Vers une étude comparative de différentes approches de classification automatique de textes provenant des secteurs métiers

M B Billami, M Kandi, L Nicolaieff, K Ducharlet, C Gosset, S Rey, C Bortolaso, M Derras

Vers une étude comparative de différentes approches de classification automatique de textes provenant des secteurs métiers

M. B. Billami, M. Kandi, L. Nicolaieff, K. Ducharlet, C. Gosset, S. Rey, C. Bortolaso, M. Derras

Berger-Levrault, 64 Rue Jean Rostand, 31670 Labège, France

{mb.billami, mohamed.kandi, lina.nicolaieff, kevin.ducharlet, camille.gosset, stephanie.rey, christophe.bortolaso, mustapha.derras}@berger-levrault.com

Résumé

De nos jours, la classification automatique de textes est en passe de devenir un domaine de recherche de plus en plus appliqué aux secteurs métiers. De fait, nous assistons à un intérêt croissant des entreprises pour l'exploration des contenus textuels et le traitement automatique du langage naturel. Des techniques de classification ont été proposées impliquant l'entraînement de modèles sur des corpus de données provenant du domaine général. Cependant, ces modèles peuvent se retrouver en difficulté lorsqu'ils sont utilisés pour des métiers spécifiques ayant un vocabulaire spécialisé. Dans cet article, nous proposons plusieurs modèles de classification automatique ayant différents types de représentation de documents pour une application à des secteurs métiers. L'évaluation de nos modèles est effectuée sur un corpus français et validée sur deux corpus de référence pour l'anglais.

Mots-clés

Vocabulaire de spécialité, Types de représentation de documents, Plongements lexicaux, Termes-clés, Classification automatique de textes.

Abstract

Nowadays, automatic text classification is becoming a research field increasingly applied to business sectors. In fact, we foresee a growing interest from companies in exploring text content and automatic natural language processing. Classification techniques have been proposed involving training models on data corpus obtained from the general domain. However, these models tend to run into difficulties when used for specific business domains involving a specialized vocabulary. In this article, we propose different automatic classification models for application to business sectors with different types of document representation. The evaluation of our models is performed on a French corpus and validated on two references corpus for English.

Keywords

Specialty vocabulary, Types of document representation, Word Embeddings, Key-terms, Automatic text classification.

1 Introduction

Face à la concurrence accrue, une entreprise doit rester à la pointe de l'innovation. Cela implique l'identification de potentiels technologiques et d'avancées scientifiques, mais également l'exploration de nouveaux marchés variés et spécifiques. Depuis une vingtaine d'années, nous sommes confrontés à une quantité croissante de données à traiter au quotidien. Tous les secteurs industriels et professionnels, ainsi que toutes les activités scientifiques, sociales, politiques et technologiques génèrent de grandes quantités d'information. Il est ainsi de plus en plus compliqué pour une organisation de se tenir au courant des nouveautés et innovations de ses différents cœurs de marché, que ce soit dans les domaines scientifiques, technologiques ou métiers. Une solution face à ce déferlement d'information est l'automatisation de la veille. Cela implique le recueil et la classification automatique des informations selon les métiers de l'entreprise et l'identification des innovations scientifiques et technologiques à suivre. Ainsi, la classification ou la catégorisation automatique de textes est un domaine de recherche qui peut répondre parfaitement à cette problématique de veille technologique.

La classification automatique de textes peut être considérée comme une procédure permettant d'affecter des documents à un ensemble de catégories prédéfinies. La classification est ainsi utilisée pour organiser, gérer et classer des documents. Dans cet article, nous nous intéressons à la mise au point d'un système automatique de veille innovation et marché. Un tel système implique plusieurs problématiques relatives à la collecte d'information sur le web (*web scraping*) et à la classification des documents collectés. La difficulté principale réside dans le vocabulaire utilisé qui est à la fois : (1) spécifique à des domaines d'activité très pointus et (2) en constante évolution pour cibler pertinemment ce qui relève d'une innovation.

Premièrement, les domaines de spécialités marchés des entreprises peuvent être très pointus avec des vocabulaires métiers très spécifiques, dont les lexiques et/ou les bases de connaissances de spécialité ne sont pas toujours accessibles. De plus, pour une entreprise travaillant dans de multiples secteurs d'activités (comme le *Médico-Social*, les logiciels de *Gestion Financière et de Ressources Humaines*, la *Gestion des*

Ressources Matérielles et Roulantes, et *les Outils de Relation avec les Citoyens*), le sens métier des termes « *pilotage (gouvernance)* », « *facture (dépense)* », voire « *permis de conduire (examen)* » se révèlent souvent plus importants que le sens commun « *pilotage (véhicule)* », «*facture (justificatif)* » et « *permis de conduire (document)* ». Il est ainsi complexe d'utiliser d'une manière directe des modèles de classification entraînés sur des corpus génériques pour permettre une classification de documents provenant de tels domaines métiers, et surtout quand la langue traitée est le français (plus de complexité et peu de corpus métiers fournis pour l'apprentissage). De plus, dans notre problématique, une catégorisation en classes multiples peut être pertinente. Par exemple, un article sur les « *smart city* »[1] peut à la fois concerner la gestion de la relation citoyen à travers les questions de démocratie participative, mais également la gestion des ressources et équipements de la ville comme l'éclairage grâce à l'IoT (*Internet of Things*). Il n'est ainsi pas possible d'utiliser un seul classificateur binaire pour ce genre d'informations multi-spécialités.

Deuxièmement, le vocabulaire spécifique à l'innovation est en constante évolution. Par exemple, plusieurs termes issus des tendances technologiques stratégiques de Gartner[2] ne sont pas présents dans des ressources génériques comme DBpedia [21]. Les termes tels que « *Visualisation de données* », « *3D* », « *réalité mixte* » ou « *Hyperautomation* » ne sont pas actuellement référencés dans DBpedia. De nouvelles avancées technologiques, scientifiques ou en innovations sociales font ainsi émerger chaque jour de nouveaux mots/termes et concepts. Par exemple, des expressions polylexicales telles que « *ville du quart d'heure* », ou « *inclusion numérique* » sont émergées très récemment. Un apprentissage statique devient vite une limite pour une application de veille sur l'innovation dans un cadre métier.

Afin de répondre à ces enjeux, nous proposons dans cet article trois approches de classification/catégorisation automatique de textes : deux approches sont supervisées et une approche est non supervisée. Nous évaluons les méthodes proposées sur un corpus français spécifique à notre cas d'usage de veille métier et innovation. Ensuite, nous validons ces méthodes pour une application sur des corpus de référence anglais et nous proposons une comparaison à plusieurs systèmes état-de-l'art.

Après avoir présenté en section 2 les travaux antérieurs de la classification automatique de textes, nous décrivons en section 3 plus en détail notre problème de classification de documents traitant de thématiques métiers spécifiques. Ensuite, en section 4, nous présentons l'architecture générale de notre système de classification. Par la suite, dans la section 5, nous décrivons notre méthodologie de travail avec la proposition de 3 méthodes de classification automatique de textes. Les différents corpus que nous utilisons sont présentés en section 6. Enfin, nous discutons les résultats d'évaluation en section 7 avant de conclure sur notre travail en section 8.

