Ontologie et base de connaissances pour le pré-traitement et post-traitement en fouille de données

Laurent Brisson, Martine Collard, Nicolas Pasquier

Ontologie et base de connaissances pour le pré-traitement et post-traitement en fouille de données

Laurent BRISSON,
Martine COLLARD,
Nicolas PASQUIER

Laboratoire I3S - Université de Nice
2000 route des Lucioles
06903 Sophia-Antipolis, France
{brisson,mcollard,pasquier}@i3s.unice.fr
http://www.i3s.unice.fr/execo/

Résumé. Dans cet article, nous présentons la méthodologie EXCIS (Extraction using a Conceptual Information System) orientée ontologie qui permet d'intégrer la connaissance des experts dans un processus de fouille de données. EXCIS décrit les étapes d'un processus à la manière de la méthodologie CRISP-DM, mais son originalité réside dans la construction d'un système d'information conceptuel (CIS) lié au domaine d'application qui permet d'améliorer le pré-traitement des données et l'interprétation des résultats. ExCIS est en cours de développement et cet article présente uniquement la construction du système d'information qui consiste en la création de trois éléments : une ontologie extraite à partir des données brutes, une base de données orientée "fouille" dont les attributs sont les concepts de l'ontologie et une base de connaissance.

1 Introduction

Un des défis de la fouille de données est d'extraire de l'information qui soit intéressante et utile pour les utilisateurs experts. De nombreux algorithmes ont été construits pour extraire les modèles les meilleurs au sens de critères comme la précision, la surface de ROC, le lift ou d'autres mesures. Un certain nombre de travaux portent sur des indices qui mesurent l'intéressabilité des modèles extraits (Hilderman et al., 2001; Liu et al., 1999). Ils distinguent en général l'intérêt objectif de l'intérêt subjectif. La méthode développée par Liu et al. (1999) repose sur la prise en compte des attentes de l'utilisateur. Silberschatz et al. (1995) ont proposé une méthode qui définit l'inattendu à l'aide d'un système de croyances ; dans cette approche, sont définies les croyances faibles que l'utilisateur peut changer si de nouveaux motifs sont découverts et des croyances fortes qui ne peuvent pas être remises en cause.

Dans la plupart des projets de fouille de données, la connaissance a priori est soit implicite, soit organisée dans un système conceptuel structuré. EXCIS est dédié aux situations de fouille de données dans lesquelles la connaissance de l'expert est cruciale pour l'interprétation des motifs extraits, aucune représentation conceptuelle de cette connaissance n'existe et le processus de fouille ne dispose que de bases de données opérationnelles. Dans cette approche, une

ontologie du domaine est construite en analysant les données existantes à l'aide des experts dont le rôle est très important. Le processus de conception de l'ontologie est dirigé en vue de faciliter non seulement la préparation des jeux de données à fouiller, mais également l'interprétation des résultats. L'objectif central dans EXCIS, est de fournir une solution de manière à ce que le processus d'extraction puisse faire usage d'un système d'information conceptuel (CIS : Conceptual Information System) pour optimiser la qualité de la connaissance extraite. Nous considérons le paradigme de CIS comme défini par Stumme (2000). Le CIS fournit la structure d'information utile pour les tâches de fouille qui se succèdent. Il contient un schéma conceptuel définissant une *Ontologie* étendue par une *Base de Connaissance* (ensemble d'informations factuelles sur le domaine d'intérêt) et une *Base de données orientée fouille*(MOBD : Mining-Oriented relational DataBase). L'extraction de l'ontologie et la construction de la base sont également dirigées par l'objectif de fouille.

Les ontologies (Gruber, 2002) fournissent un support formel pour exprimer les croyances (Silberschatz et al., 1995) et la connaissance a priori sur un domaine. Des ontologies ne sont pas disponibles sur tous les domaines ; elles doivent être construites spécifiquement en interrogeant les experts et en analysant les données existantes. L'extraction de structures ontologiques à partir des données est très similaire au processus de recherche d'un schéma conceptuel dans une base de données opérationelle. Différentes méthodes ont été proposées par Kashyap (1999), Johannesson (1994), Stojanovic et al. (2002) et Rubin et al. (2002). Elles sont basées sur le postulat selon lequel la connaissance stockée dans les données relationnelles est suffisante pour produire une construction intelligente de l'ontologie. Elles appliquent en général une correspondance entre concepts ontologiques et tables relationnelles de sorte que l'ontologie est très similaire au schéma conceptuel de la base de données. Dans EXCIS, l'ontologie fournit une représentation conceptuelle du domaine d'application principalement extraite par l'analyse des données opérationnelles. Les caractéristiques principales de la méthodologie sont :

