre. Elle est basée sur Enhydra, un serveur Web d'objets dynamiques en Java, et Postgres, une base de données relationnelle.

2. Base lexicale

2.1. Structure de la base lexicale

Les données lexicales du GDEF sont stockées dans deux bases distinctes : une base estonienne et une base française. La première contient l'essentiel des informations figurant dans les articles du dictionnaire, à l'exception des équivalents français et de leurs informations grammaticales « permanentes », non dépendantes du contexte d'emploi (genre des substantifs, pluriels irréguliers des substantifs et des adjectifs, féminins irréguliers des adjectifs, *h* aspiré, etc.). L'insertion dans un article du dictionnaire d'un équivalent français et l'affichage des informations le concernant se font

au moyen d'un lien reliant un élément XML de l'article de la base estonienne à un article de la base française. Cette technique permet une gestion centralisée des diverses informations concernant les mots français, ce qui permet d'éviter des erreurs, de corriger immédiatement dans l'ensemble du dictionnaire les données fautives, et d'ajouter ultérieurement des informations non prévues ou non disponibles au départ (classe de conjugaison des verbes, auxiliaire utilisé à la conjugaison active, prononciation).

2.2. Données lexicographiques de départ

Ces deux bases de données ont été constituées en extrayant automatiquement les informations pertinentes de plusieurs bases lexicales existantes.

Pour l'estonien, nous avons utilisé le grand dictionnaire estonien-russe (EVS) en cours de rédaction à l'Institut de la langue estonienne de Tallinn (3 volumes publiés, de A à P, achèvement prévu en 2007). À partir de la version XML de ce dictionnaire, nous avons extrait les mots vedettes et leurs informations morphologiques, les subdivisions sémantiques de chaque article avec les indications sémantiques correspondantes, les indications de domaine de spécialité et de registre, les exemples et les locutions. Nous disposons ainsi d'une nomenclature de 40 000 mots estoniens (de A à P) et d'une première version de chaque article, que les rédacteurs n'ont plus qu'à adapter et à compléter avec les informations françaises.

Environ 6 330 articles estoniens ont en outre été pourvus d'une indication de fréquence tirée du Dictionnaire des fréquences de l'estonien écrit (Kaalep et Muischnek 2002). Cette information nous permet de rédiger en priorité les articles sur les mots les plus fréquents, afin de maximiser l'utilité du dictionnaire en cours d'élaboration qui est immédiatement consultable sur Internet.

La base de données française a été extraite de Morphalou, dictionnaire des formes fléchies du français élaboré par l'ATILF et comprenant environ 66 000 lemmes à catégorie grammaticale unique (caractéristique qui rendait cette base parfaitement adaptée à l'usage que nous voulions en faire). Nous en avons extrait, outre les lemmes, la catégorie grammaticale, le genre des substantifs, ainsi que les pluriels et féminins « irréguliers » (selon nos critères). L'ampleur de la base française ainsi constituée permet d'ores et déjà un fonctionnement satisfaisant du système d'établissement des liens, même s'il nous reste encore à ajouter les données absentes de Morphalou (*h* aspiré, classe de conjugaison, mots composés, etc.).

3. Méthodes de travail et solutions informatiques

3.1. Procédure de rédaction

Le travail de rédaction se déroule par groupes, chacun d'eux rassemblant au moins un rédacteur estonien et un rédacteur français, qui sont amenés tous les deux à intervenir sur l'article. La réalisation d'un article comporte trois étapes. Dans un premier temps, le rédacteur — Français ou Estonien — contrôle la pertinence de la structure sémantique proposée par la base de données, insère les équivalents français en spécifiant éventuellement leurs contextes d'emploi, leur registre et leurs rections, il adapte au besoin le choix d'exemples et de locutions proposé et en fournit une traduction. Il peut consulter son coéquipier locuteur natif de l'autre langue. Il doit en outre tester les équivalents sur des exemples relevés dans deux corpus estoniens en ligne (corpus de l'estonien écrit de l'Université de Tartu et corpus de l'Institut de la langue estonienne) ainsi que sur Google. Dans une deuxième étape, l'article est révisé par un autre membre du groupe. Tout article rédigé par un Français est révisé par un Estonien et vice-versa. La nature de la révision dépend évidemment de la langue maternelle du réviseur. Un Estonien vérifiera surtout la pertinence des distinctions sémantiques, la correction des indications sémantiques et contextuelles rédigées en estonien par le rédacteur français, la pertinence des exemples et des locutions. Un réviseur français s'attachera surtout à vérifier la pertinence des équivalents et des indications concernant leurs contextes d'emploi. Une fois révisé, l'article est soumis à une ultime vérification, opérée par l'un des deux validateurs français. Le nombre de validateurs est volontairement limité afin de garantir un minimum de cohérence dans la présentation des articles. Une fois validé, l'article devient accessible à la consultation par tous les visiteurs du site.

3.2. Interface d'édition

L'interface d'édition (formulaire de saisie) permet d'éditer des structures relativement complexes grâce à des outils de gestion de listes. L'ajout et la suppression de champs ou de blocs de champs dans le formulaire se font très simplement, au moyen d'un bouton à cliquer. Il est également possible de modifier l'ordre des blocs au sein de certaines listes de blocs. Un module spécifique permet d'établir les liens vers la base française : lorsque le rédacteur saisit un équivalent français dans la fenêtre surgissante de ce module, le système recherche les articles correspondants. S'il n'en trouve qu'un seul, il place automatiquement le lien dans le champ adéquat du formulaire. S'il en

trouve plusieurs, il affiche un résumé des différents articles (avec leur catégorie grammaticale et leur genre) et propose au rédacteur de choisir celui vers lequel il veut établir le lien. S'il n'en trouve aucun, il propose au rédacteur de créer lui-même l'article français. Lors de la sauvegarde, les informations saisies dans le formulaire sont converties automatiquement au format XML.

3.3. Gestion des contributions

Pour gérer le travail de rédaction-révision-validation, la plate-forme permet de définir des groupes et des droits d'accès. Il existe trois groupes prédéfinis (réviseurs, validateurs, administrateurs) et il est possible de créer un nombre illimité de groupes de travail. Si l'utilisateur n'est pas logué, il n'appartient à aucun groupe. Il peut simplement consulter les ressources disponibles sur la plate-forme. Lorsqu'il est logué, il accède à l'interface d'édition des articles. Les réviseurs peuvent modifier et réviser les contributions des autres membres de leur groupe de travail. Les validateurs peuvent modifier toutes les contributions et valider les contributions révisées. Les administrateurs peuvent gérer les utilisateurs et les groupes, ajouter de nouvelles ressources sur la plate-forme, etc.

Les données sont gérées de la façon suivante : les données lexicographiques, constituant l'article proprement dit, sont encapsulées dans une *contribution*, qui contient également un certain nombre de méta-informations sur l'article : auteur, réviseur, validateur, dates de création, révision, validation, ainsi qu'un historique de toutes les modifications.

La contribution comporte un statut, qui évolue pendant le cycle de rédaction en passant successivement par quatre états : « non-finie », « finie », « révisée », « validée ». Une contribution validée est en principe définitive, mais il est possible de la modifier en en faisant une copie, qui repasse alors par toutes les étapes du processus de rédaction. Une fois cette copie validée, ses données remplacent celles de la contribution d'origine.

4. Gestion et suivi du travail

4.1. Attribution des articles

L'attribution des articles aux rédacteurs se fait au moyen d'une interface spécifique accessible aux validateurs. Cette interface permet d'attribuer des paquets d'articles constitués selon différents critères combinables entre eux. Il est par exemple possible d'attribuer des tranches alphabétiques avec ou sans prise en compte de la fréquence, des groupes de termes relevant d'un même domaine

de spécialité, des séries de mots composés comportant le même élément final, de réattribuer tous les articles non finis d'un auteur donné à un autre auteur, etc. La liste complète des articles concernés est affichée pour vérification avant l'attribution.

4.2. Tableau résumé

Pour permettre le suivi de l'avancement du travail et le calcul de la rémunération des rédacteurs, il est possible d'obtenir un tableau résumé de toutes les contributions finies, révisées et validées sur une période donnée. Cette fonctionnalité est accessible à tous les membres permanents de l'équipe, ce qui leur permet notamment de comparer leur productivité à celle des autres rédacteurs.

4.3. Exportation et importation des données

La plate-forme dispose d'une fonction d'exportation, qui peut être utilisée par exemple pour la relecture à l'écran ou sur papier de groupes d'articles. Il est possible de combiner plusieurs critères de recherche pour constituer la liste des articles à exporter. L'utilisateur peut aussi définir le format d'exportation : XML, HTML, format texte et format PDF pour l'impression.

Il est également possible d'importer des articles, à condition que ceux-ci soient au format XML. Cette fonctionnalité pourrait permettre aux lexicographes qui le désirent de travailler en local puis de réimporter leurs contributions une fois le travail terminé et lorsqu'une connexion à l'Internet est accessible. En pratique, nous l'utilisons surtout pour ramener des contributions à un statut antérieur (par exemple dans le cas d'une contribution validée par erreur) ou lorsque le format des données doit être modifié, soit pour ajouter de nouvelles informations, soit pour raffiner certaines parties de la structure. Dans ce cas, les contributions concernées sont exportées au format XML, modifiées, puis réimportées sur la plate-forme.

5. Outils annexes

5.1 Forum

Pour faciliter la communication entre les rédacteurs et avec le public, un forum a été installé sur le serveur du projet. Il est constitué de deux parties : l'une réservée aux membres du projet et l'autre d'accès public. La partie réservée aux membres comprend plusieurs sous-forums : sur l'un figure la dernière version du protocole de rédaction, un autre est réservé à la discussion sur les problèmes

rédactionnels généraux non encore couverts par le protocole, un troisième permet de discuter des futurs développements de la plate-forme, un quatrième est consacré au partage de liens thématiques vers des glossaires ou autres ressources en lignes, et un cinquième, organisé selon l'ordre alphabétique estonien, permet de discuter de problèmes concernant des mots précis.

Bibliographie

A. Dictionnaires

Grand dictionnaire estonien français (GDEF) : http://www.estfra.ee/

Eesti-vene sõnaraamat [dictionnaire estonien-russe] I-III. Tallinn. Eesti Keele Instituut. 1997-2004.

Kaalep, H-J., Muischnek, K. (2002) *Eesti kirjakeele sagedussõnastik* [dictionnaire des fréquences de l'estonien écrit]. Tartu, Tartu Ülikooli kirjastus.

Morphalou, lexique morphologique ouvert du français : http://actarus.atilf.fr/morphalou/

B. Autres

Corpus de l'Institut de la langue estonienne (Tallinn) : http://www.eki.ee/corpus/

Corpus de l'estonien écrit (Université de Tartu) : http://www.cl.ut.ee/korpused/kasutajaliides/index.html.en

Mangeot, Mathieu (2001) *Environnements centralisés et distribués pour lexicographes et lexicologues en contexte multilingue*. Thèse de doctorat, Spécialité Informatique, Université Joseph Fourier Grenoble I, 2001, 280 p.

Mangeot, Mathieu ; Thevenin, David (2004) 'Online Generic Editing of Heterogeneous Dictionary Entries in Papillon Project.' in Proc. COLING 2004, ISSCO, Université de Genève, 23-27 August 2004, vol 2/2, 1029-1035.

Mangeot, Mathieu ; Sérasset, Gilles ; Lafourcade, Mathieu (2003) 'Construction collaborative de données lexicales multilingues, le projet Papillon.' *Les dictionnaires électroniques : pour les personnes, les machines ou pour les deux ?* ed. M. Zock and J. Carroll, *TAL Traitement Automatique des Langues*, Vol. 44 : 2/2003, 151-176.

MotàMot project: conversion of a French-Khmer published dictionary for building a multilingual lexical system

Mathieu Mangeot
GETALP-LIG Laboratory, 41 rue des mathématiques BP 53
F-38041 GRENOBLE CEDEX 9
France
Email: Mathieu.Mangeot@imag.fr

Abstract

Economic issues related to the information processing techniques are very important. The development of such technologies is a major asset for developing countries like Cambodia and Laos, and emerging ones like Vietnam, Malaysia and Thailand.
The MotAMot project aims to computerize an under-resourced language: Khmer, spoken mainly in Cambodia. The main goal of the project is the development of a multilingual lexical system targeted for Khmer. The macrostructure is a pivot one with each word sense of each language linked to a pivot axi. The microstructure comes from a simplification of the explanatory and combinatory dictionary.
The lexical system has been initialized with data coming mainly from the conversion of the French-Khmer bilingual dictionary of Denis Richer from Word to XML format. The French part was completed with pronunciation and parts-of-speech coming from the FeM French-english-Malay dictionary. The Khmer headwords noted in IPA in the Richer dictionary were converted to Khmer writing with OpenFST, a finite state transducer tool.
The resulting resource is available online for lookup, editing, download and remote programming via a REST API on a Jibiki platform.

Keywords: French-Khmer dictionary, MotÀMot, Jibiki

1. Introduction

Economic issues related to the information processing techniques are very important. The development of such technologies is a major asset for developing countries like Cambodia and Laos, and emerging ones like Vietnam, Malaysia and Thailand.
The MotAMot project aims to computerize an under-resourced language: Khmer. This language is spoken mainly in Cambodia.
As indicated by V. Berment in his Ph.D. thesis (Berment, 2004), "With the development of personal computers and networks, the computer becomes now a tool for writing and communicating in the same way as paper and printing were before. Word processing, email, and even more advanced systems such as dictation or voice synthesis are widely used tools. The idea then is needed than appropriate information technologies tools must be added to traditional means without which the targeted goals can not be achieved any more". The computerization of a language such as Khmer occupies a central place in this wider context.
The main goal of the project is the development of a multilingual lexical system targeted for Khmer. The lexical system has been initialized with data coming mainly from the conversion of the French-Khmer bilingual dictionary of Denis Richer from Word to XML format.

2. Presentation of the resource to be built

2.1. Microstructure of the entries

The microstructure of the monolingual volumes is based on the Explanatory and Combinatorial Lexicography. Each entry is based on the vocable. A vocable is either a group of lexies (word meaning) or an idiom.
Each entry consists of a headword and a pronunciation, followed by a list of word meaning (lexies). Each lexie is described either by a lexico-semantic formula (Melcuk & Polguère, 2006 ; Polguère, 2006) or a free gloss. For terms, it is possible to describe their domain. Then comes a translation link pointing to an axie (entry) of the pivot volume. It is followed by a list of examples and idiomatic expressions. Finally, a generic field is used to store additional information from reused dictionaries.
To cope with different contributor skill levels, the editing interface adapts itself and displays an appropriate granularity of information. For example, a novice contributor will be prompted for a simple gloss to characterize a lexical unit, while an expert linguist will describe a complete semantic formula. Similarly, only specific contributors have access to the list of lexical functions.

2.2. Macrostructure of the lexical database

The macrostructure is composed of a monolingual volume for each language and a central pivot volume. However, in order not to confuse users, they contribute through an interface with a classic view of a bilingual dictionary. Each bilingual link language A ➔ language B added via this interface (step 1 in Figure 1) is actually translated into the background by creating two interlingual links as well as an axie link representing the original translation to finally get: language A ➔ pivot axie ➔ language B (step 2 in Figure 1).

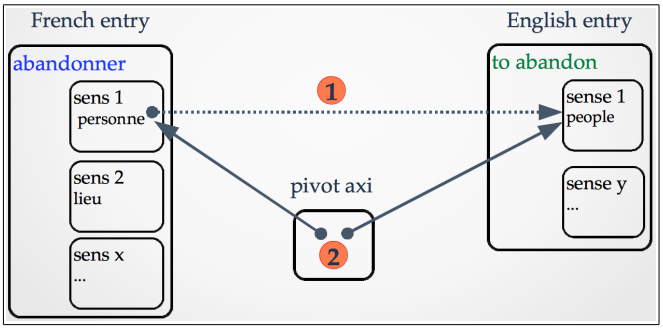


Figure 1: Building bilingual links

If a contributor wants to add a translation link between a vocable Va of language A and a vocable Vb of language B, s/he can establish this link at different levels. The ideal solution is to connect a word meaning (lexie) La of the vocable Va to another word meaning Lb of the vocable Vb. In this case, the link is bijective and Lb is also connected to La (point 1 in Figure 2).

If the vocable Vb has not yet defined specific word meanings or if the contributor can not choose between word meanings, s/he can connect directly the word meaning La to the vocable Vb. In this case, a new word meaning Lb is created with a draft level of quality and the link as well as the word meaning are marked for refinement (point 2 in Figure 2)..

In the case of reusing existing data, it is often impossible to link information to a specific word meaning. In this case, we add to the end of the vocable Va the information that one of the meanings of vocable Va can be connected to a meaning of vocable Vb but this information will not be added to Vb (point 3 in Figure 2). It will of course be marked as to be refined with emergency!

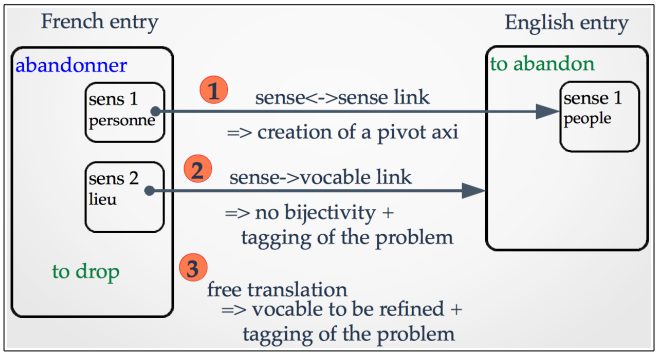


Figure 2: Different types of links

With the pivot macrostructure, if two links language A ➔ language B and language B ➔ language C exist, then it will automatically create a link language A ➔ language C whose level will be marked as a draft and to be revised.

2.3. Levels for contributions and contributors

Each information of each entry is assigned a quality level. The levels range from 1 star for a draft (recovered data whose quality is not known) to 5 stars, quality certified by an expert (e.g.: a link translation validated by a sworn translator).

Similarly, contributors will be assigned a skill level (1 to 5 stars as well). 1 star is the level of an unknown novice in the community and 5 stars is the level of a recognized expert.

Then, when a level 3 contributor revises a level 2 entry, the entry automatically goes up to level 3. Similarly, if the work of a contributor is systematically validated without corrections by other contributors to the next level, it can automatically switch to the next level after a certain threshold (e.g. 10 contributions).

To go further, we plan to analyse the work of the contributors. If a person contributes massively eg on a particular area, the system will automatically send regular proposals for contribution in the domain.

3. Preparation of existing data

In order to encourage contributions, it is preferable to provide a skeleton dictionary to modify later rather than an empty dictionary. For each language involved, a list of words of this language will be recovered to create an initial list of entries. It will always be possible to create a new article, but the creations will be subject to verification.

3.1. Conversion of a French-Khmer dictionary into XML

3.1.1. Description of the dictionary

For the Khmer language, there is at present a French-Khmer dictionary (Richer et al., 2007), which started in the late 90s, was completed in 2006 by a small group of computer scientists gathered in the "Pays Perdu" NPO created by Denis Richer, french ethnolinguist established in Siem Reap (Cambodia). This first version of the dictionary was published in spring 2007 and has 13,249 entries. Each entry is composed by a French headword, in some cases a part-of-speech and a list of word senses. Each word sense has a gloss in French and a translation in Khmer. The dictionary was originally encoded in Word format. An example of the original file can bee seen on Figure 3.

abondant, e (fruits, riz…) — (pluie) (trempé-humide)	(dāel) sambō (dāel) cō̤k-coam
abonnement (magazine) — (téléphone)	cīəw-prā cam kā- baŋ-sēvā
abonner (sur pied-nom (s'inscrire)-commercer)	coh-chmuəh-cīəw
abonner (s') (à un magazine) — (au téléphone)	cīəw-prā cam baŋ-sēvā
abord (adv.) (d'—)	mun-dambōŋ / dāəm-lāəj
aborder (accoster) — (qqn) (appeler-arrêter) — (commencer-discuter-sur-sujet-un)	cōl-cā t / cōl-cət haw-baŋchop phdāəm-cɔ̄cɛk-ampī- paŋə-hā- mūəj
aboutir (arriver à destination, déboucher) — (avoir-résultat) — (devenir) (aller-être)	tə̄w-dal mīən-lɔ̄ttəphā l tə̄w-cīə

Figure 3: Extract of the dictionary in Word format

3.1.2. Conversion of the file into XML

We first converted the dictionary from Word format to a lexical resource in XML format according to the methodology described in (Mangeot & Enguehard, 2013).

The Open Document Format has the great advantage of being based on XML. Instead of a conversion, the contents of the XML document has been retrieved, then transformed to obtain what we want.

A document in ODF format is actually a zip archive containing multiple files including the text content in XML. This content is stored in the "content.xml" file in the archive. To retrieve this file, just follow some clever manipulations. On MacOs, create an empty folder and then copy the .odt file inside. Then, open a terminal and run the "unzip" command to unzip the file. On Windows, you must change the .odt file extension into .zip and then open the. zip archive.

The file "content.xml" can now be extracted from the archive and then renamed and placed in another location. It becomes the base file on which we will continue our work. The next step consists in editing this file with a "raw" text editor with syntax highlighting and regular expressions based search and replace.

One may first think that since the source file "content.xml" is already in XML, it may be enough to write an XSLT stylesheet to convert the file into an XML dictionary, but the XML used in the source file is completely different from the XML targeted. Indeed, the source file comes from a word processor. It is designed for styling a document and not for structuring a dictionary entry. Therefore, it is finally easier to convert the XML file "by hands" with regular expressions than to write an XSLT stylesheet for automatically converting the source file.

3.1.3. Tagging each part of information

All pieces of information were explicitly tagged by replacing the ODF markup by a new "homemade" tagset. Each piece of information is usually distinguished from others in the original file with a different style. Search/replace operations were performed for each type of information.

The original dictionary file has tow columns. The French is on the left one and the Khmer on the right one. Each entry has one or several word senses. Each word sense is written on one line and has a khmer translation on the right column. Therefore, each line was tagged with "sens" tag (see Figure 4).

```
<article>
 <vedette>abondant</vedette>
 <sens>
  <glose>abondant, e (fruits, riz…)</glose>
  <traduction><api>(dāel) sambō</api>
  </traduction></sens>
 <sens>
  <glose>(pluie) (trempé-humide)</glose>
  <traduction><api>(dāel) cō̤k-coam</api>
  </traduction></sens>
</article>
```

Figure 4: French entry "abondant" after XML conversion

On the original Word file, feminine forms of adjectives are separated from the masculine with a comma. The feminine form were separated in order to clearly identify the headword of each entry.

The original form of the entry is duplicated in the gloss field.

The XML file obtained is a unique volume containing French entries translated into Khmer.

3.2. Restructuring and links reification

3.2.1. Conversion into the MotÀMot structure

The XML data obtained is then converted into the structure chosen for the MotÀMot project.

In files coming from word processors, the data structure is usually implied. New structural elements must be added to move towards a more standardized structure. Concerning standards, Lexical Markup Framework (LMF) (Romary et al., 2004) became an ISO standard in November 2008 (Francopoulo et al., 2009). It suits ideally our goals. As it is a meta-model and not a format, the principle of the LMF model can be applied to the entry structure and keeping our tags without using the LMF syntax (Enguehard & Mangeot, 2013). The entry tag corresponds to the LexicalEntry object; the head tag corresponds to the Form object; The headword tag corresponds to the Lemma object; the sense tag corresponds to the Sense object; The gloss tag corresponds to the Definition object; the refaxie tag corresponds to the Equivalent object (Figure 6).

The core meta-model LMF is shown in Figure 5.

Unique identifiers are also added to entries and word senses (Figure 6). They will be used afterwards for linking entries.

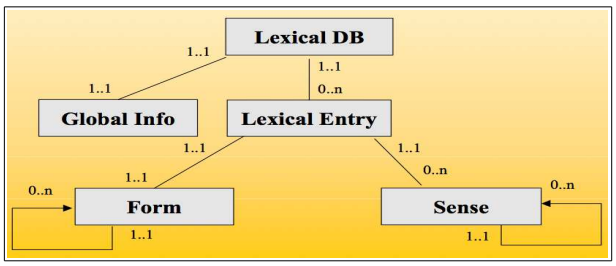


Figure 5: LMF core metamodel

```
<m:entry id="fra.abondant.27.e" level="">
 <m:head>
  <m:headword>abondant</m:headword>
  <m:pos></m:pos>
 </m:head>
 <m:sense id="s1" level="">
  <m:gloss>abondant, e (fruits, riz…)</m:gloss>
  <m:translations>
   <m:translation>sambō</m:translation>
  </m:translations>
 </m:sense>

  <m:sense id="s2" level="">
   <m:gloss>(pluie) (trempé-humide)</m:gloss>
   <m:translations>
    <m:translation>cǭk-coam</m:translation>
   </m:translations>
  </m:sense>
 </m:entry>
```

Figure 6: French entry "abondant" after restructuring

3.2.2. Link reification

We then converted this unique volume into three volumes : one French, one pivot and one Khmer by eliciting the translation links. From each French->Khmer translation, we obtain 2 links : one French->Pivot and one Pivot->Khmer.

The entry `fra.abondant.27.e` has 2 Khmer translations: "`sambō`" & "`cǭk-coam`".

They were elicited into the 4 following links:

1. Sense `s1` of French entry `fra.abondant.27.e` linked to Axi entry `axi.[fra:abondant,khm:sambō].27.1.e`;
2. Axi entry `axi.[fra:abondant,khm:sambō].27.1.e` linked to Khmer entry `sambō`;
3. Sense s2 of French entry `fra.abondant.27.e` linked to Axi entry `axi.[fra:abondant,khm:cǭk-coam].27.2.e`;
4. Axi entry `axi.[fra:abondant,khm:cǭk-coam].27.2.e` linked to Khmer entry `cǭk-coam`.

Figure 7 shows a French entry after reification.

The axi volume is obtained by gathering all the axies in the same file.

The Khmer volume is obtained by gathering all the Khmer entries and by merging them when they have the same Khmer word as headword (at this stage, the comparison is based on the IPA of the Khmer word). Each link to an axie constitutes a sense.

The volume will then be sorted by Unicode sort order of the Khmer headword in Khmer writing after the process described in part 4.

The French volume contains 13,249 entries; the Khmer volume contains 23,766 khmer entries; the axie volume contains 32,402 axies linking to both a French and a Khmer entry.

3.2.3. Adding data in the French volume

The French entries were enriched with data from the FeM French-English-Malay dictionary : the pronunciation of the headword and the part-of-speech for the non-homonymous vocables.

```
<m:entry id="fra.abondant.27.e" level="">
  <m:head>
    <m:headword>abondant</m:headword>
    <m:pronunciation>ABON-DAN-</m:pronunciation>
    <m:pos>adj.</m:pos>
    <m:fem_form>abondante</m:fem_form>
    <m:fem_pron>ABON-DAN-T</m:fem_pron>
  </m:head>
  <m:sense id="s1" level="">
   <m:gloss>abondant, e (fruits, riz…)</m:gloss>
   <m:refaxie idrefaxie="axi.[fra:abondant,
khm:sambō].27.1.e" />
 </m:sense>

 <m:sense id="s2" level="">
     <m:gloss>(pluie) (trempé-humide)</m:gloss>
     <m:refaxie idrefaxie="axi.[fra:abondant,
 khm:cǭk-coam].27.2.e" />
  </m:sense>
 </m:entry>
```

Figure 7: French entry "abondant" after reification

4. Conversion of the IPA in Khmer alphabet

The Khmer part is a simple phonetic transcription of the Khmer translations written in a special API font (SILSophia IPA93) created by the Summer Institute of Linguistics. We developed a program of transcription from the phonetic form to the Khmer writing[1] based on a finite state transducer (FST): OpenFst[2]. This program is available online for use at the following URL:

http://jibiki.univ-savoie.fr/khmer/

The source code is also available for download from the same page. This work is licensed under the Creative Commons Zero license, public domain.

4.1. IPA normalisation

The first step is to normalize the IPA notation. It consists in two different parts. The first part is a correction of mistakes in the IPA notation. Some IPA letters that do not

1 http://jibiki.univ-savoie.fr/khmer/
2 http://www.openfst.org/

exist in Khmer pronunciation were replaced by the correct ones.
The second part is the normalisation of combined letters. In Unicode, some letters with diacritics can be written by combining a basic character with different marks (combining macron, ring below, etc.) but also exist in one global character. When one character exists, the combining characters where replaced by the equivalent global character.

Original	Replacement
Corrections	
y	j
f	hv
n	ŋ
Normalisations	
a + ◌̄	ā
o + ◌̄ + ◌̥	ō̥

Table 1: replacement operations for API normalisation

4.2. Intermediate notation

During this step, two operations are performed at the same time.
The first one consists in grouping the series of API letters that correspond to one letter in Khmer.
The longest matches are processed before.
The second one consists in tagging the type of syllable. In Khmer, each consonant can be written in two different ways following the composition of the syllable in which the consonant is included. These two variants are considered as two different letters with different Unicode codes. We call the two variants A and B and we tag each consonant with its type.
For tagging the type of consonants, several heuristics are applied in the following order:

In the original file, Khmer words noted in IPA are separated by the '-' character. This feature was used in the conversion when word beginning and endings require special treatments. Some vowels beginning a word like “a” and “ɔ” are specific Khmer letters.

- a	-a
- ɔ t	- ɔt t
- ā	-a ā

The consonant “l” ending a word is a type B consonant.

l -	l B -

There are three types of vowels:
- some are located after a type A consonant;

ā e	A āe
ā o	A āo

- other after a type B consonant;

u ɔ	B uɔ
ō a	B ōa

- the third category can be located indifferently after the two types of consonants. Hopefully, only two vowels are in this case.

ū ɔ	ūɔ
◌̲ ɯ ɔ	◌̲ ɯɔ

After the three and two IPA letter combinations, come the unique letters:

e	A e
ū	B ū

Next, the consonants are grouped:

c h	ch
k h	kh
p h	ph
t h	th

For some consonants followed by third type vowels, it is possible to attribute them automatically a type.

b ūɔ	b A ūɔ
ch ūɔ	ch B ūɔ
d ūɔ	d A ūɔ
t ūɔ	t ūɔ
b ◌̲ ɯɔ	b B ◌̲ ɯɔ
k ūɔ	k B ūɔ
t ◌̲ ɯɔ	t B ◌̲ ɯɔ

4.3. Khmer generation

The third step consists in generating Khmer writing from the intermediate notation with the two types of consonants.
The vowels are replaced with the same Khmer letters:

◌̄ɑ	អ◌̄
◌̲ ɯɔ	◌ុ◌

The consonants are replaced following their type:

kh + v + A	ខ្វ
kh + v + B	ឃ្វ

The Khmer language writing system has many ambiguities,. Some letters (most of them word endings) are not pronounced. The result is far from perfect. Nevertheless, it was still added to entries and will then be reviewed by khmer native speakers.

5. Availability on the Web with the Jibiki platform

Jibiki is a generic platform for manipulating online lexical resources (Mangeot, 2001) with users and group management, consultation of heterogeneous resources and generic dictionary articles edition. This is a community website originally developed for the Papillon project (Mangeot et al., 2003).

Jibiki distinguishes form other lexical resources development platforms such as TshwaneLex from tshwanedje.com, the Dictionary Publishing System from IDM or FieldWork from the SIL on the following points:

- it is completely free and open-source;
- it allows to browse and edit any kind of XML lexical resource without modifying the XML structure or tags (so far, more than 100 resources were successfully imported into a Jibiki platform);
- it allows to manipulate complex macrostructures such as pivot ones or lexical networks;
- it is available online from a simple Web browser;
- it is remotely programmable via a REST API.

5.1. Description of the platform

The platform is programmed entirely in Java based on the environment "Enhydra". All data is stored in XML format in a database (Postgres). This website mainly offers two services: a unified interface for simultaneous access to many heterogeneous resources (monolingual, bilingual dictionaries multilingual databases, etc...) and a specific editing interface to contribute directly to the dictionaries available on the platform.

Several lexical resources construction projects used or still use this platform successfully (Mangeot & Chalvin 2006; Sérasset et al., 2006). The source code for this platform is published under the LGPL open-source licence and available for free download from the open forge of the LIG laboratory[3].

An instance of the platform has been adapted specifically to the MotÀMot project. The language of the interface is French. It is also envisaged to localize it in Khmer.

5.2. Visualisation of Khmer text in browsers

Unfortunately, widespread web browsers based on the Webkit engine such as Safari or Chrome but also word processors such as OpenOffice family are still not able to display correctly all the Khmer letters. Some letters have to be composed. The order must change. In the following example, the last letter of the word "jɔ̄n-hɔh" (plane) has a dash circle which is in fact an empty placeholder for a vowel. On the same system with the same configuration (MacOs X 10.6.8), Firefox is able to fill in the placeholder (Figure 9) but not Safari (Figure 8).

យអណ្ហាះ /jɔ̄n-hɔh/

Figure 8: Khmer word not correctly displayed in Safari

យអណ្ហាះ /jɔ̄n-hɔh/

Figure 9: Khmer word correctly displayed in Firefox

5.3. Volume Lookup

Two different lookup interfaces are available to the user. The volume lookup allows the user to lookup a word or prefix on a specific volume, French or Khmer. The Khmer entries can be searched from their headword in IPA or Khmer writing. On the left part of the result window, the volume headwords are displayed, sorted in alphabetical order. An infinite scroll allows the user to browse the entire volume. On the right part of the window, the entries previously selected on the left part are displayed (see Figure 10).

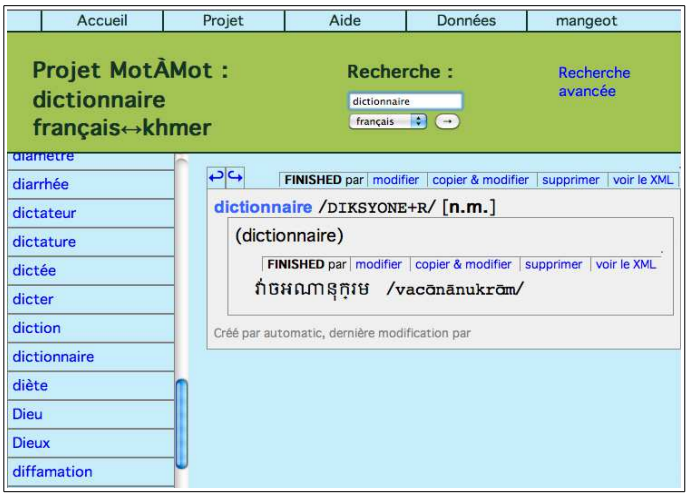


Figure 10 : Entry "dictionnaire" of the French-Khmer dictionary in Jibiki

5.4. Advanced lookup

The advanced lookup is available for complex multi-criteria queries. For example, it is possible to lookup an entry with a specific part-of-speech, and created by a specific author. On the left part of the result window, the headwords of the matching entries are displayed, sorted in alphabetical order. An infinite scroll allows the user to browse all the matching entries. On the right part, the entries previously selected on the left part are displayed.

3 http://jibiki.ligforge.imag.fr

5.5. Online editing

The editing module is based on an HTML interface model instantiated with the entry to be edited (see Figure 11). The model can be generated automatically from a description of the structure of the entry using an XML schema. It can then be modified to improve the screen rendering. The only information required to publish a dictionary article is the XML schema representing the structure of this entry. In addition, it is possible to edit any type of dictionary provided that it is encoded in XML. Figure 8 shows the editing interface for the French entry "dictionnaire". The link to the pivot axie is indicated in the `refaxie` field of the `sense` block. If a contributor wants to add a new link to a Khmer entry, s/he will use the curved arrow button on the left of the `reflexie` field in the `translation` block. Then s/he will be able to look up a word in the Khmer volume. When a Khmer entry is selected, an axie is generated in the background and the appropriate links are added to the French and Khmer entries. The resulting axie link is then shown in the `refaxie` field of the `sense` block.

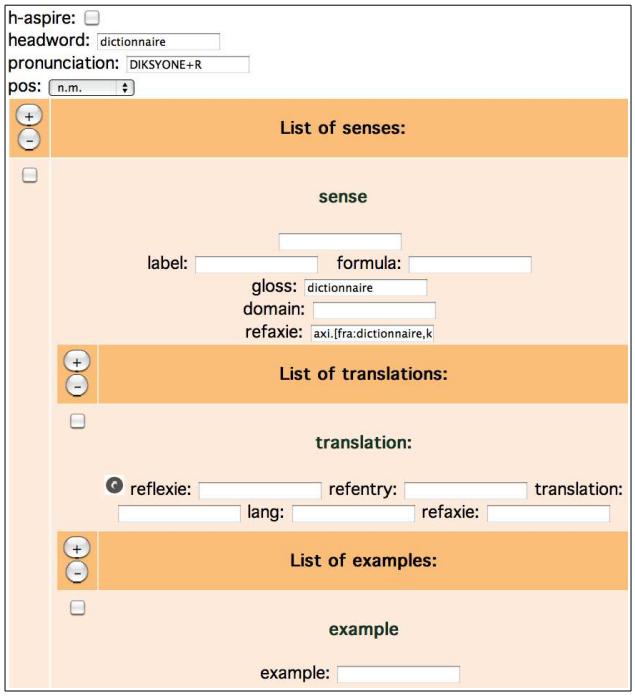


Figure 11: Editing interface of the French entry "dictionnaire"

5.6. XML data available for download

The XML data of the dictionary (French, Khmer and axies volumes) is available for download with a CreativeCommons CC by license via the "Données" tab on the MotÀMot website. Data exports from the database will be regularly performed and the new versions available for download.

5.7. Remote access via a REST API

Once dictionaries are uploaded into the Jibiki server, they can be accessed via a REST API[4]. Lookup commands are available for querying indexed information: headword, pronunciation, part-of-speech, domain, example, idiom, translation, etc. (see Figure 12). The API can also be used for editing entries provided that the user previously registered in the website.

Querying entries

URL	api/[dictionary]/[lang]/[criteria]/[string]	
URL	api/[dictionary]/[lang]/[criteria]/[string]/[key]	
Method	GET	
Querystring	strategy=	strategy
	count=	count
	startIndex=	startIndex
Returns	200 OK & XML (entry)	
	404 Not Found	

The searches with the [key] part give the value of the key for the corresponding entries. For example, searching an entry with *api/*/eng/cdm-headword/trial/cdm-pos* will give a list of parts-of-speech for the "trial" entry.

It is possible to query all the dictionaries by putting a * instead of the name of the dictionary. It is also possible to do the same for the [lang] and [key] part of the URL.

The **handle** criteria is special as it does not give a list of entry handles but the full entry XML content.

Figure 12: API documentation for querying entries

6. Conclusion

The MotÀMot project has now ended. The project website is available at the following URL:
http://jibiki.univ-savoie.fr/motamot/
Users can lookup and edit French and Khmer dictionary entries. They can also download the resulting XML data. The platform can be used remotely from other applications via a REST API.
The quality of the data can be increased subsequently. Particularly the Khmer entries and headwords that were generated automatically need revision by Khmer native speakers. Of course, people can already voluntarily contribute, but it is very difficult to do so without at least a community manager that can help and motivate contributors. We are now looking for funds in order to found a second project focused on the data revision.

7. Acknowledgements

The MotÀMot project has been partly funded by the Agence Universitaire de la Francophonie. We also thank Vanra IENG for his work on the project.

8. References

Berment, V. (2004). Méthodes pour informatiser des langues et des groupes de langues peu dotées. Thèse de nouveau doctorat, Université Joseph Fourier, Grenoble, France.

Enguehard, Ch. & Mangeot, M. (2013). LMF for a selection of African Languages. Chapter 7, book "LMF:

4 http://jibiki.univ-savoie.fr/motamot/Api.po

Lexical Markup Framework, theory and practice", Ed. Gil Francopoulo, Hermès science, Paris, France, 17 p.

Francopoulo, G., Bel, N., George, M., Calzolari, N., Monachini, M., Pet, M., and Soria, C. (2009). Multilingual resources for NLP in the lexical markup framework (LMF). Language Resources and Evaluation, 43(1), pages 57–70, March.

Mangeot, M. (2001). Environnements centralisés et distribués pour lexicographes et lexicologues en contexte multilingue. Thèse de nouveau doctorat, spécialité informatique, Université Joseph Fourier (Grenoble 1), septembre 2001, 280 p.

Mangeot, M., Sérasset, G. & Lafourcade, M. (2003). Construction collaborative de données lexicales multilingues, le projet Papillon. Revue TAL, édition spéciale, Les dictionnaires électroniques : pour les personnes, les machines ou pour les deux ? (Electronic dictionaries: for humans, machines or both?) Ed. Michael Zock & John Carroll, Vol. 44:2/2003, pp. 151-176.

Mangeot, M. & Chalvin, A. (2006). Dictionary Building with the Jibiki Platform: the GDEF case. Proc. of LREC 2006, Genoa, Italy, 23-25 May 2006, pp 1666-1669.

Mangeot, M. & Enguehard, Ch. (2013). Ressources Lexicales : contenu, construction, utilisation, évaluation. Chapitre 8 : Des dictionnaires éditoriaux aux représentations XML standardisées, Ed. Nuria Gala, Michael Zock. Hermès science, Paris, 24 p.

Mel'čuk, I. & Polguère, A. (2006). Dérivations sémantiques et collocations dans le DiCo/LAF. Langue française, numéro spécial sur la collocation « Collocations, corpus, dictionnaires », sous la direction de P. Blumenthal et F. J. Hausmann, 150, juin 2006, 66-83.

Polguère, A. (2006). Structural properties of Lexical Systems: Monolingual and Multilingual Perspectives. Proceedings of the Workshop on Multilingual Language Resources and Interoperability (COLING/ACL 2006), Sydney, 50-59.

Richer, D., Keo, T. & Vania, I. (2007). Dictionnaire Français-Khmer (en phonétique), D.R. Edition, ISBN-13:890-0-9.