[1] http://www.envirolex.fr/smart-city-et-ville-du-quart-dheure-paris-veut-se-transformer/

[2] https://www.gartner.com/smarterwithgartner/gartner-top-10-strategic-technology-trends-for-2020/

2 État de l'art

Plusieurs travaux de recherche sur la classification de documents textuels ont été proposés. Trois grandes catégories d'approches émergent [5, 7, 9, 18], à savoir : (1) les approches supervisées [1, 2, 8, 10, 16]; (2) les approches non supervisées [6, 14, 23]; et (3) les approches semi-supervisées ou hybrides [12, 13]. Beaucoup de systèmes de classification automatique de documents textuels existent aujourd'hui. Un ensemble de corpus de référence est mis à disposition de la communauté scientifique pour permettre la validation et la comparaison de ces systèmes.

La plupart des systèmes de classification de ces dernières années ont été évalués principalement sur des corpus de référence anglais, tels que : Reuters-21578[3], 20-Newsgroups[4], WebKB[5], voire IMDb[6]. Pour les autres langues, des corpus d'évaluation existent aussi, tels que DEFT'08 [11] pour le français, TanCorp [4] pour le chinois ou Kalimat Corpus[7] pour l'arabe. Ci-après, nous présentons quelques systèmes état-de-l'art ayant été appliqués pour l'anglais et avec lesquels nous effectuons notre étude comparative. Pour un état-de-l'art complet, le lecteur pourra consulter le travail mené par Dhar et al. [7].

Labani et al. [20] ont proposé un modèle, appelé *Critère de discrimination relative multivariée* (MRDC), pour obtenir les caractéristiques de classification de textes. L'importance de leur modèle réside dans la réduction des caractéristiques redondantes basées sur les concepts de pertinence maximale et de redondance minimale. Dans ce modèle, pour chaque token, la fréquence des documents a également été prise en compte. Ce modèle se propose ainsi comme une méthode de filtrage. Son évaluation a été effectuée sur trois corpus anglais, à savoir : (a) Reuters-21578, (b) 20-Newsgroups et (c) WebKB. Une meilleure précision a été obtenue en utilisant le classificateur Multinomial Naïve Bayes (MNB). Par ailleurs, dans les travaux de Jiang et al. [15], une approche hybride de classification a été introduite, basée sur un réseau DBN (*deep belief network*) et une régression *softmax*. Le DBN a été utilisé pour résoudre les problèmes de calcul matriciel à haute dimension et à dispersion.

Kowsari et al. [19] ont proposé la méthode RMDL (*Random Multimodel Deep Learning*). Cette dernière a la capacité de déterminer les frontières entre classes par suite de l'obtention de l'architecture d'apprentissage profond la plus appropriée. Cette méthode permet d'améliorer les performances de classification grâce à des ensembles d'architectures d'apprentissage profond. Wu et al. [24], quant à eux, ont travaillé sur une technique d'équilibre entre la surpondération et la sous-pondération dans un système de pondération supervisée de termes. Ils ont généré quatre règles basées sur les paramètres du modèle qu'ils ont définis. Les documents ont

[3] https://archive.ics.uci.edu/ml/datasets/reuters-21578+text+categorization+collection

[4] https://archive.ics.uci.edu/ml/datasets/Twenty+Newsgroups

[5] http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/webkb-data.gtar.gz

[6] https://datasets.imdbws.com/

[7] https://sourceforge.net/projects/kalimat/files/latest/download

été classés selon 5 méthodes de défuzzification, à savoir : (a) *Centroid*, (b) *Bisector*, (c) *Smallest of Maxima* (SOM), (d) *Mean of Maxima* (MOM) et (e) *Largest of Maxima* (LOM). Parmi ces méthodes, *Centroid* a obtenu les meilleurs résultats.

Jin et al. [16] ont étudié un modèle de classification conceptuellement simple en exploitant des plongements lexicaux (*word embeddings*) basés sur les classes de textes. Leur hypothèse réside dans le fait que les mots présentent des caractéristiques de distribution différentes selon les classes de textes. Sur la base de cette hypothèse, ils ont entraîné des représentations de mots pour chaque classe. Pour la prédiction de la classe d'un nouveau document, un calcul probabiliste est effectué. La classe sélectionnée est celle qui maximise les probabilités de sélection des vecteurs d'*embeddings* de ses mots.

Jiang et al. [14] ont travaillé sur le concept d'un algorithme à base de similarité pour regrouper les caractéristiques de textes où les tokens sont regroupés en différents groupes en fonction du test de similarité. Des tokens similaires sont associés à une fonction d'appartenance ainsi qu'à une moyenne et un écart en fonction des résultats de leur similarité. Dogan et Uysal [8] ont présenté une étude exhaustive sur l'utilisation de différentes composantes de la fréquence des termes et cela pour 7 méthodes de pondération supervisée de termes parmi lesquelles 6 méthodes existaient déjà et se sont dérivées les unes des autres. L'évaluation des différentes méthodes a été effectuée sur les corpus Reuters-21578 et 20-Newsgroups en utilisant deux algorithmes de classification, à savoir : les Machines à vecteurs de support (SVM) et Rocchio. Ce dernier est l'un des classificateurs basés sur les centroïdes de classes.

Pour le français, la classification automatique de textes a déjà présenté son intérêt par le passé. La 4ème édition des campagnes d'évaluation DEFT (DÉfi Fouille de Textes) datant de 2008 [11] a été concentrée sur cette problématique. La tâche de classification avait pour objectif de catégoriser des textes en 9 classes, à savoir : (a) *Sports*, (b) *International*, (c) *Art*, (d) *Économie*, (e) *Littérature*, (f) *Politique française*, (g) *Sciences*, (h) *Société* et (i) *Télévision*. Le corpus DEFT'2008 a été constitué à partir de deux sources distinctes : *Le Monde* et *la Wikipédia francophone*. Pour chacune de ces sources, un article est identifié s'il apparaît avec l'une des 9 classes. Les résultats obtenus durant cette édition ont montré que les systèmes d'apprentissage supervisé et à base des SVM semblent être très performants. De plus, le prétraitement des textes par lemmatisation de tokens n'était pas indispensable. Cependant, la réduction de l'espace vectoriel par l'utilisation de l'information mutuelle afin de projeter les textes s'est montrée significative. Les meilleures techniques utilisées durant la campagne 2008 pour le français semblent être pertinentes même pour l'anglais puisque des techniques similaires avec des résultats similaires ont été présentés en 2019 par Dogan et Uysal [8].

Dans cet article, nous nous intéressons à une étude comparative de plusieurs méthodes de classification de textes d'actualités écrits en français. Ces textes proviennent d'un corpus de données que nous avons nous-mêmes créé en effectuant une collecte automatique à partir de sources web ciblées. Nous appelons ce corpus par la suite le corpus BL-News. Les méthodes que nous proposons sont indépendantes de la langue. De ce fait, nous présentons une évaluation de nos méthodes sur notre corpus français et une validation sur deux corpus de référence anglais, avec lesquels plusieurs systèmes état-de-l'art ont été validés, à savoir : Reuters-21578 et 20-Newsgroups. Par ailleurs, les méthodes que nous proposons utilisent (a) soit des techniques d'apprentissage supervisé, (b) soit des techniques d'apprentissage non supervisé et à base de similarité sémantique. Pour les systèmes avec un apprentissage supervisé, différentes approches sont proposées.

