Prise en compte des « Points de Vue » pour l'annotation d'un processus d'Extraction de Connaissances à partir de Données

Hicham Behja(*),(**),(***), Brigitte Trousse(**), Abdelaziz Marzak(***)

(*) ENSAM Meknes Marjane 2 ; B.P. 4024
Beni Mhammed Meknes Maroc
(**) INRIA Sophia Antipolis Projet AxIS,
BP 93, 06902 Sophia Antipolis, France
(***){Nom.Prenom}@sophia.inria.fr
(***) Faculté des Sciences Ben M'sik de Casablanca
Avenue Driss Harti B.P 6621 Casablanca, Maroc
marzak@hotmail.com

Résumé. Dans cet article on propose une nouvelle approche qui rend explicite la notion de point de vue dans une analyse multivues issue d'un processus d'Extraction de Connaissances à partir de Données (ECD). Par point de vue, nous entendons la vision particulière d'un analyste lors de son processus ECD, vision référant à un corps de connaissances qui lui est spécifique. On cherche, d'une part, à faciliter la réutilisabilité et l'adaptabilité du processus, et d'autre part à garder une trace des points de vues sous-jacents aux analyses faites. Le processus d'ECD sera vu comme un processus de génération et de transformation de vues qui seront annotées par des métadonnées pour garder la sémantique de la connaissance extraite. Un positionnement de notre approche vis-à-vis des travaux méthodologiques du processus d'ECD sera donné. Des éléments de modélisation du processus ECD basé sur les points de vue seront décrits au niveau ontologique. Enfin, on illustrera notre approche sur l'analyse des usages d'un site web à partir des fichiers log, selon le point de vue fiabilité.

1 Introduction

Le processus d'ECD est un processus itératif et interactif, constitué principalement de trois grandes étapes: prétraitement, fouille de données et postraitement (Fayyad *et al.,* 1996(a)). Il se présente comme un processus complexe tant au niveau des techniques et méthodes qu'au niveau des données manipulées (Gancarski et Trousse, 2004).

Dans le cadre de nos recherches sur l'analyse des usages d'un système d'information (Tanasa et Trousse, 2004), nous nous intéressons au processus d'ECD appliqué aux données Web. Ces données peuvent être hétérogènes (textes, hypertexte, images, vidéo, etc.), incohérentes, évolutives, incomplètes mais en général, on peut dire qu'elles sont bien structurées dans le sens où elles respectent un format bien connu par les analystes d'ECD (comme le format CLF «Common Log File »). Mais cette structure reste relative et liée au point de vue de l'analyste d'ECD. Les mêmes données peuvent être vues mal structurées du point de vue de l'analyste des comportements des utilisateurs d'un système d'information (SI) basé sur le Web. En effet ces données sont relativement pauvres pour répondre aux objectifs de l'analyste en termes de comportement utilisateur. Cette approche nécessite de plus amples informations sur les utilisateurs qui se présentent comme les acteurs centraux d'une analyse des comportements d'un SI basé sur le Web. Ces objectifs passent,

essentiellement, par la définition d’ontologies ou de métadonnées qui annotent les attributs mis en jeu, ce qui est peu souvent disponible.

Par ailleurs, le processus d’ECD est mené par un ou plusieurs experts et par conséquent plusieurs types de connaissances et de savoir faire sont mis en jeu. Ces experts ont la tâche, d’une part, de réduire la masse d’informations à traiter, surtout dans l’étape de prétraitement et d’autre part, d’orienter et de sélectionner les meilleures solutions et configurations des plans d’exécutions durant le cycle de l’extraction de la connaissance.

L’intégration des deux types de connaissances, à la fois du domaine analysé et du domaine de l’analyste (Cf. Fig. 1), manipulés dans un processus d’ECD sont très importantes dans la découverte de nouvelles connaissances. Si la connaissance du domaine analysé est variée, il en est de même des analystes d’ECD qui vont la manipuler : chacun a ses intérêts particuliers ; chacun regarde des propriétés, méthodes particulières des objets du processus et chacun organise les objets selon sa propre vision. Ceci passe par l’annotation du processus d’extraction via des métadonnées et plus généralement par la construction d’ontologies.