Romary L., Salmon-Alt S., Francopoulo G. (2004). Standards going concrete: from LMF to Morphalou. Workshop on Electronic Dictionaries, COLING 2004, Geneva, Switzerland.

Sérasset, S., Brunet-Manquat, F. & Chiocchetti, E. (2006). Multilingual legal terminology on the Jibiki platform: the LexALP project. Coling-ACL 2006: Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, Sydney.

Computerization of African languages-French dictionaries

Chantal Enguehard, Mathieu Mangeot

Laboratoire LINA
2 rue de la Houssinière, BP 92208
F-44322 NANTES CEDEX 03 France

Laboratoire GETALP-LIG
41 rue des mathématiques BP 53
F-38041 GRENOBLE CEDEX 9 France

Email: Chantal.Enguehard@univ-nantes.fr, Mathieu.Mangeot@imag.fr

Abstract

This paper relates work done during the DiLAF project. It consists in converting 5 bilingual African language-French dictionaries originally in Word format into XML following the LMF model. The languages processed are Bambara, Hausa, Kanuri, Tamajaq and Songhai-zarma, still considered as under-resourced languages concerning Natural Language Processing tools. Once converted, the dictionaries are available online on the Jibiki platform for lookup and modification.

The DiLAF project is first presented. A description of each dictionary follows. Then, the conversion methodology from .doc format to XML files is presented. A specific point on the usage of Unicode follows. Then, each step of the conversion into XML and LMF is detailed. The last part presents the Jibiki lexical resources management platform used for the project.

Keywords: DiLAF, dictionary, Jibiki

1. Introduction

The work behind this paper has been done during the DiLAF project to computerize African languages-French dictionaries (Bambara, Hausa, Kanuri, Tamajaq, Zarma) in order to disseminate them widely and extend their coverage. We present a methodology for converting dictionaries from Word .doc format in a structured XML format following the Unicode character encodings and Lexical Markup Framework (LMF) standards. The natural language processing of African languages is in its infancy. It is our duty to help our colleagues from the South in this way. This requires, among other things, the publication of articles, that are a valuable resource for under-resourced languages.

Many studies have been conducted in the past in this area. However, it seemed interesting to redefine a new methodology taking into account recent developments such as the Open Document Format (ODF) or LMF standards. On the other hand, we wanted to develop the simplest possible method based solely on free and open source tools so that it can be reused by many. This method can also be used for other dictionaries and by extension, any text document (language resource at large) to be converted to XML.

2. Presentation of the DiLAF project

If access to computers is considered as the main indicator of the digital divide in Africa, we must recognize that the availability of resources in African languages is a handicap with incalculable consequences for the development of Information Technology and Communication Technologies (ICT). Most languages in francophone West Africa area are under-resourced (π-language) (Berment, 2004): electronic resources are scarce, poorly distributed or absent, making use of these languages difficult when it comes to introducing them into the education system and especially develop their use in writing in the administration and daily life.

Dictionaries are the cornerstone of processing natural language, be it in the mother tongue or in a foreign language. The primary function of communication is conveying meaning, yet meaning is primarily conveyed through vocabulary. As David Wilkins, a british linguist (1972) wrote “so aptly “While without grammar little can be conveyed, without vocabulary nothing can be conveyed".

Thus, to help bridge this gap, we are engaged with colleagues from North and South to improve the equipment of some African languages through, among others, the computerization of printed dictionaries of African languages.

The DiLAF project aims to convert published dictionaries into XML format for their sustainability and sharing (Streiter et al., 2006). This international project brings together partners from Burkina Faso (CNRST), France (LIG & LINA), Mali (National Resource Centre of the Non-Formal Education) and Niger (INDRAP, Department of Education, and University of Niamey).

Based on work already done by lexicographers we formed multidisciplinary teams of linguists, computer scientists and educators. Five dictionaries were converted and integrated into the Jibiki lexical resources management platform (Mangeot, 2001). These dictionaries are therefore available on the Internet[1] under a Creative Commons license:

- Bambara-French dict. Charles Bailleul, 1996 edition;
- Hausa-French dict. for basic cycle, 2008 Soutéba;
- Kanuri-French dict. for basic cycle, 2004 Soutéba;
- Tamajaq-French dict. for basic cycle, 2007 Soutéba;
- Zarma-French dict. for basic cycle, 2007 Soutéba.

1 http://dilaf.org/

The aim of these usage dictionaries is to popularize the written form of the daily use of African languages in the pure lexicographical tradition (Matoré, 1973) (Eluerd, 2000). Departing from interventionist approaches of normative dictionaries (Mortureux, 1997), the present descriptive dictionaries remain open to contributions and their online availability online will, hopefully, develop a sense of pride among users of these languages. Similarly, they will participate in the development of a literate environment conducive to increase the literacy whose low level undermines the achievements of progress in other sectors.

3. Presentation of the dictionaries

Four of the five dictionaries have been produced by the Soutéba project (program to support basic education) with funding from the German cooperation and support of the European Union. These dictionaries for basic education have a simple structure because they were designed for children of primary school class in a bilingual school (education is given there in a national language and in French). Most terms of lexicology, such as lexical labels, parts-of-speech, synonyms, antonyms, genres, dialectal variations, etc. are noted in the language in question in the dictionary, contributing to forge and disseminate a meta-language in the local language, a specialized terminology. The entries are listed in alphabetical order, even for Tamajaq (although it is usual for this language to sort entries based on lexical roots) because the vowels are written explicitly (this mode of classification was preferred because it is well known by children).

3.1. Hausa-French dictionary

The Hausa-French dictionary includes 7,823 entries sorted according to the following lexicographical order: a b ɓ c d ɗ e f fy g gw gy h i j k kw ky ƙ ƙw ƙy l m n o p r s sh t ts u w y y̆ z (République du Niger, 1999a).

They are structured with different patterns according to the part-of-speech. All entries are typographical, followed by the pronunciation (tones are marked with diacritics placed on vowels) and part-of-speech. On the semantic level, there is a definition in Hausa, a usage example (identified by the use of italics), and the equivalent in French. For a noun, the gender, feminine, plurals and sometimes dialectal variants are noted. For verbs, it is sometimes necessary to specify the degree to calculate morphological derivatives. Morpho-phonological variants of feminine and plural adjectives derivations are also written.

Example:

> **jaki** [jàakíi] *s.* **babbar dabbar gida mai kamar doki, wadda ba ta kai tsawon doki ba amma ta fi shi dogayen kunnuwa**. *Ya aza wa jaki kaya za ya tafi kasuwa. Jin.:* n. *Sg.:* **jaka**. *Jam.:* **jakai, jakuna**. *Far.:* âne

3.2. Kanuri-French dictionary

The Kanuri-French dictionary includes 5,994 entries sorted according to the following lexicographical order: a b c d e ə f g h i j k l m n ny o p r ɾ s sh t u w y z (République du Niger, 1999b).

The orthographic form of the entry is followed by an indication of pronunciation targeting rating tones. The part-of-speech is shown in italics, followed by a definition, a usage example, a French translation and meaning in French. Additional information may appear as variants.

Example:

> **abərwa** [àbə̀ɾwà] *cu.* **Kəska təngəri, kalu ngəwua dawulan tada cakkidə.** *Kəryende kannua nangaro, abərwa cakkiwawo.* [Fa.: ananas]

3.3. Soŋay Zarma-French dictionary

The Zarma-French dictionary includes 6916 entries sorted according to the following lexicographical order: a ã b c d e ẽ f g h i ĩ j k l m n ŋ ɲ o õ p r s t u ũ w y z (République du Niger, 1999d).

Each entry has an orthographic form followed by a phonetic transcription in which the tones are rated according to the conventions already set for the Kanuri. The part-of-speech specify explicitly the transitivity or intransitivity of verbs. For some entries, antonyms, synonyms and references are indicated. A gloss in French, a definition and an example end the entry.

Example:

> **ɲagas** [ɲágás] *mteeb.* ● *brusquement (détaler)* ● sanniize no kaŋ ga cabe kaŋ boro na zuray sambu nda gaabi sahã-din ● *Za zankey di hansu-kaaro no i te ɲagas*

3.4. Tamajaq-French dictionary

The Tamajaq-French dictionary includes 5,205 entries sorted according to the following lexicographical order: a â ă ə b c d ḍ e ê f g ğ h i î j ǰ ɣ k l ḷ m n ŋ o ô q r s ṣ š t ṭ u û w x y z ẓ (République du Niger, 1999c)

The orthographic form of the entry is followed by the part-of-speech and a gloss in French displayed in italics. For nouns, morphological information about the state of annexation is often included, the plural and gender are also explicitly stated. A definition and an example of usage follow. Other information may appear as variants, synonyms, etc. As Tamajaq is not a tonal language, phonetics does not appear.

Example:

> **əbeɣla** *sn.* mulet ♦ **Ag-anɣer əd tabagawt.** *Ibeɣlan wər tăn-tăha tămalăya.* anammelu.: **făkr-ejăḍ.** təmust.: **yy.** iget.: **ibəɣlan**.

3.5. Bambara-French dictionary

The Bambara-French dictionary of Father Charles Bailleul (1996 edition) includes more than 10,000 entries sorted according to the following lexicographical order: a b c d e ɛ f g h i j k l m n ɲ ŋ o ɔ p r s t u w y z.

This dictionary is primarily intended for French speakers wishing to improve Bambara but it is also a resource for Bambara speakers. In the words of the author himself, the

dictionary "plays the role of a working tool for literacy, education and Bambara culture." To date, it can be considered as the most comprehensive dictionary of the language. It is also used by specialist of other varieties of this language like Dyula (Burkina Faso, Côte d'Ivoire) and Malinké (Guinea, Gambia, Sierra Leone, Liberia, etc.).

4. General conversion methodology and tools

The main objective is to convert dictionaries from a word processor format adapted for human use into XML and explicitly mark all the information so one can use them automatically in natural language processing tasks. The constraints are, on the one hand, working with free, open and multi-platform tools and on the other hand define a simple process that can be understood and then performed independently by linguists having no computer knowledge except regular expressions.

4.1. Conversion methodology

The conversion methodology follows these steps:

1. conversion of problematic characters to Unicode;
2. conversion of OpenOffice format to XML;
3. identification and explicit tagging of each part of information (headword, part-of-speech, etc.);
4. XML validation and manual correction of errors in the data (closed lists of values, references);
5. entries structuring following the LMF standard.

4.2. Tools used

The first tool allows one to edit files in original format and then convert them into XML. For this step OpenOffice (or LibreOffice) is ideal. It is free and open source. Furthermore, besides the ISO standard Open Document Format (odt), it can open many Microsoft formats (rtf, .doc as well as .docx). Finally, the XML produced is simple, especially compared to Office Open XML Microsoft.

OpenOffice has a regular expression engine for search/replace functions. This tool can be used for many steps before converting to XML. However, the search/replace may be problematic because during replacements, the boundaries of text styles can be changed (part of a word is suddenly in italics). Because we rely on styles to convert to XML, we limit the conversion to Unicode characters to keep styles intact.

Then, we need an editor to modify the files. For these operations, the XML editors are not very useful because they do not directly change the plain text with regular expressions and most are not able to edit large files like dictionaries. We recommend using a simple "raw" text editor supporting regular expressions and syntax highlighting.

For XML validation and verification steps, a web browser such as FireFox does it very nicely. It is able to detect and display the XML validation errors and can interpret CSS and XSLT style to enhance the display.

4.3. Incremental backups needed

The methodology intends to make backups at each stage and keep track of all search/replace operations done in order to go back when errors resulting from improper action are identified. Sometimes it happens that an error is noticed long after being made. If an error can not be corrected simply by a new search/replace, it is possible to go back from a previous version.

Despite all precautions, sometimes errors are detected very late and it is very difficult to go back. If the error can not be detected automatically, it will require manual correction. One must keep in mind that nobody is perfect and yet others even better trained had to forget the possibility to automatically correct all the errors in the conversion process.

5. Use of Unicode

5.1. Characters conversion to Unicode

Although the alphabets of languages on which we have worked (Enguehard 2009) are mainly of Latin origin, new characters needed to note specific sounds in some languages with a single character has been adopted by linguists in a series of meetings. Thus, each of the alphabets we previously presented comprises at least one of these special characters: ɓ ɗ ƎɛɣƘɲŋɔƴ. Characters composed of a Latin character and a diacritical mark were also created: âêîôûãẽĩõũḍḷṣṭẓǧǰšr.

Although most of these characters are present for several years in the Unicode standard (based on the work of the ISO 10646 (Haralambous 2004)), dictionaries were written using old hacked fonts. A methodology has been defined to identify and replace the inadequate characters

abirillu [àbìríllù] *m.* • *avril* • annasaara handu taacanta kaŋ go marsu nda me game ra • *Abirillu, 15, 1974 no Sayni Kunce na hino sambu* • *f/b.* abirillo, abirilley

abiyanso [àbìyànsôo] *m.* • *aéroport* • batama kaŋ ra abiyey ga zumbu • *Tilbeeri nda Dooso sinda abiyanso kaŋ ra abiyo beeri ga zumbu* • *f/b.* abiyansa, abiyansey

abiyo [àbíyò] *m.* • *avion* • naarumay hari no kaŋ ra i ga boro nda jinay daŋ a ma deesi nd'ey • *Jidda no abiyey ga alfujaajey zumandi* • *him.* beene-hi • *f/b.* abiya, abiyey

abunaadam [àbúnăadàm] *m.* • *être humain, personne* • *f/b.* abunaadamo, abunaadamey • *di.* adamayze

Figure 1: Excerpt of the Zarma-French dictionary in original format.

with the ones defined in the Unicode standard. It implies that all identified characters are recorded in a file so one can easily repeat this operation if necessary. Table 1 shows part of the list for Zarma. There is no automatic method that will detect these problematic characters. It is imperative to look at the data.

Origin	*Unicode*
§	ã
é	ẽ
$	ɲ
ù	ŋ
£	Ɲ

Table 1: Partial view of the Unicode correspondence table for Zarma.

5.2. Digraphs lexicographical order

Digraphs can be easily typed using two characters but their use changes the sort order which determines the lexicographic presentation of dictionary entries. Thus, for Hausa and Kanuri, the digraph 'sh' is located after the letter 's'. So, in the Hausa dictionary, the word "sha" (drink) is located after the word "suya" (fried), and, in Kanuri, the word "suwuttu" (undo) precedes the name "shadda" (basin).

These subtle differences can hardly be processed by software and require that digraphs appear as a proper sign in the Unicode repertoire. Some used by other languages are already there, sometimes under their different letter cases: 'DZ' (U+01F1), 'Dz' (U+01F2), 'dz' (U+01F3) are used in Slovak; 'NJ' (U+01CA), 'Nj' (U+01BC), 'nj' (U+01CC) in Croatian and for transcribing the letter " Њ " of the Serbian Cyrillic alphabet, etc.

It would be necessary to complete the Unicode standard with digraphs of Hausa and Kanuri alphabets in their various letter cases.

fy	Fy	FY
gw	Gw	GW
gy	Gy	GY
ky	Ky	KY
kw	Kw	KW
ƙy	Ƙy	ƘY
ƙw	Ƙw	ƘW
sh	Sh	SH
ts	Ts	TS

Table 2: Hausa and Kanuri digraphs missing in Unicode.

5.3. Characters with diacritics

Some characters with diacritics are included in Unicode as a unique sign, others can only be obtained by composition.

Thus, vowels with tilde 'a', 'i', 'o' and 'u' can be found in Unicode in their lowercase and uppercase forms while the 'e' with a tilde is missing and must be composed with the character 'e' or 'E' followed by the tilde accent (U+303), which can cause renderings different from other letters with tilde when viewing or printing (tilde at a different height for example).

Letter ǰ with caron exists in Unicode as a sign ǰ (U+1F0), but its capitalized form Ɉ must be composed with the letter J and caron sign (U+30C).

The characters ẽ, Ẽ et Ɉ should be added to the Unicode standard.

5.4. Letter case change

Word processors usually provide the letter case change function, but do not always realize it the correct way.

Thus, we found during our work that OpenOffice Writer software (3.2.1 version) fails in transforming 'ɾ' to 'Ɽ' from lowercase to uppercase or vice versa (the character remains unchanged) while Notepad++ (5.8.6 version) fails in transforming ǰ in Ɉ.

6. Conversion of the format towards XML

Figure 1 shows an excerpt of the Zarma-French dictionary in the original .odt format. All the following examples are based on this dictionary.

The Open Document Format has the great advantage of being based on XML. Instead of a conversion, we will actually retrieve the contents of the XML document, then transform it to get what we want.

A document in ODF format is actually a zip archive containing multiple files including the text content in XML. This content is stored in the “content.xml” file in the archive. To retrieve this file, some clever manipulations must be followed. On MacOs, one has to create an empty folder and then copy the .odt file inside. Then, with a terminal, the “unzip” command must be launched to unzip the file. On Windows, the .odt file extension must be changed into .zip and then the. zip archive can be opened.

The file “content.xml” can now be extracted from the archive and then renamed and placed in another location. It becomes the base file on which we will continue our work. The next step consists in editing this file with a “raw” text editor.

One may first think that since the source file “content.xml” is already in XML, it may be enough to write an XSLT stylesheet to convert the file into an XML dictionary, but the XML used in the source file is completely different from the XML targeted. Indeed, the source file comes from a word processor. It is designed for styling a document and not for structuring a dictionary entry. Therefore, it is finally easier to convert the XML file “by hands” with regular expressions than to write an XSLT stylesheet for automatically converting the source file.

7. Explicit tagging of the information

```
<text:span text:style-name="Phonetic_20_form">
<text:span text:style-name="T7">[àbìyànsôo]</text:
span></text:span>
```

Figure 2: Part of an entry (prononciation) in XML ODF format

This step consists in tagging explicitly all pieces of information. Each piece of information is usually distinguished from others in the original file with a different style. Figure 2 shows a part of the "abiyanso" entry (airport) in the Zarma-French dictionary. The style used to indicate the pronunciation is "Phonetic_form".

After locating the pieces of information, one must choose a set of tags to mark them.

This raises the question of the choice of the language used for tags. The choice of English as the international language of research may be privileged. But in our case, English is not a language present in our dictionaries and furthermore, it is not mastered by all linguists colleagues working on the project. The use of French solves this problem since all partners master the language. However, in the case of under-resourced languages computerization projects, we believe that it is important to encourage partners to use the words of their language to define the name of the tags. This may possibly give rise to the creation of new terms that did not exist in these languages. From a political perspective, it helps to move away from a post-colonial vision of the social status of African languages and brings new value to these languages.

The set of tag now defined, the next step is to replace the ODF markup by this new “homemade” tagset.

Simply perform search/replace operations for each type of information. For the example, the following regular expression (perl syntax) removes the tag "T7":

```
s/<text:span text:style-name="T7">([^<]+)
<\/text:span>//g
```

The second expression replaces the tag "Phonetic_form" with "ciiyaŋ":

```
s/<text:span text:style-name="Phonetic_20_form">
([^<+)<\/text:span>/<ciiyaŋ>$1<\/ciiyaŋ>/g
```

```
<sanniize>abiyanso</sanniize><ciiyaŋ>[àbìyànsôo]
</ciiyaŋ><kanandi>m.</kanandi><bareyaŋ>aéroport
</bareyaŋ><feeriji>batama kaŋ ra abiyey ga
zumbu</feeriji><silmaŋ>Tilbeeri nda Dooso sinda
abiyanso kaŋ ra abiyo beeri ga zumbu</silmaŋ>
<f>abiyansa</f><b>abiyansey</b>
```

Figure 4: Entry converted with « homemade » tags

Replacing all tags leads to the result in Figure 4.

8. Correction of the data

At this stage, several corrections are performed on the data.

8.1. XML Validation

In order to use XML tools, our file must be well formed. The manipulations of the previous step almost always introduce XML syntax errors. FireFox includes an XML parser and is also able to indicate exactly where the errors are located in the file.

Once the error is located, one has to check if it is not repeated elsewhere in the file. If this is the case, a regular expression must be written to correct the error in a systematic way instead of doing it by hand. In our case, the following regular expression can solve the problem:

```
s/<sanniize([^<+)<\/sanniize>/<sanniize>$1<\/sanniize>/g
```

The XML file is now well formed. It is then possible to manipulate it with XML tools.

abarba [ábàrbà] *m.* type de banane **banaana dumi no kaŋ i ga haagu ga ŋwa** *Abarba gani ŋwaayaŋ ga hin ga te boro se gunde-kuubi* budde abarbaa abarbey
abirillu [àbìrílìù] *m.* avril **annasaara handu taacanta kaŋ go marsu nda me game ra** *Abirillu, 15, 1974 no Sayni Kunce na hino sambu* abirillo abirilley
abiyanso [àbìyànsôo] *m.* aéroport **batama kaŋ ra abiyey ga zumbu** *Tilbeeri nda Dooso sinda abiyanso kaŋ ra abiyo beeri ga zumbu* abiyansa abiyansey
abiyo [àbíyò] *m.* avion **naarumay hari no kaŋ ra i ga boro nda jinay daŋ a ma deesi nd'ey** *Jidda no abiyey ga alfujaajey zumandi* beene-hi abiya abiyey
abunaadam [àbúnǎadàm] *m.* être humain, personne abunaadamo abunaadamey adamayze

Figure 3: Compact view in a browser

8.2. Verification of closed lists of values

The stage of verification of information taking their value in a closed list is important. Some errors come from bad handling in the previous steps, while others were present in the original file before conversion. For example, a dictionary uses parts-of-speech, a termbase uses a list of domains, etc. Make a copy of the file and keep only the values to check is a systematic approach for verification. In the example of Figure 4, the part-of-speech marked by "kanandi" can be extracted with the following expression:

```
s/^*<kanandi>([^<]+)<\/kanandi>*$/$1/
```

The resulting list must then be sorted alphabetically. TextWrangler and Notepad + + plugin with its TextFX have the necessary commands. If the editor does not offer this option, OpenOffice Calc spreadsheet can be used. This approach is then used to quickly detect irregularities. If a value appears only once, it is very likely that this is a mistake. In the dictionary used in the examples, we corrected "alteeb" to "alteeb.", "Dah." to "dab.", "m/tsif." to "m / tsif.", etc.

8.3. Simple corrections

A CSS style sheet can be set to view the data directly in a browser. A compact display with a different style for each type of information helps to detect structuring errors in an entry. In the example of Figure 3, we see immediately that definition (in bold) and example (in italics) are lacking for the entry "abunaadam".

With an XSL stylesheet, one can modify the data before display like adding a unique identifier for each entry, then,

for each reference define a hypertext link to the corresponding entry. When the linguist browses the file, s/he can click on the hyperlinks to verify that the references are also entries of the dictionary like the entry "abunaadam" in Figure 3 with a reference to the entry "adamayse".
It is essential to scrutinize the data to detect some errors, even if they can be fixed automatically thereafter with regular expressions. The data visualization step is also very important from a pedagogical point of view. It allows to show the benefits of XML encoding the data, in particular that several forms (style) can be associated with the same information (data). By learning the basics of CSS, the lexicographers can modify the style sheets themselves.

9. Structure of the entries

The entries can now be restructured. In files coming from word processors, the data structure is usually implied. We will have to add new structural elements to move towards a more standardized structure, allowing subsequent reuse. Concerning standards, LMF (Romary et al., 2004) became an ISO standard in November 2008 (Francopoulo et al., 2009). It suits ideally our goals. As it is a meta-model and not a format, we can apply the principle of the LMF model to our entry structure and keep our tags without using the LMF syntax. The core meta-model LMF is shown in Figure 5. The object "Lexical Entry" contains a "Form" and one or more "Sense" objects.

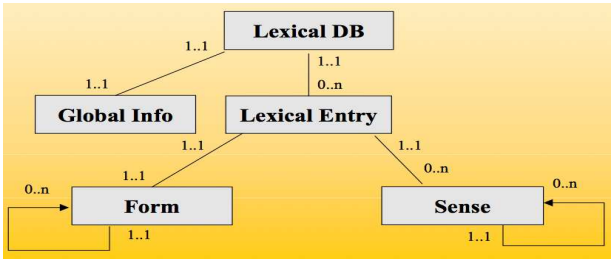


Figure 5: LMF kernel meta-model

Our lexical entries must now follow this meta-model. Figures 6 and 7 show an example of an entry before and after the addition of structuring tags. The "article" tag corresponds to the "Lexical Entry" object; the “bloc-vedette” tag correspond to the "Form" object and the "bloc-semantique" tag is the "Sense" object.

```
<sanniize>abiyanso</sanniize>
<ciiyaŋ>[àbìyànsôo]</ciiyaŋ>
<kanandi>m.</kanandi>
<bareyaŋ>aéroport<bareyaŋ>
<feeriji>batama kaŋ ra abiyey ga zumbu.</feeriji>
<silmaŋ>Tilbeeri nda Dooso sinda abiyanso kaŋ ra abiyo beeri ga zumbu</silmaŋ>
<f>abiyansa</f><b>abiyansey</b>
```

Figure 6: Zarma entry before structuring

```
<article>
 <bloc-vedette>
  <sanniize>abiyanso</sanniize>
  <ciiyaŋ>[àbìyànsôo]</ciiyaŋ>
 </bloc-vedette>
 <bloc-grammatical>
  <kanandi>m.</kanandi>
  <f>abiyansa</f><b>abiyansey</b>
  <bloc-sémantique>
   <bareyaŋ>aéroport<bareyaŋ>
   <feeriji>batama kaŋ ra abiyey ga zumbu.</feeriji>
   <silmaŋ>Tilbeeri nda Dooso sinda abiyanso kaŋ ra abiyo beeri ga zumbu</silmaŋ>
  </bloc-sémantique>
 </bloc-grammatical>
</article>
```

Figure 7: Entry after structuring following LMF model

```
<LexicalEntry id="abiyanso">
 <Lemma>
  <feat att="writtenForm" val="abiyanso"/>
  <feat att="phoneticForm" val="àbìyànsôo"/>
 </Lemma>
 <feat att="partOfSpeech" val="m."/>
 <Sense id="1">
  <Equivalent>
   <feat att="language" val="fra"/>
   <feat att="writtenForm" val="aéroport"/>
  </Equivalent>
  <Definition>
      <feat att="writtenForm" val="batama kaŋ ra abiyey ga zumbu"/>
  </Definition>
  <Context>
   <TextRepresentation>
    <feat att="language" val="dje"/>
      <feat att="writtenForm" val="Tilbeeri nda Dooso sinda abiyanso kaŋ ra abiyo beeri ga zumbu."/>
   </TextRepresentation>
  </Context>
 </Sense>
</LexicalEntry>
```

Figure 8: Zarma entry in LMF syntax

A simple XSLT stylesheet is then provided for download with each dictionary.

```
<xsl:template match="article">
   <LexicalEntry id="{sanniize}{sanniize/@lamba}">
     <xsl:apply-templates />
   </LexicalEntry>
</xsl:template>
<xsl:template match="bloc-vedette">
   <Lemma>
     <xsl:apply-templates />
   </Lemma>
</xsl:template>
<xsl:template match="sanniize">
      <feat att="writtenForm" val="{.}"/>
</xsl:template>
<xsl:template match="ciiya">
   <feat att="phoneticForm" val="{.}"/>
</xsl:template>
```

Figure 9: excerpt of the Zarma XSL stylesheet for producing LMF syntax

It converts each dictionary into the LMF syntax (see Figure 8). For more detailed information about this part, refer to (Enguehard & Mangeot, 2013).
The next step planned is to convert the resources into the Lemon format[2] and integrate them into dbnary[3] (Sérasset, 2014), the team database for linked data.

10. Web access via the Jibiki platform

10.1. Presentation of the platform

Jibiki (Mangeot et al., 2003; Mangeot et al., 2006; Mangeot, 2006) is a generic platform for handling online lexical resources with users and groups management. It was originally developed for the Papillon Project. The platform is programmed entirely in Java based on a the "Enhydra" environment. All data is stored in XML format in a Postgres database. This website mainly offers two services: a unified interface for simultaneous access to many heterogeneous resources (monolingual or bilingual dictionaries, multilingual databases, etc.) and a specific editing interface for contributing directly to the dictionaries available on the platform.
Several lexical resources construction projects used or still use this platform successfully. This is the case for the GDEF project (Chalvin et al., 2006) building an Estonian-French bilingual dictionary[4], the LexALP project about multilingual terminology on the Alpine Convention or more recently MotÀMot project on southeast Asias' languages[5]. The source code for this platform is freely available for download from the forge of the LIG laboratory[6].
An instance of the platform has been adapted specifically to DiLAF project[1] because, in addition to dictionaries, specific project information must be accessible to visitors:
- presentation of the project and partners;
- general methodology form converting published dictionaries to LMF format;
- stylesheets for different tools or tasks to be performed: tutorial on regular expressions, methodology of converting a document that uses fonts not conform to the Unicode standard to a document conforming to the Unicode standard, list of software used (exclusively open-source), methodology to monitor the project;
- presentation of each dictionary: original authors, principles that governed the construction of the dictionary, language, alphabet, structure of the lexical entries, etc.
- dictionaries in LMF format.

It is also envisaged to localize the platform for each language of the project.

10.2. Lookup interfaces

Three different interfaces are available to the user:
- the generic lookup allows the user to lookup a word or a prefix of a word in all the dictionaries available on the platform. The language of the word must be specified.
- the volume lookup allows the user to lookup a word or prefix on a specific volume. On the left part of the result window, the volume headwords are displayed, sorted in alphabetical order. An infinite scroll allows the user to browse the entire volume. On the right part of the window, the entries previously selected on the left part are displayed.
- the advanced lookup is available for complex multi-criteria queries. For example, it is possible to lookup an entry with a specific part-of-speech, and created by a specific author. On the left part of the result window, the headwords of the matching entries are displayed, sorted in alphabetical order. An infinite scroll allows the user to browse all the matching entries. On the right part, the entries previously selected on the left part are displayed.

10.3. Editing process

The editor (Mangeot et al., 2004) is based on an HTML interface model instantiated with the lexical entry to be published. The model is generated automatically from an XML schema describing the entry structure. It can then be modified to improve the rendering on the screen. Therefore, it is possible to edit any type of dictionary entry provided that it is encoded in XML.
The editing process can be adapted for specific needs through levels and status. A quality level (eg: from 1 to 5 stars, an entry with 1 star is a draft and one with 5 stars is certified by a linguist) can be assigned to each contribution. Similarly, a competence level can be assigned to each contributor (1 star is a beginner and 5 stars is a certified linguist). Then, when a 3 stars level user edits a 2 stars entry, the entry level raises to 3 stars.
Status can also be assigned to entries and roles to users. For example, in order to produce a high quality dictionary, an entry must follow 3 steps: creation by a registered user, revision by a reviewer and validation by a validator.

2 http://lemon-model.net/
3 http://dbnary.forge.imag.fr/
4 http://estfra.ee
5 http://jibiki.univ-savoie.fr/motamot/
6 http://jibiki.ligforge.imag.fr

10.4. Remote access via a REST API

Once dictionaries are uploaded into the Jibiki server, they can be accessed via a REST API. Lookup commands are available for querying indexed information: headword, pronunciation, part-of-speech, domain, example, idiom, translation, etc. The API can also be used for editing entries. The user must be previously registered in the website.

11. Conclusion

We presented a methodology for dictionaries conversion from word processing files to XML format. The DiLAF project does not stop in so good way. Before distributing dictionaries, there are still manual correction steps and possibly data addition. For example, examples of the Zarma-French dictionary will be translated into French. Once dictionaries are converted, we can then extend their coverage through a system of contribution / editing / validation that can be done online live on the Jibiki platform. The low Internet access in Africa will require us to develop alternative methods. We can then use the data as raw material to increase the computerization of these languages: morphological analysers, spell-checkers, machine translation systems, etc.

12. Acknowledgements

The DiLAF project is funded by the Fonds Francophone des Inforoutes of the International Organization of the Francophonie. We also thank all the linguists of the team without whom this project would not have been possible: Sumana Kane, Issouf Modi, Michel, Radji, Rakia, Mamadou Lamine Sanogo.

13. References

Berment V. (2004). Méthodes pour informatiser des langues et des groupes de langues « peu dotées ». Thèse de nouveau doctorat, spécialité informatique, Université Joseph Fourier Grenoble I, Grenoble, France.

Buseman A., Buseman K., Jordan D. & Coward D. (2000). The linguist's shoebox: tutorial and user's guide: integrated data management and analysis for the field linguist, volume viii. Waxhaw, North Carolina: SIL International.

Chalvin, A. & Mangeot, M. (2006) Méthodes et outils pour la lexicographie bilingue en ligne : le cas du Grand Dictionnaire Estonien-Français. Actes d'EURALEX 2006, Turin, Italie, 6-9 septembre 2006, 6 p.

Eluerd R. (2000). La Lexicologie. Paris : PUF, Que sais-je ?

Enguehard C. (2009). Les langues d'Afrique de l'ouest : de l'imprimante au traitement automatique des langues. Sciences et Techniques du Langage, 6, 29–50.

Enguehard C. & Mangeot M. (2013) *LMF for a selection of African Languages*. Chapter 7, book "LMF: Lexical Markup Framework, theory and practice", Ed. Gil Francopoulo, Hermès science, Paris.

Francopoulo G., Bel N., George M., Calzolari N., Monachini M., Pet M. & Soria C. (2009). Multilingual resources for nlp in the lexical markup framework (LMF). Language Resources and Evaluation, 43, 57–70. 10.1007/s10579-008-9077-5.

Haralambous Y. (2004). Fontes et codages. O'Reilly France.

Mangeot M. (2001). Environnements centralisés et distribués pour lexicographes et lexicologues en contexte multilingue. Thèse de nouveau doctorat, spécialité informatique, Université Joseph Fourier Grenoble I, Grenoble, France.

Mangeot, M., Sérasset, G. and Lafourcade, M. (2003) Construction collaborative de données lexicales multilingues, le projet Papillon. (Papillon, a project for collaborative building of multilingual lexical resources). special issue of the journal TAL (Electronic dictionaries: for humans, machines or both?, edited by M. Zock and J. Carroll), Vol. 44:2/2003, pp. 151-176. 2003.

Mangeot, M. et Thevenin, D. (2004). Online Generic Editing of Heterogeneous Dictionary Entries in Papillon Project. Proc. of COLING 2004, ISSCO, Genève, Switzerland, 23-27 August, vol 2/2, pp 1029-1035.

Mangeot M. & Chalvin A. (2006). Dictionary building with the Jibiki platform: the GDEF case. In LREC 2006, p. 1666–1669, Genova, Italy.

Mathieu Mangeot (2006) Dictionary Building with the Jibiki Platform. Software Demonstration, Proc. EURALEX, Torino, Italy, 6-9 September 2006, 5 p.

Matoré G. (1973). La Méthode en lexicologie. Paris, France : Didier.

Mortureux, Marie-F. (1997). La lexicologie entre langue et discours. Paris, SEDES.

République du Niger (1999a). Alphabet haoussa, arrêté 212-99.

République du Niger (1999b). Alphabet kanouri, arrêté 213-99.

République du Niger (1999c). Alphabet tamajaq, arrêté 214-99.

République du Niger (1999d). Alphabet zarma, arrêté 215-99.

Romary L., Salmon-Alt S. & Francopoulo G. (2004). Standards going concrete: from LMF to Morphalou. In Proceedings of the COLING Workshop on Enhancing and Using Electronic Dictionaries, ElectricDict '04, p. 22–28, Stroudsburg, PA, USA: Association for Computational Linguistics.

Sérasset, G. (2014) Dbnary: Wiktionary as a Lemon Based RDF Multilingual Lexical Resource. Semantic Web Journal - Special issue on Multilingual Linked Open Data, 2014. (to appear).

Streiter O., Scannell K. & Stuflesser M. (2006). Implementing NLP projects for non-central languages: Instructions for funding bodies, strategies for developers. In Machine Translation, volume 20.

Wilkins, David A. (1972). Linguistics in Language Teaching. Cambridge, MA: MIT Press.

Collaborative construction of a good quality, broad coverage and copyright free Japanese-French dictionary

Mathieu Mangeot-Nagata

Laboratoire LIG, équipe GETALP	Department of Digital Media
Bâtiment IMAG CS 40700	Hosei University, 3-7-2 Kajino Cho,
38058 GRENOBLE CEDEX 9, FRANCE	Koganei City, Tokyo, JAPAN 184-8584

mathieu.mangeot@imag.fr

Abstract

Although French and Japanese are regarded as well-resourced languages concerning tools and linguistic resources, the French-Japanese couple is considered an under-resourced language pair regarding its availability on the Web. Indeed, there are few bilingual electronic lexical resources of quality and which are both royalty and copyright free. French-Japanese bilingual aligned corpora and machine translation systems are logically equally rare.

Fortunately, there are printed French-Japanese dictionaries of good quality and which are sufficiently old to be royalty-free. It should be possible to reuse these resources as part of our project to build a good quality and broad coverage dictionary available on the Web. In order to update this data whose vocabulary might be old, we could reuse existing electronic resources such as Wikipedia or Japanese-English electronic resources. The resulting resource could be then available on the Web for lookup and correction by voluntary contributors. This methodology could be applied to other language couples in a similar situation with good printed dictionaries but few electronic resources.

We first conduct an inventory of Japanese bilingual dictionaries (printed or electronic) with their historical evolution. Then, we describe the resource we want to build. The next part concerns the conversion of three resources: the Cesselin Japanese-French printed dictionary, the language links between Japanese, French and English Wikipedia pages and the JMdict Japanese-English electronic dictionary. The Cesselin dictionary has been scanned, OCRized and parsed to detect headwords and entries. Then several error correction were performed on French and Japanese. New entries were created from Wikipedia links and finally, missing JMdict dictionary entries missing in the result resource were converted and added. Finally, we released the resource on a Web site built around the Jibiki platform allowing articles to be viewed and edited online. A French-Japanese bilingual corpus and an active reading moduel are also available. The resulting resources (dictionaries and corpora) are available for download on the project website. The data is released under public domain.

1. Introduction

This research project is located in the field of natural language processing (NLP), at the intersection of computer science and linguistics, specifically multilingual lexicography and lexicology.

Concerning the Web, although French and Japanese are two well-resourced languages (Berment, 2004), this is not the case of the French-Japanese couple:

- Electronic French-Japanese bilingual dictionaries (denshi jishô) cannot be copied to a computer or reused;

- There is a French-Japanese dictionary on the Web[1], but it only contains 40 000 entries, no examples and is not available for download.

There are collaborative Web dictionaries, such as the Japanese-English JMdict project led by Jim Breen (2004) that contains over 173,000 entries. These resources are freely downloadable. It is therefore possible to carry out such projects.

During a first stay in Japan from November 2001 to March 2004, we had already noticed the lack of French-Japanese bilingual resources on the Web. This gave rise to the Papillon project about the construction of a multilingual lexical database with a pivot structure (Sérasset et al., 2001). Since then, progress has been made in several areas (technical, theoretical, social) (Mangeot, 2006) but the actual production of data has made very little progress. On the other hand, there is a new trend in reusing existing lexical resources (word sense disambiguation, using open source resources (Wiktionary, dbpedia), merging with ontologies, etc.). Although they allow the coverage of existing resources to be consolidated and expanded, these experiments still use data created by hand by professional lexicographers. There are printed French-Japanese dictionaries of good quality and which are sufficiently old to be royalty-free. It should be possible to reuse these resources as part of our project to build a good quality and broad coverage dictionary available on the Web.

Based on this observation, we defined the following project to build a rich multilingual lexical system with priority for French and Japanese languages. The construction will be done first by reusing existing resources (printed Japanese-French dictionaries, Japanese-other language dictionaries, Wikipedia) and automatic operations (scanning and corrections, calculating translation links) and then by volunteer contributors working as a community on the Web. They will have to contribute to dictionary articles according to their level of expertise and knowledge in the field of lexicography or bilingual translation.

The resulting resources will be royalty-free and intended for use both by humans via conventional bilingual dictionaries and by machines for automatic language processing tools (analysis, machine translation, etc.).

First, we will conduct an inventory of French-Japanese bilingual dictionaries, then describe the resource we want to build. The following sections concern the conversion of three resources: the Cesselin printed dictionary, the language links between Wikipedia pages and the JMdict electronic dictionary. Finally, we conclude with the release of the resource on a Web site built around the Jibiki platform allowing articles to be viewed and edited online.

2. State of the art of Japanese bilingual dictionaries

Although French and Japanese are regarded as well-resourced languages concerning tools and linguistic resources, the French-Japanese couple is considered an under-resourced language pair (Berment, 2004). Indeed, there are few bilingual electronic lexical resources of quality and which are both royalty and copyright free. French-Japanese bilingual aligned corpora and machine translation systems are logically equally rare.

For historical as well as practical reasons, Japanese people are quick to put the emphasis on English. The English-Japanese couple is one of the best equipped at present with very substantial resources such as the EDR dictionary (1993) and machine translation systems among the best performers.

2.1 French-Japanese Printed Dictionaries

In this section, we present the dictionaries that are the most significant ones, either for historical reasons or in the innovations they bring. It should be noted that until recently there were two distinct lexicographical traditions depending on whether the writing team was of French or Japanese mother tongue and also that the dictionaries were unidirectional (one language to another but not vice versa). This was the case until 2009, with the release of the Assimil bidirectional dictionary (Hisamatsu et al., 2009). Moreover, until the 1950s, French mother tongue authors were all Catholic missionaries. The primary objective was to translate the Bible into Japanese.

2.1.1. Japanese→French dictionaries

1603: Vocabvlario da Lingoa de Iapam (Nippo jisho). This Japanese →Portuguese dictionary contains 32,293 articles written by the Portuguese Jesuit missionaries and is considered Japan's first bilingual dictionary.

1862: Translation of the Nippo jisho into French by Léon Pagès (1814-1886). He takes care of adding a katakana transcription of Japanese words (Griolet, 2008).

1904: Lemaréchal dictionary written by Jean Lemaréchal (1842-1912) containing around 60,000 articles on 1,008 pages. He abandons the Latin transcription adapted to French for the more widespread Hepburn romaji (Griolet, 2008).