3 Définition du problème

Comme le démontre l'état de l'art, beaucoup de modèles de classification automatique de textes ont déjà fait leur preuve. Cependant, notre cas d'application a plusieurs particularités pour lesquelles le système de classification automatique doit s'adapter et répondre mieux à ce besoin. Dans cette section, nous présentons trois grandes contraintes sur notre cas d'usage.

3.1 Vocabulaire métier

Les documents que nous souhaitons traiter font référence à des thématiques spécifiques en rapport avec des métiers bien précis. Cette contrainte implique la présence d'un vocabulaire technique qu'il va falloir reconnaître, traiter et pondérer. Nous nous intéressons ainsi à la gestion dans les secteurs métiers suivants : (a) *Éducation*, (b) *Gammes de Gestion* (*financière et ressources humaines*), (c) *Médico-social*, (d) *Relation avec les Citoyens* et (e) *Asset & Fleet* (*ressources matérielles et roulantes*). Le vocabulaire métier de ces thématiques est composé d'une part de mots simples et d'autre part de multi-mots, voire d'expressions. Par exemple, « *préadmission* (*Médico-social*) », « *feuille d'émargement* (*Éducation*) », « *Collectivité territoriale* (*Gammes de Gestion*, *Relation avec les Citoyens*) », « *Internet des Objets* (*Asset & Fleet*) », « *Pacte Civil de Solidarité – PACS* (*Relation Citoyen*) », etc. Nous avons associé manuellement à chaque thématique un ensemble de termes-clés dont ces exemples font partie. Nous appelons chaque ensemble comme étant un lexique d'une thématique donnée.

Afin de quantifier la spécificité de nos lexiques, nous avons mesuré la couverture de la base DBPedia sur les termes-clés de chacun. Le tableau 1 présente le nombre de termes-clés de chaque lexique avec le pourcentage de couverture de la base DBpedia contenant actuellement plus de 4 millions d'entrées.

Lexique	Taille du lexique	Couverture DBpedia (%)
Asset & Fleet	180	29
Éducation	98	16
Gammes de Gestion	122	27
Médico-social	**284**	14
Relation Citoyen	216	30
Innovation	125	**43**
Total	**1 025**	**43**

Table 1 – Couverture des lexiques par DBpedia

En plus des lexiques thématiques, nous avons aussi un lexique « *Innovation* ». Ce dernier fait référence à la catégorie

transversale *Innovation* de tous nos secteurs métiers**.** Cette catégorie tient compte de l'information innovante pouvant se retrouver dans le lot de documents abordant les thématiques métier. En effet, nous nous intéressons ici non seulement à classer l'information relative aux métiers mais aussi à vérifier si elle évoque de l'innovation. Les résultats du tableau 1 montrent que la couverture totale du vocabulaire de nos lexiques représente 43 %, ce qui reste faible. Par exemple, « *5G* », « *devops* », «*générateur de formulaires électroniques*» ne sont pas reconnus actuellement dans DBPedia.

3.2 Classification en classes multiples

En plus de la catégorie transversale « *Innovation* », un document (article) peut être classifié dans une ou plusieurs thématiques. Nous nous retrouvons donc dans le cadre d'une classification multiple. Le nombre de catégories (classes) associées à un document n'est pas fixe. Cette classification se différencie de la classification standard qui attribue une seule étiquette par instance (document).

3.3 Évolution dynamique dans le temps

Nous avons, d'une part, les lexiques métiers qui sont amenés à évoluer dans le temps et, d'autre part, l'aspect *Innovation* que nous voulons faire ressortir du lot des documents à traiter. Cette évolution de l'ensemble des lexiques est importante à prendre en considération pour une meilleure classification.

4 Description du système

Nous avons mis en place un système expérimental pour notre étude comparative des méthodes de classification de textes. L'architecture de ce système est illustrée dans la figure 1.

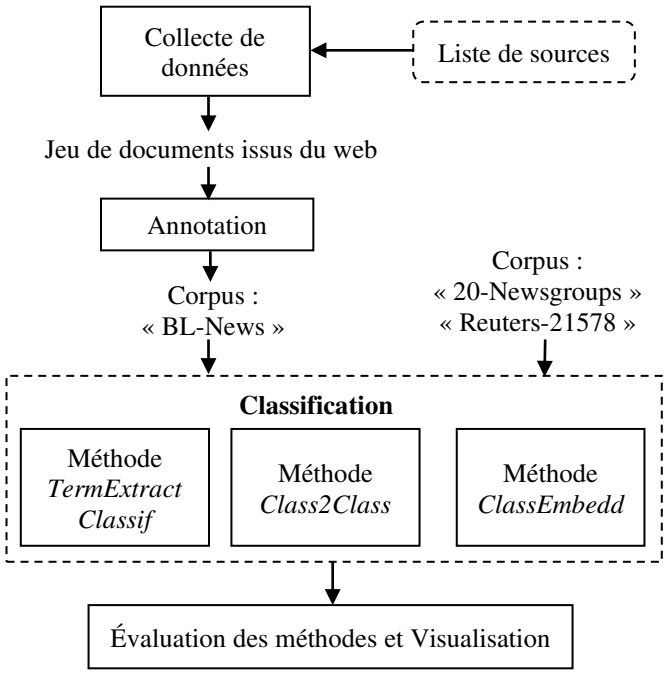

Figure 1 – Architecture du système de classification

Tout d'abord, nous avons développé un module de collecte automatique et périodique de documents issus du web. La liste de sources, constituée de sites d'actualités fréquentés par les experts métiers, est éditée en amont par un expert. Le fonctionnement de la collecte de données est expliqué en détail dans la section 6.1.

Ensuite, nous avons lancé une campagne d'annotation des documents collectés à laquelle neuf annotateurs humains y ont participé. Chacun des annotateurs est un locuteur natif du français et possède la connaissance nécessaire pour juger la pertinence d'associer un document à une thématique donnée. Par ailleurs, chaque document est attribué par un seul annotateur à une ou plusieurs thématiques métiers, voire parfois à aucune. L'annotateur précise aussi si le document traite d'un sujet innovant ou non (cf. *catégorie transversale*). Les résultats de cette campagne sont présentés dans la section 6.2. BL-News représente ainsi le corpus construit. Comme cité en section 2, la validation de nos méthodes de classification est appliquée sur deux corpus anglais : 20-Newsgroups et Reuters-21578.

Les trois corpus sont utilisés pour entraîner (dans le cadre d'une supervision) et évaluer les méthodes de classification automatique. L'implémentation de nos méthodes est effectuée en langage Python et fait appel à l'utilisation de plusieurs librairies *open-source* pour le traitement du langage naturel et l'apprentissage automatique, telles que *spaCy*[8] [13], *PKE* (*Python Keyphrase Extraction*)[9] [2] et *Scikit-learn*[10] [3]. Enfin, une interface web basée sur *Django*[11] a été développée pour faciliter l'édition de la liste de sources et l'annotation des documents. La visualisation et l'analyse des résultats se repose sur la pile *ELK* (*Elasticsearch, Logstash et Kibana*)[12].

5 Méthodologie

Dans cette section, nous présentons trois méthodes de classification automatique de textes. La première méthode repose sur un apprentissage supervisé et utilise des techniques d'extraction de termes-clés pour créer des représentations conceptuelles de documents. La deuxième méthode repose elle-aussi sur un apprentissage supervisé mais vise plutôt à proposer plusieurs classificateurs binaires et utilise des techniques de réduction de dimensionnalité. La troisième méthode, quant à elle, repose sur un apprentissage non supervisé et utilise les plongements lexicaux pour la création d'embeddings représentant les classes et les documents dans un même espace de représentation vectorielle.