- Conceptualisation de la connaissance implicite : le CIS est conçu pour que les tâches de fouille utilisent l'ontologie, la base de connaissance et la MODB
- Adaptation à la méthodologie CRISP-DM avec :
  - le pré-traitement des jeux de données à l'aide du CIS.
  - le post-traitement des modèles extraits pour filtrer les informations surprenantes et/ou utilisables.
  - l'évolution incrémentale de la connaissance stockée dans le CIS.

Ce projet est actuellement mis en oeuvre dans le cadre d'une étude sur des données de la Caisse Nationale d'Allocations Familiales (CNAF). L'objectif de l'étude est l'amélioration des relations entre l'organisme et les allocataires. Nous disposons de deux sources d'information : une base de données des allocataires et de leurs contacts avec les caisses d'une part et la connaissance des agents des caisses, experts en matière de processus de gestion, comportements et habitudes au sein de l'organisme.

Cet article est organisé comme suit. La Section 2 donne une vue générale de l'approche EXCIS. La Section 3 décrit les structures conceptuelles de l'ontologie. Dans la Section 4, nous donnons une description détaillée de la construction du CIS. La Section 5 est dédiée à la construction de la base de connaissance et aux mécanismes d'inférence à l'aide du serveur Cyc. La Section 6 donne une conclusion.

2 Vue générale de l'approche EXCIS

ExCIS intègre la connaissance experte tout au long du processus de fouille : dans une première phase, la connaissance est structurée et organisée dans le CIS, puis dans les phases qui suivent, elle est exploitée et étendue.

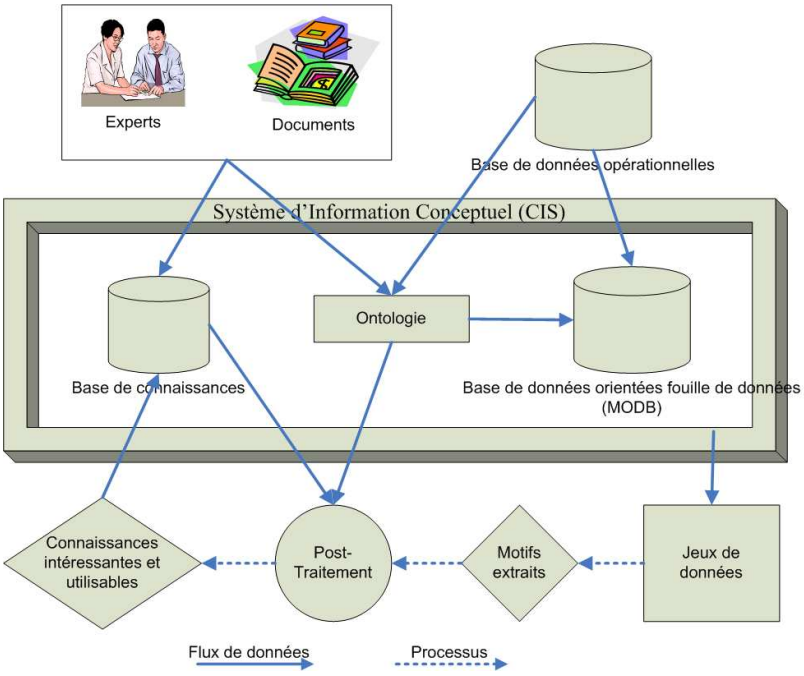

FIG. 1 – *Clustering hiérarchique des conditions.*

Le processus global présenté par la figure 1 montre :

- La construction du CIS où :
  - L'ontologie est extraite en analysant la base de données et en dialoguant avec les experts
  - La base de connaissance est déduite dans un premier temps des dialogues avec les experts
  - La nouvelle base de données MODB est construite.
- L'étape de pré-traitement dans laquelle des jeux de données spécifiques sont construits pour des tâches de fouille particulières
- L'étape de fouille standard où les motifs sont extraits
- L'étape de post-traitement où les motifs découverts peuvent être interprétés et/ou filtrés par comparaison à la fois avec la connaissance a priori stockée dans le CIS et les attentes individuelles des experts.