1939: Cesselin dictionary (Cesselin, 1939) written by Gustave Cesselin (1873-1944), containing 82,500 articles on 2,340 pages. It is considered "the best from the perspective of those who study the Japanese language in depth, as it provides many examples presented in alphabetical form." (Griolet, 2008).

2009: Assimil Japanese dictionary. This dictionary (Hisamatsu et al., 2009) is to our knowledge the first two-way bilingual dictionary (French→Japanese and Japanese→French). It contains 24,000 articles on 1,280 pages. It also contains 135,000 words, phrases and translations, 35,000 usage examples. All words and phrases are transcribed into romaji. This makes it a very useful tool for French speakers learning Japanese.

2.1.2. French→Japanese Dictionaries

1864: futsugo meiyo dictionary (elucidation of the French language) by Hidetoshi Murakami (1811-1890). This scholar is considered the first Japanese to have learned French, through a French-Dutch dictionary (Koichi, 2010).

1866: French-English-Japanese dictionary by Father Eugene Mermet de Cachon (1828-1871) containing about 5,300 articles on 433 pages. The Japanese was reviewed by Léon Pagès.

1887: “dictionnaire universel français-japonais” (French-Japanese universal dictionary) by Nakae Katsusuke and Nomura Yasuaki. This dictionary is a Japanese translation of the French monolingual dictionary Petit Littré. It also marks the first integration of multiple word senses.

1905: French-Japanese dictionary by Emile Raguet (1854-1929) and Tota Ono, 1,048 pages.

1953: second edition of the previous French-Japanese dictionary (Raguet & Martin, 1953) called "Raguet-Martin", revised and expanded by Jean Marie Martin (1886-1975). This dictionary contains about 50,000 entries on 1,445 pages. It "remains the only major Japanese dictionary to submit translations in alphabetical form, and therefore intelligible without knowing ideograms." (Griolet, 2008).

1983: “dictionnaire franco-japonais de notre époque” (Franco-Japanese dictionary of our time) published by Mikasa-shobo. Contains about 42,000 articles on 1,763 pages. The romaji is indicated for the Japanese as well as the pronunciation of French words.

1988: Shôgakukan-Robert dictionary. This dictionary, the result of the Japanese translation of the monolingual French Robert dictionary is a considerable work. It remains to this day the largest French-Japanese dictionary printed since it contains more than 100,000 articles. Unfortunately, there is no romaji (Latin transcription) or furigana (kanji pronunciation). It is intended primarily for Japanese-speaking users.

2.1.3. Electronic Dictionaries (denshi-jishō)

The Crown French → Japanese dictionary (Sanseido, 1978) contains 47,000 articles. There is neither romaji nor furigana (see Figure 1).

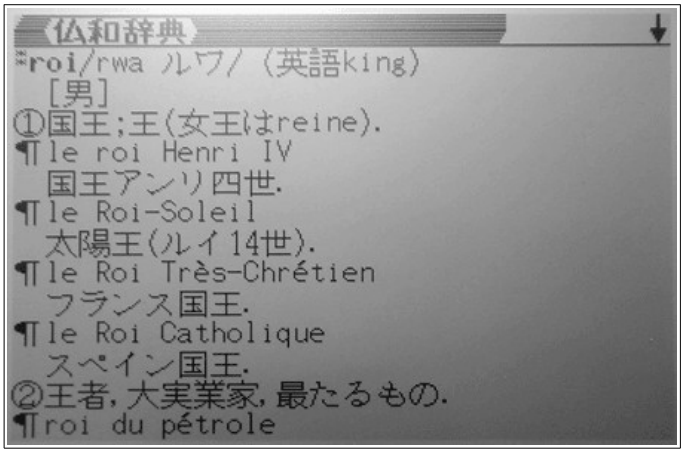


Figure 1 : **Screenshot of the Crown dictionary in electronic version**

The Japanese → French Concise dictionary (Sanseido) contains 38,000 articles. There is neither romaji nor furigana (see Figure 2).

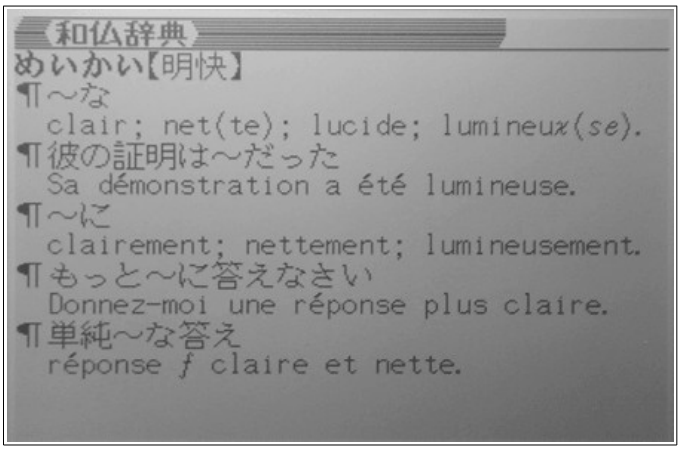


Figure 2 : **Screenshot of the Concise dictionary in electronic version**

Francophones with a level sufficient to read Japanese certainly need a wider coverage dictionary. These dictionaries are thus designed for Japanese-speaking users.

2.1.4. Conclusion

Electronic dictionaries are not reusable outside the medium in which they are sold. In addition, they are designed for Japanese-speaking users and their coverage is not very wide.

The only existing French-Japanese dictionaries of good quality and with broad coverage are publishing dictionaries that exist only in paper format, for which there is no online lookup interface. However, some are old enough to be copyright free.

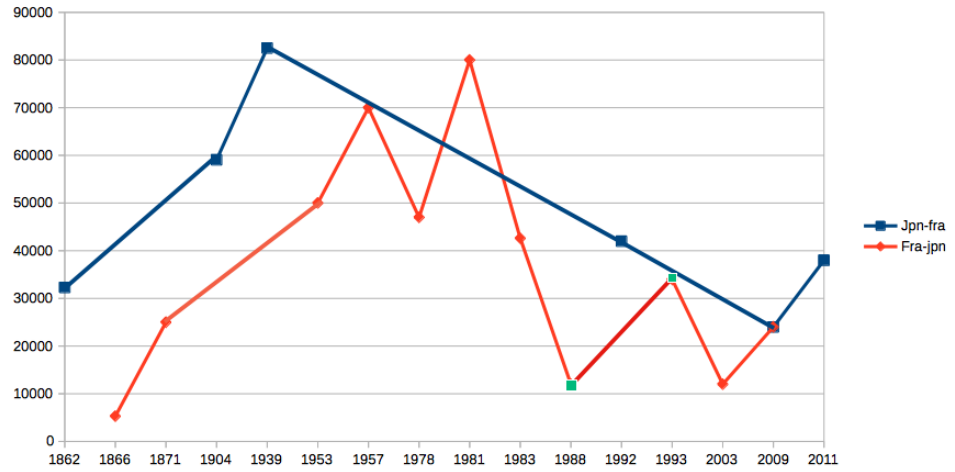


Figure 3 : N**umber of articles in French-Japanese dictionaries**

Figure 3 shows the changing number of articles in the French-Japanese dictionaries based on years. We note a peak in the 1950s. It must be possible to reuse some of these resources in our project to build a dictionary of good quality and broad coverage available on the web, provided that it is updated with modern vocabulary.

2.2 Wiktionary Projects

The French Wiktionary currently has 2.2 million articles, including 1.2 million French articles and slightly fewer than 7,000 Japanese translations, of which about 1/2 are proper nouns (often a simple transcript in Japanese syllabary of the headword). There are also a few translations that come from the dictionnaire-japonais.com website (see 2.4.2). Translations are indicated at the entry level and not at the sense level. There is no translation of context description (gloss, examples, etc.), or information on the Japanese translation (part-of-speech, etc.).

The Japanese Wiktionary has 83,000 articles with 26,000 Japanese articles and 2,800 French articles translated into Japanese. There are also inflected forms or oral conjugated forms, e.g. 32 articles for the verb "aimer" (to like/love). The coverage is very insufficient.

Wiktionary projects are interesting and fashionable but they have several limitations:

The structure of the articles is free. It is not possible to use the same precise microstructure for all articles.

Although it is possible to describe, in a language A Wiktionary, a word sense of a language B in language A, the initial interface is not designed for writing bilingual dictionaries. For example, the description of the reverse language link A → B must be done by hand in language B Wiktionary.

It is not possible to automatically add existing data from other sources in order to build a draft to be refined later.

The contributions are anonymous. It is not possible to use a quality level for data or a review / validation system.

Although the success of the Wikipedia project might logically lead us to think that it would be the same for the collaborative construction of quality bilingual or multilingual dictionaries, this is not the case. We quote in this regard, Larry Sender, co-founder of Wikipedia:

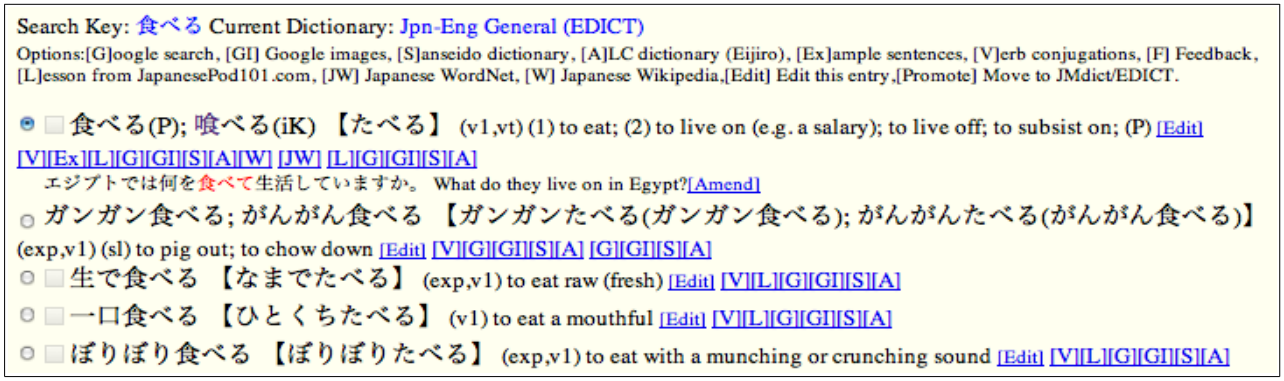
"To try to develop a dictionary by collaboration among random Internet users, particularly in a completely uncontrolled wiki format, now strikes me as a nonstarter."

Indeed, every Wikipedia article can be written by a specialist in the field in question, but for a general language dictionary, it is not possible to find a specialist for only a few articles. Only linguists who are specialists in the language and professional translators (after being trained in lexicography) can write an entire article.

2.3 Online Japanese–other language Resources

2.3.1. Japanese → English Dictionary: JMdict

The JMdict[2] (Japanese-Multilingual Dictionary) (Breen, 2004) is a project led by Jim Breen. It contains 173,000 Japanese entries translated into English with additional translations in other languages: German (from WaDokuJiten), 31,000 French equivalents from the dico FJ project (see 2.4.1), Russian, Dutch, etc.


Search Key: 食べる Current Dictionary: Jpn-Eng General (EDICT)
Options:[G]oogle search, [GI] Google images, [S]anseido dictionary, [A]LC dictionary (Eijiro), [Ex]ample sentences, [V]erb conjugations, [F] Feedback, [L]esson from JapanesePod101.com, [JW] Japanese WordNet, [W] Japanese Wikipedia,[Edit] Edit this entry,[Promote] Move to JMdict/EDICT.

食べる(P); 喰べる(iK) 【たべる】 (v1,vt) (1) to eat; (2) to live on (e.g. a salary); to live off; to subsist on; (P) [Edit]
[V][Ex][L][G][GI][S][A][W] [JW] [L][G][GI][S][A]
エジプトでは何を食べて生活していますか。 What do they live on in Egypt?[Amend]
ガンガン食べる; がんがん食べる 【ガンガンたべる(ガンガン食べる); がんがんたべる(がんがん食べる)】 (exp,v1) (sl) to pig out; to chow down [Edit] [V][G][GI][S][A] [G][GI][S][A]
生で食べる 【なまでたべる】 (exp,v1) to eat raw (fresh) [Edit] [V][L][G][GI][S][A]
一口食べる 【ひとくちたべる】 (v1) to eat a mouthful [Edit] [V][L][G][GI][S][A]
ぼりぼり食べる 【ぼりぼりたべる】 (exp,v1) to eat with a munching or crunching sound [Edit] [V][L][G][GI][S][A]

Figure 4 : **"食べる" (taberu) article from the Jmdict dictionary**

Advantages: broad coverage resource, royalty-free and available for download. It is also regularly revised and updated.

Disadvantages: unidirectional Japanese → other language dictionary. There is no English → Japanese reverse dictionary. The microstructure is limited: the translation contexts are not described. It also lacks a definition and examples.

2.3.2. Japanese → German Dictionary: WaDokuJiten

The WaDokuJiTen[3] from Ulrich Apel (Apel, 2002) consists of more than 280,000 entries. Its wide coverage as well as its microstructure are more developed than those of the Jmdict.

Nr.	Japanisch	Lesung	Deutsch	Worttyp	
1	食べる	たべる	[1] essen; speisen; zu sich nehmen; fressen; probieren. [2] leben von.	下一他	
2	食べるのを遠慮する	たべるのをえんりょする	nicht essen.	サ変自	
3	食べるとしゃきしゃきする	たべるとしゃきしゃきする	beim essen knusprig sein.	サ変自	

Figure 5 : **“食べる” (taberu) article from the WaDokuJiTen dictionary**

Advantages: more complete than the JMdict in terms of coverage and information, free of charge and available for download.

Disadvantages: like the JMdict, the dictionary is unidirectional. It also does not contain usage examples to illustrate the translation contexts.

This dictionary is to date the most comprehensive Japanese-other language resource available for free. It constitutes a goal to reach for our resource in terms of coverage.

2.4 French-Japanese Resources Available Online

2.4.1. Dico FJ Project

The dico FJ dictionary project, a pioneering one in the field of Japanese-French resources on the Web, was launched in early 2000 by Jean-Marc Desperrier (Desperrier, 2002). It contains just over 10,000 entries that come mainly from translation of the Japanese-English dictionary JMdict by Jim Breen. There has not been any change since 2003.

Advantages: royalty-free and available for download.

Disadvantages: more disadvantages than the Jmdict. Translation errors arose because some contributors with a poor level of Japanese translated the English translations directly into French instead of translating the Japanese headword, increasing the number of misinterpretations.

2.4.2. Dictionnaire-japonais.com

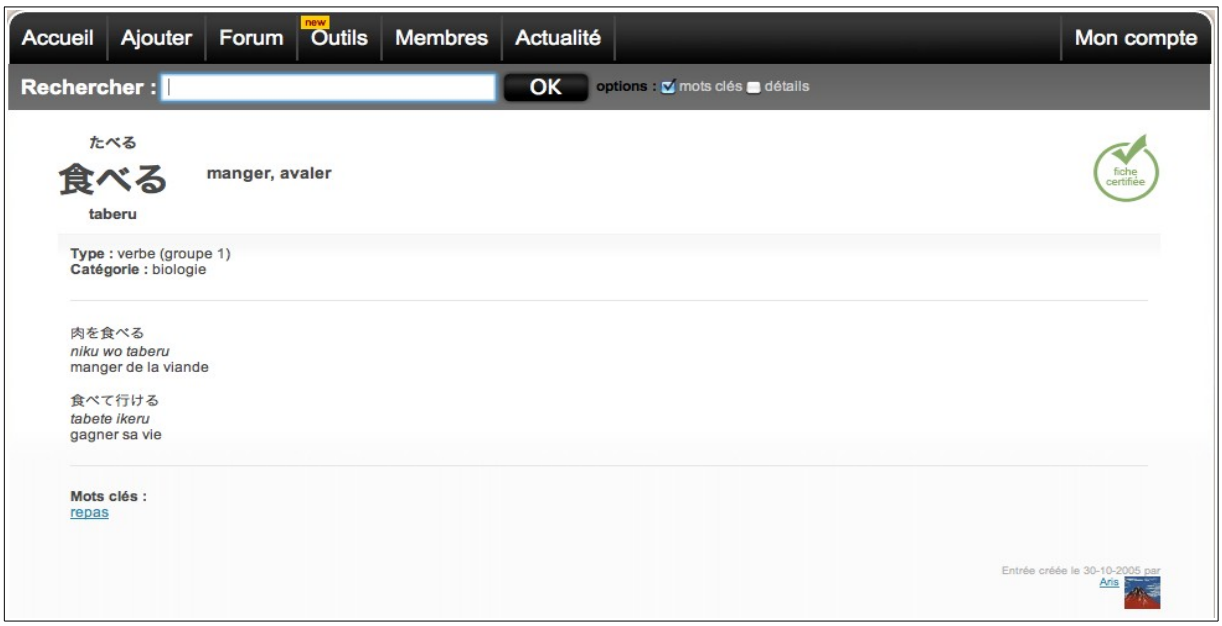


Figure 6: **食べる (taberu) Article from dictionnaire-japonais.com**

The dictionnaire-japonais.com1 project currently contains just over 40,000 words. This represents clear progress compared to other Japanese-French online dictionary projects. Each user can contribute directly by adding entries. The community of contributors appears to be quite active, as evidenced by the activity on the project forum. The information available for each entry is relatively limited, to a "grammatical type," a "category" (field), a language level, and sometimes an "origin of the word" (etymology).

Advantages: available online, the coverage is wider than the dico FJ, there is an active community of volunteer contributors.

Disadvantages: added to the disadvantages of the dico FJ dictionary, the data is not freely available for download.

2.5 Conclusion

The French-Japanese dictionaries available online do not have a broad coverage and they are all oriented from Japanese to French.

Most French-Japanese dictionaries also lack information in order to be used by both French-speaking and Japanese-speaking users. For example, to our knowledge, there is no dictionary with both kanji (ideograms), kana (syllabic) and romaji (transcription in the Roman alphabet).

Furigana is used to display the pronunciation in kana (Japanese syllabary) above the words with kanjis. See Figure 13 for an example. Dictionaries for Japanese-speaking users do not indicate the romaji and most of the time, the examples are written with kanji without furigana. Beginner learners of Japanese cannot read them. Dictionaries for French speaking users do not indicate the pronunciation of French words or the gender (masculine/feminine), which are essential to Japanese-speaking readers. They also lack some important information such as that concerning counters (we do not count objects in the same way: one car = ichi dai, one dog = ippiki, etc.) or language levels.

In conclusion, for personal use, it is possible to find printed dictionaries (or their electronic version) of reasonably good quality if one knows how to read kanji; but when one looks for a free dictionary or a resource that is reusable in other tools, there is no choice but to use an English-Japanese dictionary, which, as we know, can only increase the misunderstandings and translation errors.

However, the JMdict and WadokuJiten projects show that it is possible to carry out collaborative dictionary construction projects online.

At this point, we can redefine our project in this way: to retrieve and convert quality copyright-free French-Japanese printed dictionaries thanks to an optical recognition process and to make them available online for users who will be able to correct the remaining errors and update the data.

3. Description of the resource to build

3.1 History of the project

In 2001, already faced with the same problem of a lack of French-Japanese bilingual lexical resources, we launched the Papillon Project (Mangeot et al., 2004), which allowed us to move forward on theoretical aspects with the definition of a pivot macrostructure.

In 2003, the launch of the GDEF project (Mangeot & Chalvin 2006) of an Estonian-French bilingual dictionary was an opportunity to make progress with the software part via Jibiki, a generic online platform for lexical resource management.

In 2010 the MotÀMot project (Mangeot, 2014) of a French-Khmer dictionary was the opportunity to work on defining quality levels.

In 2012, the DiLAF project (Enguehard & Mangeot, 2014) led us to define a precise methodology for data recovery from standard text processors (Word).

3.2 Microstructure of the Articles

3.2.1. General Microstructure

Generally speaking, our articles will be based on a combination of a lexeme and a part-of-speech. The articles will therefore not have any grammatical block. The articles' structuring will follow that defined by the Lexical Markup Framework (LMF) standard (Francopoulo et al., 2009): each article contains a form block which includes information related to the form: headword, pronunciation, part-of-speech, and a semantic block with a list of sense blocks. Each sense block describes a word meaning. It also contains the translation in another language as well as a list of examples. Each example is translated into the other language.

The data come primarily from the conversion of existing resources. At first, the structure of articles will therefore follow that of the original resource. Then, in the medium term, our goal is to strive for a richer microstructure based on the Explanatory and Combinatorial Lexicography (Mel'čuk et al., 1995), part of the Meaning-Text Theory (MTT). For each word sense (formally a lexie), add a semantic formula that can be seen as a formal definition. In the case of a predicative lexie, the formula describes the predicate and its arguments and also the schema that describes the syntactic realization of arguments. Then, add a list of lexico-semantic functions. There are 56 basic functions applicable to any language that can be combined with each other. Finally, add a list of examples and possibly idioms.

We plan to follow the structure of the lexical networks currently developed by Alain Polguère and his colleagues at ATILF laboratory in Nancy. For French, a new resource, called “Réseau Lexical du français” is under construction (Polguère, 2013). For Japanese, during Papillon project (Mangeot & Kuroda, 2003), 100 entries were created by hands by Mutsuko Tomokiyo, Japanese linguist, in order to verify the validity of the theory. The most difficult part of the conversion from the existing microstructure to a richer one based on the Explanatory and Combinatorial Lexicography (ECL) is the handling of the translation links. In the ECL, each word sense is an entry by itself, called a lexie. In a multilingual dictionary, translation links must be represented by links between two lexies, but in traditional bilingual dictionaries, they are usually represented by the word alone. Therefore, one has to decide to which word sense the link refers to. A first step will be to create automatically a new lexie for each target of translation link. Then, contributors will be able to merge lexies in the target language (Sérasset & Mangeot, 2001).

3.2.2. Japanese Articles

The articles of most Japanese dictionaries are based on the lexeme. There are no homograph articles. We will follow this division.

Regarding the headword, each kanji has several possible pronunciations. Therefore, it is necessary to indicate the pronunciation. This is usually done by using the hiragana syllabary. If we simply add kanji and hiragana, beginners in Japanese cannot easily read the articles. We must also use a Latin transcription of the Japanese called romaji. There are several methods

for official romaji: the oldest and most widely used is the Hepburn method, introduced by the American missionary James Hepburn in 1887. The Kunrei is a method introduced by the Japanese Ministry of Education and described as the ISO standard number 3602:1989. It is based on Japanese phonology. Its main advantage is that it illustrates better the grammar, while Hepburn's biggest problem is that it changes radical verbs, which does not reflect the underlying morphology of Japanese. However, non-native speakers prefer the Hepburn method because it gives a better indication of English pronunciation. In our dictionary, the goal is to help beginner learners of Japanese who are non-native speakers, not to indicate to native speakers how to write romaji. We therefore choose the Hepburn method.

It is unfortunately not possible to automatically generate the romaji only from the hiragana pronunciation. Indeed, the " う " (u) letter is transcribed in two different ways depending on whether it extends a vowel "o" or not. The combination of the two vowels "o" and "u" can be written in two different ways in romaji depending on whether the "u" is the beginning of a morpheme (kanji) or not. For example, the word “東京” (Tokyo) is written in hiragana "とうきょう" and romaji "Tōkyō" (not "toukyou"), while the word "子牛" (calf) is written in hiragana " こ う し "and romaji "koushi" (not "kōshi"). We therefore choose to represent each headword with three parts:

- a first part "Japanese-headword" indicates the Japanese word as it appears in the texts. This can be a kanji word only (for nouns), a combination of kanji and hiragana (for verbs and adjectives in particular), a word in hiragana only (adverbs), a word in katakana (word of foreign origin) or any combination of the three writing systems (kanji, hiragana and katakana);
- a second part "hiragana-headword" always indicates the pronunciation of the word in hiragana (even if the Japanese-headword is already a word in hiragana);
- a third part "romaji-headword" indicates the transcription of the Japanese-headword in modern Hepburn romaji. This part can itself contain two versions: one for display that can contain spaces, dashes or dots and one for lookup that contains only letters.

In the rest of the article, for each Japanese text segment, we add the pronunciation in hiragana above the text (called furigana) and a transcription in Hepburn romaji.

3.2.3. French Articles

Traditional lexicography distinguishes two vocables as homographs if there is no clear semantic link between them (Polguère, 2008). But in practice, it frequently happens that dictionaries with the same source language follow a different division between homograph vocables. We believe that this distinction is arbitrary. Furthermore, to our knowledge, there are no objective criteria for assessing automatically a semantic link between two words with automatic processing tools. We therefore choose not to distinguish homograph vocables if they belong to the same grammatical category. The combination of a lexeme and a part-of-speech therefore constitutes a single article.

For the non-French-speaking users of the dictionary, pronunciation of the headword will be added. The gender (masculine / feminine) of each noun representing a French translation in a Japanese article will be added.

3.3 Quality Levels

Each article is assigned a level of quality. The levels range from 1 star for a draft (converted data whose quality is unknown) to 5 stars, for an article certified by an expert (e.g., translation link validated by a professional translator).

Likewise, contributors will be assigned a skill level (1-5 stars as well). 1 star being the level of an unknown novice in the community and 5 stars being the level of a recognized expert.

Then, for example, when a level 3 contributor revises a level 2 article, the article automatically moves up to level 3. Similarly, if the work of a contributor is systematically validated without corrections by other senior contributors, s/he can automatically switch to the next level after a certain threshold (e.g., 10 contributions).

To go further, we plan to analyze the work of the contributors. If a person contributes massively on, say, a specific domain, the system will automatically send regular contribution proposals in this domain.

4. Conversion of the Cesselin Dictionary

4.1 Presentation of the Dictionary

haichi (配置) n.m. Placement, arrangement, f. disposition, mise en ordre, répartition ...**suru** (--する) v.t: Arranger, placer, répartir, disposer. **Endō ni junsa wo—suru** (沿道に巡者を--する) Poster des agents de police le long de la route. **Teitai ni—suru** (梯隊に--する) Disposer en échelons.

Figure 7: **Scan of « 配置 » (haichi) article of Cesselin dictionary**

The "Cesselin" (Cesselin, 1944) is a French → Japanese dictionary developed by Gustave Cesselin, an apostolic missionary who died in 1944 having spent his entire career in Japan. The dictionary contains 2,365 pages and over 82,600 articles. The headword is noted in romaji and Japanese (kanji or kana). It is followed by a part-of-speech in French and a list of French translations. Next comes a list of phrases containing the headword in romaji and Japanese (kana and kanji). Each phrase is translated into French. The article concludes with a list of examples, each noted in romaji, Japanese and translated into French (see Figure 7).

4.2 Copyright Negotiation

To ensure that a book is copyright free, one must verify that all the authors have been dead for a fixed term. In the case of a dictionary written by a significant number of authors, such verification may be difficult. Furthermore, the duration varies from one country to another. In Japan, it is currently 50 years. In France, it is 70 years, a duration shared with many countries.

Gustave Cesselin is the only official author of his dictionary. He died in 1944. Thus, in Japan as in France, the "Cesselin" dictionary is copyright free.

If the targeted dictionary is not copyright free, another solution is to negotiate directly with the copyright holders. In the case of the missionaries, who were the most numerous authors until the 1950s, the holders are the congregations.

In the case of the "Raguet Martin" French-Japanese dictionary (Raguet & Martin, 1953), Jean-Marie Martin died in 1975. We must therefore wait 10 years in Japan and 30 years in France for the dictionary to be finally copyright-free. So we contacted the “Missions Étrangères de Paris”, the congregation holding the copyright of this dictionary. An agreement was concluded for the use of the "Raguet Martin" data on our website.

4.3 Dictionary Scanning

Several scanning techniques were tested:

- manual scanning with a flat scanner. The resulting image has a black band in the middle of the two pages that hides some of the characters at the beginning of each line, so potentially headwords. The result is not usable. Moreover, the operation is tedious because it requires lifting of the dictionary and turning of each page by hand.
- scanner with camera on top. The procedure is quick (3 hours) because it is not necessary to move the book. Unfortunately, a small bend remains in the middle of the two pages.
- scanner with one camera above and another at the side. The side camera is used to calculate the thickness of the book and then automatically straighten the curvature in the middle of the two pages. We could not test this type of machine.
- book cutting and automatic scan. This technique gives the best scan quality since there is no curvature or black area. Moreover, it is very fast because automatic. However, it requires a copy to be sacrificed because the binding is cut in order to scan the book page by page.

4.4 Optical Character Recognition

Once the dictionary scanning has been achieved in the best possible conditions, we must find optical reader software that is able to:

- recognize multiple languages simultaneously;
- allow the training of the optical recognition;
- recognize the kanji.

We tested a dozen types of software. Abbyy was the only one to comply with most of our requirements. Its major drawback is that it does not allow the training of the optical recognition for ideograms. We then performed two passes. The first pass was performed by choosing only French as the recognition language and training the optical recognition on the first page. The French text parts were correctly recognized with very few errors. Text portions in Japanese (kanji or kana) were not recognized at all. The second treatment was performed by choosing both French and Japanese as recognition languages, which forbade us to train the recognition (because Japanese uses ideograms). The French text parts were recognized with a higher error rate than during the first pass and the Japanese text parts were recognized with a standard error rate. It takes about 12 hours for a pass throughout the whole dictionary.

The results of the process are then exported into OpenXML (.docx) or OpenDocumentFormat (odt). These documents are actually zip archives. They are then decompressed and the XML files containing the text of each page of the dictionary are extracted. The result files of the two passes are then merged using an algorithm for calculating string distance such as Levenshtein.

4.5 Headword Detection

The most important part of the conversion of a dictionary is located in the detection of headwords. They delimit the entries and are also access keys. This part consists of three phases:

1. Headword extraction from the header of each page. This operation is very important because these headwords will be used to delimit the alphabetical order of the headwords in the page. In the .docx or .odt archives, the pages' headers are separated from the main text. A tool was written to automatically extract these headers. Then, the headers were verified (is it romaji and are they sorted in alphabetical order?) and corrected if necessary. In the example of Figure 8 for page 323 of the Cesselin, the headers' headwords are 'haichai' and 'haigeki'.

2. Comparison of each beginning of a line in strict alphabetical order done first with the headers: header 1 < Word < header 2 and then between the detected headwords: headword 1 < headword 2 < headword 3. In Figure 8, the following headwords are extracted: haichai, haichi, haichi, haichüritsu, haidan

3. Approximate comparison of each beginning of a line. To take into account OCR errors, a second comparison is performed. This time, an accepted percentage of errors is introduced by using the Levenshtein string distance algorithm. In Figure 8, the headwords « iichi » and « haicbutsu » are extracted and automatically corrected to become « haichi » and « haichutsu ».

A problem of over-detection remains. Indeed, some phrases or examples included in an article start at the beginning of a line and reuse the article headword. For example, on page 1,038 of the Cesselin, "kuwadate iru" was detected as a headword but it is an example of the article "kuwadateru", and "Kuwashiku monoshiberu" was detected as a headword but it is a phrase of the article "kuwashii".

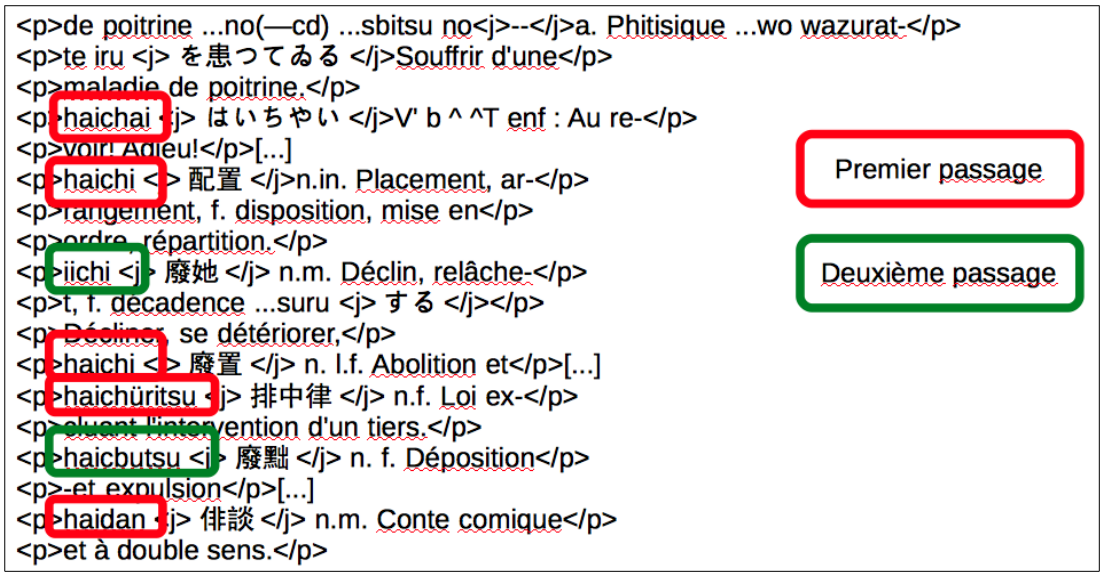


Figure 8: **Headword detection for page 323 of the Cesselin dictionary**

4.6 Post-OCR Corrections

After the optical recognition, it is possible to automatically correct text characters depending on the language in the text.

4.6.1. French

Corrections to the French mainly focus on the use of diacritics. Examples are: Â + ' ' ⇒ À; Etre ⇒ Être; ç[^àaou] ⇒ c; etc.

4.6.2. Romaji

The romaji uses only macron as a diacritical mark on the vowels: ā,ī,ū,ē,ō. Other diacritic letters are automatically converted: à ⇒ a; äâ ⇒ ā, etc.

Some character strings do not exist in romaji. They are also converted: lt + [aiueo] ⇒ h + [aiueo]; rn + [aiueo] ⇒ m +[aiueo], etc.

4.6.3. Japanese

Errors in the Japanese text segments appear mainly at the beginning or end of a segment. When there is a change of language between French and Japanese, the software detects this change too late. Therefore, the first and last characters of a Japanese segment are sometimes replaced by ASCII characters. Some patterns are often repeated. Below, there are three examples of replacement. The <j> opening tag indicates the start and the </ j> end tag the end of the Japanese segment:

9 ⇒ り; ｜c ⇒ に; = ⇒ニ

v') ⇒ い　Ex : <j>と忙はし</j>v') ⇒ <j>と忙はしい</j>)

't) ⇒ す　Ex : <j>心を越</j>'t)⇒ <j>心を越す</j>)

4.6.4. Priority Lists for Corrections

In order to prioritize corrective work for the contributors when the resource is online, priority lists were calculated from word frequency lists in a Japanese monolingual corpus. The frequency list of words used (JapFreqList_5109_Novels) comes from a corpus of 5,109 Japanese novels of modern Japanese literature. It was built for the cbJisho[4] project and can be downloaded by following this link[5]. It contains 188,218 entries. The highest frequency is the Japanese comma "," with 26,244,137 occurrences.

To prioritize the remaining work on the headwords with undetected kanji, a new frequency list based on the hiragana was generated from the JapFreqList list. The JMdict dictionary is first queried via the REST[6] Application Programming Interface (API) of the Jibiki platform16. Then, the frequencies obtained for each hiragana are summed and the resulting list is sorted. The last step is to compare this list with the one of undetected kanji headwords in the Cesselin.

4.7 Modernization of the Japanese

Since the dictionary was published, the Japanese language has undergone many changes, including in its writing. The goal of the project is not to reproduce the Cesselin as faithfully as possible, but to build a modern dictionary reflecting the current use of the language. This is why we decided to modernize the Japanese automatically where possible.

4.7.1. Romaji

The romaji used at the time of writing the Cesselin, based on Hepburn transcription, is called romajikwa. It uses the letters "kwa" and "gwa" to transcribe certain syllables that are now simplified into "ka" and "ga". Example: kwaikwan ⇒ kaikan. The letter "m" was also used to transcribe the hiragana letter “ん” before "m, b, p" consonants, because it is actually pronounced "m". Now, the letter "n" is used everywhere. Example: jimbōchō ⇒ jinbōchō; gumma ⇒ gunma.

The transcription of the "n" consonant is ambiguous in Japanese as it can be the final syllable "ん" or the beginning of a syllable (na, ni, nu, ne, no). It is customary to note in romaji when the n is a final syllable. In the Cesselin, a dash is added after "n". But the dash is also sometimes used to separate syllables. On the other hand, current romaji uses other characters. The modern Hepburn uses an apostrophe “'”. Francophones use the dot “.”. So we replaced the dash marking a "ん" with a dot. Example: ran-i ⇒ ran.i

4.7.2. Kanji Simplification

At the time of the writing of the dictionary, there was no encoding table for the kanjis. The first Japanese encoding standard, JIS (Japanese Industrial Standard) 208 appeared in 1978. It contains about 7,000 characters. This encoding was widely used during the computerization of Japanese. So, the kanjis not included in this encoding gradually disappeared. To remedy this situation, other coding standards appeared, such as JIS 212 in 1990 which specifies 6,067 characters, and JIS 213, which specifies 11,233 characters. But the damage was already done and most of the current Japanese words use only the JIS 208 kanjis. We have therefore replaced all the kanjis that were not included in the JIS 208 with their JIS 208 variant. In the same way, we have replaced the JIS 208 variants by those of the lowest grade, when available.

JIS 213 → JIS 208 replacements: 状⇒状,歩⇒歩,黄⇒黄,縁⇒縁,晩⇒晩,黒⇒黒

JIS 212 → JIS 208 replacements: 啞⇒唖,掻⇒掻,頬⇒頬,上⇒上,伙⇒火

JIS 208 variants replacements: 阪 8⇒坂 3,弐⇒一 1,埜 10⇒野 2,亰⇒京 2,區⇒区 3

Some ideograms were even not included in any JIS. They are part of the Han Chinese characters. Some are actually used in the original Cesselin dictionary but the others come from character recognition errors. For these two series, we had to find a JIS 208 equivalent by hand.

Examples of Han characters included in the Cesselin: 絕⇒絶, 說⇒説,靑⇒青.

Examples of Han characters coming from errors: 內⇒内,戶⇒戸,出⇒出.

4.7.3. Japanese

The Japanese language has also evolved since the 1950s, particularly the verb endings. The evolution was already indicated in the romaji (romajikwai → romaji), but the hiragana used for the verb endings in Japanese kept track of old pronunciations. Examples: the "ふ" (fu) ending is pronounced "u"; the “へる” (heru) ending is pronounced "eru". Therefore this situation created a mismatch between the romaji and hiragana. So we changed the hiragana to match the romaji and thus the modern pronunciation. E.g.: kokitsukau 扱 使 ふ ⇒ 扱 使 う; kikikaeru 切 替 へ る ⇒ 切 替 え る.

Some hiragana letters were also replaced systematically: ゐ⇒い (i);ゑ⇒え(e). Example: 光ってゐる ⇒ 光っている.

The sokuon, a letter indicating a geminate consonant noted with a macron in Hepburn romaji, is usually noted with a small tsu "っ". In the Cesselin dictionary, the sokuon is noted with a standard size tsu "つ". In the following verifications, all variants with and without small tsu are generated to avoid over-detection problems.

4.8 Information Tagging

Once all corrections have been made, it is time to move on to the markup of each information part . Initially, only segments containing Japanese (kana or kanji) are tagged. The result of the previous steps also tagged out the headword in romaji and Japanese. All other segments must be tagged.

We first listed all the abbreviations used in the dictionary, which allowed us to tag the etymology, part-of-speech, domain and language levels. We also included in the list frequent errors from the optical reading. Example: "n.in." instead of" n.m." in Figure 9.

Then, the Japanese segments are preceded with segments in romaji. The latter begin either with "..." or a capital letter. Japanese segments are also followed by French segments ending with a dot. These rules allowed us to tag the examples.

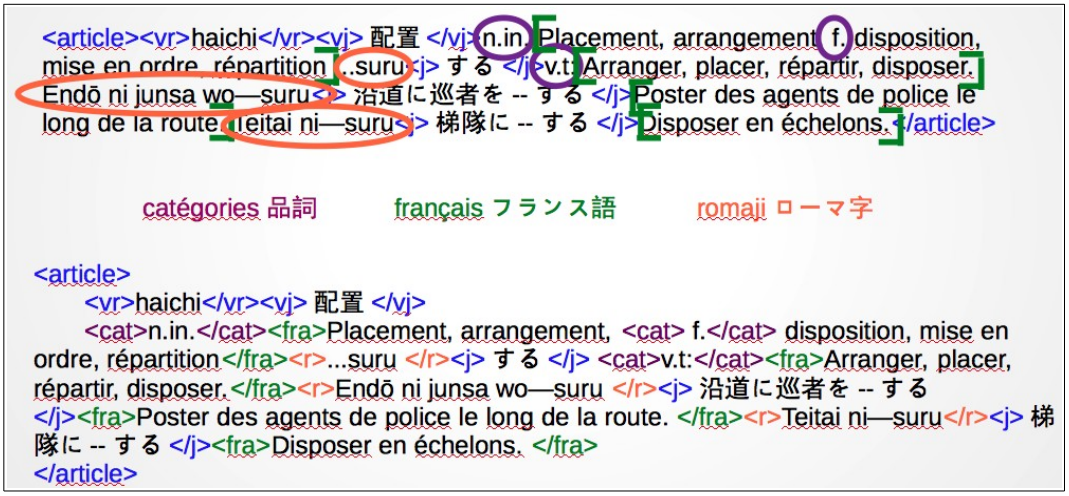


Figure 9: **Information tagging for « 配置 » (haichi) entry of the Cesselin dictionary**

Finally, the only remaining segment to tag was the French translation of the headword.

Note: because of optical recognition errors, it is complicated to use a language detection tool to differentiate between romaji and French.

4.9 Entry Structuring

During this step, the aim is to structure the articles according to the normative part of the LMF standard (Francopoulo et al., 2009). This will allow automatic export into the informative part of the standard (LMF syntax). The normative part specifies the structure of different blocks but gives no constraint on how to represent them (XML elements or attributes, name of the elements, etc.). It is therefore possible for the resource to comply with the LMF standard while keeping its own tags.