5.1 Approche supervisée par extraction de termes-clés

Cette approche se fonde principalement sur une extraction de termes-clés. Nous l'appelons *TermExtractClassif*. Chaque thématique de nos secteurs métiers est caractérisée par des termes-clés. Un document est potentiellement pertinent s'il contient suffisamment de termes-clés dans son titre et/ou son contenu.

[8] https://spacy.io/
[9] https://github.com/boudinfl/pke/
[10] https://scikit-learn.org/stable/
[11] https://www.djangoproject.com/
[12] https://www.elastic.co/fr/what-is/elk-stack/

Dans l'idéal, il serait intéressant d'énumérer en amont tous les termes-clés d'une thématique et leur associer des poids d'importance. Ensuite, ces termes-clés pourraient être utilisés pour mesurer la pertinence des documents vis-à-vis de la thématique. Cependant, il n'est pas raisonnable de construire manuellement une liste suffisamment exhaustive de termes-clés pour des thématiques métiers spécifiques. De plus, si nous prenons en compte l'aspect *innovation*, la liste est amenée à évoluer dans le temps. La méthode *TermExtractClassif* permet de générer une liste de termes-clés pondérés à partir d'un corpus de documents annotés manuellement. Le principe du processus de cette génération est illustré dans la figure 2.

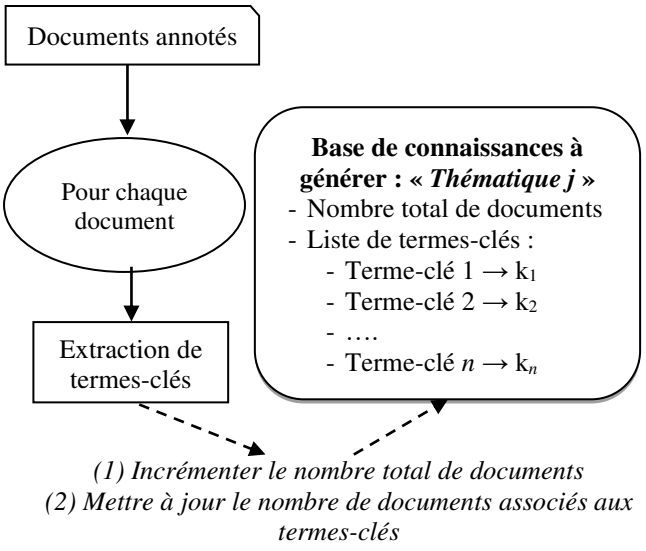

Figure 2 – Construction/enrichissement de la base de connaissances

Pour chaque document annoté en catégorie/thématique par les annotateurs humains, nous effectuons une extraction de termes-clés à partir du titre et du contenu, à l'aide de la bibliothèque *PKE* (*Python Keyphrase Extraction*) [2]. La liste de termes-clés d'une thématique donnée est ainsi constituée de termes extraits automatiquement à partir du corpus. Afin de quantifier l'importance de chaque terme, nous lui attribuons le nombre de documents dans lesquels il apparaît. La base de connaissances d'une thématique donnée est ensuite construite. Nous effectuons cette opération pour toutes les thématiques. Chaque base de connaissances est ainsi utilisée pour mesurer la pertinence de nouveaux documents à classer. La figure 3 illustre ce processus de prédiction. Quand un nouveau document '*d'* arrive, nous effectuons une extraction de termes-clés. Ensuite, nous vérifions si ces termes-clés sont présents dans chaque base de connaissances. Le score du document '*d'* pour la thématique '*j'* est calculé comme suit :

$$score\ (d,j) = -\sum_{i=1}^{n}\left((k_i/T_j) * \log\ (k_i/T_j)\right) \qquad (1)$$

k_i représente le nombre de documents contenant le terme-clé *i* dans le corpus annoté ; T_j représente le nombre total de documents annotés de la thématique *j* ; et *n* représente le nombre de termes-clés dans le document *d*. Nous mesurons un score de pertinence pour le titre et pour le contenu. Un document est considéré comme pertinent si son score de titre et/ou de contenu est supérieur à un seuil fixé au préalable.

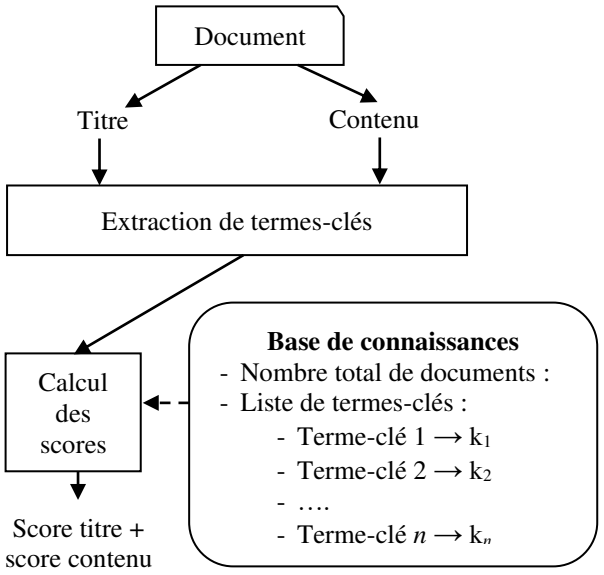

Figure 3 – Approche de classification avec TermExtractClassif

5.2 Approche supervisée : des classificateurs binaires et une analyse discriminante linéaire

La taille des vecteurs caractéristiques permettant de représenter les documents est un élément clé dans la classification de textes. En effet, avoir à disposition un très grand corpus de données, et en appliquant un TF-IDF standard (fréquence du terme-fréquence inverse du document) [17], va engendrer la création de vecteurs avec un très grand nombre de dimensions. L'approche que nous proposons cherche à avoir plus de précision et moins de temps de traitement dans la classification de textes. Pour, cela, nous utilisons des techniques de réduction de dimensionalité avec un top n des caractéristiques provenant du TF-IDF, l'application d'un seuil de variance et une analyse discriminante linéaire (*Linear Discriminant Analysis - LDA*).

Les différentes étapes de cette approche sont présentées dans la figure 4. Nous appelons cette approche par la suite *Class2Class*.