La MODB est dite générique car elle tient le rôle d'une sorte de réservoir de données à partir duquel des jeux spécifiques peuvent être générés. L'idée sous-jacente dans le CIS est de construire des structures qui procurent plus de souplesse à la fois dans les travaux de pré-

traitement et dans le post-traitement des modèles découverts. Les structures hiérarchiques et les liens de généralisation/spécialisation entre les concepts ontologiques jouent un rôle central :

- Ils permettent de réduire la taille des modèles extraits comme les ensembles de règles souvent volumineux
- Ils fournissent également un outil pour l'interprétation des résultats (classes) d'algorithmes de classification non supervisée.

Pour des données numériques ou catégorielles, ils fournissent des niveaux de granularité différents très utiles dans le pré-traitement et dans le post-traitement.

Exemple Supposons que 10 et 11 soient définis dans l'ontologie comme des concepts (NUMJOUR) qui héritent d'un concept plus général «NUMSEMAINE=2», alors les règles Règle1 et Règle2 peuvent être généralisées et remplacées par la Règle3 dont la partie gauche est plus générale.

```
Règle 1: If NUMJOUR = 10 and MOTIF = "Appel Entrant"
            then OBJECTIF = "Demande credit"
Règle 2: If NUMJOUR = 11 and MOTIF = "Appel Entrant"
            then OBJECTIF = "Demande credit"
Règle 3: If NUMSEMAINE = 2 and MOTIF = 'Appel Entrant"
            then OBJECTIF = "Demande credit"
```

3 Structures conceptuelles de l'ontologie

3.1 Ontologie

Dans l'approche EXCIS, l'ontologie du domaine est un outil essentiel à la fois pour améliorer le processus de fouille et pour interpréter ses résultats. L'ontologie est définie par un ensemble de concepts et de relations entre concepts qui sont découverts en analysant les données. Elle apporte un soutien dans l'étape de pré-traitement pour construire la MODB et dans le post-traitement pour raffiner les motifs extraits. Comme montré par la figure 2, les relations de généralisation/spécialisation entre concepts ontologiques fournissent une information importante ; ils peuvent être largement exploités pour réduire la taille des motifs découverts. Par exemple, un ensemble de règles de dépendances (règles attribut-valeur) peut être réduit par généralisation sur les attributs ou par généralisation sur les valeurs. Aussi, les règles de construction de l'ontologie sont les suivantes :

- Distinguer concept-attribut et concept-valeur.
- Etablir une correspondance entre attributs sources et concepts attributs d'une part et valeurs sources et concepts valeurs d'autre part
- Définir des hiérarchies de concepts.

Cette ontologie a deux caractéristiques importantes inhérentes à l'objectif de foullle de données :

- Elle ne contient aucune instance puisque les valeurs sont organisées en hiérarchies et considérées comme des concepts ; les instances sont uniquement présentes dans la base de données finale MODB.
- Chaque concept a uniquement deux propriétés génériques.

La MODB est une base de données relationnelle dont le rôle est de stocker les données de granularité plus fine issues de la base de données opérationnelle. Les attributs de la MODB sont ceux qui sont identifiés comme pertinents a priori pour la fouille et ses instances sont des enregistrements de valeurs de granularité le plus fine.

3.2 Notion de concept

Dans une ontologie EXCIS, un concept doit faire référence à un paradigme du domaine utile pour le processus de fouille. Un concept d'EXCIS est caractérisé par les deux propriétés suivantes : son rôle dans un motif extrait (attribut ou valeur) et une propriété booléenne qui indique sa présence dans la MODB. Un *concept-attribut* correspond à une propriété d'une donnée initiale et un *concept-valeur* correspond aux valeurs d'une telle propriété. Un concept qui est présent dans la MODB est appelé un *concept concret*. Un concept qui n'est pas concret, mais est utile pendant l'étape de post-traitement est appelé un *concept abstrait*. Par exemple sur la figure 2 «Nombre Enfants» est un concept-attribut abstrait et «3 Enfants» est un concept-valeur concret.