The informative part of the standard provides an example of LMF syntax, but we consider that it is not convenient to use (Enguehard & Mangeot, 2013). So we choose to use our own tags. The structuring consists primarily in gathering information about the word form into one "<forme>" block and each word sense into a "<sens>" block. Examples in the Cesselin are not attached to a particular word sense. Therefore, we did not separate the examples into different sense blocks.

<article>
<forme><vedette><vr>haichi</vr><vj> 配置 </vj></vedette><cat>n.in.</cat></forme>
<sens><fra>Placement, arrangement, <cat> f.</cat> disposition, mise en ordre, répartition</fra></sens>
<exemples>
<exemple>
<r>...suru </r><j> する </j><cat>v.t:</cat><fra>Arranger, placer, répartir, disposer.</fra>
</exemple>
<exemple>
<r>Endō ni junsa wo—suru </r><j> 沿道に巡者を -- する </j><fra>Poster des agents de police le long de la route.</fra>
</exemple>
<exemple>
<r>Teitai ni—suru</r><j> 梯隊に -- する </j><fra>Disposer en échelons.</fra>
</exemple>
</exemples>
</article>

Figure 10 : **Structuring of « 配置 » (haichi) article of the Cesselin dictionary**

4.10 Headword Verification

Once the dictionary is structured, at this stage, we need to detect as many potential errors as possible in order to mark them so they can be easily corrected later online by users of the dictionary. We also add additional information such as furigana for Japanese segments.

A first detection consists in certifying the presence of the headwords in other dictionaries. To this end, we have used the 'super daijirin' English-Japanese dictionary (Matsumura, 2006) included in MacOS to program a tool to automate the dictionary lookup. We have also downloaded and installed the JMdict for a better verification.

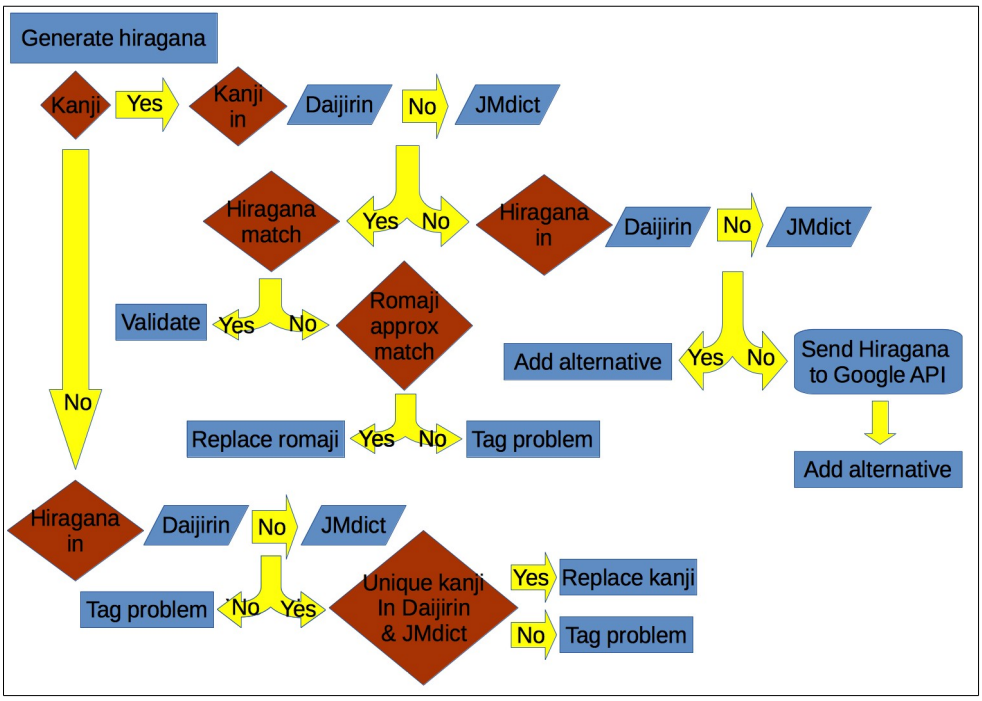


Figure 11: **Flow chart for the headword verification algorithm**

The algorithm is as follows:

- Generation of the hiragana from the headword in romaji and adding it in the block form of the article;

- Verification of the presence of the headword in kanji in the 'super daijirin'.

- if not present, verification in the JMdict.

- if the headword in kanji is attested in one of these two dictionaries, then check whether the hiragana corresponds.

- if not, convert the hiragana found in one of the verification dictionaries into romaji and then calculate the approximate distance between the two romaji using the Levenshtein algorithm.

- if the distance is small, replace the romaji and hiragana of the headword by those found in the verification dictionary

- if the distance is large, identify the problem for later correction.

- if the headword in kanji is not attested, use the hiragana to seek an alternative proposal in the verification dictionaries

- if an alternative proposal is found, then add it in the article as a possible alternative.

- if no alternative is found, use the Google transcription API to offer an alternative.

- if the headword in kanji is empty due to an OCR error, check the hiragana in the two verification dictionaries

- if there is only a single headword in kanji in the two verification dictionaries, then replace the empty headword in kanji by that found in the verification dictionary.

Optical reading errors are frequent with letters with macron in romaji. It is used to generate the hiragana. Therefore, for comparison with the hiragana with the verification dictionaries, all variants with and without long vowel are generated (ā/a, ī/i, ū/u, ē/e, o/ō). This avoids over-detection problems.

The page number in the original printed version of the Cesselin is added to each article. This will later allow a link to be displayed to the PDF file of the page scanned, allowing contributors to correct OCR errors by viewing the source file directly.

Finally, approximately one out of two headwords has been certified in another dictionary, and about 10% of headwords in kanji remain empty.

4.11 Error Detection for French

The detection of potential errors in French is carried out using the tree tagger[7] morphological analyzer. Each French sentence is sent to the parser. If an unknown word is detected, it is tagged for subsequent correction.

Example: "L'un des huit enfers glacés du boaddhisme.". "boaddhisme" was not recognized by the analyzer. In this case, it is an OCR error. The correct word is "bouddhisme".

This step could be refined further. Indeed, all the Latin words (eg: names of plants) were not recognized by the analyzer. On the other hand, it should be possible to automatically correct errors such as the one mentioned in the example.

4.12 Error detection for Japanese

The Japanese analyzers cannot detect spelling errors such as the ones found in French, as the Japanese language does not use separators and all kanji have a meaning. For the detection of potential errors, we compared the romaji transcription with the Japanese version. First we converted the romaji segment into hiragana and then we used the Mecab[8] Japanese morphological analyzer to generate the furigana of the kanji included in the corresponding Japanese text segment. We then compared the hiragana resulting from the conversion of romaji with the furigana coming from the analyzer. Comparing the hiragana is done by generating all the variants as shown in 4.10. When a difference is found, it is tagged for subsequent correction.

Example: 早いが重宝 [はやいがちょうほう] and « hayai ga jūhō » [はやいがじゅうほう] ;

Example: の徴 [のしるし] and « no kizashi » [のきざし].

Variants with and without sokuon “っ” are generated during the comparison (see 4.7.3).

At this stage, the hiragana was added to the Japanese examples thanks to the output of the analyzer.

This step could also be improved. Indeed, the two examples given above are not real errors but come from the fact that there may be several possible furigana for the same kanji. It should therefore be interesting to use the output of an analyzer that provides all possible solutions.

5. Conversion of Wikipedia Links

The coverage of the Cesselin dictionary is already substantial: more than 82,000 articles. However, its release date is 1939, before World War II that resulted in the occupation of Japan by the US military from 1945 to 1952. Since that time, many English words have been incorporated into Japanese after being transcribed into katakana. A modern Japanese dictionary cannot ignore them. We therefore reused two free resources available to complete the first set of data from the Cesselin: Wikipedia and JMdict.

We had originally planned to use Wikipedia, but we found that most Japanese translations in the French Wiktionary actually came from translation links between Wikipedia pages. Thus, we preferred to directly use the original resource. So we picked the links of the Japanese Wikipedia to the French and English Wikipedia pages.

5.1 Conversion Process

For each language, Wikipedia offers to download all data as a database export in SQL format[9]. We download the information on each page such as the ID and title of the page (jawiki-latest-page.sql.gz) and links to pages in other languages (jawiki-latest-langlinks.sql.gz). Next, we import that data into a MySQL database and then we create a new table called "translation" initially containing the identifier of each page, its title, the title of the linked pages for French and English ones and the language of the linked page (French or English).

5.2 Extraction of the hiragana

The title of Japanese wikipedia pages are in Japanese (kanji + kana). There is no kanji reading (furigana) or pronunciation (romaji). We must therefore find a way to obtain them. The use of a morphological analyzer is not appropriate here because there are many proper

names in the titles of pages and they are not all in the analyzer dictionary. However, in the first sentence of every Japanese wikipedia page, the hiragana reading of the title is indicated in parentheses.

With Wikipedia API[10], we automatically recover a sample of each page that contains the first sentence and we analyze it to extract the hiragana, that we include in a new field in the table "translation" created in the previous step.

5.3 Generation of the romaji

As explained in 3.2.2, the romaji cannot be automatically generated from the hiragana if it contains the pair of letters “おう” or “うう”. Furthermore, while Japanese is a language without separators, it is customary to add spaces between words and capitalization at the beginning of proper names in the romaji transcriptions, which greatly facilitates reading. If the romaji is generated from the hiragana (or kanji), there will be no spaces.

We observed that pages that are translations in other languages of Japanese proper nouns often indicate in the first sentence the Japanese spelling and the romaji of these nouns.

Example: Le parc quasi national d'Abashiri (網走国定公園, Abashiri Kokutei Kōen) est un parc quasi national situé sur la côte Nord-Est de l'île de Hokkaidō au Japon.

As with the extraction of hiragana, we use the Wikipedia API to extract the romaji translations of Japanese pages.

When no romaji can be found in the translation page, we will seek the readings of each kanji that composes the Japanese word in the Kanjidic dictionary[11]. Several types of readings are associated to one kanji: onyomi (Chinese origin), kunyomi (Japanese origin), okurigana (with additional hiragana for termination), nanomi (for proper names).

Compositional phenomena in Japanese must then be taken into account: rendaku (consonant harmonization, e.g. か → が); sokuon (geminate consonant: つ → っ); renjo (doubling of the 'n' character, e.g. ん あ → な); etc.

Then, the romaji generation is performed based on the hiragana conversion.

E.g.: 東京 + [とうきょう] 東 = とう; 京=きょう ; とう+きょう => tōkyō

E.g.: 子牛 + [こうし] 子= こ; 牛=うし ; こ+うし => koushi

5.4 Selection of the headwords to import

At this point, articles that will be imported into the Cesselin dictionary have to be selected. The aim is not to import all the articles in order to "make up the numbers," but only those whose headwords are attested in other resources. First, we will select the articles available with French translations. Articles with English translations will only be imported after the conversion of the JMdict dictionary. The selection algorithm is the following:

- Check that the page title (headword in Japanese) is in the Daijirin dictionary;

  - If so, check whether the page title is already in the Cesselin;

    - If so, add the link to Wikipedia articles in the Cesselin article,

    - If not, check whether the romaji is in the Cesselin;

      - If so, check whether all the Japanese headwords of the Cesselin articles were verified during step 4.10 ;

- If all the headwords have been verified then import the article;

- If the romaji is not in the Cesselin then import the article.

This algorithm does not guarantee a selection without problems. Indeed, because of potential errors in optical character recognition, it is possible that a headword may be imported when it is already in the Cesselin, if the kanji is empty and if the romaji has an error. In this case, once the romaji and kanji of the original Cesselin article are verified and corrected, a search for Japanese-headword + hiragana-headword duplicates will allow us to remove duplicate articles. Another possible problem is not importing an article because the romaji is already in the Cesselin but not the kanji. In this case, once all the Japanese-headwords associated to that romaji have been verified, it will be possible to re-import the missing article.

A total of 23,456 articles were generated from Wikipedia links and imported into the Cesselin. Of these, 20,825 articles are translated into French and 2,631 articles are translated into English. The latter articles were imported after the JMdict conversion (see following section).

6. Conversion of the JMdict dictionary

JMdict is a Japanese → English dictionary (see Figure 4). From the beginning, in order to increase the coverage of our dictionary, we decided to import JMdict entries, although most translations are in English and not in French. On the one hand, English being close to French and much studied, most Francophones can understand written English and secondly, due to the lack of French-Japanese lexical resources, many French speaking learners of Japanese use English-Japanese dictionaries anyway. However, despite having the technical possibility of crossing JMdict with a French-English dictionary, we decided to leave the English translation as is, to avoid misinterpretations. Contributors may themselves offer online translation into French when they look up these articles.

Each JMdict article contains for the headword, a list of words in kanji (k_ele) and a list of words in hiragana or katakana (r_ele). Each word in kana is sometimes followed by a restriction list to indicate to which word in kanji it corresponds. On the other hand, if, in Japanese text, the word is written in hiragana or katakana, the word list in kanji may be empty. Therefore, reading the headwords (kanji + hiragana reading) is not immediate and requires the correct correspondences to be computed. There is no transcription in romaji.

For every headword, we then clarified the information and generated a list of headwords consistent with the structure chosen for the Cesselin (see 3.2.2): if the k_ele list is empty, we copy the items from the r_ele list into it. If an r_ele element is written in katakana, it is copied in the k_ele list and then replaced by the hiragana in the r_ele list. Then each element of the k_ele list is a Japanese headword to which is linked an r_ele item in hiragana. If there are several readings for the same word in kanji, the Japanese headword is duplicated for each corresponding hiragana. Then, the romaji generation algorithm from section 5.3 is reused to generate the romaji from hiragana and kanji.

Next, the Japanese kanji headwords have been simplified as in Section 4.7.2 and the resulting duplicated headwords then removed.

Finally, the algorithm of Section 5.4 is reused to select articles to import. When importing articles in the Cesselin dictionary, the unique identifier of the article in the JMdict is stored for future reference.

A list of articles to be translated primarily in French is also generated from the JapFreqList_5109_Novels frequency list (see 4.6.4).

A total of 47,810 articles from the JMdict, of which 2,521 translated into French and 45,289 translated into English, were imported in the Cesselin.

7. Online release on the Jibiki platform

7.1 Site Description

The Project[12] web site is built around the Jibiki platform (Mangeot, 2006) for the management of heterogeneous lexical resources online. The platform is programmed using Enhydra, a java object server based on a 3/tiers architecture. The database used for the data layer is Postgres. This platform, developed and continuously improved since 2001, is freely available with an LGPL licence on the LIG laboratory forge[13].

In addition to the functions for resource management (dictionary lookup and editing), several pages were added:

- a blog as homepage;
- a bilingual aligned corpus lookup interface (KWIC);
- a module for active reading.
- a download page for retrieving all the data (dictionaries and corpora).

The site is available in three languages: French, English and Japanese.

7.1.1. Home page: blog

The home page consists of a standard consultation interface (see 7.1.4), a short text presenting the project, lists of articles to be corrected as a priority and a blog programmed using WordPress.

Two lists of articles are displayed: the articles of the Cesselin dictionary in which headword kanjis were not recognized during the optical reading (see 4.6.4) and the articles from the JMdict whose translation is in English and must be translated into French (see 6). For each list, the ten most frequent words are displayed. A button for calculating lists is provided, in order to update the lists if corrections have been made.

The blog allows the presentation of the latest news about the project and of the best contributor of the month. Each article is translated into the three languages of the site: French, English and Japanese.

7.1.2. Bilingual Aligned Corpora

The consultation module of the French-Japanese aligned bilingual corpora is programmed in perl using the IMS Open Corpus Workbench[14] platform and its lookup tool Corpus Query Processor (CQP). This allows the use of a regular expressions language to build complex queries. For example, the query "inter(ê|e)(t|ss)(é|e)(r|s)?" will obtain the words “intérêt”, “intérêts”, “intéressé”, “intéresser”, “intéressée”, “intéressées”. It is also possible to combine the levels of annotation. For example, the query [(lemma="sous.+") & (Cat="V. *")] will get all the conjugated forms of verbs beginning with the prefix "sous". This module is derived from a first version used in the GDEF project for an Estonian-French corpus[15].

The data come on the one hand from the OPUS project[16] for software (KDE, OpenOffice), the Koran and movie subtitles (OpenSubtitles) and on the other hand from corpora we built

ourselves with texts found on the Web (Le Monde Diplomatique, the Bible, a Franco-Japanese tax convention and the Universal Declaration of Human Rights).

These data, aligned at the paragraph level, were then tagged using Tree tagger for French (see 4.11) and Mecab for Japanese (see 4.12). The furigana was added to the Japanese text, allowing queries to be written at the kanji reading level. For example, a search for the word "はな" (hana) will obtain the words 花 (flower), 鼻 (nose), etc.

Figure 12 shows the search result for the Japanese word "配置" (haichi) in the “Le Monde Diplomatique” corpus. The corpus is parallel. Thus, it is also possible to search for French words or lemmas.

At present, the whole corpus size is approximately 6 million words including:

- Newspapers: Le Monde Diplomatique (288,745 words);
- Legal texts: Franco-Japanese Tax Agreement (17,443 words) and Universal Declaration of Human Rights (2,208 words);
- Software: KDE 4 (1,179,000 words) and OpenOffice 3 (569,903 words);
- Religious texts: the Bible (904,914 words) and the Koran (Tanzil) (192,905 words);
- Movie subtitles (OpenSubtitles) (4,714,000 words).

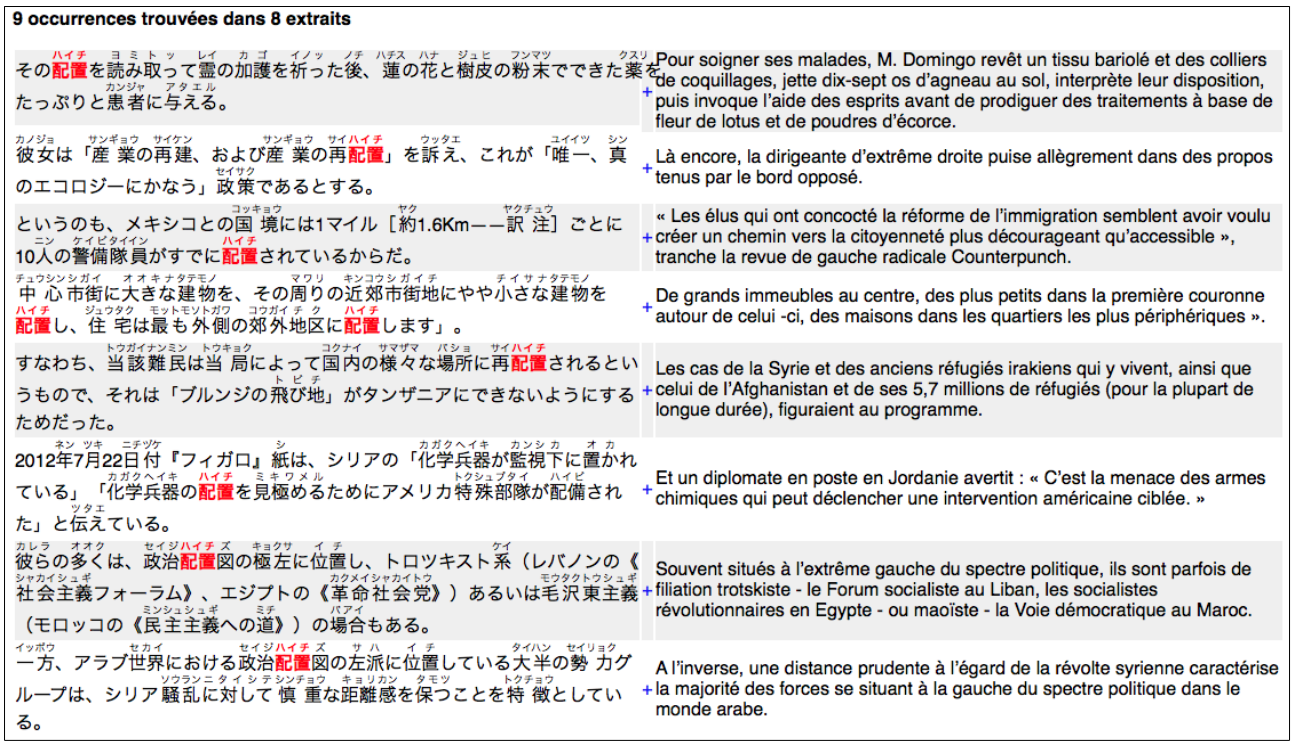


Figure 12: **Lookup result for the word « 配置 » (haichi) in the *Le Monde Diplomatique* corpus**

7.1.3. Active Reading Module

The active reading module offers a reading aid for a user who knows a language but does not master it. The user enters a text in that language and the module will display a word-for-word translation. In our case, the French speaking user can enter a Japanese text and the Japanese speaking user a French text. Then, the module adds the pronunciation of words or furigana in the case of Japanese and translations of each word. In order to avoid interfering

with the reading, the translations are displayed only when the user points at a word with the mouse. See Figure 13 for an example of the display of a Japanese text with the mouse pointed at the word "時代" (jidai).

The module first sends the text to a morphological analyzer: Tree tagger for French (see 4.11) and Mecab for Japanese (see 4.12), then retrieves the lemma for each word. It then uses the REST[6] API of the Jibiki platform[17] to look up different resources. For Japanese text, the furigana is obtained with the morphological analyzer and translations with the Cesselin dictionary. For French text, for the moment translations are available in English, until we obtain a French → Japanese dictionary. The pronunciation and translation are obtained using the FeM dictionary (Gut et al., 1996).

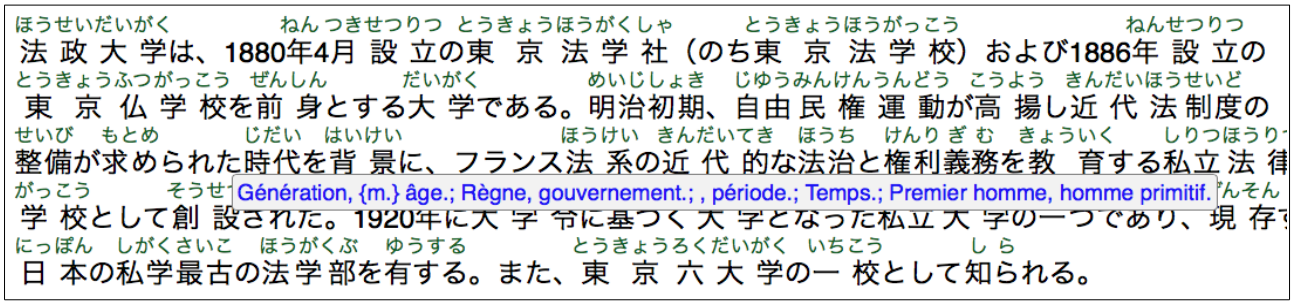


Figure 13: **Display of a Japanese text with the mouse on « 時代 » (jidai) word**

7.1.4. Standard Lookup

The standard lookup interface allows a user to look up Japanese words typed in romaji, kana or kanji. It displays an advanced view of a printed dictionary: the left side displays the headwords in the immediate vicinity of the targeted word, listed alphabetically. An infinite scroll can be used to browse all the dictionary headwords in alphabetical order. When a headword is clicked on on the left side, the full article is displayed on the right side.

A menu is displayed at the top right of each article. It includes links to the editing form, the modification history, the XML source of the article, the original scanned page of the dictionary in PDF format and the result of the headword lookup in the bilingual corpus.

Errors or anomalies detected previously are displayed with a special color background. Headwords not attested and the romaji that do not match their Japanese counterpart are in orange background like “ 廃 她 ” (haichi) in Figure 14. French errors are in yellow background (see Figure 15). English translations from the JMdict or Wikipedia non-translated into French are in green background.

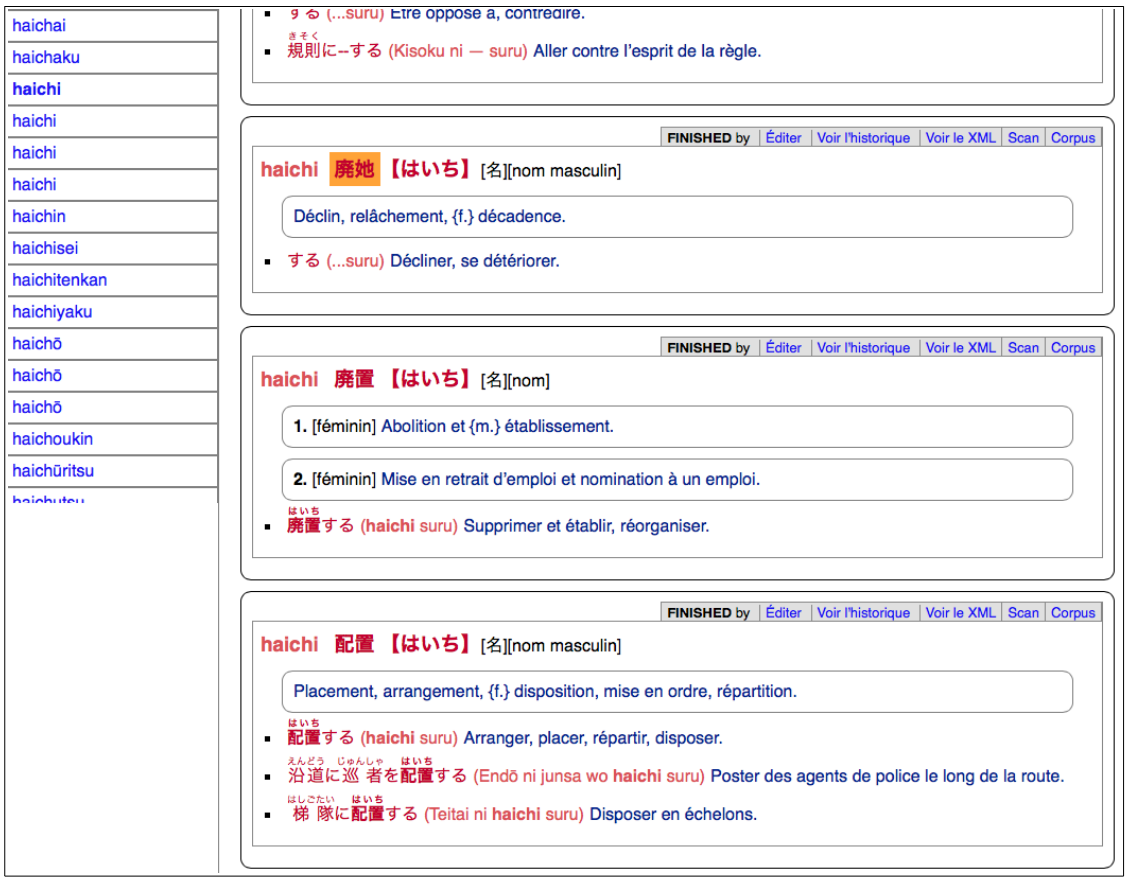


Figure 14: **Result of the lookup of « haichi » through the standard interface**

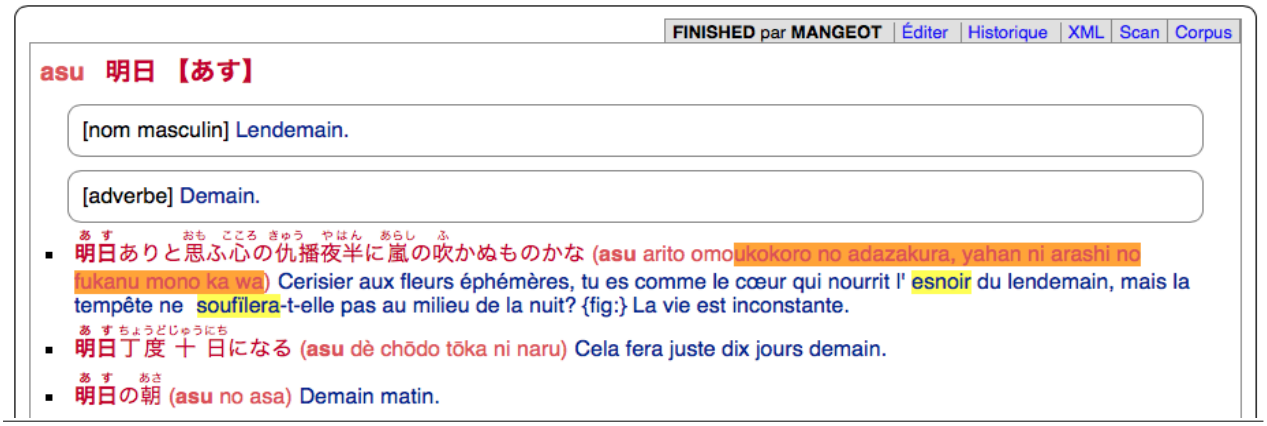


Figure 15: **« 明日 » (asu) article with errors in French (yellow background) & Japanese (orange)**

7.1.5. Advanced Lookup

The advanced consultation interface allows multi-criteria searches on all lexical resources installed on the platform. The criteria apply directly to the data (headword, pronunciation, part of speech, translations, examples, etc.) but also on metadata (author, status, article ID, contribution ID, etc.). These can be combined in a single search. It is possible for example to look for articles in which the romaji starts with a "b" and which pertain to the botany domain (see Figure 16). The search results are displayed in an alphabetical list on the left side of the window. If the number of results is greater than 100, an AJAX request is made to obtain from

the server the following results in alphabetical order, as with the standard consultation interface.

Figure 16: **Advanced Lookup Interface with combination of two search criteria**

7.2 Online Editing

To edit an article online, the user must be previously registered on the Jibiki platform.

Most of the editing work to be done is the correction of the post-OCR errors in both French and Japanese. These corrections can be achieved by any speaker of these two languages. Therefore, one does not need to be an expert lexicographer to be able to edit an entry. Thus, in order to encourage the contributions, we decided to let the edition open to anyone registered on the platform. Of course, we provide several mechanisms of quality control if we discover that someone introduces many errors in its contributions:

- New entries cannot be created. Contributors must edit existing entries.
- A version control mechanism such as the one of Wikipedia project is implemented on the platform. It is possible to cancel the modifications of one specific user.
- If too many problems are detected, a system of different levels for contributors (contributor, reviewer, validator) can be set up such as the one used in the GDEF project (Mangeot & Chalvin 2006).

After one year of project life, the dictionary counts 9,000 modifications. No contributor has been yet identified for introducing errors in the data.

7.2.1. Quick Editing

When viewing an article, it is possible to perform small modifications directly on the Web page. To do this, a double-click on the segment to modify turns it into a text field with an 'ok' button on the right to validate the edition (see Figure 17).

This editor is programmed using AJAX[18] technology and it uses the REST[6] API of the Jibiki platform16 to interact with the server. When clicking on the 'ok' button, the new string is sent to the server with the Xpath[19] of the edited segment and the article ID.

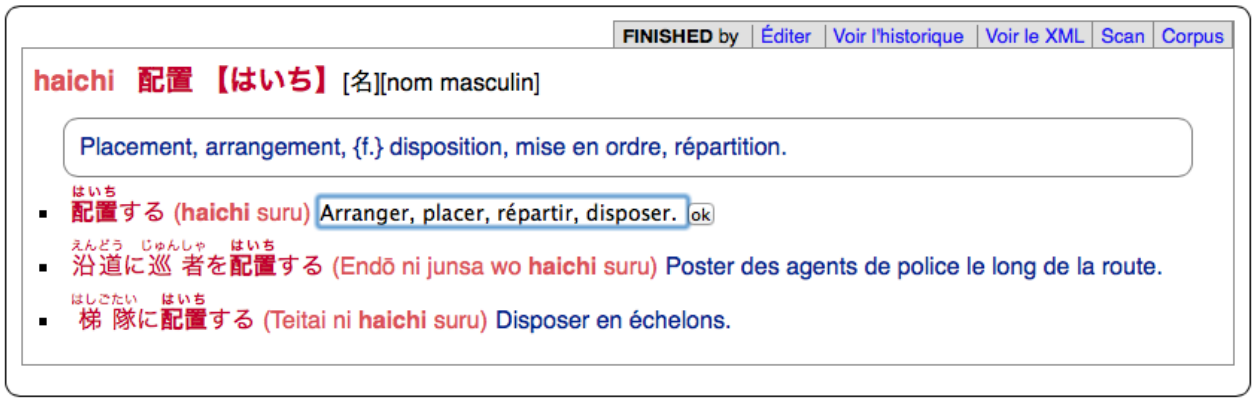


Figure 17: **Quick Editing of the French translation of an example of « 配置 » (haichi) article**

7.2.2. Full Editing Form

The full editing interface is automatically generated from the articles noted as XML schema[20] (Mangeot & Thevenin, 2004). It consists of a standard HTML form with interactors for different data types (text field for free text, drop down menu for closed lists of values, check boxes for booleans, radio buttons for choices of values, etc.), as well as more complex interactors to manage lists of objects (for example, adding or deleting one example in a list of examples by pressing the buttons "+" and "-", as in Figure 18).

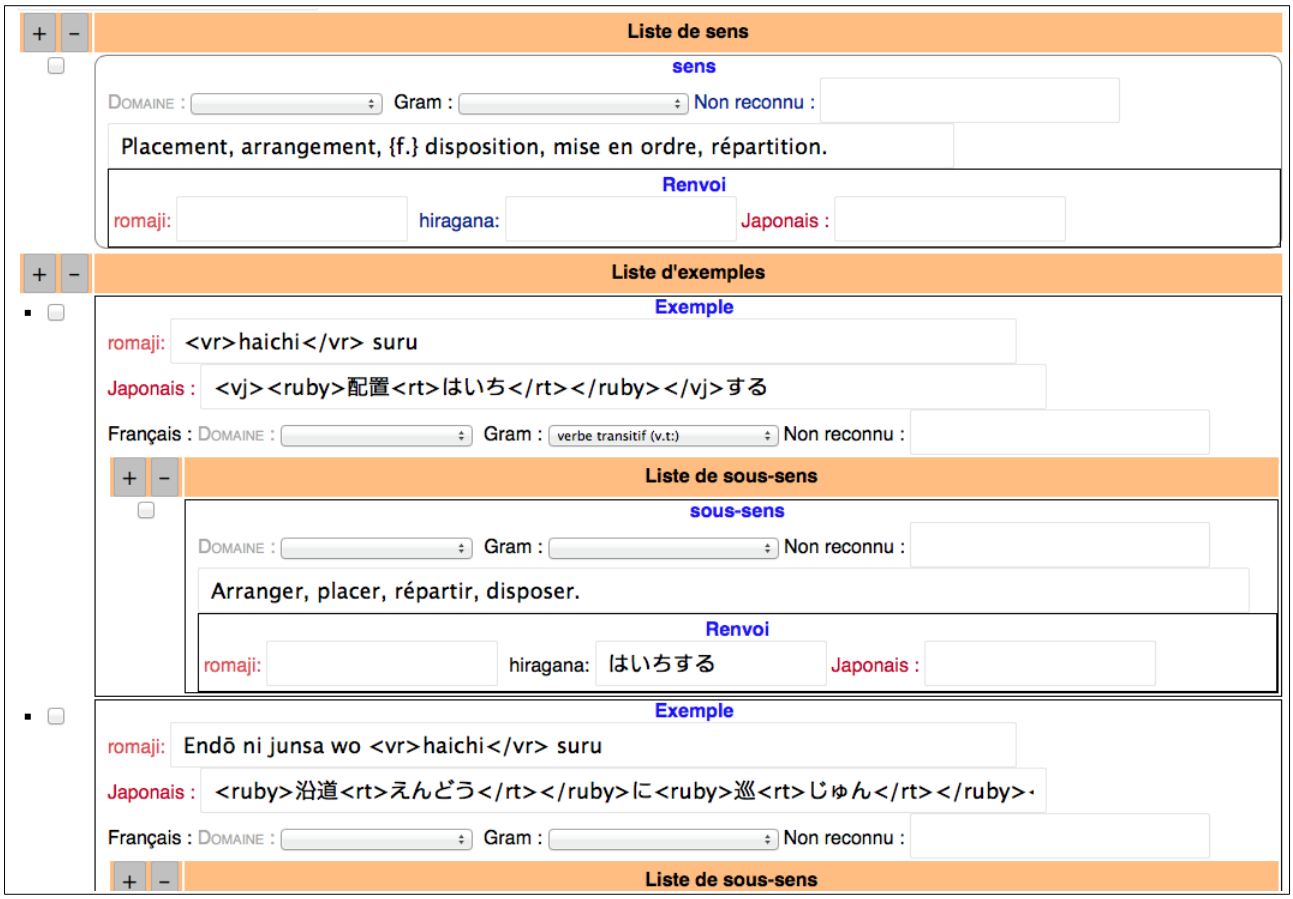


Figure 18: **Full editing form for the examples of the « 配置 » (haichi) article**

To access the full editing form, simply click on the link "Edit" menu at the top right of each article. The user can then edit the targeted article. At the end of the work, s/he previews the changes. They can be temporarily saved by attributing the "draft" status to the article. Modifications can also be cancelled or saved permanently in the database. All previous versions are kept in the database. This allows the administrator to go back if systematic errors are detected for a specific contributor or search criteria.

7.3 Statistics

7.3.1. Number of articles

The dictionary contains 153,897 articles in total. Among these,

- 82,663 articles come from the Cesselin dictionary, of which:
  - 9 270 are articles (11.21%) with headwords in kanji not recognized by the OCR;
  - 29 967 are articles (36.25%) with headwords not yet verified;
- 47,721 articles come from the JMdict dictionary;
- 23,512 articles come from links between Wikipedia pages.

Of the 153,897 articles, 45,144 articles (29.33%) are translated into English (and therefore must be translated into French).

7.3.2. Number of contributions

After seven months of online availability, on the 25 february 2016, the site had recorded 6,470 modifications of articles, among them:

- 1,059 headwords in kanji added,
- 225 headwords in kanji verified,
- 2,801 translations in French.

7.3.3. Number of visits

Seven months after its opening on July 22, 2015, the site had registered 1,288 visits by the 22 February 2015.

8. Conclusion

We have shown in this paper that it is possible to launch a project for the collaborative construction of dictionaries on the Web, using copyright free resources, and which allows a usable dictionary to be obtained immediately. The site has been open for only three months, but the high number of contributions already made shows that the experiment is a success.

The methodology described in this article to convert a printed dictionary into XML can be reapplied to any printed resource (of which there are a lot!).

The constitution of this resource is a starting point for future research.

Regarding the production of data, we plan to launch a similar process to retrieve a French → Japanese dictionary. We also plan to expand the current resource by adding new information (counters, quantifiers, frequencies of occurrences in corpora, etc.).

Regarding the use of data, obtaining a French → Japanese resource will allow us to experiment the convergence towards a pivot macrostructure (Mangeot et al., 2004). The

examples and their translation can be reused to build an aligned bilingual corpus which can then be used for example to build a statistical machine translation system such as Moses.

9. Acknowledgements

This project was made possible through the Hosei International Fund (HIF) program, which allowed us to be welcomed at Hosei University, Tokyo from October 2014 to August 2015.

References

A. Dictionaries

Cesselin G. 1940. *Dictionnaire japonais-français*, Maruzen, Tokyo, juillet 1940, 2340 p.

Gut Y., Puteri R., Megat R., Zaharin Y., Chuah Choy K., Salina A. S., Boitet Ch., Nédobejkine, N. , Lafourcade M. et al. 1996. *Kamus Perancis-Melayu Dewan, dictionnaire français-malais.* Dewan Bahasa Dan Pustaka, Kuala Lumpur, 667 p.

Hisamatsu K., Obataya Y., Hayakawa, F. et al. 2009. *Dictionnaire Japonais-Français / Français-Japonais*, Assimil, Paris, 1280 p. ISBN 978-2-7005-0445-3.

Matsumura A. 2006. *Daijirin Japanese-English dictionary (大辞林)*, 3rd edition, Sanseido, Tokyo, 2974 p. ISBN 4-385-13905-9.

Raguet É. & Martin J-M. 1953. *Dictionnaire français-japonais*, Hakusuisha, Tokyo, 1467 p.

B. Other literature

Apel U. 2002. WaDokuJT - A Japanese-German Dictionary Database. Papillon 2002 Seminar, 16-18 July 2002, NII, Tokyo, Japan, 13 p.

Berment V. 2004. *Méthodes pour informatiser des langues et des groupes de langues "peu dotées".* Thèse de nouveau doctorat, Université Joseph Fourier Grenoble I, Grenoble, France, 277 p.

Breen JW. 2004. *JMDict: a Japanese-multilingual dictionary*. In: Coling 2004 workshop on multilingual linguistic resources, Geneva, Switzerland, pp. 71-78.

Desperrier J-M. 2002. *Analyze [sic] of the results of a collaborative project for the creation of a Japanese- French dictionary.* In: Proceedings of Papillon 2002 Seminar, Tokyo, Japan.

EDR 1993. EDR Electronic Dictionary Technical Guide. Project Report, n°-042, Japan Electronic Dictionary Research Institute Ltd., 16 August 1993, 144 p.

Enguehard Ch. & Mangeot M. 2013. *LMF for a selection of African Languages.* Chapter 7, book "LMF: Lexical Markup Framework, theory and practice", Ed. Gil Francopoulo, Hermès science, Paris, France, 17 p.

Enguehard Ch., Mangeot M. 2014. *Computerization of African languages-French dictionaries.* Proc. of Collaboration and Computing for Under Resourced Languages in the Linked Open Data Era (CCURL), LREC 2014 workshop, Reykjavik, Island, 27 May 2014, 8 p.

Francopoulo G., Bel N., George M., Calzolari N., Monachini M., Pet M., Soria C. 2009. *Multilingual resources for NLP in the Lexical Markup Framework (LMF).* Language Resources and Evaluation, Vol. 43, pp. 57–70. ISBN: 10.1007/s10579-008-9077-5.