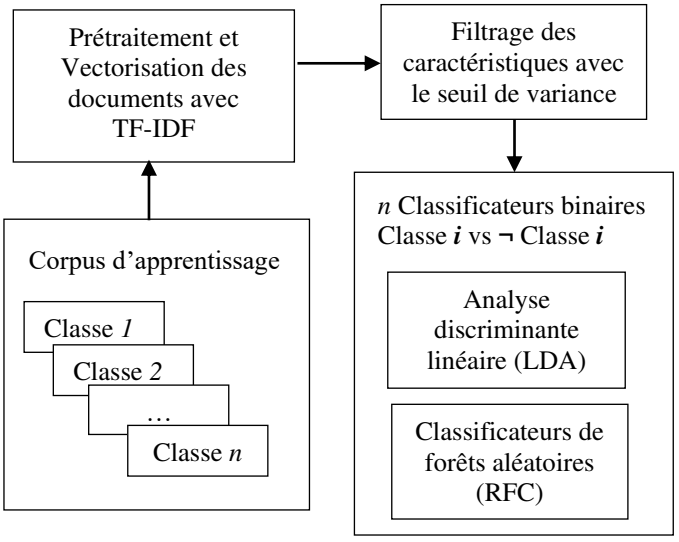

Figure 4 – Approche de classification avec Class2Class

Tout d'abord, nous effectuons un prétraitement sur les données d'apprentissage. Pour cela, nous utilisons *Spacy* [13] pour la lemmatisation. Les modèles « *fr_core_news_lg* » et « *en_core_web_lg* » sont utilisés pour le français et l'anglais, respectivement. Ensuite, nous retirons tous les nombres du corpus et nous gardons en textes seulement les mots portant du sens, à savoir : *noms communs*, *noms propres*, *adjectifs*, *adverbes* et *verbes*.
Par la suite, nous transformons les textes en vecteurs de caractéristiques à l'aide du TF-IDF en utilisant la bibliothèque *Scikit-learn* [3]. Le TF-IDF permet de donner des poids plus élevés aux termes (ou mots) qui apparaissent plus fréquemment dans un texte par rapport aux autres textes du corpus. Par conséquent, nous pouvons dire que le TF-IDF mesure la pertinence et non seulement la fréquence. Nous avons fait le choix de construire un modèle de vectorisation à base de n-grammes avec n ∈ {1, 2, 3}. Cela se justifie par le fait que plusieurs termes-clés de nos lexiques représentent des multi-mots. Nous sélectionnons un nombre maximal de 4 000 meilleures caractéristiques avec TF-IDF. Ensuite, nous utilisons une technique de filtrage, à savoir : le seuil de variance (*Variance Threshold*) qui est proposé dans *Scikit-learn*, pour la suppression des entités constantes. Ces dernières contiennent une seule valeur pour toutes les sorties de l'ensemble du corpus d'entraînement. Ils ne peuvent donc nous donner aucune information précieuse qui pourrait aider la classification de textes.

Après avoir généré le modèle TF-IDF et celui en appliquant le seuil de validation, nous nous intéressons à la création des classificateurs binaires pour chaque thématique traitée. Pour le corpus BL-News, nous créons 12 classificateurs : six avec l'analyse discriminante linéaire et six avec les forêts aléatoires (*Random Forest Classifiers - RFC*). Chacun des six classificateurs traite une thématique donnée, par exemple, un classificateur pour la détection de l'*Innovation* et un classificateur pour détecter si l'article parle d'*Éducation* ou non. Le corpus d'apprentissage d'un classificateur donné représente ainsi deux classes : une classe concerne une thématique bien précise, l'autre classe est construite par complétude du corpus. Ainsi, nous avons des corpus déséquilibrés quelle que soit la thématique à traiter. Par ailleurs, il est à noter que le *RFC* est un méta-estimateur qui ajuste un certain nombre de classificateurs type « *arbres de décision* » sur divers sous-échantillons de l'ensemble de données et utilise la moyenne pour améliorer la précision prédictive et contrôler le surajustement.

5.3 Approche non supervisée : *word embeddings* vers *document embeddings* et *class embeddings*

Nous avons mis en place une méthode non supervisée dans le but qu'elle soit indépendante de toute base d'apprentissage. Cela retire ainsi à la méthode les éventuels biais liées aux données d'apprentissage. Nous appelons cette méthode *ClassEmbedd* (cf. figure 5)**.** Cette méthode fonctionne en deux étapes : (1) tout d'abord, nous créons une représentation vectorielle pour les documents et les thématiques à l'aide de plongements lexicaux ; et (2) nous utilisons un score de similarité pour évaluer la distance entre les documents et les thématiques.

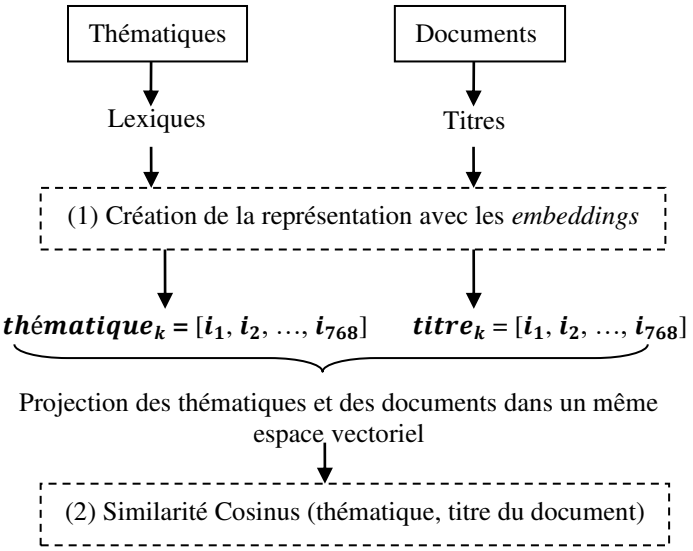

Figure 5 – Approche de classification avec ClassEmbedd

(1) – Représentation sémantique par utilisation de plongements lexicaux

Notre objectif s'est porté sur la recherche d'un modèle de représentation vectorielle permettant la comparaison de nos lexiques métiers avec les documents collectés. Nous avons fait le choix de prendre en considération le modèle *Sentence Transformers* [22], en utilisant la bibliothèque *PyTorch*[13]. Ce modèle permet d'encoder les mots des phrases en prenant en compte le contexte dans lequel les mots apparaissent. Pour ne pas lisser les informations pertinentes dans un document, il faut fournir au modèle un segment résumant l'essentiel du contenu de ce document. Pour cette raison, nous avons fait le choix d'encoder seulement les titres des documents. En effet, le titre est un élément important qui contient de l'information pertinente d'un document. Concrètement, lorsque tout le contenu d'un document est pris en considération, l'information est bruitée par les mots superflus et les résultats peuvent révéler que tous les documents vont tendre vers une représentation similaire. Une fois les documents et les termes-clés de nos lexiques encodés de la même manière, nous pouvons les projeter sur un même espace vectoriel afin de mesurer des similarités sémantiques.

(2) – Similarités sémantiques

Pour la projection, sans avoir besoin d'un corpus de documents annotés manuellement en thématiques, nous avons utilisé l'algorithme des *k*-plus proches voisins (*k-PPV*) pour un entraînement sur les termes de nos lexiques. L'idée derrière l'utilisation du *k-PPV* est de prédire l'emplacement d'un document dans l'espace des lexiques. Les termes-clés des lexiques au voisinage d'un document sont alors retournés. Un voisin proche est défini selon la distance Cosinus la plus petite par rapport à un document donné. Nous avons ensuite récupéré la liste des termes pondérés par leur distance et regroupé par thématique en calculant la distance médiane. La thématique la plus proche correspond donc à celle ayant la distance médiane la plus petite par rapport à un document donné.