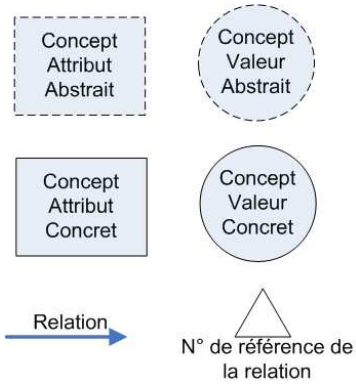

FIG. 2 – *Légende*

3.3 Relation entre concepts

Une relation est un lien orienté entre deux concepts. Etant donné que 4 types différents de concepts existent dans EXCIS et que nous distinguons les relations entre concepts d'une même hiérarchie et entre concepts de hiérarchies différentes, nous avons 32 sortes de relations entre concepts. Parmi ces relations, se distinguent particulièrement 3 cas :

- Relations de généralisation/spécialisation entre deux concepts-valeur qui sont des liens «est-une-sorte-de» entre 2 concepts-valeur (voir la relation 2 sur la figure 3)
- Relations de généralisation/spécialisation entre deux concepts-attribut qui sont des liens «est-une-sorte-de» entre 2 concepts-attribut (voir la relation 8 sur la figure 3)
- Relations de généralisation/spécialisation entre un concept-attribut et un concept-valeur qui sont des liens «est-une-valeur-de» (voir la relation 6 sur la figure 3)

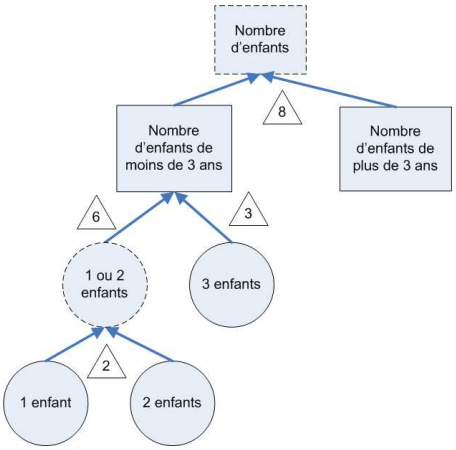

FIG. 3 – *Concepts relatifs aux enfants*

Les relations entre concepts à l'intérieur de la même hiérarchie sont énumérées dans la table 1 et les relations entre concepts de hiérarchies différentes sont énumérées dans la table 1. Les relations autorisées sont numérotées, les relations interdites sont représentées par une lettre.

TAB. 1 – *Relations entre concepts de la même hiérarchie*

Concept	Concret Valeur	Abstrait Valeur	Concret Attribut	Abstrait Attribut
Valeur Concret	1	2	3	D
Valeur Abstrait	B	2	6	5
Attribut Concret	C	C	9	7,8
Attribut Abstrait	C	C	A	10

TAB. 2 – *Relations entre concepts de hiérarchies différentes*

Concept	Concret Valeur	Abstrait Valeur	Concret Attribut	Abstrait Attribut
Valeur Concret	4	4	D	5
Valeur Abstrait	B	4	D	5
Attribut Concret	C	C	D	D
Attribut Abstrait	C	C	A	D

3.4 Description et utilisation des relations ontologiques

Avant tout, deux relations sont interdites dans EXCIS : toute relation de généralisation/spécialisation d'un concept-abstrait vers un concept concret qui a le même rôle (attribut ou valeur) car les

concepts abstraits ont été définis pour être plus généraux que les concepts concrets (voir les relations A ou B sur la figure 4), et toute relation de généralisation/spécialisation d'un concept-attribut vers un concept-valeur (voir la relation C sur la figure 4) car la relation «est-une-valeur-de» n'a pas de signification dans ce cas.

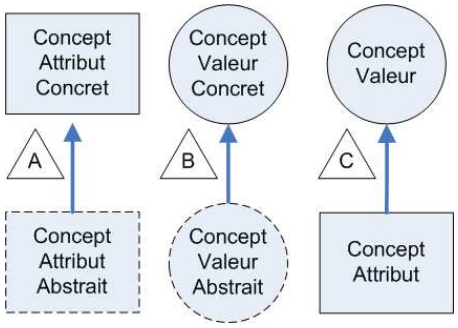

FIG. 4 – *Relations interdites*

3.4.1 Relations entre concepts-valeur

La généralisation ou la spécialisation entre concepts-valeur (voir relation 2 sur la figure 3) sont utiles de manière à généraliser des motifs pendant la phase de post-traitement. De plus, les relations entre deux concepts-valeur concrets de la même hiérarchie sont essentiels car ils permettent de choisir différents grains dans les données issues de la MODB. Si, par exemple, dans une session du processus de fouille, nous nous intéressons plus particulièrement aux types d'allocations, le grain des données sera choisi au niveau «Allocation Logement» (voir relation 1 sur la figure 5).