Griolet Pascal 2008. *Plus de « cent cinquante ans » d'histoire de l'enseignement du japonais.* Le japonais au XXIe siècle - Actes des États généraux pour l'enseignement du japonais en France, pp 47-63.

Koichi Hirao 2010. *La rénovation du dictionnaire français-japonais dans les années 1980 : le Dictionnaire général français-japonais de Hakusuisha et le Shogakukan Robert Grand Dictionnaire français-japonais* dans Heinz, Michaela (éd.), Cultures et lexicographies, Berlin, Frank & Timme, p. 103-111.

Mangeot M. 2001. *Environnements centralisés et distribués pour lexicographes et lexicologues en contexte multilingue.* Thèse de nouveau doctorat, spécialité informatique, Université Joseph Fourier Grenoble I, 280 p.

Mangeot M. 2006. *Papillon project: Retrospective and perspectives.* In P. Zweigenbaum, Ed., Acquiring and Representing Multilingual, Specialized Lexicons : the Case of Biomedicine, LREC workshop, Genoa, Italy, 6 p.

Mangeot M. 2014. *MotàMot project: conversion of a French-Khmer published dictionary for building a multilingual lexical system.* Proc. of LREC 2014, Reykjavik, Island, 28-30 May 2014, 8 p.

Mangeot M. 2015. *Construction of an open-source multilingual lexical system targeted on French and Japanese through contributive and automatic methods.* Internal Report, Hosei University, 12 p.

Mangeot M. & Chalvin A. 2006. *Dictionary building with the Jibiki platform: the GDEF case.* In LREC 2006, Genova, Italy, pp. 1666–1669.

Mangeot M. & Kuroda K. (2003) *Interlinguistic Divergences in Papillon Multilingual Dictionary.* Proc. of ASIALEX 2003, Meikai University, Urayasu, Chiba, Japan, 27-29 august 2003, pp 156-162.

Mangeot M., Sérasset G. & Lafourcade M. 2004. *Construction collaborative d'une base lexicale multilingue.* Traitement Automatique des Langues, vol. 44(2), pp. 151–176.

Mangeot M. & Thevenin D. 2004. *Online Generic Editing of Heterogeneous Dictionary Entries in Papillon Project.* Proc. of COLING 2004, ISSCO, Université de Genève, Switzerland, 23-27 August 2004, vol 2/2, pp 1029-1035.

Mel'čuk I., Clas A. & Polguère A. 1995. *Introduction à la lexicologie explicative et combinatoire.* Universités francophones et champs linguistiques. Louvain-la Neuve : AUPELF-UREF et Duculot, 256 p.

Polguère A. 2008. *Lexicologie et sémantique lexicale. Notions fondamentales.* Paramètres, 304 pages. Les Presses de l'Université de Montréal, Montréal, 2e édition. Nouvelle édition revue et augmentée.

Polguère A. 2013. *Tissage du Réseau Lexical du Français (RLF) : buts et méthodes.* 27e Congrès International de Linguistique et de Philologie Romanes (CILPR 2013), Jul 2013, Nancy, France.

Sérasset G. & Mangeot M. (2001) *Papillon Lexical Database Project: Monolingual Dictionaries & Interlingual Links.* Proc. of NLPRS 2001, Hitotsubashi Memorial Hall, National Center of Sciences, Tokyo, Japan, 27-30 November 2001, pp. 119-125.

Notes

1 http://www.dictionnaire-japonais.com
2 http://www.csse.monash.edu.au/~jwb/jmdict.html
3 http://www.wadoku.de
4 http://subs2srs.sourceforge.net/cbJisho/help.html
5 http://forum.koohii.com/viewtopic.php?pid=132030#p132030
6 https://en.wikipedia.org/wiki/Representational_State_Transfer
7 http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/
8 https://mecab.googlecode.com/svn/trunk/mecab/doc/index.html
9 https://dumps.wikimedia.org/jawiki/latest/
10 https://www.mediawiki.org/wiki/API:Main_page
11 http://www.csse.monash.edu.au/~jwb/kanjidic.html
12 http://jibiki.fr
13 https://jibiki.ligforge.imag.fr
14 http://cwb.sourceforge.net
15 http://corpus.estfra.ee
16 http://opus.lingfil.uu.se
17 http://jibiki.fr/jibiki/Api.po
18 https://en.wikipedia.org/wiki/Ajax_(programming)
19 http://www.w3.org/TR/xpath/
20 http://www.w3.org/XML/Schema

Dictionaries for Under-Resourced Languages: from Published Files to Standardized Resources Available on the Web

MATHIEU MANGEOT[1], CHANTAL ENGUEHARD[2]
[1]*GETALP-LIG laboratory, 38041 Grenoble, France (mathieu.mangeot@imag.fr);*
[2]*LINA laboratory, 44000 Nantes, France (chantal.enguehard@univ-nantes.fr)*

Abstract.

Most work in the field of natural language processing focuses on well-resourced languages. However, much remains to be done on under-resourced ones: there are few dictionaries, parsers, etc. Nevertheless, when published dictionaries are available, it is sometimes possible to find the data files used to print the dictionary (usually in Word format). A conversion process can then be applied to these files in order to obtain standardized XML lexical data. Attention must be paid to specific problems such as a lack of standardization in the alphabets or the use of hacked fonts for displaying specific characters. Next, the standardized XML data can be imported into an online lexical resources management platform. It is then available online for lookup and editing. A final step can also be performed to automatically export the data into interchange formats such as Lexical Markup Framework or lemon in order to produce linked data.

Keywords. Dictionary; lexical database; Jibiki platform; XML; LMF; Bambara; Khmer; Wolof; Niger; National Language

1 Introduction

In the field of natural language processing (NLP), most work focuses on well-resourced languages (English, French, German, Japanese, etc.): there are lexicons, dictionaries, corpora, sophisticated tools such as lemmatizers or parsers, etc. and one

can find a lot of online resources. However, much remains to be done on under-resourced languages: there are few dictionaries, parsers, etc. (and their quality is often very low) even when some have a large number of speakers (there are, for example, 250 million Bengalis, 30 million Haoussas).

Being a speaker of a poorly endowed language means limited or no access to the linguistic resources of this language: dictionaries, but also textbooks, literary, media, etc. These unmet needs affect many aspects of life: web, education, health, culture, etc. (Osborn, 2011). This difficult access to the resources compromises the capacity to express one's thoughts: freedom of expression is at stake. In addition, it negatively affects development with respect to several dimensions: the economy, culture, technology, public health, etc.

NLP also suffers from this shortage because it is very difficult to develop NLP tools or research without any linguistic resources or with limited resources. Under-resourced languages remain *terra incognita* for NLP.

The lack of quality and sustainable resources for under-resourced languages is a bottleneck that creates a vicious circle: the lack of resources discourages research in NLP, prevents the development of populations (including in education), resulting in a low level of education and a lack of "language workers" (linguists, lexicologists, novelists, reporters, etc.). Finally there are not enough linguists and lexicologists to produce good quality resources...

The United Nations Educational, Scientific and Cultural Organization (UNESCO) has mentioned several times this dimension and the richness of linguistic diversity. It has called upon Member States "*to promote the preservation and protection of all languages used by peoples of the world*". In 2005, during the conference organized in Bamako (Mali), it added that language is a critical factor in the ability to communicate.

Finally, the access to linguistic resources appears as an ethical issue. Promoting and creating linguistics resources for under-resourced languages meets this need.

Thus, we consider that there are two major reasons for NLP to develop research and tools which help in the creation and promotion of resources for under-resourced languages: the human development of the populations and the scientific issues of NLP.

When printed dictionaries for these languages are available, it is sometimes possible to find the data files used to print them (usually in Word format). A conversion process can then be applied to these files in order to obtain standardized XML lexical data that is next put online for lookup and editing or exported into interchange formats such as Lexical Markup Framework (LMF) or lemon, in order to produce linked data.

If the data files cannot be found either because they were lost or non-existent (in the case of old dictionaries), it is also possible to perform an optical character recognition process on the scanned files from the printed books. The result of the OCR process is then saved into Word or ODF format and processed as with the previous data files. In order to cope with the remaining OCR errors, it is possible to use a lemmatiser if it exists for the processed language, like in the Japanese-French Jibiki.fr project[1] (Mangeot, 2016).

In the first section of this article, specific issues are tackled concerning available dictionaries for under-resourced languages, and the resources to which the conversion methodology was applied are presented. The following section details technical problems such as the lack of standardization in the alphabets or the misuse of unicode characters with hacked fonts. The third section focuses on the conversion process of a published dictionary into a standardized XML file with a discussion about the choice of the standard format, the description of the process itself and the different steps it involves. The last section describes the use of Jibiki, an online lexical resources management platform for lookup and editing of the previously converted dictionaries.

2 Dictionaries for under-resourced languages

2.1 SPECIFIC ISSUES OF UNDER-RESOURCED LANGUAGES

Languages are more or less well-resourced in terms of their support by tools: adapted keyboard, spell-checker, speech synthesis, machine translation, etc. A classification based on the estimation of the electronic resources and tools defines three classes: well-resourced languages or τ-languages (eg: English, French), moderately-resourced languages or μ-languages (eg: Portuguese or Swedish), and under-resourced languages or π-languages (eg: Bambara, Kanuri or Khmer) (Berment, 2004).

The term under-resourced languages covers contrasting situations. We mention here three of them:

— it is the official language of a country, as is Irish (or Irish Gaelic) in Ireland. It is also the case for languages spoken by the majority of the population such as Khmer in Cambodia.

— it is a language without official status, that became a regional language: for example Basque and Breton in France; Ladin in Italy, Cornish in the United Kingdom.

— it is a national language of a country whose official language (used at school, or to write the laws) is different and often comes from a former colonizer state (Calvet, 1996). This is the case of African languages on which we have worked and that are spoken in Niger, Mali and Burkina Faso. In these three countries, the official language is French.

The dictionaries presented in this article concern Khmer, the official language of Cambodia and also eight African languages from countries where French is the official language: Bambara, Fulfulde, Hausa, Kanuri, Tamajaq, Wolof and Zarma. They are under-resourced languages whose socio-economic context is characterized by limited resources:

— there are few linguists who have an under-resourced language as their mother tongue and who exercise their professional activity in that language.

— the budget for the development of linguistic resources is low.

The governmental investment dedicated to language planning and, in particular, the development of electronic language resources is therefore very limited. The few studies that are conducted are characterized by a discontinuity in the time and spatial spread, which affects their sustainability and reuse (Streiter, 2006).

Because of the scarcity of linguistic research, descriptions of these languages are incomplete and many questions remain.

Developing lexical resources from scratch requires substantial budgets, qualified and available people, and the ability to lead a project for several years, conditions that cannot be met in many countries where there are few dictionaries which are generally not compiled by professional lexicographers. Therefore, the dictionaries on which we have worked contain numerous errors or incompleteness and are likely to evolve.

This contrasts sharply with the published dictionaries of well-resourced languages like French or English. For example, Larousse or Harrap's are firms employing dozens of professionals who regularly review their dictionaries over several decades.

However, there are some published dictionaries (often bilingual) which can be reused to achieve in only a few weeks, at low cost, a first version of an electronic resource. Collecting the digital files constitutes an important advance. However, in their absence, the dictionary can be keyed in again when only a printed copy has been collected.

Whatever its format (electronic or print), a dictionary represents a sum of important knowledge that can be recovered and reused. In all cases the authors or publisher of the initial dictionary should be involved in the project in order to obtain their agreement that the lexical resource that will be produced can be widely distributed in electronic form, visible on the Internet.

2.2 DICTIONARIES WRITTEN BY A SINGLE AUTHOR

Many of the dictionaries written by a single author are bilingual because their author, originally from a τ-language, aims to promote a π-language. Some were written by clerics in charge of the evangelism of populations in colonized countries ("pères blancs" in Africa, Portuguese Jesuits in Asia).

There are also dictionaries developed by literate people, often linguists, wishing to serve their mother tongue. This is the case of the elementary Hausa-French dictionary written by Abdou Minjinguini (Minjinguini, 2003) and the monolingual Zarma dictionary written by Issoufi Alzouma Oumarou (Oumarou, 1997).

The conversion methodology (described in section 4) has been applied successfully to the following two dictionaries.

2.2.1 The French-Khmer dictionary

abondant, e (fruits, riz…) — (pluie) (trempé-humide)	(daːel) sɑmboː (daːel) coːk-coam
abonnement (magazine) — (téléphone)	ciːəw-prāː cam kaː- bɑŋ-seːvaː
abonner (sur pied-nom (s'inscrire)-commercer)	coh-chmuəh-ciːəw
abonner (s') (à un magazine) — (au téléphone)	ciːəw-prāː cam bɑŋ-seːvaː
abord (adv.) (d'—)	mun-dɑmboːŋ / daːəm-laːəj
aborder (accoster) — (qqn) (appeler-arrêter) — (commencer-discuter-sur-sujet-un)	coːl-cāː t / coːl-cət haw-bɑŋchop phdaːəm-coːcɛːk-ɑmpɨː paŋə-haː- muːəj
aboutir (arriver à destination, déboucher) — (avoir-résultat) — (devenir) (aller-être)	təːw-dɑl miːən-loːttəphāː l təːw-ciːə

Figure 1: Excerpt from the French-Khmer dictionary in Word format

The French-Khmer dictionary[2] project (Richer et al., 2007), started in the late 1990s, was completed in 2006 by a small group of computer scientists gathered in the "Pays Perdu" NPO created by Denis Richer, a French ethnolinguist established in Siem Reap (Cambodia). This first version of the dictionary was published in spring 2007 and has 13,249 entries. Each entry is composed of a French headword, in some cases a part-of-speech and a list of word senses. Each word sense has a gloss in French and a translation in Khmer. The Khmer translation is noted in International Phonetic Alphabet and not in Khmer writing. The dictionary was originally encoded in Word format. An example of the original file can be seen in Figure 1.

2.2.2 The Bambara-French dictionary

The Bambara-French dictionary[3] of Father Charles Bailleul (1996 edition) includes more than 10,000 entries. This dictionary is primarily intended for French speakers wishing to improve Bambara but it is also a resource for Bambara speakers. In the

words of the author himself, the dictionary "plays the role of a working tool for literacy, education and Bambara culture." To date, it can be considered as the most comprehensive dictionary of the language. It is also used by specialists of other varieties of this language such as Dyula (Burkina Faso, Côte d'Ivoire) and Malinké (Guinea, Gambia, Sierra Leone, Liberia, etc.).

2.3 DICTIONARIES BUILT BY PROJECTS

Dictionaries built by projects have several authors. The group of authors usually defines some principles about the structure and the definition of closed lists of values such as grammatical classes.

The most recent dictionaries of this type are built with lexicography tools such as Linguist Shoebox/Toolbox[4] (Buseman et al., 2000) and FieldWorks Language Explorer (FLEx)[5] of the Summer Institute of Linguistics (SIL) or TshwaneLex (TLex)[6].

Even if these tools are able to export content to an XML structure, it is often the case that this operation has not been performed or the files produced by the lexicographical tools are not available. Only Word files can be used for printing.

The conversion methodology (described in section 4) has been applied successfully to the following six dictionaries.

2.3.1 The Niger SouTeBa dictionaries

In the DiLAF project[2] (Enguehard and Mangeot, 2014), we worked on five dictionaries written in five national languages of Niger and French. They have been produced by the Soutéba project (a program to support basic education) with funding from German cooperation and the support of the European Union. The software used for building these dictionaries is the SIL toolbox. They have a simple structure because they were designed for children in primary school classes in a bilingual school (education is given there in a national language and in French). Most terms of lexicology, such as lexical labels, parts-of-speech, synonyms, antonyms, genres, dialectal variations, etc. are noted in the language in question in the dictionary,

contributing to forge and disseminate a meta-language in the local language, a specialized terminology. The entries are listed in alphabetical order, even for Tamajaq (although it is usual for this language to sort entries based on lexical roots) because the vowels are written explicitly (this mode of classification was preferred because it is well known by children).

The Fulfulde-French dictionary[7] includes 4,305 entries. The orthographic form of the entry is followed by a part-of-speech. Next, follows a definition and an example in Fulfulde. Some grammatical information is mentioned, such as the plural form (mbahdi). The entry ends with a French gloss (far). Here is an example:

> **saabi** *jokk*. helmere jokkondiroore konngi ngam hollude hujja. *saabi o sooda ngawri wadi o soonni nga'ari makko*. mbahdi : saabi. *far :* à cause de.

The conversion of this dictionary was particularly difficult because the original word files disappeared and the dictionary was re-keyed in by hand from a printed version. Thus many structuring errors could be found.

The Hausa-French dictionary[2] includes 7,823 entries. The orthographic form of the headword is followed by the pronunciation (tones are marked with diacritics placed on vowels) and part-of-speech. On the semantic level, there is a definition in Hausa, a usage example (identified by the use of italics), and the equivalent in French. Here is an example:

> **jaki** [jàakíi] *s.* babbar dabbar gida mai kamar doki, wadda ba ta kai tsawon doki ba amma ta fi shi dogayen kunnuwa. *Ya aza wa jaki kaya za ya tafi kasuwa. Jin.:* n. *Sg.:* **jaka**. *Jam.:* **jakai, jakuna**. *Far.:* âne

The Kanuri-French dictionary[2] includes 5,994 entries. The orthographic form of the entry is followed by an indication of pronunciation relating to tones. The part-of-speech is shown in italics, followed by a definition, a usage example, a French translation and meaning in French. Additional information may appear as variants. Here is an example:

abərwa [àbɔ̀rwà] *cu.* **Kəska təngəri, kalu ngəwua dawulan tada cakkidə.** *Kəryende kannua nangaro, abərwa cakkiwawo.* [Fa.: ananas]

The Tamajaq-French dictionary[2] includes 5,205 entries. The orthographic form of the entry is followed by the part-of-speech and a gloss in French displayed in italics. For nouns, morphological information about the state of annexation is often included, the plural and gender are also explicitly stated. A definition and an example of usage follow. Other information may appear as variants, synonyms, etc. As Tamajaq is not a tonal language, phonetics does not appear. Here is an example:

əbeɣla *sn.* mulet ♦ **Ag-anɣer əd tabagawt.** *Ibeɣlan wər tăn-tăha tămalăɣa.* anammelu.: **făkr-ejăḍ.** təmust.: **yy.** iget.: **ibəɣlan**.

The Zarma-French dictionary[2] includes 6,916 entries. Each entry has an orthographic form followed by a phonetic transcription in which the tones are rated according to the conventions already set for the Kanuri. The part-of-speech specifies explicitly the transitivity or intransitivity of verbs. For some entries, antonyms, synonyms and references are indicated. A gloss in French, a definition and an example end the entry. Here is an example:

ɲagas [ɲágás] *mteeb.* ● *brusquement (détaler)* ● sanniize no kaŋ ga cabe kaŋ boro na zuray sambu nda gaabi sahã-din ● *Za zankey di hansu-kaaro no i te ɲagas*

2.3.2 The Wolof database

The Wolof database (Cissé, 2013) is a multifunctional lexical database from which it is possible to extract a monolingual Wolof dictionary as well as a bilingual Wolof-French dictionary. Each word sense has its own entry in the database. Polysemic words will then have several entries. The database was built with the SIL Toolbox tool. Each entry has several administrative fields (or meta-information): status of the entry, comments, author of the entry.

3 Technical difficulties

3.1 LANGUAGES WRITTEN BUT POORLY STANDARDIZED

While the fact that a language is under-resourced has been defined solely in terms of its equipment in IT tools and resources, the linguistic knowledge of this language is often scarce: there are few studies on the language, they are inaccessible because they are not published in journals or conference proceedings, and they are not available online. Moreover, these languages are also not very present in schools, either as a subject of study or as a teaching language. However, some exceptions are worth mentioning:

In Niger, experimental schools were established in the 1980s. The teaching is entirely delivered in a national language in the first half of the primary cycle. During the second half of the cycle, French appears and the national language is studied as a subject. In the final year, the teaching takes place in French only. It will be the same for the rest of school: middle school, high school and higher education (Programme Décennal du Développement de l'Éducation, 2003).

In Ecuador, the Shuar people (called improperly Jivaro) was structured in a Federation of Shuar Centres in 1964. In the 1970s the Federation organized, among other initiatives, the establishment of primary schools in villages with support through radio programs. These schools are bilingual. Instruction is provided almost entirely in Shuar in the first two years to move towards parity in education in Shuar and Spanish in the final year (Calvet, 1987).

Beyond the success of these approaches to children's literacy, the creation of a school curriculum promotes the writing and editing of textbooks which constitute text corpora written by language specialists, sometimes linguists, with a good knowledge of the written language. Such corpora could be exploited by linguists to build lexical resources. Their size remains small.

Other national language texts emerge. They are written by journalists and authors (of stories, novels, etc), who have mostly no access to language resources. Therefore,

these texts are written in a somewhat standardized language with particularly many spelling variations. These corpora thus cannot be used as a data source to automatically build lexical resources.

In this context of deprivation, the reuse of a published dictionary is a first step that will accelerate the creation of usable resources for natural language processing applications. In some dictionaries, word spelling is standardized and complies with linguistic studies; in others, such as (Bailleul, 1996), variants are explicitly marked and located geographically, while the official spelling is reported in addition to the usual spellings. Furthermore, the definitions and examples of use are a corpus of sentences and many entries are accompanied by morphological information for calculating the different forms of the same entry.

3.2 SPECIAL CHARACTERS

Set up in the 1960s, long before Unicode, the alphabets of most African languages use special characters which were absent from the standard character tables at that time. Although the alphabets of languages on which we have worked (Enguehard, 2009) are mainly of Latin origin, new characters needed to note specific sounds in some languages with a single character have been adopted by linguists in a series of meetings. Thus, each of the alphabets of the African languages we have worked on includes at least one of these special characters: ɓ ɗ ə ɛ ɣ ƙ ɲ ŋ ɔ ʋ. The character b with hook (ɓ, Ɓ) appears in the Hausa alphabet (République du Niger, 1999a) while the character eng (velar n) (ŋ, Ŋ) appears in the Bambara, Tamajaq (République du Niger, 1999c) and Soŋai-Zarma (République du Niger, 1999d) alphabets. Characters composed of a Latin character and a diacritical mark have also been created (such as â, ă, ã, ḍ, ğ or š).

Targeted primarily for printing texts on paper, fonts displaying these special characters have been created by redrawing the glyphs of certain characters (Chanard & Popescu-Belis, 2001). These fonts have for decades allowed texts to be published in national languages, but they prohibit natural language processing on these texts (Enguehard, 2009). The habit of using them is being installed, and the source files of

the dictionaries we have collected use such fonts. It is therefore necessary to convert them to Unicode so that the character encoding meets the international standards.

3.3 CHARACTER CONVERSION TO UNICODE

Establishing replacement characters can be tricky if the original fonts are not available, which is the most common situation. In this case, it is preferable to have a printed version to be sure to establish proper conversions. This step can be more subtle when the same character is used to display different glyphs. For example, the ampersand & is redrawn as t with a dot below ṭ of the Tamajaq alphabet in the 'Albasa Tamjq' font, as d with a hook ɗ of the Hausa alphabet in the 'AlbasaRockwellhau' and 'Hausa' fonts and as open e ɛ of the Bambara alphabet in the 'Times New Bambara' and 'Arial Bambara' fonts. It may also happen that different fonts have identical names. Finally, the same character may be used within the same document to display different glyphs. For example in the Tamajaq-French bilingual dictionary (Programme de soutien à l'éducation de base, 2007), the p lowercase character (U+0070) was used as such in the parts of the entries in French, and redesigned as a schwa ə in the 'Tamajaq Literacy2 TT20.4 SILSop' font for the Tamajaq parts.

Table 1 shows part of the list for Zarma. There is no automatic method that will detect these problematic characters. It is imperative to look at the data.

Origin	*Unicode*
§	ã
$	ɲ
ù	ŋ
£	Ɲ

Table 1: Partial view of the Unicode correspondence table for Zarma.

Figure 2 shows two entries. In their original version, special characters initially entered with a hacked font are not readable. In the Unicode version these special characters have been transformed to comply with Unicode.

äœaruf *sny. pardon* ¥ Agamay n pkpnni dpffpr erk ärät. Musa as ypwät empji-net dpffpr pnki ypgmäy dä£-as äœaruf. *An:* tptubt. *Sf:* ä . *Gt:* äœuruf. *TW:* tpsureft

ăṣaruf cat=sny. pardon ▸ Agamay n əkənni dəffər erk ărăt. Musa as yəwăt eməji-net dəffər ənki yəgmăy dăɣ-as ăṣaruf. *An:* tətubt. *Sf:* ă . *Gt:* ăṣuruf. *TW:* təsureft.

Figure 2: Tamajaq lexical entry "ăṣaruf" in published format then in Unicode format

3.4 DIGRAPH LEXICOGRAPHICAL ORDER

Digraphs can be easily typed using two characters but their use changes the sort order which determines the lexicographic presentation of dictionary entries. Thus, for Hausa and Kanuri, the digraph 'sh' is located after the letter 's'. So, in the Hausa dictionary, the word "sha" (drink) is located after the word "suya" (fried), and, in Kanuri, the word "suwuttu" (undo) precedes the noun "shadda" (basin).

These subtle differences can hardly be processed by software and require that digraphs appear as a proper sign in the Unicode repertoire. Some used by other languages are already there, sometimes under their different letter cases: 'DZ' (U+01F1), 'Dz' (U+01F2), 'dz' (U+01F3) are used in Slovak; 'NJ' (U+01CA), 'Nj' (U+01BC), 'nj' (U+01CC) in Croatian and for transcribing the letter " Њ " of the Serbian Cyrillic alphabet, etc.

It would be necessary to complete the Unicode standard with digraphs of Hausa and Kanuri alphabets in their various letter cases.

fy	Fy	FY
gw	Gw	GW
gy	Gy	GY
ky	Ky	KY
kw	Kw	KW
ƙy	Ky	KY
ƙw	Kw	KW
sh	Sh	SH
ts	Ts	TS

Table 2: Hausa and Kanuri digraphs missing in Unicode

3.5 CHARACTERS WITH DIACRITICS

Certain characters with diacritics are included in Unicode as a unique sign, whereas others can only be obtained by composition.

Thus, vowels with tilde 'a', 'i', 'o' and 'u' can be found in Unicode in their lowercase and uppercase forms while the 'e' with a tilde is missing and must be composed with the character 'e' or 'E' followed by the tilde accent (U+303), which can cause renderings which are different from other letters with tilde when viewing or printing (tilde at a different height for example).

Letter j with caron exists in Unicode as a sign ǰ (U+1F0), but its capitalized form J̌ must be composed with the letter J and the caron sign (U+30C).

The characters ẽ, Ẽ and J̃ should be added to the Unicode standard.

Concerning letter case change, word processors usually provide this functionality, but do not always realize it in the correct way. Thus, we have found during our work that the OpenOffice Writer software (3.2.1 version) fails in transforming 'ɾ' to 'ᴙ' from lowercase to uppercase or vice versa (the character remains unchanged) while Notepad++ (5.8.6 version) fails in transforming 'ǰ' to 'J̌'.

4 Conversion process into a standardized format

4.1 LEXICOGRAPHIC FOUNDATIONS

For recovered dictionaries (such as the DiLAF project, see the central format below), we do not know exactly which lexicographical choices led to the nomenclature and microstructure. In the DiLAF project, the aim is to make existing resources available to the public. The initial lexicographical choices of the authors are therefore virtually unchanged.

- Lexicographical order is implemented by a specific function directly on the database following the orders defined by the alphabets of the concerned languages;
- The macrostructure is not altered: each paper volume is represented by an electronic volume;
- The microstructure is sometimes slightly modified according to the dictionary.

Traditional lexicography distinguishes two vocables as homographs if they have no clear semantic link with each other (Polguère, 2008). But in practice, it frequently happens that dictionaries of the same languages follow a division into different homograph vocables.

We believe that this distinction is actually arbitrary. To our knowledge, there are also no specific criteria to evaluate a semantic link between two words that can be implemented easily with NLP tools, let alone for under-resourced languages. When it is possible, we have therefore chosen to group homograph vocables into a single entry. However, we distinguish entries with different parts-of-speech. The combination of a lemma and a part-of-speech therefore constitutes a single entry.

For lexical databases generated from retrieved data (such as the MotÀMot project, see the target format below), the Jibiki platform allows great freedom in the definition of the macrostructure (ref paper links). It is possible to design complex macrostructures composed of several layers of volumes related to each other (see PiVAX (Nguyen et al. 2007) or proAxie (Zhang and Mangeot, 2013) macrostructures). However, the foundations of a Jibiki structure are based on a traditional approach to designing dictionaries, the onomasiological approach: from the lexical unit to the word meanings and gathering word meanings into interlingual links (axies) at the interlingual pivot level. Axies are not concepts, although they tend to become ones.

The OntoLex W3C working group, and more generally linked data and lemon projects are based on a semasiological approach (from the concept to the word

meaning). As stated in the first point of the OntoLex declaration: “The mission of the Ontology-Lexicon community group is to develop models for the representation of lexica […] relative to ontologies. These lexicon models are intended to represent lexical entries containing information about how ontology elements […] are realized in multiple languages. It is certainly possible to define a macro-structure based on a semasiological approach with Jibiki, but the tool has not been designed for that and it has not yet been experimented. For a more detailed discussion, see (Zhang et al. 2014).

Another difference between these two approaches is that, for the onomasiological one, the aim is to build consistent and high-quality resources. This can be done only if thorough verifications are carried out on the nomenclature (choice of entries) and the data. Several possibilities exist: to indicate for each entry a quality level, to show the history of the origin of the data, to establish a process of revision / validation, etc. For the semasiological approach, the main goal is to obtain the broadest coverage in terms of quantity of entries and number of languages. That said, we believe that in the future these two approaches, complementary, should become closer or even merge.

4.2 CHOICE OF THE STANDARD

Our goal is to convert published dictionaries to make them available to the natural language processing (NLP) scientific community. Thus, the final format must be based on standards. We studied two main standards: The Text Encoding Initiative and the Lexical Markup Framework.

4.2.1 The Text Encoding Initiative

The TEI is led by a consortium gathering American and European public research organizations. Its goal is to define an exchange format for exchanging, creating and storing annotated texts with a standardized tagset. The latest version is P5, published in 2007. Each TEI encoded document must begin with a header described in chapter 2.

Chapter 9 focuses on dictionaries. To address the problem of the structuring of entries, the TEI provides a binary solution: an article can be represented by a *<entry>* element whose structure is very rigid and codified; the element *<entryFree>* may be preferred because it admits the insertion of any elements in any order in the entry.

In practice, the *<entry>* is too burdensome to use. Therefore, lexicographers prefer to use the *<entryFree>*, but it is too loose to allow effective standardization and exchange of data encoded using the TEI.

TEI has had real success in encoding corpora. This is unfortunately not the case for dictionaries. The proposed solution (dichotomy between the *<entry>* and *<entryFree>*) is not satisfactory. As a consequence, very few dictionaries are encoded with TEI.

4.2.2 Lexical Markup Framework: the normative part

Lexical Markup Framework (Romary et al., 2004) is a meta-model separating the lexical parts, grammatical and semantic. The main class is a *Lexical Resource*. It contains a class that describes the meta-information on the lexicon *Global Information* and one or more *Lexicons*. The lexicon contains one or more *Lexical Entries*. A lexical entry contains one or more *Forms* of the entry and one or more *Senses*. The forms contain spelling variants *Form representations*. The senses can in turn contain other senses recursively. They can contain *Definitions* that contain *Text Representations* and narrative descriptions or *Statements*. The *Representation* class allows one to link different *Form Representations* and their occurrences in a *Text Representation*. This meta-model became an ISO standard under the number 24613:2008 in November 2008 (Francopoulo et al., 2009).

We would draw the reader's attention to the fact that LMF is separated into two parts. The normative part describes the meta-model but does not specify the data representation scheme (which elements and attribute names to use). This is described in the informative part of LMF, which is not included in the standard itself. Thus, it is possible to obtain a resource using its own elements and attributes names but still following the normative part of LMF. Some people enjoy that freedom (and we are

among them), while others would have preferred the informative part to be included in the standard in order for all LMF resources to follow the same syntax.

4.2.3 Lexical Markup Framework: the informative part

The informative parts of LMF give precise information as to how to encode a lexical resource in XML: the structure that must be followed and the elements and attributes that must be used. As it can be suitable for an interchange format, we prefer not to use it as a working format for the following reasons:

- The name of the elements are in English. While it can be understood by the majority of the researchers in the world, it can be difficult to understand for the people working directly on the dictionary. When developing a resource, they are the most important people to take into consideration!
- Objects of different nature can be put at the same level. We think that, in order to be clearly understandable, an XML format should avoid putting objects of different nature at the same indentation level. The siblings of an element must be of the same nature. The LMF format does not respect this principle. The object <*GlobalInformation*> which isa meta-information about the lexicon is a sibling of the object <*Lexicon*>, which is the resource itself.
- Free text is stored in attribute values. In XML, it is customary to include items from closed lists as attribute values, and frame the free texts by using markup tags. This general principle is not respected in the informative part of LMF since all the information is stored in textual attributes. This choice has the effect of prohibiting the minimal information display via a browser for example.

4.3 PRESENTATION OF THE METHODOLOGY

4.3.1 The conversion process

The conversion methodology proceeds in several steps and requires successive transformations of the published dictionary to several XML files called copy, central,

target and export formats. We also take into account the fact that lexicographers will revise and develop the produced resources.

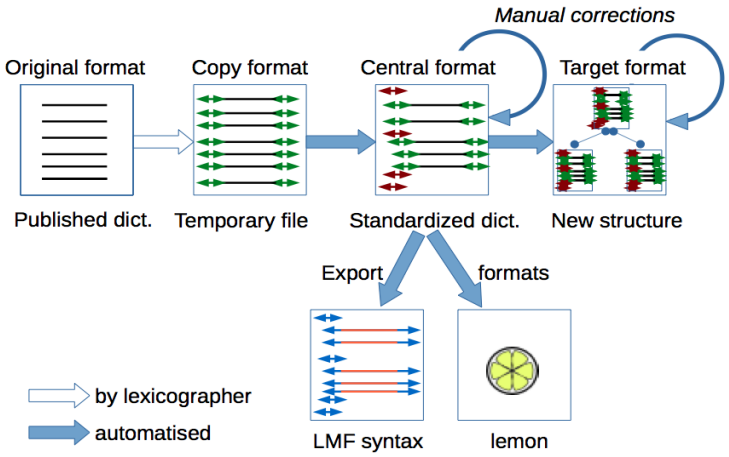


Figure 3: Conversion process

NLP experts develop conversion programs to process the transformation from copy format to central format, and from central format to target format. When they conceive these programs they get the opportunity to detect new errors and inconsistencies that are reported for subsequent correction.

The methodology is simple and based exclusively on freely available tools such as Open/Libre Office, NotePad++ and FireFox. In the following parts, we will briefly present the necessary steps of the methodology to transform one format to another. For more information, it is possible to refer to the technical manual stored on the DiLAF website (see the “Méthodologie” section of the “Projet” page). In the following section, the methodology steps necessary in order to obtain each format will also be explained.

4.3.2 The copy format

The copy format is a structural copy of the published dictionary in a valid XML format: the nature of each information part is identified and pieces of information are bracketed by XML markups according to this nature: it could be a part-of-speech, a

definition, an example, etc. The transformation of the published dictionary to the copy format is performed by lexicographers with the support of NLP experts. This step requires the resolution of many problems, including the conversion of special characters to Unicode, the identification of each information part, the definition of a set of markup tags and finally the explicit tagging of information by adding tags (Mangeot & Enguehard, 2013).

In order to bracket each information part composing an entry (definition, lexical label, phonetic, synonyms, translations, etc...), a set of elements must be chosen. This raises the question of the choice of the language used for the elements. English, the international language of research may be favoured. But in many cases, it is not present in the dictionaries. Furthermore, it is not mastered by all linguists working on the project. In the case of under-resourced language computerization projects, it is important to encourage partners to use terms in their own language (or mother tongue) to define the names of the elements. This may eventually lead to the creation of new terms that did not exist in these languages and contribute to the transfer of knowledge and ideas and, consequently, to scientific and technological development (Diki-Kidiri, 2004). From a political point of view, it participates in moving away from a post-colonial vision of the social status of these languages and contributes to their valorization.

For example, Table 3 shows the element names chosen for the Kanuri-French dictionary (Programme de soutien à l'éducation de base, 2004) in the DiLAF project:

Name of the element in Kanuri	English equivalent
kalma	headword
bowodu	pronunciation
naptu_curo_nahauyen	part-of-speech
maana	definition
misal	example
kalakta	translation
maana_tiloa	synonym
fərəm	antonym

bowodu_gade	variant
mane	cross-reference

Table 3: Element names chosen for the Kanuri-French dictionary. The conversion process from the published format to the copy format consists of four main steps:

1. Retrieving the published format in XML format (either Open Document Format or OpenXML from Microsoft). A text document in Open Document format (.odt) or Open XML (.docx) is in fact a zip archive containing several files and folders. The core document containing the text must be extracted from the zip archive. For a .odt document, the file is called content.xml; for a .docx document, the file is word/document.xml.

2. Applying an XSL stylesheet to simplify the XML. The ODF or OpenXML formats are very verbose and difficult to read. Furthermore, much of the information concerning styles is not useful for the conversion process. Therefore, a first simplification of the XML can be made with an XSL stylesheet. More precisely, the “text:p” elements are replaced by “p” elements, and “text:span” elements are replaced by the value of the “text:style-name attributes; headers, sections, columns breaks and page breaks are removed.

3. Tagging each information part by converting the XML with regular expressions. This part is the most important one and takes up most of the conversion time. One must recognize how each information part is tagged and replace those tags by the new tag set defined previously.

4. Checking the validity of the result file. First, the well-formedness of the XML file must be checked. A simple web browser can be used for this task, as long as it gives the line numbers where the errors are located. Once the file is well formed, the structural validity can be checked. Before this step, it is necessary to specify the structure of the copy format (either with a Document Type Definition or an XML schema). Then an XML parser is used to check the validity.

Annex 1 shows the result of the conversion process from the published format to the copy format for a Kanuri entry.

When a first valid version of the copy format is available, various checks are made using programs (counting the number of occurrences of each tag, checking the embeddedness of the markups, counting the number of closed list values such as parts of speech, etc.) and errors are reported to lexicographers, who can make the corrections. The copy format does not alter the structure nor the order of the information of the original format, but improves readability by explicitly labelling every information part. Finally, it is designed to disappear in favor of the other formats.

4.3.3 The central format

The central format respects the normative core of LMF. Consequently the original order of the information (that was still kept in the copy format) is changed. The structure of the entries follows the LMF meta-model but the tag names do not necessarily follow the informative part of LMF. It is obtained by applying an XSLT program that performs structural changes on the copy format. During this step, it is sometimes necessary to clarify the nomenclature.

The basic unit constituting an article may vary from one dictionary to another: the lexeme (lemmatised surface form without part-of-speech), the vocable, etc. When the basic unit is the vocable, homonym vocables have multiple entries. It is necessary to distinguish these entries in order to indicate possible links (synonymy, homonyms, etc.) between them. It is also necessary to identify each word sense of an article. It is therefore necessary to build a unique identifier for each article and each word sense in an article. When the nomenclature choices have not been respected throughout the dictionary, it is sometimes also necessary to perform certain changes in the nomenclature: merging two articles, for example when word senses of the same vocable have been described in two different articles; splitting an article into two, for example when an article includes two different parts-of-speech, etc. Finally, within the microstructure, restructuring is sometimes required, such as moving morphological information described in a semantic block to the form block.

These treatments are performed by perl programs. Markup tag names are preserved from the copy format.

Depending on the goals of the project, especially if there is a target format, the central format can be frozen and proposed for download without further modification. It represents the standardized electronic version of the published dictionary.

A detailed example of a Kanuri entry in central format is available in Annex 2.

4.3.4 The target format

The target format is specific to each project. Each target format is defined from the needs of a project. The central format is then converted automatically into the target format.

The resources in central formats are electronic versions of printed dictionaries, mainly monolingual or bilingual. The use of computers has helped to overcome the constraints of the paper form. The impossibility of inverting bilingual dictionaries led to a model having a "pivot" consisting of an axis (interlingual meanings). This leads to the definition of new macrostructures based on this pivot such as Papillon (Mangeot et al., 2003) and Pivax macrostructures (Zhang et al., 2014). Several lexical resources can be used and merged to enhance the quality and add information that is not available in the converted resource. The result is a new resource that will then be corrected and completed online by voluntary or paid contributors.

For example, in the case of the French-Khmer dictionary of the MotÀMot project (Mangeot, 2014), the French volume has been completed with information from two other existing resources: the pronunciation of the entries was taken from the FeM dictionary and the list of French entries from the GDEF dictionary (taken from the Morphalou lexicon, taken from the TLFi).

In the future, the target format will be the only one to evolve. It can then be uploaded onto an online lexical resource management platform in order to be readable and editable online by lexicographers who will be able to correct and enhance it directly (by adding new lexical entries, adding various information, translations, examples,

etc..). It would then be easy to generate a new export format dictionary by processing again the appropriate program on the target format dictionary.

4.3.5 Export formats

Export formats depend also on the needs of each project. But, as the central format respects the LMF normative part, it is easy to generate a format that follows the syntax of the informative part of the LMF standard. It is obtained by processing the central format with an XSLT program. The transformations are limited to changing the name of an element, to add an additional level with a child element, and to convert a text node into an attribute value.

Nevertheless, there is still an important issue: the data categories. In order to be 100% compliant with the LMF standard, the data categories such as the list of parts-of-speech must follow the ISOcat Data Category Registry (DCR)[8]. We encountered parts of speech that are missing such as "ideophone", appearing in the list of parts of speech in Hausa and Kanuri dictionaries. Thus, it appears necessary to enrich this list or to allow a modular definition of this list with a sublist for each language.

A detailed example of a Kanuri entry in LMF syntax format is available in Annex 3.