[13] https://pytorch.org/

6 Corpus de travail

6.1 Collecte de données

Afin d'évaluer nos méthodes dans le contexte spécifique de la veille métier et innovation, nous avons constitué un jeu de documents textuels issus de sites web d'actualités utilisés par les experts métiers dans notre cas d'utilisation. L'abondance et l'ajout régulier de nouvelles sources rendent impossible le développement d'une méthode spécifique à chaque cas pour l'automatisation de l'extraction du contenu web. Cependant, la qualité de l'extraction à partir d'une méthode générique est dégradée par la variabilité structurelle des sources, ainsi que d'autres problèmes (*restriction du contenu*, *demande de cookies*, etc.). Ce manque de fiabilité dans un cas réel de production peut avoir un impact sur la qualité de la classification, nous avons donc choisi de constituer un jeu de données réaliste pour l'évaluation de nos méthodes. Ce jeu a été généré à partir de 80 sources différentes liées aux différents métiers de notre cas d'étude. L'approche retenue consiste en deux étapes : (1) obtenir, à partir de sources fournies, la liste des derniers articles (documents) publiés ; et (2) extraire, pour chaque article, le contenu textuel d'intérêt.

Nous avons réparti les sources en deux catégories selon qu'elles mettent ou non à disposition un flux RSS (*Really Simple Syndication*). La bibliothèque Python *feedparser*[14] a été utilisée pour collecter les articles depuis les flux RSS des sources (55 sources concernées). Dans le cas où aucun flux RSS n'est fourni (25 sources concernées), nous utilisons les *xpath*, qui permettent de cibler un ensemble de balises dans le code HTML (*HyperText Markup Language*) d'une page web, pour extraire la liste des articles. Pour se faire, nous nous sommes appuyés sur l'implémentation Python de *Selenium*[15]. L'extraction du contenu d'intérêt des articles est réalisée par la bibliothèque Python *Newspaper3k*[16] dont l'approche, similaire à celle employée par les extensions de lisibilité des navigateurs, consiste à isoler le texte d'intérêt des éléments périphériques en s'appuyant sur les types de balises, la longueur de leurs contenus et leur proximité.

6.2 Campagne d'annotation

Nous avons organisé une campagne d'annotation pour un total de 764 documents : 9 annotateurs humains ont participé à la campagne. Nous rappelons que le corps BL-News contient 5 thématiques métier (*Asset & Fleet*, *Éducation*, *Gammes de Gestion*, *Médico-Social*, *Relation avec les Citoyens*), s'ajoute à cela une thématique transversale, à savoir : l'*Innovation*.

Étant donné un document, l'objectif est d'attribuer manuellement une étiquette « *pertinent* » ou « *non pertinent* » pour chacune des thématiques. De plus, il faut préciser si le document traite un sujet *innovant* ou non. Un document peut être associé à une ou plusieurs thématiques, voire aucune. Les résultats de la campagne sont illustrés dans le tableau 2.

Thématique	Nombre de documents	Taille du lexique en termes-clés
Asset & Fleet	103	180
Éducation	30	98
Gammes de Gestion	173	122
Médico-social	128	**284**
Relation Citoyen	174	216
Innovation	**226**	125
Total des documents	**764**	**1 025**

Table 2 – Description du corpus BL-News

6.3 Corpus anglais de référence

Nous utilisons deux corpus état-de-l'art pour la langue anglaise, à savoir : Reuters-21578 et 20-Newsgroups. Le corpus Reuters-21578 est un ensemble d'actualités, pour le domaine de l'économie, tirées par l'agence de presse Reuters. La version originale de ce corpus contient 21 578 documents d'actualité organisés en 135 catégories. Pour ce corpus, nous avons utilisé la version proposée dans la bibliothèque NLTK[17] (*Natural Language Toolkit*) pour laquelle seulement 90 catégories sont prises en considération en respectant le mode de découpage *ModApte* entre l'apprentissage et le test. Le tableau 3 décrit la liste des 10 classes les mieux couvertes en termes d'exemples par le corpus Reuters-21578.

Numéro	Classe	Apprentissage	Test	Total
1	*earn*	**2 877**	**1 087**	**3 964**
2	*acq*	1 650	719	2 369
3	*money-fx*	538	179	717
4	*grain*	433	149	582
5	*crude*	389	189	578
6	*trade*	368	117	485
7	*interest*	347	131	478
8	*ship*	197	89	286
9	*wheat*	212	71	283
10	*corn*	181	56	237
Total	–	**7 192**	**2 787**	**9 979**

Table 3 – Top-10 des classes du corpus Reuters-21578 avec le nombre de documents de chaque classe

Par rapport à l'ensemble des données du corpus Reuters-21578, le top-10 des classes couvre 74,87 % des données (75,04 % pour l'apprentissage et 74,44 % pour le test). Sachant que le corpus Reuters-21578 traité décrit 90 classes.

Le corpus 20-Newsgroups, quant à lui, se compose de près de 20 000 documents ayant été collectés à partir de 20 groupes différents d'actualités constituant ainsi 20 classes différentes. Pour l'application de nos méthodes sur ce corpus, nous avons opté pour l'utilisation de la version proposée dans la bibliothèque *Scikit-learn*[18] [3]. Le tableau 4 décrit la liste des 10 classes les mieux couvertes en termes d'exemples par le corpus 20-Newsgroups.

[14] https://pypi.org/project/feedparser/
[15] https://selenium-python.readthedocs.io/locating-elements.html
[16] https://github.com/codelucas/newspaper/
[17] https://www.nltk.org/book/ch02.html
[18] https://scikit-learn.org/stable/modules/generated/sklearn.datasets.fetch_20newsgroups.html

Numéro	Classe	App.	Test	Total
1	*rec.sport.hockey*	**600**	**399**	**999**
2	*soc.religion.christian*	599	398	997
3	*rec.motorcycles*	598	398	996
4	*rec.sport.baseball*	597	397	994
5	*sci.crypt*	595	396	991
6	*rec.autos*	594	396	990
7	*sci.med*	594	396	990
8	*comp.windows.x*	593	395	988
9	*sci.space*	593	394	987
10	*comp.os.ms-windows.misc*	591	394	985
Total	–	**5 954**	**3 963**	**9 917**

Table 4 – Top-10 des classes du corpus 20-Newsgroups avec le nombre de documents de chaque classe

Par rapport à l'ensemble des données du corpus 20-Newsgroups, le top-10 des classes couvre 52,62 % des données (avec une même partition entre l'apprentissage et le test). Il est à noter que le corpus 20-Newsgroups que nous traitons décrit 20 classes.

7 Résultats et discussion

Dans cette section, nous présentons tout d'abord les mesures d'évaluation que nous avons utilisées (cf. sous-section 7.1) avant de présenter les résultats obtenus (cf. sous-section 7.2).

7.1 Mesures d'évaluation

Souvent, la précision (*P*), le rappel (*R*) et la F-Mesure (*F*) sont utilisés pour évaluer les performances des systèmes de classification automatique de textes. Nous avons aussi le taux d'exactitude (Taux), appelé souvent en anglais *accuracy*.

La précision permet de répondre à la question : Quelle proportion d'identifications positives est effectivement correcte ? Le rappel, quant à lui, répond à la question : Quelle proportion de résultats positifs réels est identifiée correctement ? La F-Mesure est une moyenne harmonique de la précision et du rappel. Elle permet de mesurer la capacité d'un système à donner toutes les solutions pertinentes et à refuser les autres. Enfin, le taux d'exactitude se fonde sur la distinction « *correct*/*incorrect* », toute nuance ou gradation exclues.