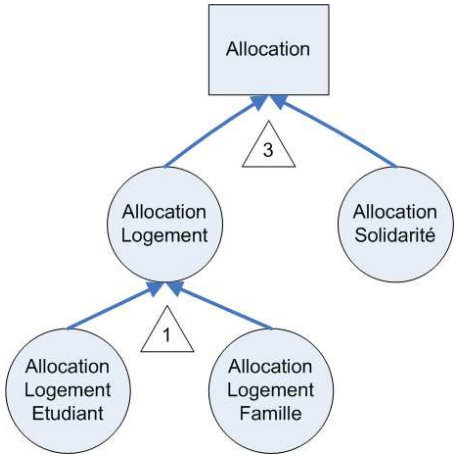

FIG. 5 – *Concepts relatifs aux allocations*

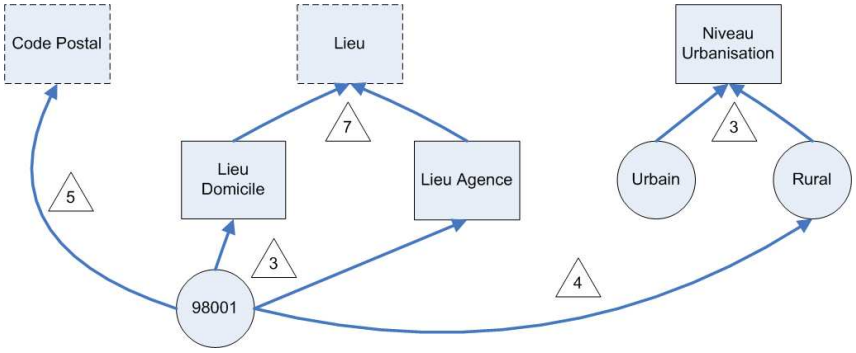

FIG. 6 – *Concepts relatifs à la localisation*

3.4.2 Relations entre concepts-attribut

La généralisation ou la spécialisation entre concepts-attribut sont également utiles de manière à généraliser des motifs pendant la phase de post-traitement. Cependant, ceci demande de procéder avec précaution car dans certains cas un attribut doit être remplacé par un autre attribut plus général (voir relation 7 sur la figure 6) et dans d'autres cas, une nouvelle valeur doit être calculée (voir relation 8 sur la figure 3). Par exemple, sur la figure 3 «Nombre Enfants» est la somme des valeurs de tous ses sous-concepts.

Les relations entre deux concepts-attribut concrets d'une même hiérarchie sont spécifiques ; elles doivent en effet être vérifiées pendant la génération des jeux de données.

La méthode EXCIS interdit des relations entre concepts-attribut de différentes hiérarchies Les relations entre concepts-attribut sont uniquement des relations de généralisation ; les concepts qui sont sémantiquement proches doivent être placés dans la même hiérarchie. Par exemple, «Lieu Domicile» et «Lieu Agence» sont dans la même hiérarchie (voir relation 7 sur la figure 6).

3.4.3 Relations entre concepts-valeur et concepts-attribut

Ces relations sont essentielles pour construire les données et fournir des vues sémantiques différentes pendant la phase de post-traitement (voir relation 5 sur la figure 6). Par exemple «98001» est à la fois un «Lieu Domicile» et un «Code Postal» (voir relation 3 sur la figure 6).

Chaque concept-valeur est lié à des concepts-attribut dans la même hiérarchie. EXCIS interdit des relations entre un concept-valeur et des concepts-attribut concrets de différentes hiérarchies ; en effet, si une telle relation existait, cela signifierait qu'un concept-valeur «est-une-valeur-de» deux différents concepts-attribut. Si ces concepts-attribut sont sémantiquement proches, ils doivent être placés dans une même hiérarchie et s'ils sont totalement différents, ils ne peuvent pas être en relation avec les même concepts-valeur.

4 Le Système d'Information Conceptuel (CIS)

Soit A l'ensemble des attributs de la base de données d'origine, C l'ensemble des concepts de l'ontologie et C_z l'ensemble des concepts associés à l'attribut $z \in A$. C est défini par $\bigcup_{z \in A} C_z$.

ExCIS diffère de CRISP-DM lors de la phase de préparation des données. Dans cette phase CRISP-DM décrit cinq tâches : sélection, nettoyage, construction, intégration et formatage des données. Les tâches de sélection et de formattage sont identiques dans les deux méthodes. Cependant dans ExCIS les tâches de nettoyage, construction et intégration des données sont fusionnées afin de pouvoir créer les concepts de l'ontologie et construire la MODB.