Linked data is a set of strongly interconnected graphs. In the case of data coming from dictionaries, the graphs are lexical networks where nodes represent the lexemes of one or more languages, and links represent the relationships between these lexemes (translation, synonymy, etc.). A lexical network can be monolingual or multilingual.

Although lexical networks have many advantages, they are not suitable for all usages. Usually, they are not browsable in alphabetical order. But we need that possibility to have an idea of the content of a lexical repository, whatever its nature. On the other hand, in a lexical network, the concept of volume is missing, which prevents the creation of a resource in a simple way when studying a new language. When importing previously existing dictionary data into a lexical network, the editorial responsibility of the dictionary editors is also lost. Most of the time, there is no guarantee of quality, nor a clear view of which entry should be in the network or not.

Once again, while it is important to produce linked data in order to provide machines with easy access to our data, this should not be directly used as a working format.

In order to produce linked data, the ideal export format is the lemon model[9]. It is very close to LMF (Eckle-Kohler et al., 2014). Apart from slight modifications for some element names, the biggest difference is the word senses. In LMF, there is a fixed set of senses whereas in lemon, each word sense is linked to an ontology concept. In the case of LMF, the *semasiologic* (from word to meaning) approach has been chosen whereas in the case of lemon, the *onomasiologic* one (from concept to word) has been chosen. In the case of bilingual dictionaries or multilingual databases, we prefer to follow the *semasiologic* approach and use *axes* (interlingual acceptions) instead of concepts in order to link word senses of different languages (Mangeot et al., 2003). This is the choice adopted by dbnary[10] (Sérasset, 2014), the team database for linked data.

5 Web access via a resource management platform

5.1 DESCRIPTION OF THE PLATFORM

Jibiki (Mangeot et al., 2006; Mangeot, 2006) is a generic platform for handling online lexical resources with user and group management. It was originally developed for the Papillon Project. The platform is programmed entirely in Java based on the "Enhydra" environment. All data is stored in XML format in a Postgres database. This website mainly offers two services: a unified interface for simultaneous access to many heterogeneous resources (monolingual or bilingual dictionaries, multilingual databases, etc.) and a specific editing interface for contributing directly to the dictionaries available on the platform.

Several lexical resource construction projects are using this platform successfully for consultation or edition (DiLAF[2], GDEF[11], Jibiki.fr[12], MotÀMot[1], Papillon[6], Pivax[13]).

The source code for this platform is freely available for download from github[14]. A docker image is also available[15].

5.2 IMPORTING A RESOURCE ON THE PLATFORM

The Jibiki lexical resource management platform is able to handle any resource provided that it is encoded in XML. Indeed, a system of common pointers in heterogeneous structures can manipulate the resources without changing their structure. Each pointer is indexed in a database and can perform a quick search. This system is called Common Markup Dictionary (CDM). There are predefined common pointers for lexical items that are commonly found in most dictionaries. It is also possible to define specific pointers for a resource.

Therefore, dictionaries in many formats can be imported (copy, central, target or LMF formats). It may be a structured dictionary following the recommendations of the TEI, such as the LMF standard, etc.

In order to import a resource into the Jibiki platform, it has to be described in XML metadata files. The dictionary metadata (authors, dates, licence, etc.) and macrostructure (languages, volumes, links between volumes) are described in the dictionary metadata file. Each volume and microstructure is described in a separate volume metadata file.

The entry microstructure of each volume is described using common pointers identifying the same kind of information (entry, entry id, headword, pronunciation, part-of-speech, definition, domain, word sense, translation, translation link, example, etc.). These CDM pointers use the XPath standard. They allow the resource to be handled without any format conversion that would involve a loss of information.

An HMTL interface simplifies the creation of the metadata files. The dictionary metadata file is filled in by hand. The volume metadata files are automatically generated by a program that analyses the XML data and produces several description files (number of entries, CDM pointers, XML schema of an entry, XSL stylesheet,

XML template for an empty entry, etc.). The information can then be corrected by the user via an HTML interface.

When the metadata files are ready, the resource can be automatically imported into the Jibiki platform. It is then instantly accessible for lookup and editing.

5.3 ONLINE LOOKUP AND EDITING

5.3.1 Lookup interfaces

Three different interfaces are available to the user:

- the generic lookup allows the user to look up a word or a prefix of a word in all the dictionaries available on the platform. The language of the word must be specified.

- the volume lookup allows the user to look up a word or prefix on a specific volume. In the left hand part of the result window, the volume headwords are displayed, sorted in lexicographical (alphabetical) order. An infinite scroll allows the user to browse the entire volume. In the right hand part of the window, the entries previously selected on the left are displayed.

- the advanced lookup is available for complex multi-criteria queries. For example, it is possible to look up an entry with a specific part-of-speech, and created by a specific author. On the left of the result window, the headwords of the matching entries are displayed, sorted in alphabetical order. A scroll bar allows the user to browse all the matching entries. On the right, the entries previously selected on the left are displayed.

5.3.2 Editing process

The editing module (Mangeot et al., 2004) is based on an HTML interface model instantiated with the lexical entry to be published. The model is generated automatically from an XML schema describing the entry structure. It can then be modified to improve the rendering on the screen. Therefore, it is possible to edit any type of dictionary entry provided that it is encoded in XML.

We would like to stress here the fact that although Jibiki provides all the tools necessary for a correction/revision/validation process of the data online, it is completely illusory to imagine that the quality of a dictionary will be improved by allowing potential voluntary contributors to act alone.

First, a real project must be defined with a community animator, a group of editors and validators must be selected based on their skills, and especially contributors must be motivated.

Second, the success of Wikipedia might lead us to think that the same can be obtained for the construction of a quality dictionary, but various experiences have shown us that it is not the case.

We also quote Larry Sender, founder of Wikipedia on the subject:

"To try to develop a dictionary by collaboration among random Internet users, particularly in a completely uncontrolled wiki format, now strikes me as a nonstarter."

Each Wikipedia article can be written by a specialist in his or her field, but for a general dictionary, it is not possible to find a specialist for some articles only. Only linguists who are specialists in the language as a whole can really help (after being trained in lexicography). Moreover, in the case of under-resourced languages, linguists who are specialists in these languages are few and far between. They are often very busy and cannot work on a project if not financed.

5.3.3 Remote access via an API

Once dictionaries are uploaded onto the Jibiki server, they can be accessed via a REST API. Lookup commands are available for querying indexed information: headword, pronunciation, part-of-speech, domain, example, idiom, translation, etc. The API can also be used for editing entries. The user must be previously registered on the website.

5.4 THE DILAF METHODOLOGY AND WEBSITE

Until now, we have focussed this article on the conversion methodology, but the DiLAF methodology is not limited to these technical subjects. It also includes the constitution of some documentation concerning each dictionary (*a minima*: origin of the dictionary, alphabet, parts of speech list, markup list) and the chosen licence. This documentation and the publication of our methodology is central to the scientific quality of the dictionaries.

We have applied our methodology to several dictionaries. It appears that the best results are obtained when a linguist and a computer scientist cooperate closely. This multidisciplinary work leads to scientific questioning and discussions and is also a good opportunity to detect inconsistencies that may occur in dictionaries. With such a team a dictionary can be converted and documented within one month of work. We stress that this methodology has been successful in converting entire dictionaries.

For the moment, the DiLAF website available at dilaf.org presents five dictionaries and their documentation: Bambara, Hausa, Kanuri, Tamajaq, Zarma. These dictionaries were downloaded 260 times between January 2014 and September 2015. Two additional ones are in progress (Wolof and Fulfulde).

The website is based on the Jibiki platform. People can have a look at the dictionaries or download them (in central format). It has been designed in accordance with our African partners: the interface is simple, free and designed to be used without any technical skills. We can see in Figure 4 that the website displays all the entries of the dictionary on the left hand side in order to incite the user to discover words s/he does not know, as a person usually does when leafing through a paper dictionary.

Figure 4: The entry "bannadu" on the DiLAF website

6 Conclusion

Faced with the lack of online, free, and good quality resources for under-resourced languages, we searched for an approach to help to transform the vicious circle caused by this shortage into a virtuous one.

We found that some good quality dictionaries are produced by local linguists or lexicographers, but that this knowledge usually remains unavailable online. We decided to carry out some research that could meet two objectives. First, we developed a methodology to transform a good quality dictionary (linguistically speaking) into an online resource that could be useful to both NLP specialists and populations. This methodology was designed to use only free tools and limited knowledge and to be off line. It is written simply (in French), in order to achieve the second goal: to make it possible for non-NLP specialists to put new dictionaries

online. In addition, by following this methodology, people would learn new pieces of knowledge they are not familiar with: using regular expressions, designing a DTD for an XML document, checking and correcting some non Unicode characters, etc.

In a context of poverty where speakers often have never seen a dictionary of their language, but where access to the Web is improving, the benefits for the population are considerable. Furthermore, the visibility of the results is an additional motivation for those involved in the conversion process.

The converted dictionaries are incomplete, and corrections are needed. However, they are a stepping stone to other developments: bilingual corpus construction; development of morphological analyzers; etc. The availability of these resources can motivate new researchers, and thus increase the potential for research.

Acknowledgements

The MotÀMot project that produced the French-Khmer dictionary has been partly funded by the Agence Universitaire de la Francophonie. We thank especially Vanra Ieng.

The DiLAF project that produced the Bambara-French, Hausa-French, Kanuri-French, Tamajaq-French and Zarma-French dictionaries has been funded by the Fonds Francophone des Inforoutes of the International Organization of Francophonie. We thank especially Soumana Kané, Issouf Modi, Michel Maï Moussa Maï, Mahamou Raji Adamou, Rakiatou Rabé, Mamadou Lamine Sanogo.

The ALFFA project that produced the Fulfulde-French dictionary is funded by the French National Agency for Research (ANR). We thank especially Mariam Barry and Alicia Boucard.

Appendix

Appendix 1: Kanuri entry *bannadu (2)* in copy format

```
<article>
    <kalma lamba="2">bannadu</kalma>                    lexical entry number 2
```

```
    <bowodu>[bànnàḍú]</bowodu>                                          phonetic
    <naptu_curo_nahauyen>kkye3.</naptu_curo_nahauyen>                   part of speech
    <maana>Diwiro yal alamdu.</maana>                                   definition
    <misal>                                                             example
        <version tələm="ka">Gənanjun bannaje, ku tadanju rakce          example in Kanuri
kələnju rojiwawo.</version>
        <version tələm="fa">Durant son jeune âge il l'a mal             equivalent of the
éduqué, aujourd'hui son fils n'arrive pas à se prendre en               example in French
charge.</version>
    </misal>
    <maana_tiloa>lànḍú</maana_tiloa>                                    synonym
    <kalakta tələm="fa">éduquer (mal)</kalakta>                         equivalent in French
</article>
```

Informative elements in green were added during conversion from published format.

Appendix 2: Kanuri entry *bannadu (2)* in central format

```
<article id="bannadu2">                                                 article with identifier
    <bloc-vedette>
        <kalma lamba="2">bannadu</kalma>                                lexical entry number 2
        <bowodu>bànnàḍú</bowodu>                                        phonetic
    </bloc-vedette>
    <naptu_curo_nahauyen>kkye3.</naptu_curo_nahauyen>                   part of speech
    <bloc-semantique  id="bannadu2.1">
        <kalakta tələm="fra">éduquer(mal)</kalakta>                     equivalent in French
        <maana>Diwiro yal alamdu.</maana>                               definition
        <misal>                                                         example
          <version tələm="kau">Gənanjun bannaje, ku tadanju             example in Kanuri
rakce kələnju rojiwawo.</version>
          <version tələm="fra">Durant son jeune âge il l'a mal éduqué,  equivalent of the
aujourd'hui son fils n'arrive pas à se prendre en charge.</version>     example in French
        </misal>
        <maana_tiloa>lànḍú</maana_tiloa>
    </bloc-semantique>                                                  synonym
</article>
```

Structuring elements in red were added during conversion from copy format.

Appendix 3: Kanuri entry *bannadu (2)* in LMF syntax

```
<LexicalEntry id="bannadu2">                                            article with identifier
    <Lemma>
        <feat att="writtenForm" val="bannadu"/>                         written form
        <feat att="phoneticForm" val="bànnàḍú"/>                        phonetic
    </Lemma>
    <feat att="partOfSpeech" val="kkye3."/>                             part of speech
    <Sense id="1">
        <Equivalent>
            <feat att="language" val="fra"/>                            equivalent in French
```

```
            <feat att="writtenForm" val="éduquer(mal)"/>
        </Equivalent>
        <Definition>
            <feat att="writtenForm" val="Diwiro yal alamdu."/>          definition
        </Definition>
        <Context>                                                       example
            <TextRepresentation>
                <feat att="language" val="kau"/>                        example in Kanuri
                <feat att="writtenForm" val="Gənanjun bannaje,
ku tadanju rakce kəlanju rojiwawo."/></TextRepresentation>
            <TextRepresentation>
                <feat att="language" val="fra"/>
                <feat att="writtenForm" val="Durant son jeune âge il    equivalent of the
l'a mal éduqué, aujourd'hui son fils n'arrive pas à se prendre en charge."/>  example in French
            </TextRepresentation>
        </Context>
        <SenseRelation targets="l̀ànd́ú">
            <feat att="type" val="synonym"/>
        </SenseRelation>
    </Sense>                                                            synonymous
</LexicalEntry>
```

Notes

References

Bailleul, C. (1996). *Dictionnaire bambara-français*.

Berment V. (2004). *Méthodes pour informatiser des langues et des groupes de langues « peu dotées »*. Thèse de nouveau doctorat, spécialité informatique, Université Joseph Fourier Grenoble I, Grenoble, France.

Buseman A., Buseman K., Jordan D., Coward D. (2000). *The linguist's shoebox: tutorial and user's guide: integrated data management and analysis for the field linguist*, volume viii. Waxhaw, North Carolina: SIL International.

Calvet, L.-J. (1987). *La guerre des langues*. Paris, Payot.

Calvet, L.-J. (1996). *Les politiques linguistiques*. Paris, PUF.

Chalvin A., Mangeot M. (2006) Méthodes et outils pour la lexicographie bilingue en ligne : le cas du Grand Dictionnaire Estonien-Français. *Proceedings of the EURALEX 2006 conference*, Torino, Italy, 6-9 September, 6 p.

Chanard, C. Popescu-Belis, A. (2001). Encodage informatique multilingue : application au contexte du Niger. *Cahiers du Rifal* (cont. Terminologies Nouvelles), n. 22, pp 33-45.

Cissé, M.T. (2013) Le dictionnaire électronique unilingue wolof et bilingue wolof-français : une création continuée, Repères DoRiF n.3 - Projets de recherche sur le multi / plurilinguisme et alentours..., septembre 2013
http://www.dorif.it/ezine/ezine_articles.php?id=127

Diki-Kidiri M. (2004). Multilinguisme et politiques linguistiques en Afrique. *Colloque Développement durable, leçons et perspectives,* Ouagadougou, Burkina-Faso.

Eckle-Kohler J., McCrae J. P., Chiarcos C. (2014). lemonUby – a large, interlinked syntactically-rich lexical resource for ontologies. *Semantic Web Journal*.

Enguehard C. (2009). Les langues d'Afrique de l'ouest : de l'imprimante au traitement automatique des langues. *Sciences et Techniques du Langage*, Vol. 6, pp 29–50.

Enguehard C., Mangeot M. (2013) LMF for a selection of African Languages. In G. Francopoulo (ed.) *LMF: Lexical Markup Framework*, Hermès science, Paris.

Enguehard C., Mangeot M. (2014) Computerization of African languages-French dictionaries *Proceedings of Collaboration and Computing for Under Resourced Languages in the Linked Open Data Era (CCURL), LREC 2014 workshop*, Reykjavik, Island, 27 May 2014.

Francopoulo G., Bel N., George M., Calzolari N., Monachini M., Pet M., Soria C. (2009). Multilingual resources for NLP in the Lexical Markup Framework (LMF). *Language Resources and Evaluation,* Vol. 43, pp. 57–70. ISBN: 10.1007/s10579-008-9077-5.

Mangeot, M., Sérasset, G., Lafourcade, M. (2003) Construction collaborative de données lexicales multilingues, le projet Papillon. (Papillon, a project for collaborative building of multilingual lexical resources). In M. Zock and J. Carroll (eds). *Special issue of the TAL journal: Electronic dictionaries: for humans, machines or both?*, Vol. 44:2/2003, pp. 151-176. 2003.

Mangeot, M., Thevenin, D. (2004). Online Generic Editing of Heterogeneous Dictionary Entries in Papillon Project. *Proceedings of the COLING 2004 conference*, ISSCO, Genève, Switzerland, 23-27 August, vol 2/2, pp 1029-1035.

Mangeot M., Chalvin A. (2006). Dictionary building with the Jibiki platform: the GDEF case. *Proceedings of the Language Resources and Evaluation Conference 2006 (LREC)*, p. 1666–1669, Genova, Italy. Available from European Language Resources Association, Paris.

Mangeot M. (2006). Dictionary Building with the Jibiki Platform. Software Demonstration, *Proceedings of the EURALEX 2006 conference*, Torino, Italy, 6-9 September 2006, 5 p.

Mangeot M., Enguehard C. (2013). Des dictionnaires éditoriaux aux représentations XML standardisées. In N. Gala & M. Zock (eds.), *Ressources Lexicales : contenu, construction, utilisation, évaluation,* Linguisticae Investigationes Supplementa, John Benjamins Publishing, Amsterdam, Pays-Bas, 24 p.

Mangeot M. (2014). MotàMot project: conversion of a French-Khmer published dictionary for building a multilingual lexical system. *Proceedings of the Language Resources and Evaluation Conference 2014 (LREC)*, Reykjavik, Island, 28-30 May 2014 (to appear). Available from European Language Resources Association, Paris.

Mangeot M. (2016) Collaborative Construction of a Good Quality, Broad Coverage and Copyright Free Japanese-French Dictionary. *International Journal of Lexicography,* Volume 31, Issue 1, 1 March 2018, Pages 78–112; doi: 10.1093/ijl/ecw035; 35 p.

Mijinguini, A. (2003). *Dictionnaire élémentaire hausa-français*.

Nguyen H-T., Boitet Ch., Sérasset G. (2007) *PIVAX, an online contributive lexical database for heterogeneous MT systems using a lexical pivot*. Proc of SNLP 2007, December 2007, Pattaya, Thailand. pp.337-342.

Osborn, D. (2011). *Les langues africaines à l'ère du numérique*. Laval, Canada: Presses de l'Université Laval.

Oumaru, I. A. (1997). Zarma ciine - kaamuusu kayna. Editions Alpha.

Polguère A. (2008) *Lexicologie et s émantique lexicale. Notions fondamentales*. Paramètres, 304 pages. Les Presses de l'Universit é de Montréal, Montréal, 2e édition. Nouvelle édition revue et augmentée.

Programme Décennal du Développement de l'Éducation (PDDE), Niger. (2003). Enseignement bilingue, pages 121-132.

Programme de soutien à l'éducation de base (Soutéba). (2004). *Dictionnaire kanouri-français destiné pour le cycle de base 1,* Niamey, Niger.

Programme de soutien à l'éducation de base (Soutéba). (2007). *Dictionnaire tamajaq-français destiné à l'enseignement du cycle de base 1*, Niamey, Niger.

République du Niger (1999a). *Alphabet haoussa*, arrêté 212-99.

République du Niger (1999b). *Alphabet kanouri*, arrêté 213-99.

République du Niger (1999c). *Alphabet tamajaq*, arrêté 214-99.

République du Niger (1999d). *Alphabet zarma*, arrêté 215-99.

Richer, D., Keo, T., Vanra, I. (2007). *Dictionnaire Français-Khmer (en phonétique)*, D.R. Edition, ISBN-13:890-0-9.

Romary L., Salmon-Alt S., Francopoulo G. (2004). Standards going concrete: from LMF to Morphalou. *Proceedings of the COLING Workshop on Enhancing and Using Electronic Dictionaries, ElectricDict '04*, p. 22–28, Stroudsburg, PA, USA: Association for Computational Linguistics.

Sérasset, G. (2014) Dbnary: Wiktionary as a Lemon Based RDF Multilingual Lexical Resource. *Semantic Web Journal - Special issue on Multilingual Linked Open Data*, 2014. (to appear).

Streiter O., Scannell K., Stuflesser M. (2006). Implementing NLP projects for non-central languages: Instructions for funding bodies, strategies for developers. *In Machine Translation, volume 20*.

Zhang Y., Mangeot, M. (2013) Gestion des terminologies riches : L'exemple des acronymes Procs of TALN-RECITAL 2013, Les Sables-d'Olonne, France, 17-21 june 2013, 6 p.

Zhang Y., Mangeot M., Bellynck V., Boitet C. (2014). Jibiki-LINKS: a tool between traditional dictionaries and lexical networks for modeling lexical resources. *Proceedings of the 4th Workshop on Cognitive Aspects of the Lexicon (CogALex)*, Aug 2014, Dublin, Ireland.

[1] http://jibiki.fr/

[2] http://jibiki.univ-savoie.fr/motamot/

[3] http://www.dilaf.org/

[4] http://www.sil.org/computing/toolbox/

[5] http://fieldworks.sil.org/flex/

[6] http://tshwanedje.com/tshwanelex/

[7] http://www.papillon-dictionary.org/

[8] http://www.isocat.org/rest/dcs/119.html

[9] http://www.lemon-model.net

[10] http://kaiko.getalp.org/about-dbnary/

[11] http://www.estfra.ee/

[12] http://jibiki.fr

[13] http://www.getalp.org/pivax/

[14] http://github.com/mangeot/jibiki

[15] https://hub.docker.com/r/mangeot/jibiki/

Exploitation d'une base lexicale dans le cadre de la conception de l'ENPA Innovalangues

Mathieu Mangeot[1,4] Valérie Bellynck[1] Emmanuelle Eggers[4]
Mathieu Loiseau[2,4] Yoann Goudin[3,4]
(1) LIG, Bâtiment IMAG, 700 avenue Centrale - Domaine Universitaire, 38400 Saint-Martin-d'Hères, France
(2) LIDILEM, Université Grenoble Alpes, CS40700, 38058 Grenoble cedex 9
(3) CERLOM, INALCO, 2 rue de Lille, 75007 Paris
(4) IDEFI Innovalangues, Maison des Langues et des Cultures, 1141 avenue Centrale - Domaine Universitaire, 38400 Saint-Martin-d'Hères, France
valerie.bellynck@imag.fr, emmanuelle.eggers@univ-grenoble-alpes.fr, mathieu.loiseau@univ-grenoble-alpes.fr, mathieu.mangeot@imag.fr, yoann.goudin@univ-grenoble-alpes.fr

RÉSUMÉ

Le projet Innovalangues a pour but la conception d'un environnement numérique personnalisé d'apprentissage des langues (ENPA). Dans ce cadre, plusieurs modules tels des jeux sérieux ou des générateurs d'exercices ont besoin d'une base lexicale. Il est intéressant également de donner la possibilité aux apprenants de gérer leurs propres lexiques. Pour éviter que chacun ne développe sa propre base lexicale, il apparaît rapidement indispensable de développer une seule base lexicale commune qui puisse servir aux modules (machines) comme aux apprenants (humains). Après une analyse des besoins autour de scénarios d'utilisation, nous proposons une architecture de base lexicale multilingue. Nous avons ensuite réalisé un prototype fonctionnel qui s'intègre à l'ENPA et permet à un apprenant de consulter des ressources lexicales existantes et de créer son propre lexique. Le prototype LexInnova utilise la plate-forme Jibiki de gestion de ressources lexicales hétérogènes à distance via son interface de programmation (API) REST.

ABSTRACT

Operating a lexical database in the framework of the Innovalangues language learning platform

The Innovalangues project aims to design a personalized digital environment for language learning. In this context, several modules such as serious games or exercise generators need a lexical database. It is also interesting to provide solutions for learners to manage their own lexicons. To prevent each one to develop its own lexical database, it quickly appears essential to develop one common lexical database that can be used for modules (machines) and learners (humans). After an analysis around usage scenarios, we set up a multilingual lexical database architecture. We then built a working prototype that integrates with the environment and allows a learner to look up existing lexical resources and create its own lexicon. The LexInnova prototype uses the Jibiki platform for managing heterogeneous lexical resources via its REST application programming interface.

MOTS-CLÉS : Innovalangues, base lexicale, dictionnaire, lexique, Jibiki, LexInnova
KEYWORDS : Innovalangues, lexical database, dictionary, lexicon, Jibiki, LexInnova

1 Introduction

L'un des principaux objectifs du projet IDEFI Innovalangues est de mettre à disposition un environnement numérique personnalisé d'apprentissage des langues (ENPA). Outre les parcours d'apprentissage, l'écosystème numérique proposé par Innovalangues fournira aux apprenants des outils pour soutenir l'apprentissage : des jeux, des générateurs d'exercices, un chat, un test de positionnement (test automatique adaptatif pour l'orientation de l'apprenant vers un niveau cible en fin de formation), etc.

Le projet est divisé en plusieurs lots (donnant lieu à des livrables du projet) dont l'objectif est de produire des outils répondant à des problématiques didactiques et liés ou intégrés à l'ENPA.

L'une des tâches fondamentales de l'apprentissage des langues est de travailler le lexique[1]. Dès lors, il est rapidement apparu que la réalisation des divers lots (outils) conduisait au développement de d'autant de bases lexicales avec des contraintes spécifiques pour chacune, sans qu'il n'y ait de lien pratique entre eux. De plus, les apprenants peuvent souhaiter également créer leurs propres lexiques d'apprentissage. Cette situation a rapidement posé un problème de mise en œuvre informatique (opérationnalisation), ce qui a mené à la définition d'un sous projet transversal au projet Innovalangues (« chantier interlot »), que nous avons appelé LexInnova.

Nous présentons d'abord le projet Innovalangues et les besoins les plus emblématiques en termes de ressources lexicales. La partie suivante concerne le cahier des charges de l'outil que nous souhaitons construire ainsi que les différents scénarios d'utilisation des ressources lexicales. Ensuite, nous spécifions la base lexicale multilingue visée. Enfin, dans la dernière partie, nous détaillons le fonctionnement du prototype fonctionnel que nous avons implémenté afin de montrer les possibilités qu'offre l'intégration un tel outil.

2 Présentation du projet Innovalangues

Le projet IDEFI Innovalangues, initié en 2012, est piloté par le service LANSAD de l'Université Grenoble-Alpes. Il tire son origine des difficultés des étudiants à atteindre un niveau B2 du Cadre Européen Commun de Référence pour les Langues (CECRL) (CECRL, 2000) alors qu'il s'agit du niveau cible pour le baccalauréat[2] (Masperi et Quintin, 2014a : 62) . À partir de ce constat, le projet a pour but de fournir aux établissements supérieurs des moyens (dispositifs, méthodes, contenus) pour porter le degré de maîtrise en langues des étudiants à un niveau B2 certifié. Pour y parvenir, Innovalangues s'est fixé trois objectifs principaux :

- « se doter d'un environnement (numérique) personnalisé d'apprentissage (ENPA), “open source”, [proposant des contenus sous licence libre], au service de la collaboration, particulièrement flexible et hautement adaptable aux différents contextes rencontrés sur le terrain de la formation » (Masperi et Quintin, 2014b : 8) ;
- proposer un espace dédié aux enseignants et ainsi « capitaliser les résultats de la recherche en didactique des langues » (Masperi et Quintin, 2014a : 72) par des actions de formation et / ou via l'outillage techno-pédagogique ;
- créer une communauté de pratiques pour « étendre la dynamique de recherche-action en langues entreprise au niveau local » (Masperi et Quintin, 2014a : 72).

Le projet se déroule principalement en deux phases. La première phase est orientée conception/implémentation. Elle est prise en charge au sein de « lots » impliquant chercheurs en didactique, enseignants, concepteurs, développeurs. Les lots sont les suivants :

[1] Voir par exemple (Luste-Chaa, 2009 : 2.4) pour une présentation historique de la place du lexique en FLE.

[2] http://www.education.gouv.fr/cid206/les-langues-vivantes-etrangeres.html#Au_lycée___général,_technologique_et_professionnel

- SELF (Système d'Évaluation en Langues à visée Formative) vise à mettre en œuvre une méthodologie de création de tests de positionnement et à fournir la plate-forme permettant de les accueillir pour six langues (anglais, italien, mandarin, japonais, espagnol, français langue étrangère).
- Innovason regroupe les activités issues de :
    - THEMPPO (Thématique Prosodie & Production Orale) qui visait à proposer des outils et des pratiques centrées sur le développement des compétences prosodiques de la production orale en langue étrangère.
    - et COCA (Compétence Orale : Conception et Assistance) qui se fixait pour objectif de développer des solutions technologiques pour assister l'enseignant dans la conception d'activités de compréhension de l'oral.
- PARCOURS est centré sur le développement d'un module de l'ENPA (plate-forme fondée sur Claroline Connect) destiné à accueillir les parcours et réalisations des autres lots, mis à disposition de l'ensemble des acteurs de la formation.
- GAMER (Gaming Application for Multilingual Educational Resources) a pour objectif premier de concevoir et développer des jeux pour l'enseignement/apprentissage des langues, mais aussi de proposer des formations de formateurs sur l'usage du jeu en classe de langues.

Dans une seconde phase, les « équipes langues » (anglais, espagnol, italien, japonais, mandarin) s'approprient les outils conceptuels et technologiques conçus par les lots pour développer des contenus de formation. Pour organiser les développements autant que pour laisser mûrir les réflexions, les changements de phases ne sont pas complètement synchronisés (la seconde phase du lot SELF ne commence pas en même temps que celle du lot GAMER) et les phases ne sont pas hermétiques : équipes « lots » et « langues » collaborent durant les deux phases. Ces collaborations peuvent faire émerger un besoin différent et un chantier est alors créé pour explorer ce besoin. LexInnova s'inscrit dans ce type de travaux : il s'agit d'un chantier émergent visant à proposer des solutions concernant des enjeux partagés par plusieurs lots. L'équipe responsable de ce chantier est constituée d'enseignants de langue, de didacticiens et d'informaticiens spécialistes du TAL, tous issus de différentes équipes, d'Innovalangues et d'ailleurs.

Au sein du projet Innovalangues, les attentes en termes d'outillage du lexique s'expriment différemment : pour certains, les besoins principaux portent sur le contenu informationnel (voir par exemple Innovason, pour qui la transcription phonétique des mots du lexique semble primordiale), alors que d'autres utilisateurs potentiels sont plus tournés vers les fonctionnalités offerte par le système. Par exemple, pour le chantier Kinéphones, l'une des demandes concernait « l'écriture en couleurs » : une représentation différente des transcriptions phonétiques alignées sur la graphie où chaque phonème est associé à une couleur sur le modèle affranchi des contraintes éditoriales matérielles des tableaux de mots de l'approche Silent Way de Caleb Gattegno dématérialisé par Kinéphones[3].

3 Recueil des besoins

Après avoir recueilli les informations linguistiques pertinentes pour les collègues du projet par le biais d'une page du wiki interne d'Innovalangues, nous avons décidé, dans une perspective de « techniques » de conception centrées utilisateur (Bastien et Scapin, 2004 : 461), d'élaborer conjointement des scénarios d'utilisation visant à préciser les besoins fonctionnels. Ceux-ci s'appuient sur des cas pratiques et peuvent intégrer des éléments biographiques exemplifiant les interactions entre langue « maternelle » et langue « cible ».

[3] http://kinephones.u-grenoble3.fr/#/en_gb/1/table

3.1 Espagnol

Dans l'exemple de scénario suivant, légèrement retravaillé pour que le lecteur puisse en comprendre les enjeux, nous tentons de montrer comment la structure des lexiques et les besoins fonctionnels peuvent être intégrés à un scénario.

Anne étudie l'espagnol débutant depuis deux mois en LANSAD. Son enseignant a choisi de faire créer un lexique de groupe à tour de rôle par des étudiants en tandem. Pour ce faire, cette semaine, Anne et son partenaire travaillent sur l'ENPA. Ils relisent les notes prises durant le cours à la lumière des ressources proposées par l'enseignant et choisissent les mots qu'ils veulent inclure dans le lexique de groupe (d'un simple clic, quand la ressource est intégrée à l'ENPA). Dans certains cas, ils ajoutent à la notice de l'entrée un commentaire qui a été fait pendant la classe (ex : « Attention, en espagnol, '*padres*' signifie non seulement '*pères*', mais aussi '*parents*', ex : "*Llamo a mis padres cada semana*." ; (voir aussi hermanos, tíos) » dans l'article '*padre*').

Ils profitent également de ce travail à destination du groupe pour mettre à jour leur propre lexique et cliquent sur les mots qu'ils souhaitent rajouter dans leur lexique respectif (lexique personnel qui alimente le lexique de groupe et *vice-versa*). Anne souhaite par exemple s'approprier le mot « tío » (oncle). Elle clique sur ce mot, un menu déroulant s'ouvre. Elle peut alors consulter la ou les définitions ainsi que les exemples correspondant à ce mot (structure), puis importer ces données telles quelles ou en les modifiant. Anne décide par exemple de modifier l'exemple associé à « tío » pour son lexique personnel ; son oncle s'appelant Bernard, pour retenir « tío » elle veut l'associer à « Bernard ». Elle va donc choisir de mettre comme exemple personnel « Mi tío se llama Bernard » (besoin fonctionnel). Anne va également intégrer dans son lexique personnel des mots pris en notes durant le cours et durant l'écoute des enregistrements audio mis à disposition dans le cadre de sa formation. Anne, pour réviser le lexique de groupe et son lexique personnel, peut lancer des générateurs d'exercices (lien avec l'ENPA). La semaine suivante, un autre tandem se chargera du lexique de la semaine.

Eunice, bilingue français-portugais, est inscrite dans le même cours. Elle va consulter les mots nouvellement arrivés dans le lexique de groupe et faire des exercices à partir de ces mots (besoin fonctionnel). Elle choisira de rajouter une information spécifique dans son lexique personnel « Attention : accent sur le "i" de "tío" » (contrairement au portugais).

3.2 Mandarin

Les scénarios pour un lexique aussi distant que celui du mandarin ne diffèrent fondamentalement pas de ceux conçus pour les autres langues. En revanche, il existe un besoin impérieux que la prise en charge du lexique puisse au-delà — ou plus précisément en-deçà — de l'entrée dans le lexique, traiter un certain nombre d'éléments tels que les sinogrammes dont chaque entrée est constituée. En effet, en sus de la compétence lexicale au cœur de notre démarche, la compétence graphique est également traitée à travers des informations relatives à la combinatoire des sinogrammes, et à un niveau encore inférieur incluant les composants graphiques, les traits ou encore les différentes lectures sino-xéniques des sinogrammes dont sont composées les entrées du lexique.

Ainsi, Pierre, apprenant en mandarin, pourra accomplir les mêmes opérations que celles décrites plus haut.家 *jiā* – « famille », « maison » – vu en classe, figurera parmi les entrées de son lexique personnel. Plus tard, au détour d'une ressource en compréhension écrite, 國 家 *guojiā*, « pays » portera l'information complémentaire selon laquelle cette entrée est programmée au cours de la formation, le système étant capable d'intégrer le lexique des unités à venir – le *lexique cible* – incitant ainsi les apprenants à s'y intéresser au plus tôt.

L'apprenant pourra également confronter les sinogrammes de 國 家 *guójiā* à d'autres lexiques compilés par des institutions, qui associent à un niveau donné une liste de sinogrammes choisis selon différents principes. En effet, le paradigme didactique dominant du mandarin langue étrangère établit le sinogramme comme l'unité didactique fondamentale (Bellassen, 2010) induisant une chronopédagogie qui programme l'enseignement-apprentissage des sinogrammes selon un double

critère de fréquence d'un sinogramme dans le corpus et de la haute combinabilité de ce même sinogramme dans le lexique contemporain. Le corollaire d'une telle pratique fige des listes de sinogrammes par niveau qui constituent le socle de l'industrie certificative. Si l'approche didactique retenue pour Innovalangues veut rompre avec ce paradigme, nous ne pouvons pas exclure ces listes que nous étiquetons dans le système comme le *lexique institutionnel* informant l'utilisateur des coordonnées de chaque sinogramme dans différents contextes institutionnels : programmes de langue vivante de l'Éducation nationale, niveaux du test certificatif de chinois HSK etc.

Plus spécifique aux langues sinogrammiques en se distinguant du paradigme évoqué ci-dessus, LexInnova viserait à offrir une vue sur le lexique qui indiquerait les entrées qui partagent un même sinogramme. Ainsi, avec notre exemple, 家 apparaît également dans les entrées 作 家 *zuòjiā* « écrivain », 學 家 *xuéjiā* « universitaire », 文 學 家 *wénxuéjiā* « spécialiste de littérature » etc. ou encore 家 庭 *jiātíng* « famille », 家 族 *jiāzú* « clan » etc. Cette vue sur le lexique permet ainsi de traiter les dimensions suivantes : la polysémie de 家, qui en plus de vouloir dire « famille » renvoie par ailleurs à la notion d'agent lorsque qu'il se situe en dernière position. Il est même indiqué que la position de 家 dans sa première combinatoire 國 家 *guójiā* « pays » est une exception : en règle générale, et largement plus représentée dans le corpus, 家 renvoie à l'idée d'agent même si le paradigme communicationnel le présentera avant tout comme signifiant « famille ».

Au-delà de la mise en relation des différentes entrées contenant un même sinogramme, idéalement le système permettrait d'accéder aux informations suivantes : 家 est un sinogramme dont la structure est binaire verticale ⿱, constitué des composants 宀 *miǎn* « le toit », et 豕 *shǐ* « le porc ». Il est même possible d'intégrer une notice grammatologique infirmant l'édifiante étymologie orientaliste selon laquelle, pour les « Chinois des temps anciens, la famille était représentée au moyen d'un cochon sous le toit » alors que depuis le IIe siècle EC, nous savons que 家 s'est d'abord noté 豭 au moyen du composant du 豕 « porc » et d'un autre composant phonétique – 叚 *jiǎ* – encore quasi-homophone vingt siècles plus tard. A la faveur d'un processus orthographique inévitable sur une période historique aussi longue, le composant phonétique est tombé au profit d'un *composant discriminant* connu également sous le terme de « clé » : celui du toit 宀.

Enfin, le sinogramme est recontextualisé avec les sinogrammes qui partagent le même composant phonétique ou *phonophore* tel que 嫁 *jià* « prendre pour mari » ; mais également ses homophones parfaits ou au ton près tel que 加 *jiā* « ajouter » ou 價 *jià* « valeur, prix », 假 *jià* «vacances » ou *jiǎ* « faux »– qui partage le même phonophore historique de 家 *jiā* – et les liens vers les entrées contenant ces sinogrammes. Finalement, 家 *jiā* est recontextualisé en intercompréhension avec les lectures sino-xéniques et d'autres langues sinitiques, *ka* – ou plus rarement *ke* – en sino-japonais, *ga* en sino-coréen, *gia* en sino-vietnamien, et même en français à travers l'ethnonyme *hakka* – 客 家 *kèjiā* – désignant la communauté culturelle des vallées de l'arrière-pays des provinces méridionales du Fujian et du Guangdong, très présente dans la diaspora et majoritaire dans les confettis insulaires de l'empire français[4].

Toutes ces fonctionnalités et vues sur le lexique permettent ainsi de sous-traiter un maximum d'informations à la plate-forme et ainsi consacrer le plus possible du cours en présentiel à l'interaction orale plutôt qu'aux explications de vocabulaire et autres digressions graphiques et culturelles qui rendent le mandarin une langue encore plus distante pour les apprenants. Par ailleurs, une telle prise en charge du lexique permet de modéliser le profil de l'apprenant pour la génération d'exercices ainsi que les jeux évoqués ci-dessus.

[4] Le même changement phonétique d'une occlusive vélaire à une palatale est observable en français à travers l'emprunt Pékin au XVIIe siècle devenu depuis Beijing...

3.3 Relations avec GAMER

Les deux exemples précédents montrent les liens entre l'ENPA et les lexiques dans deux langues. Le cas du mandarin illustre plus spécifiquement les enjeux liés à la forme écrite (les sinogrammes) des mots-forme, alors que le cas de l'espagnol, il s'agit de l'exploitation à partir des entrées lexicales (lemmes ou mots-vedettes). C'est ce type de lien avec l'ENPA qui est au cœur de la problématique du lexique dans le lot GAMER. Dans le cadre de ces travaux, plusieurs prototypes de jeux sont en cours de développement avec un point de vue différent sur le lexique. Par exemple, dans le cadre du jeu « Magic Word[5] », dont les règles le placent dans la famille de jeux de lettres que l'on pourrait exemplifier par le Boggle (Loiseau, Zampa et Rebourgeon, 2015), il est nécessaire d'avoir une ressource lexicale qui définisse le champ des possibles (liste des formes autorisées). Ce jeu est focalisé sur ce que Portine appelle le pôle *accuracy*, c'est-à-dire sur le respect des schémas lexicaux et grammaticaux (Portine, 2013 : 162). Chaque langue prise en charge devra fournir une telle ressource contenant non-seulement « toutes » les formes de la langue, mais également les traits morphologiques associés et le lemme dont la forme est issue : certaines règles des futures versions du jeu en dépendent. Mais un autre besoin est la possibilité de lier les lemmes (associés aux formes trouvées dans le jeu) à l'activité du joueur dans les parties plus formelles de l'ENPA. Dans ce cas-là les générateurs d'exercices mentionnés dans le scénario espagnol pourraient être lancés à partir de cette liste de mots (dans ce contexte, le lexique du joueur, constitué automatiquement d'après les traces d'interaction ferait office de profil de l'apprenant).