Formellement, la description mathématique de ces mesures pour une étiquette de classe donnée i est présentée dans les équations suivantes :

$$P_i = \frac{TP_i}{TP_i + FP_i} \quad (2)$$

$$R_i = \frac{TP_i}{TP_i + FN_i} \quad (3)$$

$$F_i = \frac{2 \times P_i \times R_i}{P_i + R_i} \quad (4)$$

$$Taux_i = \frac{TP_i + TN_i}{TP_i + TN_i + FP_i + FN_i} \quad (5)$$

Dans les équations, 4 variables sont à déterminer, à savoir : $\mathrm{TP}_i, \mathrm{FP}_i, \mathrm{FN}_i$ *et* TN_i.

- TP_i représente le nombre de documents correctement classés pour la i-ème classe (vrais positifs)
- FP_i représente le nombre de documents qui sont incorrectement classés en i-ème classe (faux positifs)
- FN_i est le nombre de documents qui appartiennent à la i-ème classe, mais qui sont incorrectement classés dans une classe négative (faux négatifs)
- TN_i est le nombre de documents correctement classés dans une classe négative, non i-ème classe (vrais négatifs)

Pour le calcul des mesures sur toutes les classes, la moyenne est obtenue comme suit :

$$P = \frac{\sum_{i=1}^{k} P_i}{k} \quad (6)$$

$$R = \frac{\sum_{i=1}^{k} R_i}{k} \quad (7)$$

$$F = \frac{2 \times P \times R}{P + R} \quad (8)$$

$$Taux = \frac{\sum_{i=1}^{k} Taux_i}{\mathrm{k}} \quad (9)$$

Dans ces équations, 'k' fait référence au nombre de classes.

7.2 Résultats d'évaluation

Nous avons évalué nos méthodes sur le corpus BL-News. Pour les méthodes *TermExtractClassif* et *Class2Class*, la validation croisée avec 10 *folds* est effectuée. Pour la méthode non supervisée *ClassEmbedd*, le test est effectué sur le corpus en intégralité. Les résultats obtenus sont présentés dans le tableau 5.

Système	Type d'évaluation	F	Taux
TermExtractClassif	Validation croisée	52	71
Class2Class	Validation croisée	98	99
ClassEmbedd	Corpus intégral	57	76

Table 5 – Résultats d'évaluation de nos méthodes sur le corpus BL-News

En comparant *TermExtractClassif* avec *ClassEmbedd*, il s'avère que l'utilisation des plongements lexicaux appliqués sur les termes-clés est plus bénéfique pour la classification que la simple fréquence du nombre de documents contenant les termes-clés. De plus, le test avec *ClassEmbedd* est effectué sur tout le corpus BL-News. Par ailleurs, la méthode *Class2Class* permet d'obtenir les meilleures performances, que ce soit pour la F-Mesure ou le taux d'exactitude. En effet, l'utilisation des forêts aléatoires avec plusieurs techniques de réduction de dimensionnalité a été bénéfique pour la classification. Concrètement, après avoir obtenu un ensemble de 96 489 caractéristiques avec TF-IDF, nous sommes passés à une

sélection des 4 000 meilleures caractéristiques. Ensuite, le seuil de validation nous a permis de sélectionner un ensemble de 1 516 caractéristiques. De plus, l'analyse discriminante linéaire a permis d'obtenir le meilleur temps d'exécution, en termes de rapidité (apprentissage et prédiction sur toute la chaîne de traitement), pour la méthode *Class2Class* par rapport aux autres méthodes. Toutefois, cette méthode est supervisée et exige d'avoir un corpus d'apprentissage représentatif. Nous tenons à préciser que le vocabulaire du corpus BL-News a une nature évolutive et un apprentissage statique n'est pas une bonne solution à long terme. Cela nous emmène à valider l'ensemble de nos méthodes sur des corpus de référence anglais pour lesquels le vocabulaire est bien représenté. Nous utilisons les corpus anglais Reuters-21578 et 20-Newsgroups. Les tableaux 6 et 7 décrivent les résultats obtenus sur ces deux corpus, en faisant une comparaison avec sept systèmes état-de-l'art décrits en section 2.

Système	**P**	**R**	**F**	**Taux**
TermExtractClassif	70	61	65	87
Class2Class	85	84	84	98
ClassEmbedd	37	39	28	19
Labani et al. (2018)	77,0	74,0	75,4	–
Jiang et al. (2018)	–	–	–	86,88
Kowsari et al. (2018)	–	–	**90,69**	–
Wu et al. (2017)	–	–	–	**99,07**
Jin et al. (2016)	–	–	88,60	96,50
Dogan et Uysal (2019) SVM	–	–	87,84	–
Dogan et Uysal (2019) Rocchio	–	–	82,20	–

Table 6 – Comparaison des résultats d'évaluation par l'utilisation du corpus Reuters-21578

Système	**P**	**R**	**F**	**Taux**
TermExtractClassif	60	70	65	91
Class2Class	82	82	82	97
ClassEmbedd	64	57	58	60
Labani et al. (2018)	50,10	66,40	57,10	–
Jiang et al. (2018)	–	–	–	83,33
Kowsari et al. (2018)	–	–	87,91	–
Wu et al. (2017)	–	–	–	94,98
Jin et al. (2016)	–	–	82,70	83,10
Jiang et al. (2011)	91,80	81,70	86,46	**98,72**
Dogan et Uysal (2019) SVM	–	–	**98,54**	–
Dogan et Uysal (2019) Rocchio	–	–	98,26	–

Table 7 – Comparaison des résultats d'évaluation par l'utilisation du corpus 20-Newsgroups

La méthode *ClassEmbedd* a été adaptée pour une application sur l'anglais. En effet, nous n'avons pas de lexiques associés aux classes des deux corpus anglais. Pour cela, l'*embedding* d'une classe est représenté seulement avec l'étiquette de la classe. Cela explique d'une certaine manière la dégradation des performances obtenues. Par ailleurs, les techniques de réduction de dimensionnalité dans la méthode *Class2Class* (TF-IDF + Seuil de validation) nous ont permis de passer de 341 233 caractéristiques à 4 000 puis à 1 349 pour le corpus Reuters-21578. Pour le corpus 20-Newsgroups, nous sommes passés de 72 9451 à 4 000 puis à 1 682. Cela permet de voir que le vocabulaire de base du corpus Reuters-21578 est plus riche que celui du corpus 20-Newsgroups. Les résultats que nous avons obtenus pour l'anglais nous confirment que l'utilisation de la méthode *Class2Class* permet de retourner de meilleures performances.

8 Conclusion

Dans cet article, nous nous sommes intéressés à la problématique de classification automatique de textes et plus particulièrement dans un cadre d'application métier. La finalité de ce travail est d'arriver à distinguer les documents traitant de l'innovation ou non et appartenant à l'un des domaines suivants : É*ducation*, *Gammes de Gestion*, *Médico-Social*, *Relation avec les Citoyens* et *Asset & Fleet*. Nous avons présenté trois méthodes de classification dont chacune prend en considération une représentation différente du contenu des documents. Deux des trois méthodes sont supervisées et reposent sur l'utilisation d'un corpus de documents annoté manuellement par des humains. Nous avons évalué nos méthodes sur un corpus français décrivant des articles provenant du web et collectés à partir d'un ensemble de sources prédéfinies. Nous avons appelé ce corpus BL-News pour faire référence aux articles d'actualités traitant en partie de l'innovation sur les secteurs métiers. Ensuite, dans un but de validation de l'efficacité de nos méthodes, nous avons effectué une étude comparative avec sept systèmes état-de-l'art sur deux corpus anglais de référence, à savoir Reuters-21578 et 20-Newsgroups. Les résultats que nous avons obtenus par l'application de nos méthodes sont significatifs et comparables aux systèmes état-de-l'art. Pour l'anglais, nos méthodes offrent de meilleures performances sur le corpus 20-Newsgroups par rapport aux corpus Reuters-21578. Cela vient du fait que la distribution des exemples d'apprentissage dans 20-Newsgroups est équilibrée sur l'ensemble des classes. Toutefois, notre méthode Class2Class reste performante peu importe le niveau de balancement des classes.