4.1 Définition de la portée et sélection des attribut d'origine

Les premières étapes de la méthode ExCIS correspondent aux étapes de compréhension du domaine et de compréhension des données de la méthode CRISP-DM. Ces étapes nécessitent une interaction importante avec les experts du domaine.

1. Déterminer les objectifs : par exemple dans notre cas d'application l'objectif est d'améliorer les «relations avec l'allocataire».

2. Définir des thèmes : en analysant les données nous pouvons les regrouper en ensembles sémantiques que l'on appelle thèmes. Par exemple nous avons créé trois thèmes : les profils allocataire, les contacts (par téléphone, courier, couriel, à l'agence, ...) et les évèvements (vacances, rentrée scolaire, naissance, mariage, ...).

3. Associer à chaque thème un ensemble d'attributs avec l'aide des experts.

4.2 Analyse des données et création des concepts attributs

4. Pour chaque attribut z :

5. Considérer son nom et sa description pour :
- Associer n concepts à l'attribut.
- Nettoyer C des homonymes (concepts différents avec le même nom), des synonymes (même concepts mais noms différents comme l'âge et la date de naissance par exemple), et des attributs inutiles selon nos objectifs.

6. Etudier les valeurs (distribution, valeurs manquantes, doublons, ...) afin de :
- Raffiner C_z (ajout ou suppression de concepts) selon les informations obtenues lors de l'étude.
- Nettoyer de nouveau les homonymes, synonymes et attributs inutiles. Par exemple, en analysant les valeurs nous nous sommes rendus compte que «Allocations» était en fait 2 concepts homonymes. Ainsi nous avons créé le concept «Montant de l'allocation» et le concept «Bénéficiaire de l'allocation».

7. Pour chaque concept associé à z, créer la procédure qui générera les concepts valeurs.

Lors de l'étape 7, si le concept attribut n'existe pas nous devons créer une table avec quatre champs. Ces champs sont l'attribut associé au concept, le nom de la table contenant l'attribut dans la base de données d'origine, le domaine de valeur de l'attribut et la référence à la procédure qui générera les concepts valeurs. Il y a une unique procédure par enregistrement

dans la table. Un domaine de valeur peut être une valeur distincte ou une expression régulière ; c'est également l'entrée de la procédure. En sortie la procédure retourne des références sur les concepts valeurs. Cette procédure peut être une requête SQL ou un programme externe (script shell, C, ...). Cependant, si le concept attribut existe déjà nous avons juste à ajouter un enregistrement à la table et créer une nouvelle procédure.

4.3 Création des concepts valeurs

Lors de cette étape, toutes les procédures pour générer les concepts valeurs sont créées.

8. Donner un nom à chaque concept valeur.

9. Nettoyer homonymes et synonymes parmi les concepts valeurs.

4.4 Construction de l'ontologie

10. Identifier les relations de généralisation parmi les concepts valeurs (voir figure 5).

11. Créer les concepts abstraits et réorganiser l'ontologie avec ces nouveaux concepts. Par exemple le concept «Location» sur la figure 6.

12. Créer les relations entre concepts valeurs de hiérarchies différentes (voir relation 4 figure 6).

4.5 Construction de la base de données orientée pour la fouille (MODB)

13. Générer la base de données en utilisant les procédures définies lors de l'étape 7.

Lors de cette dernière étape un programme lit les tables créées pour chaque concept attribut et exécute chacune des procédures afin de générer la MODB.

5 La Base de Connaissances

5.1 Le serveur de connaissances Cyc

Le serveur de connaissances Cyc est constitué d'une base de connaissances et d'un moteur d'inférence développé par Cycorp[1]. Le but de Cyc est de pouvoir rassembler un grand nombre de connaissances de «sens commun» afin d'aider toutes les applications d'intelligence artificielle et de traitement du langage naturel. Le serveur Cyc est construit autour des composants suivants : une base de connaissances, un moteur d'inférences et le langage de représentation CycL.