Un autre exemple de jeu, radicalement différent dans ses objectifs, est « Game of Words[6] » (Loiseau, Hallal et Ballot, 2016). Ce jeu de « devinettes » est lui tourné vers le pôle *fluency*, centré sur les activités discursives (Portine, 2013 : 162). Entre autres activités attendues du joueur, il s'agira ici d'effectuer un enregistrement audio permettant de faire deviner un mot aux autres joueurs en respectant des règles énonciatives (dans la première version, celles du jeu *Taboo*), ce qui demande d'avoir une ressource lexicale qui associe un mot à d'autres entrées utiles pour le définir (mots interdits). Dans le cadre de ce jeu, l'une des passerelles anticipées est la possibilité pour un joueur de privilégier les mots qui figurent déjà dans son dictionnaire personnel afin de lui fournir un autre point de vue sur celui-ci. Par exemple, « Anne » pourrait demander à jouer sur les mots de son lexique personnel.

Enfin, un dernier exemple de jeu et d'intégration spécifique serait celui de *Kanji Crunch*, un jeu d'association, dont il n'existe pas de prototype à l'heure actuelle contrairement aux deux autres. Celui-ci vise à permettre au joueur d'appréhender globalement l'économie du système graphique en mandarin (Goudin et Lê, 2016). Une fois les premiers niveaux joués et les compétences de base acquises (Chaix et al., 2016), un système comme LexInnova permettrait au joueur de se confronter à des niveaux construits à partir des sinogrammes issus de son lexique de groupe (cf. scénario espagnol et section 4.2.3). Mais, par le biais de cette question du sinogramme, le mandarin pose également des questions spécifiques quant à la structure du lexique.

Conclusion

À travers les exemples proposés ci-dessus, nous avons tenté de montrer d'une part la diversité des besoins en termes de structuration du lexique, mais également les ponts qui pourraient être dressés entre différents travaux menés indépendamment du fait des fonctionnalités nécessaires à la prise en compte de ces besoins. Ceux qui ont été présentés ici sont avant tout monolingues, mais d'autres (voir par exemple le projet Check Your SMILE (Yassine-Diab, Alazard-Guiu et Loiseau, 2016)), présentent également des besoins de traduction.

Au-delà des exemples ci-dessus, la plupart des équipes du projet Innovalangues développe des contenus dans plusieurs langues, ce qui nécessitent tout un outillage pour la gestion du lexique. Dans la suite de l'article, nous présenterons nos propositions pour un tel outil.

[5] http://gamer.innovalangues.net/magicword 300
[6] http://gamer.innovalangues.net/gameofwords

4 Conception d'une base lexicale multilingue

Bien que la gestion des développements et déploiement de l'ENPA, et donc la coordination des développements informatiques, soit centralisée, les produits issus de chaque lot sont conçus et développés indépendamment, le lien se faisant par l'intégration à l'ENPA. Mais les ressources linguistiques comme les lexiques et les corpus, d'une très fine granularité, sont structurés de façon complexe et très spécifique, et leurs tailles posent le problème du passage à l'échelle, qui doit être traité au niveau de chaque chantier.

Le projet Innovalangues est multilingue, puisqu'il permet aux élèves de s'inscrire et gérer leur formation dans plusieurs langues. Chaque langue peut sembler apprise indépendamment quand on ne considère que l'inscription d'un élève à un parcours. Mais le profil de l'élève et son histoire personnelle révèlent les spécificités (caractéristiques) multilingues de son rapport aux ressources linguistiques. De plus, les travaux en intercompréhension menés de longue date au LIDILEM (Carlo et al., 2015 ; Dabène, 1994) et intégrés aux formations du LANSAD[7] soulignent à nouveau les besoins de ressources lexicales multilingues centralisées.

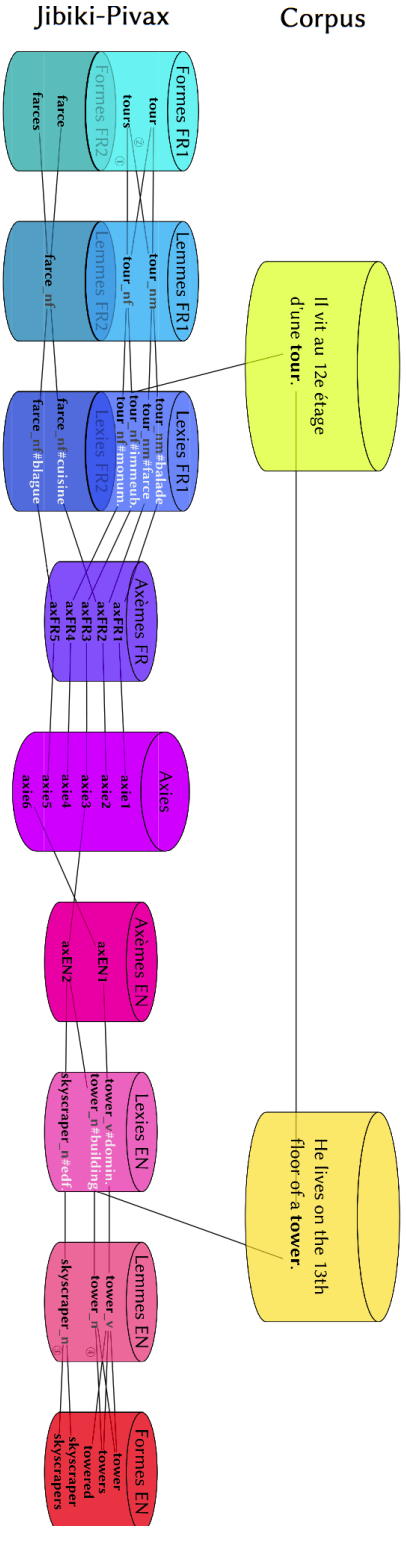
4.1 Macrostructure étendue

Les lots Innovason et GAMER exploitent tous un ou des lexique(s) réalisé(s) dans une base de données dédiée interne au lot. La diversité des besoins aurait demandé un long travail de coordination qui aurait trop ralenti chacun des chantiers, mais à l'heure où la question d'un outil centralisé pour la gestion du lexique se pose, la collecte des microstructures de chaque lexique (formes fléchies, lemmes et traits morphologiques associés[8], transcription phonétique[9], définitions L1 et L2[10] , [11], syllabes, accentuation[12], sinogrammes[13], exemples en contexte[14], etc.) a conduit à concevoir une structure de base lexicale multivolumes (cf. figure 5).

Les besoins d'accès par la forme fléchie conduisent à l'identification d'un volume pour celles-ci. L'intégration des transcriptions phonétiques doit être reliée aux formes fléchies. Cela conduit aussi à l'identification d'un volume dédié.

[7] http ://lansad.u-grenoble3.fr/version-francaise/formations-en-langues/intercomprehension-en-langues-romanes-77120.kjsp

[8] cf. Magic Word
[9] cf. InnovaSon
[10] cf. le scénario espagnol ci-dessus
[11] cf. GAMER
[12] cf. InnovaSon
[13] cf. scénario mandarin ci-dessus
[14] cf. le scénario espagnol ci-dessus



Prenons l'exemple de la graphie "tour" en français. Sa transcription phonétique sera /tur/. La forme considérée ("tour") peut-être un nom masculin, "tour_nm", ou un nom féminin, "tour_nf", gérés dans le volume dédié aux lemmes. Mais ces deux lemmes peuvent avoir des sens différents, que l'on peut préciser avec une glose, comme :

1. tour_nm#potier, tour_nm#balade, tour_nm#perimètre, tour_nm#magic, tour_nm#farce,

2. tour_nf#contrôle, tour_nf#PC, tour_nf#immeuble, tour_nf#monument.

Ces différents sens sont portés par les lexies, qui sont gérées par un volume dédié. Une lexie est donc liée à une forme pour porter un sens précis.

Il est possible qu'un autre lemme conduise au même sens. Sa lexie sera différente mais le sens "unique". Les sens partagés par plusieurs lexies sont les axèmes. Un volume dédié les stocke. Disons que dans notre exemple, "tour_nm#immeuble" soit identifié par l'axème "id-axm-3856900".

Une fois le sens monolingue ainsi précisé, pour passer au même sens dans les autres langues (et pas seulement une autre langue), on définit les axies. Formellement, une axie est une classe d'équivalence de lexies synonymes (Boitet, Mangeot, Sérasset, 2002).

Si notre axème "id-axm-3856900" est relié à l'axie "id-axi-98663940", la liaison de cette axie avec les axèmes de l'anglais, gérés dans le volume des axèmes de l'anglais, pourrait conduire à identifier l'axème "id-axm-1001123", qui est lié à la lexie "tower_n#building" mais aussi à la lexie "tower_n#skyscraper". Ces lexies conduisent à deux gloses : les noms "tower_n" et "skyscraper_n". Chacun de ces noms ont deux formes, l'une au singulier, l'autre au pluriel. Les différents volumes sollicités sont présentés dans la figure 5.

Figure 1: Proto macrostructure de la base lexicale

On peut créer une telle base lexicale dans un logiciel comme Jibiki, mais :

- d'une part, il faudra étendre les fonctionnalités offertes par l' API ;
- et d'autre part, les requêtes reliant une forme dans une langue avec les formes correspondantes dans une autre langue seront coûteuses en volume. En effet, d'un point de vue conceptuel au sens des modèles entités-associations pratiqués en conception de bases de données, les lexies réalisent une association n-m entre les lemmes et les axèmes, de même pour les axies entre les axèmes des langues considérées. Il y a encore des associations n-m entre les lemmes et les formes, et entre les formes et les transcriptions phonétiques.

Les associations n-m augmentent les possibilités de parcours de façon multiplicative, même si certaines peuvent se compenser. Il en résulte un risque de temps de réponse intolérable pour les utilisateurs dans le cadre de leur tâche d'apprentissage, pour des requêtes sollicitant la base lexicale dans toute la largeur de ses volumes. Toutefois, les régularités de la langue diminueront probablement grandement la combinatoire et on pourra travailler sur des sous-volumes pour minimiser les tailles intermédiaires.

4.2 Portée des différents types de ressources

Les scenarii proposés mettent en relief des statuts différents de certaines données de la base lexicale :

- données relatives à l'utilisateur (choisies explicitement ou collectées du fait de son activité) ;
- données relatives à des groupes d'utilisateurs ;
- données issues de ressources externes.

Ceci nous a forcés à formaliser ces statuts, définissant ainsi une macrostructure fonctionnelle pour notre prototype.

4.2.1 Dictionnaire de référence

Le choix du terme **dictionnaire de référence** provient des entités (objets) qui serviront à peupler le contenu de la base lexicale. Par transitivité, on pourra parler de « dictionnaire de référence » (concept) pour parler des informations contenues dans la base lexicale et issues des dictionnaires de référence-objets. Dès lors, se pose la question du statut des informations présentes dans la base lexicale qui ne sont pas issues des dictionnaires de référence (objets). Il semble cohérent de considérer les modifications de la base lexicales validées comme intégrées au dictionnaire de référence (concept). On en arrive à la définition suivante :

dans le cadre de LexInnova, le dictionnaire de référence est la somme des informations de la base lexicale, qui a été validée, que ce soit par le biais d'une institution externe à l'instance de LexInnova (cf. dictionnaires existants) ou par le biais du système.

4.2.2 Lexique institutionnel

U n **lexique institutionnel** a pour vocation de synthétiser le point de vue sur le lexique d'une institution. Une institution pourra être tout à fait externe à LexInnova (ex : Conseil de l'Europe) ou un groupement d'utilisateurs de LexInnova. Ce lexique constitue une forme de référentiel, il n'a pas vocation à évoluer selon le déroulement d'un cours. On pourra en revanche créer un lexique institutionnel qui traduise le déroulement générique de cours dans une institution (cf. lexique cible dans le scénario « mandarin »).

4.2.3 Lexique de groupe

U n **lexique de groupe** est un lexique dynamique qui peut être modifié par un ensemble d'utilisateurs. La configuration de base sera un enseignant propriétaire d'un lexique de groupe, qui y donne accès (y compris en écriture) à un groupe d'apprenants. Toutefois, on peut tout à fait imaginer un groupe d'apprenants se créant entre eux un lexique de groupe (par exemple pour l'argot). Cela posera à terme la question de la visibilité du lexique : un lexique de groupe doit-il être visible pour les autres groupes ? Un lexique créé par un groupe d'apprenant doit-il être consultable par les enseignants ?

Le lexique de groupe permet également de formuler des objectifs pour les membres du groupe. Chaque entrée d'un lexique de groupe pourra être taguée avec la valeur cible. Ainsi le lexique cible d'un groupe pourra évoluer tout au long de l'année en fonction des ajouts de ses membres ou d'imports issus de lexiques institutionnels ou personnels.

4.2.4 Lexique personnel

U n **lexique personnel** est un lexique qui n'a qu'un seul propriétaire. Il s'agira en général d'un apprenant, qui seul pourra modifier son contenu et décider de sa visibilité. Ici aussi le propriétaire du lexique pourra choisir les éléments cibles du lexique. Au sein de ce lexique personnel, les données pourront être étiquetées pour faire la différence entre les entrées que l'apprenant a explicitement ajoutées à son lexique et celles qui y ont été ajoutées pour une raison ou pour une autre par le système (cf. scénarios « jeu »). Quoi qu'il en soit, l'apprenant membre d'un groupe devra décider lui-même si ses objectifs coïncident avec ceux du groupe (des groupes) au(x)quel(s) il appartient. En d'autres termes, ce n'est pas parce qu'une entrée figure dans un lexique de groupe auquel a accès un apprenant qu'elle se trouvera dans son lexique personnel.

5 Résultat fonctionnel

Si les étapes précédentes nous ont permis d'affiner la modélisation du lexique, des questions restent en suspens (cf. partie précédente). Afin d'alimenter les discussions avec les futurs utilisateurs et d'orienter nos recherches, nous avons décidé d'implémenter rapidement un prototype fonctionnel. Celui-ci sert également à décider des efforts de développement futurs. Ce prototype est développé en PHP de façon à pouvoir être intégré facilement à l'ENPA développé dans le projet Innovalangues. Il se sert de la plate-forme Jibiki pour la gestion des ressources lexicales. La communication entre les deux systèmes s'effectue via l'interface de programmation de Jibiki fondée sur le protocole REST.

Jibiki (Mangeot & Chalvin, 2006) est une plate-forme générique de gestion de ressources lexicales hétérogènes en ligne développée et maintenue depuis 2001 principalement par Mathieu Mangeot. Celle-ci est programmée à l'aide de Enhydra, un serveur d'objets java basé sur une architecture 3/tiers. Pour la couche de données, la base de données utilisée est Postgres. Le logiciel est disponible gratuitement et en source ouverte sur la forge du LIG[15].

Tout type de ressource lexicale au format XML peut être importé dans la plate-forme. Les ressources sont décrites et manipulées à l'aide de pointeurs dans la structure XML, ce qui permet de les importer, les consulter et les éditer sans modifier leur structure. Il est possible de gérer tout type de microstructure. Concernant la microstructure, le système de traitement de liens inter-volumes permet de gérer différentes macrostructures : monolingue, bilingue bidirectionnelle, multilingue à structure pivot ou multi-étages type Pivax (Zhang et al., 2014).

5.1.1 Consultation des ressources

Deux interfaces de consultation sont disponibles. L'interface de consultation simple affiche une vue évoluée du dictionnaire imprimé : la partie gauche affiche les vedettes du voisinage immédiat du

[15] https : //jibiki.ligforge.imag.fr/

mot recherché classées par ordre alphabétique. Elle est équipée d'un ascenseur infini permettant de parcourir toutes les vedettes du dictionnaire selon l'ordre alphabétique. Lorsqu'on clique sur une vedette de la partie gauche, l'article complet s'affiche dans la partie droite.

L'interface de consultation avancée permet de combiner les critères de recherche. Par exemple, rechercher un mot français terminant par "er", qui soit un nom et qui appartienne au domaine de la botanique.

Lors de l'affichage de chaque article, un menu s'affiche en haut à droite. Celui-ci comprend des liens vers le formulaire d'édition, l'historique des modifications, ainsi que la vue source XML de l'article.

5.1.2 Édition des articles

Lors de l'import d'une ressource sur la plate-forme, le schéma XML représentant la structure des articles est également importé. Un formulaire d'édition des articles est ensuite automatiquement généré par la plate-forme (Mangeot & Chalvin, 2006).

Deux modes d'édition sont possibles. Lors de la consultation d'un article, l'édition sur place permet de modifier directement le texte d'un article qui s'affiche en double-cliquant sur la chaîne de caractères à modifier. Celle-ci se transforme en champ de texte avec un bouton « ok »sur la droite pour valider la saisie.

Cet éditeur est programmé à l'aide de la technologie AJAX et il utilise l'API REST de la plate-forme Jibiki pour dialoguer avec le serveur. Lors de la validation de la saisie (clic sur le bouton « ok »), la nouvelle chaîne de caractères est envoyée au serveur avec le pointeur Xpath du segment édité et l'identifiant unique de l'article (Mangeot, 2016).

Par contre, ce mode ne permet pas l'ajout de nouvelles informations, comme un exemple d'usage en contexte. Pour ajouter ou supprimer des parties d'information, il est nécessaire d'utiliser le formulaire d'édition complète. Outre les interacteurs HTML classiques (boîtes texte, menu déroulant, cases à cocher, etc.), celui-ci a été enrichi avec des interacteurs plus complexes pour gérer des listes d'objets.

5.2 Présentation du système Lexinnova

5.2.1 Architecture globale du système

Le site Web principal avec lequel l'utilisateur interagit est l'ENPA du projet Innovalangues.

L'ENPA est déjà intégré à la plate-forme Claroline Connect. Le prototype Lexinnova [16] reprend la charte graphique de l'ENPA et il est en cours d'intégration sous forme de module Claroline Connect.

Les fonctionnalités du prototype sont directement intégrées à l'ENPA sous forme de menus en haut à droite de l'écran.

Une instance spécifique de Jibiki a été déployée pour le projet Lexinnova.

Celle-ci contient :

- un dictionnaire de référence Lexinnova avec un volume pour chaque langue (pour l'instant seul l'espagnol est accessible) ;
- un dictionnaire créé pour chaque utilisateur avec un volume pour chaque langue.

Cette instance est installée sur le serveur de démonstration de l'équipe GETALP, au Laboratoire d'Informatique de Grenoble. La communication entre le prototype et l'instance de Jibiki s'effectue avec l'interface de programmation (API) REST de la plate-forme Jibiki. La figure 2 montre l'architecture du système dans son ensemble.

[16] http : //totoro.imag.fr/Lexinnova

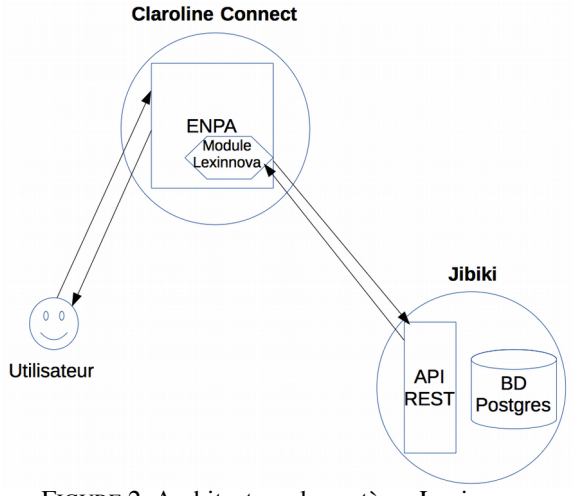


FIGURE 2: Architecture du système Lexinnova

5.3 Fonctionnement du prototype

Lors de la première visite d'un utilisateur sur l'ENPA et avant que son compte ne soit créé, deux fonctionnalités sont disponibles dans le menu Lexinnova :

- consultation des dictionnaires de référence ;
- lecture active.

Lorsque l'utilisateur crée un compte sur l'ENPA, un compte similaire est créé sur la plate-forme jibiki via un appel à l'API.

Une fois le compte créé, une nouvelle fonctionnalité est ajoutée au menu : la gestion des lexiques personnels (consultation et modification).

Si l'utilisateur a des droits spécifiques d'administration (par exemple, un enseignant ou un administrateur de l'ENPA), un second menu d'administration est ajouté à la barre des menus.

Il contient les fonctionnalités suivantes :

- gestion des utilisateurs (liste de tous les utilisateurs et suppression d'un utilisateur existant);
- gestion des lexiques (liste de tous les lexiques, création d'un nouveau lexique et suppression d'un lexique existant).

Une démonstration de ce prototype a été réalisée lors du séminaire bi-annuel Innovalangues de janvier 2016[17]. Un podcast est à présent disponible[18].

5.4 Fonctionnalités disponibles

5.4.1 Limitations du prototype

De façon à pouvoir rapidement montrer un prototype fonctionnel, nous avons choisi de limiter les fonctionnalités disponibles en nous concentrant sur certains scénarios (scénario « tío »). La réflexion

[17] http://innovalangues.fr/seminaire-bi-annuel-innovalangues-janvier-2016/
[18] http://podcast.grenet.fr/episode/lexinnova-lexiques-personnalisables-integres-a-lenpa/

sur les lexiques de groupe étant encore en cours, nous n'avons pas implémenté cette fonctionnalité. Les dictionnaires de références et les lexiques personnels ont été quant à eux complètement implémentés.

Au niveau des données, nous n'avons pas encore collecté un ensemble complet de données lexicales pour chaque langue du projet. Nous avons créé un dictionnaire de référence pour l'espagnol afin d'implémenter certains scénarios. Ce dictionnaire ne contient pour l'instant que le vocabulaire nécessaire à la réalisation des scénarios (soit une vingtaine de mots au total).

Pour le passage à l'échelle en termes de quantité de données et de nombre d'utilisateurs, des calculs ont été faits sur la base de 180 000 entrées pour chaque dictionnaire de référence et 10 000 utilisateurs inscrits. Des tests de montée en charge ont été effectués sur la base de 600 utilisateurs en ligne. Le prototype actuellement déployé peut répondre à une telle charge pour les fonctionnalités principales de gestion des ressources. Par contre, le module de lecture active nécessitant beaucoup de mémoire vive, il sera nécessaire de le déployer sur un autre serveur afin de répartir la charge.

5.4.2 Consultation des dictionnaires de référence

La consultation des dictionnaires de référence est publique. Il n'est donc pas nécessaire de se loguer pour y accéder. L'interface utilisée est la page de consultation simple de la plate-forme Jibiki : la partie gauche affiche les vedettes du voisinage immédiat du mot recherché et la partie droite affiche le ou les articles trouvés de manière détaillée. Si l'utilisateur est logué, un bouton "+" s'affiche en haut à droite de l'article. L'utilisateur peut cliquer sur ce bouton pour importer l'article dans son lexique personnel.

La figure 3 montre la fenêtre de consultation du mot "tio" dans le dictionnaire espagnol (on notera

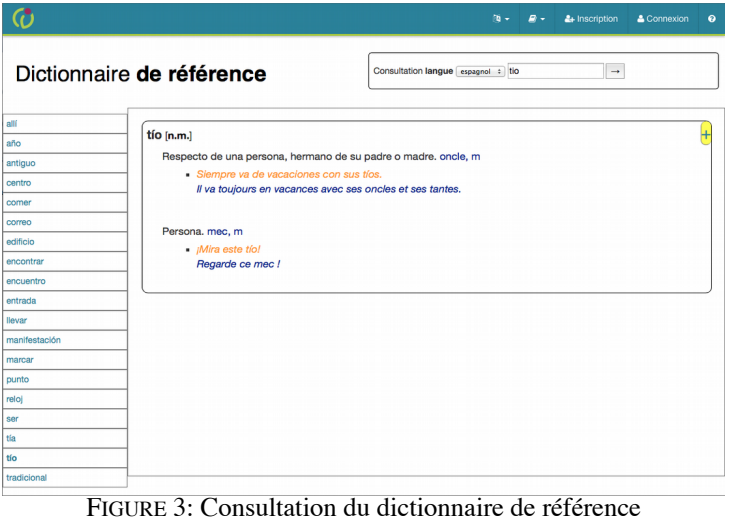
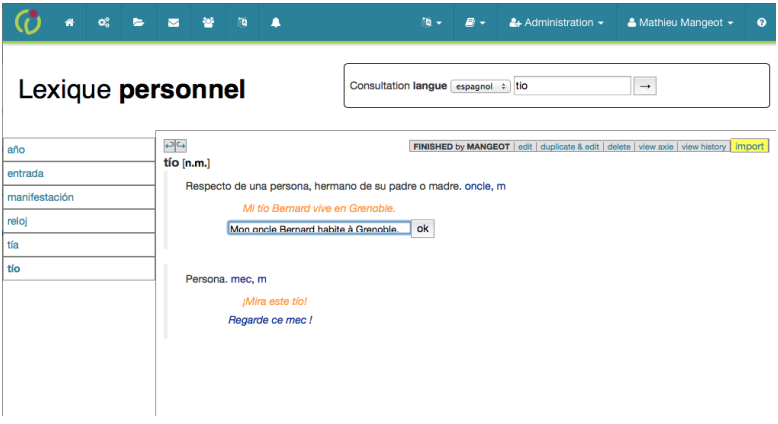


FIGURE 3: Consultation du dictionnaire de référence

que l'article « tío », accentué, est obtenu).

5.4.3 Gestion des lexiques personnels

FIGURE 4: Édition d'un exemple d'article dans le lexique personnel5



La gestion des lexiques personnels reprend également l'interface de consultation simple de la plate-forme Jibiki. Il est par contre nécessaire d'être un utilisateur enregistré pour pouvoir y accéder. Lorsque l'utilisateur consulte une entrée dans cette interface, il peut ensuite l'éditer directement en cliquant sur la zone à modifier. La technique utilisée est celle de l'édition sur place de la plate-forme Jibiki. Sur la figure 4, l'utilisateur a consulté le mot "tío" dans son lexique personnel (après l'avoir importé depuis le dictionnaire de référence). Il souhaite modifier l'exemple pour qu'il fasse référence à son histoire personnelle ce qui lui permettra d'accéder plus facilement au sens de ce mot.

5.4.4 Module de lecture active

Le module de lecture active reprend celui utilisé sur le site jibiki.fr (Mangeot 2016). Il s'agit d'une aide à la lecture pour les apprenants d'une langue étrangère. L'utilisateur saisit ou colle un texte dans la fenêtre de saisie. Le texte est ensuite envoyé à un analyseur morphologique puis chaque lemme est consulté dans son lexique personnel ou dans le dictionnaire de référence si aucune réponse n'est trouvée dans son lexique personnel. Le texte entré précédemment est réaffiché plus bas avec des informations supplémentaires provenant de l'analyse. Lorsque l'utilisateur bute sur un mot, il clique dessus et une fenêtre s'affiche avec l'article correspondant provenant de la consultation des différentes ressources (lexique personnel ou dictionnaire de référence).

Dans l'exemple de la figure 5, l'utilisateur clique sur le mot "antiguo" et l'article s'affiche sur la droite de l'écran. Si l'article provient du dictionnaire de référence, celui-ci affichera un bouton "+" permettant d'importer l'article dans le lexique personnel de l'utilisateur, comme lors de la consultation du dictionnaire de référence.

Le module de lecture active peut également afficher une transcription ou une prononciation au dessus de chaque mot en utilisant l'élément ruby[19] du standard HTML5. Pour le japonais, il est possible d'afficher le furigana ou le romaji. Pour le chinois, il est possible d'afficher le pinyin ou le bopomofo. Pour le français, il est possible d'afficher une prononciation simplifiée ou en API, etc.

[19] http://www.w3.org/TR/ruby/

FIGURE 6: Module de lecture active

6 Conclusion

La conception d'un ENPA a fait émerger la nécessité d'une grande base lexicale commune couvrant tous les besoins des différents acteurs, que se soient des machines (modules de jeux sérieux ou de générateurs d'exercices, etc.) ou des humains (apprenants). À partir de plusieurs scénarios d'utilisation, nous avons défini une structure de base lexicale multilingue permettant de répondre à ces besoins. Nous avons ensuite implémenté un prototype fonctionnel permettant la consultation automatique ou manuelle de ressources et la création de lexiques personnels. Ce prototype, programmé en PHP, interagit avec la plate-forme Jibiki de gestion de ressources lexicales.

Les perspectives principales de ce projet à moyen terme sont les suivantes :

- l'analyse des pratiques d'enseignants en langues en termes de lexiques grâce à un questionnaire en cours d'élaboration ;
- le passage à l'échelle du prototype en termes de nombre d''utilisateurs, de quantité de données et de macrostructure. Pour cela, il faudra constituer pour chaque langue un volume de grande couverture avec des données libres de droit (possibilité d'utiliser les données de wiktionary) et de relier les différents sens de chaque langue via un volume pivot ;
- un test grandeur nature au deuxième semestre 2016-2017 avec une classe d'étudiants en sciences apprenant l'anglais de spécialité à l'Université de Grenoble.

Remerciements

Cette recherche a été en grande partie financée par le projet IDEFI Innovalangues.

Références

BASTIEN, C. ET SCAPIN, D. (2004). « La conception de logiciels interactifs centrée sur l'utilisateur: étapes et méthodes », *in* Pierre Falzon (dir.), *Ergonomie*, 1re édition, Paris, Presses Universitaires de France, pp. 451–462.

BELLASSEN, J. (2010). « La didactique du chinois et la malédiction de Babel : émergence, dynamique et structuration d'une discipline » in *Études Chinoises*, Hors-série. Disponible en ligne : http://www.afec-etudeschinoises.com/IMG/pdf/Bellassen_Didactique.pdf.

BOITET, C., MANGEOT, M., SÉRASSET, G. (2002), « The PAPILLON project: cooperatively building a multilingual lexical data-base to derive open source dictionaries & lexicon », Proc of on NLP and XML (NLPXML 2002), Ed. Graham Wilcock, Nancy Ide & Laurent Romary, COLING Workshop, Taipei, Taiwan, 31 August 2002, pp. 9-15.

CARLO, M.D., ANQUETIL, M., VECCHI, S., JAMET, M.-C., MARTIN, E., PEREA, E.C., HIDALGO, R., PISHVA, Y., GILLES, F. ET ANDRADE, A.I. (2015). « REFIC – Référentiel de compétences de communication plurilingue en intercompréhension », Référentiel de compétences Projet Européen Miriadi. Disponible en ligne : http : //www.miriadi.net/refic.

CARRON T., MARTY J-C. ET MANGEOT M. (2009). *How to bring immersion in Learning Games ?* Proc. IEEE-ICALT 2009, Riga, Latvia, 15-17 July 2009, 6 p.

CECRL (2000). *Cadre européen commun de référence pour les langues : apprendre, enseigner, évaluer*, Conseil de la coopération culturelle — Comité de l'éducation — Division des langues vivantes (dir.), Édition française, Strasbourg; Paris, Conseil de l'Europe. Disponible en ligne : http : //medias.didierfle.com/media/contenuNumerique/007/4140016745.pdf.

CHAIX, C., MEUNIER-CARUS, M., PESQUET, J.-A., RONDIN, R., GOUDIN, Y. ET LOISEAU, M. (2016). « Kanji Crunch: Apprentissage dissocié des habiletés et autonomisation de la compétence graphique par le jeu vidéo en mandarin langues étrangères », 38e colloque APLIUT : *Jeux en jeu dans l'enseignement/apprentissage des langues en LANSAD*, Lyon, 2-4 juin 2016.

DABÈNE, L. (1994). « Le projet européen GALATEA: pour une didactique de l'intercompréhension en langues romanes » Jeanine Stolidi (dir.), *Recherches en linguistique hispanique*, n°22, pp. 41-45.

GOUDIN, Y. ET LÊ, T.M. (2016). « Jouer avec le sacré? Le sinogramme à l'ère du jeu sérieux » Haydée Silva et Mathieu Loiseau (dir.), *Recherches et applications*, n°59, pp. 145–160.

LOISEAU, M., HALLAL, R. ET BALLOT, P. (2016). « Game of Words: prototype of digital game focusing on oral production (and comprehension) through asynchronous interaction », EUROCALL 2016, Limassol (Chypre), 24-27 août 2016.

LOISEAU, M., ZAMPA, V. ET REBOURGEON, P. (2015). « Magic Word: premier jeu développé dans le cadre du projet Innovalangues » , *ALSIC*, vol. 18, n°2. DOI : 10.4000/alsic.2828. Disponible en ligne : http : //alsic.revues.org/2828.

LUSTE-CHAA, O. (2009). *Les acquisitions lexicales en français langue seconde: conceptions et applications*, Thèse de doctorat, Université Paul Verlaine, Metz. Disponible en ligne : http://www.theses.fr/2009METZ023L.

MANGEOT M. (2016). *Collaborative construction of a good quality broad coverage and copyright free Japanese-French dictionary* Hosei University International Found Foreign Scholar Fellowship Report Volume XVI 2013-2014, Hosei University, Tokyo, Japan, pp. 175-208.

MANGEOT M. ET CHALVIN A. (2006). *Dictionary Building with the Jibiki Platform : the GDEF case.* Proc. of LREC 2006, Genoa, Italy, 23-25 May 2006, pp. 1666-1669.

MASPERI, M. ET QUINTIN, J.-J. (2014a). « Enseigner à l'université en France, à l'ère du numérique: l'apport de dispositifs d'ingénierie innovants dans la formation en langues », *in* Cristiana Cervini et Anabel Valdivieso (dir.), *Dispositivi formativi e modalità ibride per l'apprendimento linguistico*, Bologna, CLUEB, Contesti Linguistici, pp. 61-80. Disponible en ligne : https ://www.researchgate.net/publication/271852910.

MASPERI, M. ET QUINTIN, J.-J. (2014b). « L'innovation selon Innovalangues » Elsa Del Col (dir.), *Lingua e nuova didattica*, n°1/2014, pp. 6-14.

PORTINE, H. (2013). « L'ingénierie linguistique : des technologies au service d'une didactique intégrant la cognition ? » Christian Ollivier et Laurent Puren (dir.), *Mutations technologiques, nouvelles pratiques sociales et didactique des langues*, n°54, pp. 159–168.

YASSINE-DIAB, N., ALAZARD-GUIU, C. ET LOISEAU, M. (2016). « Check your Smile, first prototype of a collaborative LSP website for technical vocabulary learning », EUROCALL 2016, Limassol (Chypre), 24-27 août 2016.

ZHANG Y., MANGEOT M., BELLYNCK V. ET BOITET C. (2014). *Jibiki-LINKS : a tool between traditional dictionaries and lexical networks for modelling lexical resources.* Proceedings of the 4th Workshop on Cognitive Aspects of the Lexicon (CogALex) 2014 (Eds. Michael Zock, Reinhard Rapp, Chu-Ren Huang), Dublin, Ireland, 23 August 2014, 12 p.

Active Reading for Intercomprehension Between Sinogramic Languages

Pierric Mazodier
Laboratoire LIG, équipe GETALP
Bâtiment IMAG CS 40700
38058 GRENOBLE CEDEX 9,
FRANCE
pierric@mazodier.fr

Mathieu Mangeot
Laboratoire LIG, équipe GETALP
Bâtiment IMAG CS 40700
38058 GRENOBLE CEDEX 9,
FRANCE
mathieu.mangeot@imag.fr

Valérie Bellynck
Laboratoire LIG, équipe GETALP
Bâtiment IMAG CS 40700
38058 GRENOBLE CEDEX 9,
FRANCE
valerie.bellynck@imag.fr

Yoann Goudin
LIDILEM – UGA / XMU
No. 422, Siming South Road,
Xiamen, Fujian, CHINA. 361005
yoanngoudin@yahoo.fr

ABSTRACT

In the domain of e-learning for a linguistic field, active reading interfaces are useful tools for learning new languages. For a learner of a given language, the goal is to assist in knowledge development in other known languages, in order to enhance their learning. An active reading program enriches the understanding of the input sentence in different ways, such as adding the pronunciation of each word or displaying a definition or a translation taken from a dictionary. By the process of analogy, the users would then be able to synthesize the information and produce themselves a translation that would be, at their convenience, both syntactically and semantically appropriate.

We propose two new features for an active reading interface that aims to help in the understanding of Sinogramic languages. We focus on Japanese and Mandarin languages. The first feature identifies the reading (*on'yomi*, *kun'yomi*…) of the sinograms; the second feature displays, for certain words of the input text, their equivalent in another language that is etymologically close to the source one. In our case, the words with *on'yomi* reading will be associated with their Mandarin equivalents, in their *Pīnyīn* form. This tool can be useful for learners of Japanese with some knowledge of Mandarin or the contrary.

CCS Concepts

• **Computing methodologies→Artificial intelligence→Natural language processing →Language resources**

Keywords

NLP; active reading; sinogramic languages; intercomprehension; analogy; Japanese; Mandarin; Jibiki platform; Mecab; HanLP; onyomi; rōmaji; furigana; Pīnyīn

1. INTRODUCTION

In the domain of e-learning for a linguistic field, active reading interfaces are useful tools for learning new languages. For a learner of a given language, the goal is to assist in knowledge development in other known languages, in order to enhance their learning. While a straightforward translation can be useful to grasp a meaning or to get a syntactically correct output sentence, it can still be semantically incorrect. Furthermore, it does not help the learner to increase her own knowledge of the language.

Instead, an active reading program enriches the understanding of the input sentence in different ways, such as adding the pronunciation of each word or displaying a definition or a translation taken from a dictionary. By the process of analogy, the users would then be able to synthesize the information and produce by themselves a better interpretation or translation that would be, at their convenience, both syntactically and semantically appropriate.

We coin sinograms as an alternative proposal to 'Chinese characters' in order to distinguish the contemporary political entity from the past cultural dominant ideology over Eastern Asia and its most visible representative: the graphic system. Consequently, Sinogramic languages (SL) [7] refer to the languages sharing a history including cultural loans through literary corpora and standards, the graphic system conveying them and so the lexicon still in use in these languages nowadays. Defined as such, SL include all the Sinitic languages among which standard Mandarin as the legitimate variant, but also typologically distinct languages such as Japanese, Korean and Vietnamese. In this paper, we will just deal with Japanese and Mandarin.

We propose two new features for an active reading interface that aims to help in the understanding of Sinogramic languages here after Japanese as a target language and Mandarin as first SL. The first feature identifies the reading type (*on'yomi*, *kun'yomi* cf. below 2.3) of the sinograms in Japanese texts ; the second feature displays, for certain words of the input text, their reading in Mandarin. In our case, the words with *on'yomi* reading will be associated with their Mandarin reading in their *Pinyin* form.

In this paper, we will proceed as follows: the second section will present an overview of the Japanese writing systems and the question of reading sinograms in Japanese texts. The third section will focus on the active reading process. Then, the Jibiki lexical resources management platform will be described in the fourth section. Lastly, we will present the new features for intercomprehension [5] between SL. In conclusion, we will discuss how these features might be improved and applied to other linguistic contexts.

2. THE JAPANESE WRITING SYSTEM AND THE READINGS OF SINOGRAMS

2.1 Japanese language *vs* 'Chinese'

As mentioned above, Japanese language [6] is typologically distinct from the contemporary Sinitic languages among which Modern Mandarin [9], but historically under cultural influence including Classical Chinese literature written in sinograms as early as the 5th century. By the 8th century, in order to write down reading information and Japanese-specific words, sentences and texts, several initiatives became soon two sets of syllabaries that are derived from the sinogramic script. In the same time, the Japanese crafted their own readings of the

sinograms and by doing so emancipated from Classical Chinese. We insist on the fact that through this historical process, 'Chinese' never refers to the legitimate language spoken nowadays known as Mandarin.

2.2 Contemporary graphic typology for Japanese

As a result today, the Japanese writing system is composed of four different types of characters:

1. ***kanji*** (漢字) are sinograms borrowed through this historical process and sometimes derived from initial forms. They are syllabic, commonly used in nouns, verbs and adjectives (e.g. 東京 respectively read as *Tō* and *kyō*, meaning the city of Tokyo) ;

2. ***hiragana*** (平仮名) is one of the both sets mentioned above. *Hiragana* are 'syllabic' – moraic is more accurate [6]– signs used for particles, pronouns, adverbs, inflections (also referred as ***okurigana*** (送り仮名), etc. (e.g. そして *soshite*, "then") ;

3. ***katakana*** (片仮名) refers to the second set and an alternate version of *hiragana*. Nowadays, they are used for non-Chinese foreign loan words (e.g. クロワッサン *kurowassan*, "croissant").

4. ***rōmaji*** (ローマ字) are the Latin characters also used nowadays in Japanese contemporary texts (e.g. ABC Mart).

Furthermore, when small *kana* (usually *hiragana*) are seen written above *kanji*, they are called ***furigana***[1] (振り仮名) and are used to indicate the reading of the sinogram below. They appear usually in texts for children or above sinograms for indicating rare readings. At last, for non-Japanese speakers, there are two main transliteration standards of *rōmaji*: the *Hepburn* and *Kunrei-shiki* (訓令式), which differs from each other only by some subtleties (e.g. *shi*/*si* for transliterating し/シ, *tsu*/*tu* for つ/ツ ...).

2.3 Reading sinograms in Japanese

Beyond this graphic typology of Japanese texts, we also have to introduce shortly how the sinograms can be read in Japanese. As already mentioned above, *kanji* have two types of reading:

1. ***on'yomi*** (音読み), referred here after as 'Chinese pronunciation' or ***on* reading** inherited from Chinese loans over the history, (e.g. 東京 *Tōkyō*, 'Tokyo') ;

2. ***kun'yomi*** (訓読み), referred here after as 'Japanese pronunciation' or ***kun* reading**, usually used for Japanese indigenious lexicon (e.g. 東 *higashi*, 'East').

The point is that beyond their belonging to differently originated lexical stocks, they also do not share the same phonological patterns. Indeed, while the latter can be polysyllabic – three for 東 *higashi* – the former will always conform the sinosyllabic structure shared by all the SL (cf. below 2.4).

Each k*anji* may have at least one reading and frequently have one or more of each of these two types of readings. In order to be complete in this overview, we have to distinguish polysinographic – composed of at least two sinograms – words with mixed readings. The are two types conventionally referred as ***jūbakoyomi*** (重箱読み) ***on+kun*** reading words – *jū-bako* ' box'– and ***yutōyomi*** (湯桶読み) ***kun+on*** reading words, *yu-tō* 'h ot pot', At last, there is a last type referred as *ateji* (当て字) where *kanji* are used to phonetically represent native or borrowed words with less regard to the underlying meaning of the characters such as in 寿司'sushi'.