Ce travail nous a permis de proposer de nouvelles approches de classification automatique de textes. Même si nous avons validé ces approches sur les corpus Reuters-21578 et 20-Newsgroups, la nature et la taille du corpus BL-News sont totalement différentes. En effet, le corpus BL-News est beaucoup plus petit que les deux corpus anglais. De plus, même si les méthodes à base d'apprentissage supervisé ont apporté de bonnes performances, elles risquent de se retrouver en difficulté avec l'apparition de nouveaux articles ayant un vocabulaire métier plus poussé. Autrement dit, le vocabulaire de nos secteurs métiers est en constante évolution et utiliser seulement une méthode à base d'apprentissage supervisé ne suffit pas puisque plusieurs termes pertinents seront considérés comme des OOV (*Out of Vocabulary*). Pour faire face à ce problème, nous souhaitons combiner l'utilisation de nos méthodes supervisées avec la méthode non supervisée à base

de plongements lexicaux. Cela nous permettrait d'augmenter la base d'apprentissage, d'effectuer des réentraînements de modèles et de prendre de meilleures décisions dans un contexte évolutif.

Remerciements

Nous tenons à remercier toutes les personnes ayant contribué au projet dans sa globalité, à savoir : la collecte des articles, la campagne d'annotation et le développement des différentes méthodes de classification. Aussi, nous souhaitons remercier toute personne ayant contribué à la rédaction et à la relecture (de près ou de loin) de cet article.

Références

[1] F. Béchet, M. El-Bèze, et J. M. Torres-Moréno, En finir avec la confusion des genres pour mieux séparer les thèmes, *Atelier DEFT (Défi Fouille de Textes) - TALN2008*, p. 161-170, 2008.

[2] F. Boudin, pke: an open source python-based keyphrase extraction toolkit, *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations*, p. 69-73, 2016.

[3] L. Buitinck *et al.*, API design for machine learning software: experiences from the scikit-learn project, *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, p. 108-122, 2013.

[4] F. Cozman, I. Cohen, et M. Cirelo, Semi-Supervised Learning of Mixture Models, *Proceedings of the International Conference on Machine Learning (ICML)*, 2003.

[5] A. P. Das, K. Nayak, et M. Nayak, A Survey on Machine Learning Based Text Categorization, *The International Organization of Scientific Research (IOSR) Journal*, 2018.

[6] S. Dey Sarkar, S. Goswami, A. Agarwal, et J. Aktar, A Novel Feature Selection Technique for Text Classification using Naïve Bayes, *International Scholarly Research Notices*, p. 1-10, 2014, doi: 10.1155/2014/717092.

[7] A. Dhar, H. Mukherjee, N. S. Dash, et K. Roy, Text categorization: past and present, *Artificial Intelligence Review*, 2020, doi: 10.1007/s10462-020-09919-1.

[8] T. Dogan et A. K. Uysal, On Term Frequency Factor in Supervised Term Weighting Schemes for Text Classification, *Arabian Journal for Science and Engineering*, p. 9545-9560, 2019, doi: 10.1007/s13369-019-03920-9.

[9] S. K. Dwivedi et C. Arya, Automatic Text Classification in Information retrieval: A Survey, *Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies*, 2016, doi: 10.1145/2905055.2905191.

[10] G. Feng, S. Li, T. Sun, et B. Zhang, A probabilistic model derived term weighting scheme for text classification, *Pattern Recognition Letters*, p. 23-29, 2018.

[11] G. Grouin, J. B. Berthelin, S. El Ayari, M. Hurault-Plantet, et S. Loiseau, Présentation de DEFT'08 (DÉfi Fouille de Textes), *Atelier DEFT (Défi Fouille de Textes) - TALN2008*, p. 1-10, 2008.

[12] D. S. Guru, M. Suhil, L. N. Raju, et N. V. Kumar, An alternative framework for univariate filter-based feature selection for text categorization, *Pattern Recognition Letters*, vol. 103, p. 23-31, 2018, doi: 10.1016/j.patrec.2017.12.025.

[13] M. Honnibal, I. Montani, S. Van Landeghem, et A. Boyd, spaCy: Industrial-strength Natural Language Processing in Python, *Zenodo*, 2020, doi: 10.5281/zenodo.1212303.

[14] J. Y. Jiang, R. J. Liou, et S. J. Lee, A Fuzzy Self-Constructing Feature Clustering Algorithm for Text Classification, *IEEE Transactions on Knowledge and Data Engineering*, p. 335-349, 2011, doi: 10.1109/TKDE.2010.122.

[15] M. Jiang *et al.*, Text classification based on deep belief network and softmax regression, *Neural Computing and Applications*, p. 61-70, 2018, doi: 10.1007/s00521-016-2401-x.

[16] P. Jin, Y. Zhang, X. Chen, et Y. Xia, Bag-of-Embeddings for Text Classification, *Proceedings of the International Joint Conference on Artificial Intelligence*, p. 2824-2830, 2016, doi: 10.5555/3060832.3061016.

[17] K. S. Jones, A statistical interpretation of term specificity and its application in retrieval, *Journal of Documentation*, vol. 28, p. 11-21, 1972.

[18] M. Kaur et V. Kumar, Optimization of Text Classification using Supervised and Unsupervised Learning Approach, *International Journal of Computer Science and Information Technologies (IJCSIT)*, vol. 6, nº 4, p. 3385-3387, 2015.

[19] K. Kowsari, M. Heidarysafa, D. E. Brown, K. J. Meimandi, et L. E. Barnes, RMDL: Random Multimodel Deep Learning for Classification, *Proceedings of the International Conference on Information System and Data Mining*, 2018, doi: 10.1145/3206098.3206111.

[20] M. Labani, P. Moradi, F. Ahmadizar, et M. Jalili, A novel multivariate filter method for feature selection in text classification problems, *Engineering Applications of Artificial Intelligence*, p. 25-37, 2018, doi: 10.1016/j.engappai.2017.12.014.

[21] J. Lehmann *et al.*, DBpedia – A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia, *Semantic Web Journal, IOS Press*, vol. 6, nº 2, p. 167-195, 2015, doi: 10.3233/SW-140134.

[22] R. Nils et G. Iryna, Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, 2019.

[23] C. Wenliang, C. Xingzhi, W. Huizhen, Z. Jingbo, et Y. Tianshun, Automatic Word Custering for Text Categorization Using Global Information, *Proceedings of the Asia Information Retrieval Symposium*, p. 1-11, 2005, doi: 10.1007/978-3-540-31871-2_1.

[24] H. Wu, X. Gu, et Y. Gu, Balancing between over-weighting and under-weighting in supervised term weighting, *Information Processing & Management*, vol. 53, nº 02, p. 547-557, 2017, doi: 10.1016/j.ipm.2016.10.003.