Le serveur Cyc nous permet de construire et de gérer l'ontologie ainsi que la base de connaissance du CIS. Nous avons choisi la technologie Cyc après une étude de langages (OWL, DAML+OIL, Frame Logic, ...) existants et de leurs outils. Cyc repose sur deux notions fondamentales : les collections et les individus. Une collection est une type de chose, une classe de choses. Les choses qui appartiennent à une collection sont appellées ses instances. A l'opposé,

[1] http ://www.cyc.com/

un individu est une chose atomique. Ces deux notions correspondent à nos besoins car dans notre ontologie l'instance d'un concept peut avoir ses propres instances. Une autre notion clef pour gérer les connaissances avec CycL est la notion de Microthéorie. Une microthéorie est un ensemble d'assertions extraites de la base de connaissances. Un des principales fonctions des microthéories est de séparer les assertions en ensembles consistants dans lesquels il n'existe aucune contradiction. Toutefois il possible qu'il y ait des contradictions entre assertions de microthéories différentes. Par conséquence, nous pouvons utiliser les microthéories afin de gérer les contradictions entre experts et de prendre en compte les opinions divergeantes des utilisateurs. Pour finir, la base de connaissance de «sens commun» de Cyc est une fonctionnalité intéressante puisque nous travaillons actuellement sur des données «sociales» fournit par les caisse d'allocation familiales.

5.2 Définition des relations entre concepts en CycL

Dans la section 3.4 nous avons montré qu'il existe trois sortes de relations utiles à l'analyse des résultats de la fouille de données. Ces relations peuvent être définies dans le serveur de connaissances Cyc en utilisant les relations binaires prédéfinies (isa, $genls$) ou des relations nouvelles à définir comme $valeurDe$ et $relationAvec$ présentées ci-dessous. Pour chacune d'entre elles nous pouvons choisir définir propriétés parmi : la reflexivité, l'irréflexivité, la symétrie, l'anti-symétrie, l'assymétrie et la transitivité.

5.2.1 isa : Une relation pour décrire les propriétés des concepts

En CycL, le prédicat isa est utilisé pour exprimer qu'une chose est instance d'une collection. Une expression de la forme *(isa X Y)* signifie que X est une instance de la collection Y. Nous utilisons la relation isa afin de décrire les concepts des propriétés : leur rôle (attribut/valeur) et leur présence dans le MODB (abstrait/concret). Ainsi, nous avons créé tous les concepts de la figure 2 dans une microthéorie appellée DataMiningMt qui diffère de la microtheorie du domaine étudié.

5.2.2 genls : Relations entre concepts attributs et concepts valeurs de la même hiérarchie

En CycL, $genls$ est utilisé pour dire qu'une collection est inclue dans une autre. Une expression de la forme *(genls X Y)* signifie que chaque instance de la collection X est aussi une instance de la collection Y. Nous utilisons $genls$ afin de définir les relations entre concepts attributs (voir les relations 7,8,9,10 dans la table 1) et les concepts valeurs de la même hiérarchie (voir les relations 1,2 dans la table 1).

5.2.3 valeurDe : Relations entre concepts valeurs et concepts attributs

Nous avons défini la relation $valeurDe$, irreflexive et assymétrique, pour représenter les relations 3,5,6 (voir table 1 et 2). Afin de permettre l'héritage à travers les relations $genls$ nous

avons dû définir de nouvelles assertions pour le moteur d'inférences Cyc à l'aide de la relation $implies$

5.2.4 relationAvec : Relations entre concepts valeurs de différentes hiérarchies

Nous avons défini la relation $relationAvec$, qui est uniquement transitive, pour représenter la relation 4 de la table 2. Afin de permettre au moteur d'inférence de traiter tous les concepts liés successivement par cette relation il est également nécessaire de créer de nouvelles assertions.

5.3 Un moteur d'inférence pour améliorer les résultats de la fouille de données

Notre objectif principal en développant le CIS est de fournir un ensemble d'outils afin d'analyser les résultats d'une fouille de données à un niveau sémantique. Une première étape est de réécrire les règles générées afin de les simplifier selon les relations définies dans l'ontologie et d'offrir à l'utilisateur un outil intuitif pour explorer ces règles. Dans une seconde étape, l'utilisateur devra exprimer ses connaissances au sein d'une microthéorie et notre algorithme sélectionnara les règles les plus intéressantes *en fonction des connaissances de l'utilisateur*. Actuellement la première étape est en cours de developpement. Pour l'illustrer voici un exemple :

Considérons les règles définies en section 2 où «NUMSEMAINE=2» est un sur-ensemble de «NUMJOUR=10» et «NUMJOUR=11» :

Afin de simplifier l'écriture de nos règles nous utilisons la notation suivante : soit A l'item NUMSEMAINE=2, A1 l'item NUMJOUR=10, A2 l'item NUMJOUR=11, B l'item MOTIF="APPEL ENTRANT" et C l'item OBJECTIF="DEMANDE CREDIT".