2.4 *On* readings and the sinosyllable

As this paper deals with features shared by SL and so Mandarin and Japanese *on* readings, we have to discriminate several categories of *on'yomi* corresponding to the different times during which the under going standard reading of a given word was borrowed from Chinese by Japanese speakers and introduced into their lexicon. Through this process, a given sinogram could be borrowed several times with diachronic variation of its reading. Three different *on* reading categories are conventionally discriminated as follows:

1. ***go'on*** (呉音) were the earliest readings introduced to Japan during the Nara period (e.g. 浪人 *rōnin*, 'ronin') ;

2. ***kan'on*** (漢音) followed (e.g. 外国人 *gaikokujin*, 'foreigner') ;

3. ***tōsō'on*** (唐宋音) came later during Heian and Edo period (e.g. 蒲団 *futon* 'futon').

Even if these readings evolved, they all conform to the sinosyllable structure as long as it was the basis of literary composition into Classical Chinese and the institution of riming. The structure is organized as Figure 1.

ONSET (Initial)	RHYME (Final)			
		Rime		
		TONE		
	GLIDE (medial)	NUCLEUS	CODA	
consonant or 0	*-j- / -y- / -w-* or 0	vowel (or 0*)	cons. *-n / -ng (/ -m*)* voc. *-j / -w* or 0	*-p* *-t* *-k*

Figure 1: The sinosyllable and its structure

Any sinosyllable – whatever Japanese *on* reading included – will conform this pattern distributed between the onset and the rhyme subdivided itself as glide (or 0), nucleus, vocalic or consonantic coda (or 0) plus tone (or 0).

Among the examples above, such as 人 read as *nin* or *jin*, *n-* and *j-* are the onset, *-i-* is the nucleus, and *-n* is the consonantic coda. Compared to the *kun* reading of the sinogram 人, *hito*, we can see that this reading does not conform the sinosyllabe distribution.

2.5 Sinogram and sinosyllable as *potential of intercomprehension*

Beyond this stricking diversity of various graphic systems in Japanese texts and diverse reading types for *kanji*, either it is challenging especially for foreign learners, there are short cuts even for this latter population as soon as learning and teaching are emancipated from language ideologies. Thus, English loans easily identified through their notation in *katakana* might provide a huge stock of lexcicon semanticly transparent and very helpful for phonological awarness. Furthermore, for foreign learners already familiar with sinograms – such as those coming from China, Taiwan, Hongkong etc., *kanji* are – with some exceptions – not only visually accessible for their meaning. The point now is that as long as Mandarin is becoming the first SL to be learnt abroad beyond the few aeras previsously mentioned, *katakana* and semantic informations provided by *kanji* are not the only *potential of intercomprehension*: sinograms are also a resource through their *on* reading and so a feature on which learners can rely in order to transfer their reading knowledge in Mandarin into Japanese.

This is our didactic statement [4] and the theory for which our tool is designed. As far as we know, such a tool does not exist and so is our contribution.

3. THE ACTIVE READING PROCESS

3.1 Description of the generic process

The active reading tool [1] proceeds as follows:

1. The text received as input is sent to a morphological analyzer to obtain the lemma and grammatical information for each word.

[1] also known as Japanese ruby characters.

2. Using the lemmas received from the analyzer, the tool then looks up a monolingual or bilingual dictionary to obtain lexical information about each lemma. It can be a definition or a translation.

3. The tool displays the input text by adding additional information either above or below the text, or by displaying words with different colors, or by adding a code that will display a pop-up window when the mouse hovers over the word.

3.2 Active reading for Japanese

3.2.1 Description of the active reading

The active reading service offers to display the reading of a Japanese text in the form of *furigana* and the French translations coming from a Japanese-French dictionary. The list of the corresponding translations from the dictionary appears in a textbox when hovering the mouse over certain words. The service is available online from the jibiki.fr website[2].

This active reading instance is tailored for the following scenario: a learner of Japanese with a good knowledge of French. It could be enlarged to learners fluent in other languages by using different dictionaries. For example, the use of Jim Breen's JMdict [2] could be useful for users fluent in English, Russian, German, Dutch, etc.

3.2.2 The morphological analysis

The parsing and analysis of the input text is performed with the Conditional Random Field based MeCab morphological analyzer and tokenizer[3].

Figure 2 shows the result of the analysis by Mecab of the sentence entered by the user in Figure 3. Each line raepresents a token. The first left column is the token itelf. Then follows a series of 8 items separated by commas. The first one is the part-of-speech + 名詞 is a noun, 動詞 is a verb, 助詞 is a particle and 記号 is a punctuation mark. The next 4 ones are used for subcategorization. Then, the next item is the lemma (eg: for the item 掛かり, the lemma is the infinitive form of the verb 掛かる). The next item is the writing form of the token in *katakana* and the last item is the reading form in *katakana* (e.g. the particle は has the writing form ハ *ha* and the reading form ワ *wa*). For our purpose, we use the lemma and the reading form.

One can see in the example of Figure 2 that the numbers are actually parsed each figure separately by Mecab. In order to render correctly the *furigana* of the numbers, we encoded a specific function. That is especially important in Japanese because of the 万 (*man*) '10 000' counting unit which differs from the Western 1000 counting unit.

3.2.3 The Japanese-French dictionary

The jibiki.fr Japanese-French dictionary[4] [8] combines 3 main sources:

• a digitized version of the *Cesselin* [3] Japanese-French dictionary edited by Gustave Cesselin and published in 1940. It contains more than 82,000 articles with detailed microstructure: French translations, examples, locutions, etc;

• completed with 47,630 Japanese-French or Japanese-English entries from the JMdict [2] for modern vocabulary and

• 23,486 Japanese-French or Japanese-English Wikipedia links for recent terminology.

The articles from JMdict and Wikipedia were imported if they appeared also in the *Super Daijirin dictionary*, in order to avoid a large number of unuseful entries. The resulting resource thus built contains more than 153,000 entries and 140,000 examples. It can be considered as the most reliable source for Japanese-French translations.

The result of the OCR process is never perfect. Thus, some articles coming from the *Cesselin* need corrections. The dictionary can be edited online by any user who detect an error. So far, more than 81,000 contributions were established in the 4 years of the project life.

Mecab analyzer is distributed with a generic dictionary called ipadic with 392,126 entries in total. Despite its consequent size, several words in our dictionary are not present in the ipadic dictionary. Thus, we decided to generate a Mecab dictionary from the content of Jibiki.fr dictionary in order to enlarge the coverage of Mecab. Our dictionary contains 22,321 entries that are not present in ipadic. The first version is still beta. On going work is on its way to produce a correct dictionary particularly concerning the subcategories of the verbs. Once this work will be completed, we will publicly release this dictionary on the Jibiki.fr project website.

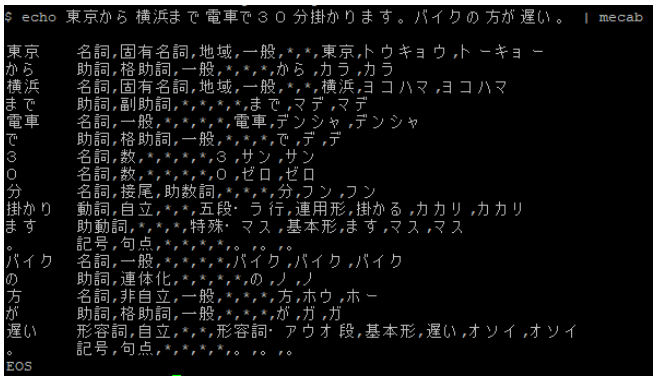


Figure 2. Parsing of a Japanese text with MeCab

4. ENRICHING THE ACTIVE READING FOR INTERCOMPREHENSION

4.1 Purpose of the new features

We will now present two new features aiming at making the active reading output display even more informative. As explained previously, Japanese language uses different writing systems and readings depending on which lexicon stock a given word belongs to: *on'yomi* for *kanji* whose pronunciation was borrowed from Chinese during different periods, *kun'yomi* for Japanese spellings, *katakana* for foreign loan words. The objective is to highlight the words of the text with different colors according to their stock, in order to tell them apart and get knowledge of the links between them and their various influences. Then, the *Pīnyīn* is displayed below the words whose *kanji* are *on'yomi*[5] (see Figure 4). This information gives some clues for intercomprehension between sinogramic languages [4].

These new features are aimed to learners of Japanese with some knowledge of Mandarin or the contrary.

Figure 3 shows the new active reading interface with these new features. The top half of the window is the same as Figures 2. The result is then still displayed on the bottom half of the window with the *furigana* on top of the Japanese sentence and a French translation when the mouse is over a word (here the word "*osoi*", translated in French by "*tardif*") but two new features can be seen:

[2] https://jibiki.fr/reading/
[3] https://taku910.github.io/mecab/
[4] http://jibiki.fr/
[5] https://jibiki.fr/reading/pierric_f.php

• The words in violet background are in *kun'yomi* (*Yokohama*, *kakari*, *osoi*);

• The words in red background are in *on'yomi* and below them, the *Pīnyīn* of the Mandarin equivalent word is displayed in red (*Tōkyō*, *densha*, *fun*, *hou*);

• The words in green background are in *katakana* here an English loan word (*baiku*).

As already mentioned above, among the Chinese imperial literary standards, there was the fundamental practice of Classical riming system. If it is abolished nowadays, this institution is still the potential of intercomprehension between SL through the conformation to the sinosyllable structure. Thus, on the Figure 3 , one can already see that a constant pattern emerges between *Pīnyīn* and *rōmaji* for Japanese words in *on'yomi*. For example, the Mandarin "*dōng*" is transcribed with the Japanese "*tō*"; the Mandarin "*diàn*" is transcribed with the Japanese "*den*", etc.

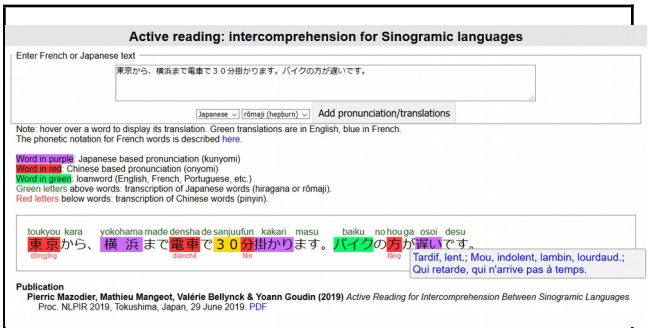


Figure 3: Active reading with on'yomi, kun'yomi and Pīnyīn

4.2 Challenges of the implementation

The implementation of such features presents some particularities to take into account:

• We made the assumption that all the *kanji* of a same word have the same nature, since there does not seem to exist any counter-example to this rule, at least in modern-day Japanese. Although MeCab provides the information about the pronunciation of the *kanji*, it does not give the *on/kun'yomi* nature. The information being also absent from the Cesselin, we have to collect it from KANJIDIC[6], an online *kanji* database from James Breen's JMdict/EDICT project [2]. The KANJIDIC has been imported also on the Jibiki platform.

• Since we lack exhaustive resources able to indicate from which foreign language a loan word written in *katakana* comes from, we will not be able to tell them apart with different colors so far.

• There are also numerous cases where a word has a special pronunciation that cannot be deduced from the pronunciations of the individual kanji it contains (e.g. 今日 *kyō*, today). These cases will appear with a different color.

4.3 Details of the implementation

We modified the active reading PHP program by incorporating a piece of code composed of four main parts: information extraction, decision making, *Pīnyīn* addition and display.

4.3.1 Program overview

The active reading's PHP code consists in the following steps, the green ones being our new features' implementation:

a. Catching user's input

b. Parsing it through MeCab morphological analyzer

c. Gathering the following elements, for each word: *furigana*, lemma (e.g. base form of verbs…), part-of-speech (e.g. verb, noun…), subcategory (e.g. transitive verb)

d. Converting *furigana*'s *katakana* into *hiragana*

e. Querying the Jibiki.fr Japanese-French dictionary for retrieving translations with the REST API

f. Querying KANJIDIC with the REST API and extracting information

g. Processing it through decision function

h. Associating the result to a highlight color

i. For *on'yomi* word, get *Pīnyīn* through HanLP[7] Mandarin morphological analyzer

On HTML side:

j. Displaying each word with: corresponding highlight color, *furigana* on top, translation in a pop-up window and, for *on'yomi* words, *Pīnyīn* below.

4.3.2 Information extraction (f)

The *on/kun'yomi* nature of the kanji has to be retrieved from KANJIDIC. The data is organized in XML tree structures that can be accessed with a Curl request to the Jibiki REST API. The entries to be checked are those of each kanji for each word in the text that possesses *furigana*. If a word also contains *hiragana* (as *okurigana*, e.g. 買い物 *kaimono*, 'shopping'), the XML structure of this said sinogram will be empty ; we nevertheless keep the *hiragana* as it is. For *katakana* words, we will use a regular expression with a pre-established list of *katakana*. Instead of immediately processing the data, we store it in an array to make it available for further use and to ensure code clarity. Considering the aforementioned hypothesis of one *yomi* per word, the *on'* and *kun'yomi* will be stored separately, to be later processed independently.

Before going to the decision part, the pronunciation data has to be formatted:

• Some pronunciations are preceded or followed by a dash (–), indicating a restriction on their position in the word (e.g. 何:なん – *nan*– (what) cannot be used at the end of a word). This information will be used to skip some cases in the decision part. We also remove the dash.

• Some also include a dot (.), usually separating the radical of a verb or adjective from its inflection (e.g. 会:あ.う *a.u*, meet). We then remove the inflection and the dot.

4.3.3 Decision making (g)

Once all the data has been collected and formatted, for each word, we will compare the different possible *on'yomi* combinations of their *kanji* (and potential *okurigana*) with the known *furigana* of the word. If there is a match, we know that the word is *on'yomi* and we label it as such. If not, we repeat the process for the *kun'yomi*. If ultimately no match is found, the word will be labeled as special pronunciation.

Once the word's reading has been figured out, the highlight color value is drawn from a correspondence array: red for *on'yomi kanji*, purple for *kun'yomi kanji*, gold for special pronunciation and green for *katakana*.

4.3.4 Pīnyīn addition (i)

To retrieve the *Pīnyīn* of the *on'yomi* words, we used HanLP parsing toolkit for Mandarin to retrieve the individual information for each *kanji* (Figure 4). HanLP is a powerful java library for processing Mandarin texts. We developed a simple

[6] http://www.edrdg.org/kanjidic/kanjidic.html

[7] https://github.com/hankcs/HanLP

java program that displays the *Pīnyīn* relative information for each sinogram received in input. The first line displays each sinogram separated by a comma; the next line displays the *Pīnyīn* with tones in numbers, the next one the *Pīnyīn* with tones with diacritics; the next one the *Pīnyīn* without tone and the other lines give more detailed information about the reading. For our purpose, we use the third line that gives us the *Pīnyīn* with tones represented by diacritics.

```
$ hanlp.sh 東京
原文(texte source) : 東,京,
拼音-数字音调 (Pinyin, ton en chiffre) : dong1,jing1,
拼音-符号音调 (Pinyin, ton traditionnel) : dōng,jīng,
拼音-无音调 (Pinyin, sans ton) : dong,jing,
声调 (ton) : 1,1,
声母 : d,j,
韵母 :  ong,ing,
输入法头 : d,j,
```

Figure 4. analysis of a sinogramic word by HanLP

4.3.5 Display (i)

On the HTML side, we need to add the highlight variable as a style attribute in the start tag of the output element in the case of *kanji* or *katakana* words. The tag is a Ruby HTML tag, which allows to write *furigana* on top of Japanese text.

In order to display *Pīnyīn* below as well, the Ruby <rtc> tag is used in conjunction with the <rt> tag for furigana and a CSS style option has to be added: {ruby-position: under;}. Unfortunately, even if the ruby tags including the <rtc> one are now part of the HTML5 specifications, few browsers implement it. At the moment, only Firefox is correctly rendering the *Pīnyīn* below the text. The webkit family (Safari, Chrome, etc.) and Opera browser are rendering the *Pīnyīn* next to the Japanese.

5. CONCLUSION AND PERSPECTIVES

This article explains the development of a first version of an active reading tool for Japanese texts enriched for intercomprehension between sinogramic languages by displaying the *Pīnyīn* of the Japanese *on'yomi* words. This work is promising. We are waiting for the feedback of potential users in order to improve it.
Several perspectives are on their way.

Concerning the implementation of the tool itself, as the code currently constitutes a draft of a more formalized future clean version, some improvements can be imagined for code optimization, performance improvements and trustful case handlings. This latter point can be challenging, since it requires exhaustive data sets of special cases (e.g. counters, Japanese names), most of them being virtually infinite. In the case of our feature, the extent of handled cases depends on the coverage of KANJIDIC. A more thorough (but costly) process would be to cross-check different sources. Other technical optimizations could be to display the *Pīnyīn* below the Japanese even on browsers that do not render the <rtc> tag correctly.

The described active reading scenario of Japanese texts could be extended at least in two different ways:

• looking up etymological information for *katakana* words from rich dictionaries (the Cesselin contains such information) and displaying precise information about the origin of the words. For example, the word "*baiku*" comes from the English word "bike".

• Computing automatically analogies between rōmaji and *Pīnyīn*.

We are working also on different active reading scenarios for other languages:

• It would be interesting to include other sinogramic languages such as Korean, Vietnamese and Sinitic languages among which Taiwanese very soon.

• A study has also been driven for Berber and Arabic speaking learners of French and English [1].

• Other language families could be addressed like Romance languages, Slavic languages, Germanic languages, pidgins and creoles, etc.

6. REFERENCES

[1] Abdellaoui, S., Bellynck, V., Mangeot, M. and Boitet, C. 2018. Outillage de l'accès aux textes par la lecture active étymologique multilingue pour apprenants berbérophones et arabophones. *Traitement Automatique des Langues Africaines TALAf 2018* (Grenoble, France, Sep. 2018).

[2] Breen, J.W. 2004. JMDict: a Japanese-multilingual dictionary. (Geneva, Switzerland, Aug. 2004), 71–78.

[3] Cesselin, Gustave 1940. *WaFutsu daijiten - Dictionnaire japonais-français*. Maruzen.

[4] Goudin, Y. 2017. *L'intercompréhension en langues sinogrammiques : théories, représentations, enjeux, et modalités d'une didactique de la variation*.

[5] Grin, F. and Conti, V. 2008. *S'entendre entre langues voisines: vers l'intercompréhension*. Georg.

[6] Labrune, L. 2012. *Japanese Phonology*. OUP.

[7] Magistry, P., Fabre, M. and Goudin, Y. 2017. Phonological Cues in Sinograms: from Psycholinguistics to Computational Modeling for Language Teaching. *Traitement Automatique des Langues*. 57, 3 (2017), 41–65.

[8] Mangeot-Nagata, M. 2016. Collaborative Construction of a Good Quality, Broad Coverage and Copyright Free Japanese-French Dictionary. *International Journal of Lexicography*. 31, 1 (Sep. 2016), 78–112. DOI:https://doi.org/10.1093/ijl/ecw035.

[9] Norman, J. 1988. *Chinese*. Cambridge University Press.

Annexe 4 : Bibliographie personnelle

Publications de Mathieu Mangeot

Nombre de documents
92

Enseignant-chercheur à l'Université Savoie Mont-Blanc et membre de l'équipe GETALP du laboratoire d'informatique de Grenoble (LIG).

Note : depuis 2014, je ne me déplace plus en avion pour assister à des conférences.

Article dans une revue

2 documents

Mathieu Mangeot-Nagata. Collaborative Construction of a Good Quality, Broad Coverage and Copyright Free Japanese-French Dictionary. *International Journal of Lexicography*, Oxford University Press (OUP), 2016, 31 (1), pp.78-112. . .

Mathieu Mangeot, Gilles Sérasset, Mathieu Lafourcade. Construction collaborative d'une base lexicale multilingue, le projet Papillon. *Traitement Automatique des Langues*, ATALA, 2003, 44 (2), pp.151-176.

Communication dans un congrès

66 documents

Pierric Mazodier, Yoann Goudin, Mathieu Mangeot, Valérie Bellynck. Active Reading for Intercomprehension Between Sinogramic Languages. *Natural Language Processing and Information Retrieval 2019*, Jun 2019, Tokushima, Japan.

Louis Lecailliez, Mathieu Mangeot. A Hypergraph Data Model for Building Multilingual Dictionary Applications. *ASIALEX 2018*, Jun 2018, Krabi, Thailand.

Mathieu Mangeot, Mutsuko Tomokiyo. Contributions et corrections dans le dictionnaire japonais- français jibiki.fr. *11e journées du réseau "Lexicologie, terminologie, traduction" LTT 2018*, Sep 2018, Grenoble, France.

Mathieu Mangeot, Valérie Bellynck. Un devin de microstructures pour importer ou normaliser des ressources lexicales. *Lexicologie Terminologie Traduction LTT 2018*, Sep

2018, Grenoble, France.

Slimane Abdellaoui, Valérie Bellynck, Mathieu Mangeot, Christian Boitet. Outillage de l'accès aux textes par la lecture active étymologique multilingue pour apprenants berbérophones et arabophones. *Traitement Automatique des Langues Africaines TALAf 2018*, Sep 2018, Grenoble, France.

Mutsuko Tomokiyo, Christian Boitet, Mathieu Mangeot. Towards an Automatic Classification of Illustrative Examples in a Large Japanese-French Dictionary Obtained by OCR. *First Workshop on Linguistic Resources for Natural Language Processing*, Aug 2018, Santa Fe, United States. pp.112 - 121.

Mouhamadou Khoulé, Mathieu Mangeot, Mamadou Nguer. Manipulation de dictionnaires d'origines diverses pour des langues peu dotées : la méthodologie iBaatukaay. *Traitement Automatique des Langues Africaines 2018*, Sep 2018, Grenoble, France.

Mutsuko Tomokiyo, Mathieu Mangeot-Nagata, Christian Boitet. Analyse et classification d'exemples illustratifs dans le dictionnaire "Cesselin" en utilisant Google Traduction et un dictionnaire UNL-UWs. *11es journées du réseau Lexicologie Terminologie Traduction 2018*, Sep 2018, Grenoble, France.

Ying Zhang, Valérie Bellynck, Mathieu Mangeot, Christian Boitet. Une nouvelle architecture intégrant les données lexicales générales, terminologiques et « situées » : Pivax-3. *Terminology & Ontology: Theories and applications TOTh 2017*, Jun 2017, Chambéry, France.

Mathieu Mangeot, Yoann Goudin, Mathieu Loiseau, Valérie Bellynck, Emmanuelle Eggers. La prise en charge du lexique pour l'apprentissage sur plateforme en ligne : scénarios d'utilisation et prises en compte des spécificités du mandarin. *Journées d'études internationales AREC 2017 — Le lexique du chinois contemporain*, 2017, Paris, France.

Valérie Bellynck, Emmanuelle Eggers, Yoann Goudin, Mathieu Loiseau, Mathieu Mangeot, et al.. Projet Lex:M : gestion de lexiques pour environnements d'apprentissage en ligne. *8e rencontres du Pôle Grenoble Cognition* , 2017, Grenoble, France.

Mutsuko Tomokiyo, Mathieu Mangeot, Christian Boitet. Development of a classifiers/quantifiers dictionary towards French-Japanese MT. *MT Summit 2017*, Sep 2017, Nagoya, Japan.

El Hadj Malick Fall, El Hadji Mamadou Nguer, Sokhna Bao-Diop, Mouhamadou Khoule, Mathieu Mangeot, et al.. Digraphie des langues ouest africaines : Latin2Ajami : un algorithme de translittération automatique. *Atelier Traitement Automatique des Langues Africaines TALAf 2016, conférence JEP-TALN-RECITAL 2016*, Jul 2016, Paris, France.

Mathieu Mangeot. Traitement Automatique des Langues Africaines 2016 Préface. *Atelier Traitement Automatique des Langues Africaines TALAf 2016, conférence JEP-TALN-RECITAL 2016*, Jul 2016, Paris, France.

Alla Lo, El Hadji Mamadou Nguer, N'Diaye Abdoulaye, Cheikh Bamba Dione, Mathieu Mangeot, et al.. Correction orthographique pour la langue wolof : état de l'art et perspectives. *JEP-TALN-RECITAL 2016: Traitement Automatique des Langues Africaines TALAF 2016*, Jul 2016, Paris, France.

Mouhamadou Khoule, Mathieu Mangeot, El Hadji Mamadou Nguer, Mame-Thierno Cissé. iBaatukaay : un projet de base lexicale multilingue contributive sur le web à structure pivot pour les langues africaines notamment sénégalaises. *Atelier Traitement Automatique des Langues Africaines TALAf 2016, conférence JEP-TALN-RECITAL 2016*, Jul 2016, Paris, France.

Mathieu Mangeot, Valérie Bellynck, Emmanuelle Eggers, Mathieu Loiseau, Yoann Goudin. Exploitation d'une base lexicale dans le cadre de la conception de l'ENPA Innovalangues. *Enseignement des Langues et TAL*, ATALA; AFCP, Jul 2016, Paris, France. pp.48-64.

El Hadji Mamadou Nguer, Mouhamadou Khoule, Mouhamad Ndiankho Thiam, Mbaye Baba Thiam, Ousmane Thiare, et al.. Dictionnaires wolof en ligne : état de l’art et perspectives.. *Colloque National sur la Recherche en Informatique et ses Applications*, Oct 2015, Thiès, Sénégal.

Laurent Besacier, Elodie Gauthier, Mathieu Mangeot, Philippe Bretier, Paul Bagshaw, et al.. Speech Technologies for African Languages: Example of a Multilingual Calculator for Education. *Interspeech 2015 (short demo paper)*, Sep 2015, Dresden, Germany.

Mathieu Mangeot. MotàMot project: conversion of a French-Khmer published dictionary for building a multilingual lexical system. *Languages Resources and Evaluation Conference*, May 2014, Reykjavik, Iceland. pp.8.

Chantal Enguehard, Mathieu Mangeot. Computerization of African languages-French dictionaries. *CCURL 2014 : Collaboration and Computing for Under Resourced Languages in the Linked Open Data Era*, May 2014, Reykjavik, Iceland. pp.121.

Chantal Enguehard, Mathieu Mangeot. Favorisons la diversité linguistique en TAL. *Journée d'étude de l'ATALA. "Ethique et Traitement Automatique des Langues"*, Nov 2014, Paris, France.

Zhang Ying, Mathieu Mangeot, Valérie Bellynck, Christian Boitet. Jibiki-LINKS: a Tool between Traditional Dictionaries and Lexical Networks for Modelling Lexical Resources. *Cognitive Aspects of the Lexicon (CogALex) 2014*, Aug 2014, Dublin, Ireland. pp.87 - 98.

Mathieu Mangeot. Construction d'un système lexical multilingue, libre de droits, centré sur le français et le japonais via des méthodes automatiques et contributives. *Journée Francophone de la Recherche*, Nov 2014, Tokyo, Japon.

Mathieu Mangeot, Chantal Enguehard. DILAF : des dictionnaires africains en ligne et une méthodologie. *Francophonie et Langues Nationales*, Nov 2014, Dakar, Sénégal.

Mathieu Mangeot, Fatiha Sadat. Actes de l'atelier TALAf 2014 sur le traitement automatique des langues africaines : Préface. *TALAf 2014 Traitement Automatique des Langues Africaines*, Jul 2014, Marseille, France.

Ying Zhang, Mathieu Mangeot. Gestion des terminologies riches : L'exemple des acronymes. *TALN-RECITAL 2013*, 2013, Sables-d'Olonne, France. pp.6.

Ying Zhang, Mathieu Mangeot. Bases lexicales multilingues : traitement des acronymes. *première journée " Jeunes Chercheurs " du réseau Lexicologie Terminologie Traduction (JCLTT 2013)*, 2013, Bruxelles, Belgique. pp.6.

Chantal Enguehard, Soumana Kané, Mathieu Mangeot, Issouf Modi, Mamadou Lamine Sanogo. Vers l'informatisation de quelques langues d'Afrique de l'Ouest. *Atelier TALAf 2012 : Traitement Automatique des Langues Africaines*, Jun 2012, Grenoble, France.

Mathieu Mangeot, Carlos Ramisch. A Serious Lexical Game for Building a Portuguese Lexical-Semantic Network. *Proceedings of the ACL 2012 3rd Workshop on The People's Web Meets NLP: Collaboratively Constructed Semantic Resources and their Applications to NLP*, Jul 2012, Jeju, South Korea.

Mathieu Mangeot, Chantal Enguehard. Actes de l'atelier sur le traitement automatique des langues africaines : écrit et oral; Préface. *TALAf 2012 Traitement Automatique des Langues Africaines*, Jun 2012, Grenoble, France.

Chantal Enguehard, Soumana Kané, Mathieu Mangeot, Issouf Modi, Mamadou Sanogo. Vers l'informatisation de quelques langues d'Afrique de l'Ouest. *4ème atelier international sur l'Amazighe et les Nouvelles Technologies*, Feb 2011, Rabat, Morocco. pp.13-32.

Mathieu Mangeot, Chantal Enguehard. Informatisation de dictionnaires langues africaines-français. *journées LTT 2011*, Sep 2011, Villetaneuse, France. 11 p.

Mohammad Daoud, Kyo Kageura, Christian Boitet, Asanobu Kitamoto, Mathieu Mangeot. Multilingual Lexical Network from the Archives of the Digital Silk Road. *6th COLING Workshop on Ontologies and Lexical Resources*, 2010, Beijing, China. pp.9.

Mathieu Mangeot, Sereysethy Touch. MotÀMot project: building a multilingual lexical system via bilingual dictionaries. *Second International Workshop on Spoken Languages*

Technologies for Under-Resourced Languages, 2010, Penang, Malaysia. pp.6.

Christophe Courtin, Stéphane Talbot, Mathieu Mangeot. Activity regulation for the participative creation of data on-line. *International Conference on Machine and Web Intelligence*, Oct 2010, Algiers, Algeria, France. pp.132-139.

Mathieu Mangeot, Hong-Thai Nguyen. Building lexical resources: towards programmable contributive platforms. *IEEE-RIVF 2009, International Conference on Computing and Communication Technologies*, 2009, Danang, Vietnam, Vietnam. pp.84-92.

Mathieu Mangeot, Hong Thai Nguyen. Projet Mot à mot : élaboration d'un système lexical multilingue par le biais de dictionnaires bilingues. *journées scientifiques LTT*, Sep 2009, Lisbonne, Portugal, France. pp.12.

Thibault Carron, Jean-Charles Marty, Mathieu Mangeot. How to bring immersion in Learning Games?. *IEEE-ICALT 2009*, Jul 2009, Riga, Latvia, Latvia. pp.6, .

Christian Boitet, Valérie Bellynck, Mathieu Mangeot, Carlos Ramisch. Towards Higher Quality Internal and Outside Multilingualization of Web Sites. *ONII-08 (Summer Workshop on Ontology, NLP, Personalization and IE/IR)*, Jul 2008, Mumbai, India. pp.8.

Mohammad Daoud, Asanobu Kitamoto, Christian Boitet, Mathieu Mangeot. CLIR-Based Collaborative Construction of Multilingual Terminological Dictionary for Cultural Resources. *ASLIB'08*, Nov 2008, London, United Kingdom. 12 p.

Antoine Chalvin, Mathieu Mangeot. Méthodes et outils pour la lexicographie bilingue en ligne : le cas du Grand Dictionnaire Estonien-Français. *EURALEX 2006*, Sep 2006, Turin, Italie. pp.x-x.

Mutsuko Tomokiyo, Peter Weyer-Brown, Mathieu Mangeot. Représentation Sémantique de Lexique pour un Dictionnaire de Traduction Manuelle et Automatique. *Actes des Aspects méthodologiques pour l'élaboration de lexiques unilingues et multilingues*, May 2006, Bertinoro, Forli, Italie. pp.12.

Mathieu Mangeot, Antoine Chalvin. Dictionary Building with the Jibiki Platform: the GDEF case. *LREC 2006*, May 2006, Genova, Italy. pp.1666-1669.

Mathieu Mangeot. Dictionary Building with the Jibiki Platform: Software Demonstration. *EURALEX 2006*, May 2006, Torino, Italy. pp.121-126.

Mathieu Mangeot. Papillon project: Retrospective and Perspectives.. *Acquiring and Representing Multilingual, Specialized Lexicons: the Case of Biomedicine, LREC workshop*, May 2006, Genoa, Italy. pp.6.

Mathieu Mangeot. Back to the roots: building a French-Japanese dictionary. *Proc. of Papillon 2005 Workshop*, Dec 2005, Chiang Rai, Thailand, France. pp.52-58.

Mathieu Mangeot, David Thevenin. Online Generic Editing of Heterogeneous Dictionary Entries in Papillon Project. *Proc. of the COLING 2004 conference*, Aug 2004, Geneva, Switzerland. pp.1029-1035.

Mathieu Mangeot, Slaven Bilac, David Thevenin. Construction collaborative d'un dictionnaire multilingue : le projet Papillon. *JSF'2003 Journées Scientifiques Francophones*, Nov 2003, National Olympic Memorial Youth Center, Tokyo, Japon. pp.3.

Mathieu Mangeot, Kyoko Kuroda. Divergences interlinguistiques dans le dictionnaire multilingue Papillon. *Proc. MTT'03 Première conférence internationale sur la Théorie Sens-Texte*, Jun 2003, École Normale Supérieure, Paris, France. pp.33-42.

Mathieu Mangeot, Kyoko Kuroda. Interlinguistic Divergences in Papillon Multilingual Dictionary. *Proc. ASIALEX'2003, Meikai University*, Aug 2003, Urayasu, Chiba, Japan. pp.33-42.

Christian Boitet, Mathieu Mangeot, Gilles Sérasset. The PAPILLON project: cooperatively building a multilingual lexical data-base to derive open source dictionaries and lexicons. *Proc. of the 2nd Workshop NLPXML 2002, Post COLING 2002 Workshop*, 2002, Taipei, Taiwan. pp.93-96.

Mathieu Mangeot. How to Import an Existing XML Dictionary Into the Papillon Platform. *Proc. of Papillon 2002 Workshop*, Jul 2002, Tokyo, Japan. pp.10.

Mathieu Mangeot. Proposal Changes for the Monolingual XML Schema. *Proc. of Papillon 2002 Workshop*, Jul 2002, Tokyo, Japan. pp.10.

Jérôme Godard, Mathieu Mangeot, Frédéric Andrès. Data Repository Organization and Recuperation Process for Multilingual Lexical Databases. *Proc. of SNLP-Oriental COCOSDA 2002*, May 2002, Hua Hin, Prachuapkirikhan, Thailand. pp.249-254.

Mathieu Mangeot. Projet Papillon : intégration de dictionnaires existants et gestion des contributions. *JST'02 Journées Science et Technologie*, Nov 2002, National Olympic Memorial Youth Center, Tokyo, Japon. pp.64-65.

Mathieu Mangeot. An XML Markup Language Framework for Lexical Databases Environments: the Dictionary Markup Language. *LREC Workshop on International Standards of Terminology and Language Resources Management*, May 2002, Las Palmas, Spain, France. pp.37-44.

Mathieu Mangeot, Gilles Sérasset, Frédéric Andrès. Frameworks, Implementation and

Open Problems for the Collaborative Building of a Multilingual Lexical Database. *Proc. of SEMANET Workshop, Post COLING 2002 Workshop*, Aug 2002, Taipei, Taiwan. pp.9-15.

Mathieu Mangeot, Gilles Sérasset. Projet Papillon : architecture du serveur Web et structure des articles. *JST-2001 Journées Science et Technologie*, Dec 2001, Tokyo, Japon. pp.149-150.

Mathieu Mangeot, Gilles Sérasset. Papillon Lexical Database Project: Monolingual Dictionaries and Interlingual Links. *NLPRS-2001*, Nov 2001, Tokyo, France. pp.119-125.

Mutsuko Tomokiyo, Mathieu Mangeot, Emmanuel Planas. Papillon: a Project of Lexical Database for English, French and Japanese, using Interlingual Links. *JST'00 Journées Science et Technologie*, Nov 2000, National Olympic Memorial Youth Center, Tokyo, Japan. pp.3.

Mathieu Mangeot. Papillon Lexical Database Project: Monolingual Dictionaries and Interlingual Links. *WAINS'7, 7th Workshop on Advanced Information Network and System*, Dec 2000, Kasetsart University, Bangkok, Thailand. pp.6.

Marie-Hélène Corréard, Mathieu Mangeot. XML- A Solution For LDBs, Eds and MRDs?. *Proc. COMPLEX'99*, Jun 1999, Pécs, Hungary. pp.6.

Mathieu Mangeot. Accès unique à des dictionnaires hétérogènes. *Proc. LTT'99*, Nov 1999, Beyrouth, Liban. pp.3.

Gilles Sérasset, Mathieu Mangeot. L'édition lexicographique dans un système générique de gestion de bases lexicales multilingues. *Natural Language Processing and Industrial Applications*, 1998, Moncton, Canada. pp.110-116.

Mathieu Mangeot. Conception, implémentation et indexation de BaLeM, une base lexicale multilingue. *TALN'98*, Jun 1998, Paris, France. pp.3.

Chapitre d'ouvrage

4 documents

Mathieu Mangeot. Collaborative construction of a good quality, broad coverage and copyright free Japanese-French dictionary. *Hosei University International Found Foreign Scholar Fellowship Report*, 2016, Volume XVI 2013-2014.

Chantal Enguehard, Mathieu Mangeot. LMF for a selection of African Languages. Francopoulo, Gil. *LMF: Lexical Markup Framework, theory and practice*, Hermès science, pp.8, 2013.

Mathieu Mangeot, Chantal Enguehard. Des dictionnaires éditoriaux aux représentations XML standardisées. Gala, Nuria and Zock, Michael. *Ressources Lexicales : contenu,*

construction, utilisation, évaluation, John Benjamins, pp.24, 2013.

Mohammad Daoud, Christian Boitet, Kyo Kageura, Asanobu Kitamoto, Mathieu Mangeot, et al.. Building Specialized Multilingual Lexical Graphs Using Community Resources. Lacroix, Zoé. *Resource Discovery*, Springer, Berlin/Heidelberg, pp.94-109, 2010, Lecture Notes in Computer Science, .

Direction d'ouvrage, Proceedings, Dossier

5 documents

Mathieu Mangeot, Emmanuel Schang. TALAf 2018 : Traitement Automatique des Langues Africaines. Sep 2018, Grenoble, France. 2018, .

Mathieu Mangeot, Cyril Grouin. TALAf 2016 : Traitement Automatique des Langues Africaines. Jul 2016, Paris, France. 2016, .

Mathieu Mangeot, Sadat Fatiha, Brigitte Bigi. Atelier TALAf 2014 : Traitement Automatique des Langues Africaines. Jul 2014, Marseille, France. 2014, .

Mathieu Mangeot, Marc Van Campenhoudt. Lexicologie, terminologie, traduction : Nouvelles recherches au cœur d'un système. Éditions du Hazard, pp.121, 2013.

Chantal Enguehard, Mathieu Mangeot, Gilles Sérasset. JEP-TALN-RECITAL 2012, Atelier TALAf 2012: Traitement Automatique des Langues Africaines. Enguehard, Chantal and Mangeot, Mathieu and Sérasset, Gilles. ATALA/AFCP, pp.x-x, 2012.

Rapport

14 documents

Mathieu Mangeot. Mise en oeuvre d'un outil de lecture active de textes latins. [Rapport de recherche] Laboratoire Litt&Arts. 2019.

Mathieu Mangeot. Jibiki REST Application Programming Interface. [Technical Report] Laboratoire d'Informatique de Grenoble. 2019.

Mathieu Mangeot. Jibiki users REST Application Programming Interface. [Technical Report] Laboratoire d'Informatique de Grenoble. 2019.

Mathieu Mangeot, Chantal Enguehard. Dictionaries for Under-Resourced Languages: from Published Files to Standardized Resources Available on the Web. [Research Report] Laboratoire d'informatique de Grenoble. 2018.

Mathieu Mangeot. Rapport de projet LexInnova : passage à l'échelle. [Rapport Technique] Université Grenoble Alpes. 2016.

Mathieu Mangeot. Interlots:LexInnova/Ressources à récupérer. [Rapport Technique] Université Grenoble - Alpes. 2016.

Mathieu Mangeot. Interlots:LexInnova/Spécifications/Structure de la base lexicale. [Rapport Technique] Université Grenoble - Alpes. 2016.

Mathieu Mangeot. Construction collaborative d'un dictionnaire japonais-français de qualité, à large couverture et libre de droits. [Rapport de recherche] Laboratoire d'Informatique de Grenoble. 2015.

Mathieu Mangeot. Construction of an open-source multilingual lexical system targeted on French and Japanese through contributive and automatic methods: Research project in Japan. [Research Report] Laboratoire d'Informatique de Grenoble. 2015.

Mathieu Mangeot. Construction d'un système lexical multilingue, libre de droits, centré sur le français et le japonais via des méthodes automatiques et contributives : Projet de recherche au Japon. [Rapport de recherche] Laboratoire d'Informatique de Grenoble. 2014.

Mathieu Mangeot, Emmanuel Giguet. Multilingual Aligned Corpora From Movie Subtitles. [Research Report] LISTIC. 2005.

Mathieu Mangeot. De l'ordre dans les dictionnaires au GETA. 1999.

Mathieu Mangeot. Nouvelle architecture pour le serveur UNL. 1999.

Mathieu Mangeot. Outils pour lexicographes naïfs (en informatique).. [Rapport de recherche] Université Joseph Fourier, Grenoble 1. 1997.

Thèse

1 document

Mathieu Mangeot. Environnements centralisés et distribués pour lexicographes et lexicologues en contexte multilingue. Autre [cs.OH]. Université Joseph-Fourier - Grenoble I, 2001. Français.