Les règles 1 et 2 sont défines par les axiomes suivants :

```
(rule (TheList A1 B C))
(rule (TheList A2 B C))
```

Définissons l'assertion suivante :

```
(implies
       (and
           (rule (TheList ?X B C))
           (rule (TheList ?Y B C))
           (genls ?X ?Z)
           (genls ?Y ?Z))
       (rule (TheList ?Z B C)))
```

Avec cette requête *(rule ?X)* le système Cyc retourne le résultat suivant, qui est la règle 3 déduite des règles 1 et 2 :

```
?X : (TheList A B C)
```

6 Conclusion

Nous avons présenté une nouvelle méthodologie ExCIS qui permet l'intégration de la connaissance des experts d'un domaine dans le processus de fouille de données. L'objectif principal est d'améliorer la qualité de la connaissance extraite et de faciliter son interprétation. ExCIS est basé sur un système d'information conceptuel (CIS) qui stocke la connaissance des experts. Le CIS joue un rôle central dans la méthodologie car il est utilisé pour générer des jeux de données avant la fouille, pour filtrer et interpréter les modèles obtenus et pour mettre à jour la connaissance des experts. Ce papier est essentiellement consacré à la description de la structure du CIS et de sa construction et évoque la manière dont il peut être utilisé pour améliorer les résultats de la fouille de données. Nous avons montré les structures ontologiques du CIS, et décrit les choix effectués pour identifier les concepts et les relations de l'ontologie en analysant des données opérationnelles. Nos travaux futurs seront consacrés au développement des techniques exploitant les informations de l'ontologie afin d'interpréter les résultats de la fouille.

7 Remerciements

Nous désirons remercier la CNAF et plus spécialement Pierre Bourgeot, Cyril Broilliard, Jacques Faveeuw, Hugues Saniel et le BGPEO pour avoir soutenu ce travail.

Références

T. Gruber (2002). *What is an Ontology ?*. http ://www-ksl.stanford.edu/kst/what-is-an-ontology.htm.

R.J. Hilderman et H.J. Hamilton (2001). *Evaluation of Interestingness Measures for Ranking Discovered Knowledge*. Proceedings 5th PAKDD conference, Lecture Notes in Computer Science 2035 :247-259.

P. Johannesson (1994). *A Method for Transforming Relational Schemas into Conceptual Schemas*. Proceedings 10th ICDE conference, M. Rusinkiewicz editor, pp. 115-122, IEEE Press.

V. Kashyap (1999). *Design and Creation of Ontologies for Environmental Information Retrieval*. Proceedings 12th workshop on Knowledge Acquisition, Modelling and Management.

B. Liu, W. Hsu, L.-F. Mun et H.-Y. Lee (1999). *Finding Interesting Patterns using User Expectations*. Knowledge and Data Engineering, 11(6) :817-832.

D.L. Rubin, M. Hewett, D.E. Oliver, T.E. Klein et R.B. Altman (2002). *Automatic Data Acquisition into Ontologies from Pharmacogenetics Relational Data Sources using Declarative Object Definitions and XML*. Proceedings 7th Pacific Symposium on Biocomputing, pp. 88-99.

A. Silberschatz et A. Tuzhilin (1995). *On Subjective Measures of Interestingness in Knowledge Discovery*. Proceedings of the First International Conference on Knowledge Discovery and Data Mining, 275-281.

L. Stojanovic, N. Stojanovic et R. Volz (2002). *Migrating Data-intensive Web Sites into the Semantic Web*. Proceedings 17th ACM Symposium on Applied Computing, pp. 1100-1107, ACM Press.

G. Stumme (2000). *Conceptual On-Line Analytical Processing*. K. Tanaka, S. Ghandeharizadeh et Y. Kambayashi editors. Information Organization and Databases, chpt. 14, Kluwer Academic Publishers, pp 191-203.

Summary

In this paper, we present the new ontology-based methodology ExCIS (Extraction using a Conceptual Information System) for integrating expert prior knowledge in a data mining process. This methodology describes guidelines for a data mining process like CRISP-DM. Its originality is to build a specific Conceptual Information System related to the application domain in order to improve datasets preparation and results interpretation.