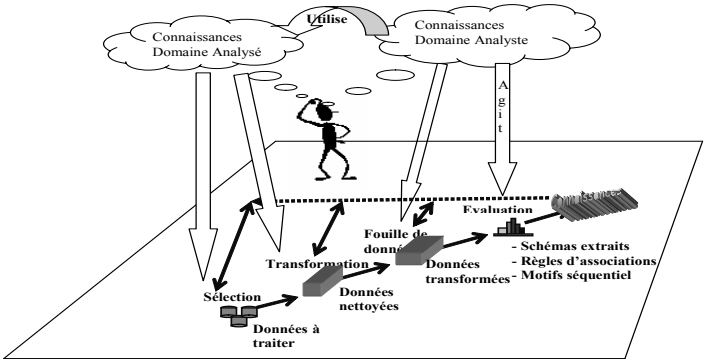

Fig. 1 –*ECD un processus complexe*

L’intégration du savoir-faire de l’expert dans le cycle d’ECD que nous proposons devra prendre en compte toutes les phases du processus de l’extraction, proposer une modélisation du processus d’ECD permettant de capitaliser la connaissance dans toutes ses étapes. En prenant en considération l’évolution et la réutilisabilité du processus on doit pouvoir planifier, ordonner, contrôler et gérer le processus. Pour cela, une modélisation multivues dans une activité d’analyse s’avère très prometteuse pour répondre à nos objectifs.

Dans cet article, on présente en section 2 un positionnement par rapport aux travaux sur l’annotation dans un processus d’ECD. La section 3 présente les définitions de base des différents concepts manipulés ainsi que le principe d’une annotation d’un processus d’ECD en termes de points de vue. Les sections 4 et 5 présentent notre modèle conceptuel des connaissances manipulées en ECD ainsi que des éléments de notre plateforme.

2 Travaux en annotations du processus d’ECD

Les travaux sur l’annotation dans un processus d’ECD, se sont focalisés sur l’acquisition, la spécification et la manipulation des connaissances du domaine analysé pourvue une utilisation et une définition des plans d’exécution spécifiés par l’analyste du processus.

La connaissance du domaine analysé dans le processus d'ECD peut concerner la sémantique des données (métadonnées) de la base de données considérée, le modèle du domaine d'application, le contexte dans lequel ces données ont été générées et le but du processus d'ECD (Piatetsky-Shapiro et Frawley, 1991). Or, les métadonnées sont décrites soit d'une façon littérale soit par des attributs qui décrivent les propriétés intrinsèques des entités considérées. L'utilisation de la connaissance sur les données manipulées dans un système d'extraction est extrêmement importante. En effet cette connaissance permet, tout d'abord, bien souvent de *limiter l'espace de recherche* des données (Yoon *et al*, 1999) afin de ne pas mobiliser, à chaque requête, l'intégralité de la base de données pour retrouver des éléments spécifiques d'usage courant. D'autre part elle permet via des métadonnées et connaissances sur les données (dictionnaire de mots, thesaurus, …) (Devedzic, 2001) de structurer, contrôler et fusionner les données (Anand, 1995) de la base de données. Ces métadonnées peuvent indiquer quelles sont les données disponibles, où elles se trouvent, ce qu'elles signifient (nom, type, unités de mesures, composants,…) comment y accéder, etc. Enfin ces connaissances sont utilisées lors de la validation et l'intégration d'une nouvelle connaissance du domaine issue du processus d'ECD. En d'autres termes, les nouvelles connaissances extraites du processus d'ECD font partie de la connaissance du domaine analysé (Devedzic, 2001).

Cependant, concernant les travaux utilisant une approche connaissance sur le domaine de l'analyste on distingue ceux qui utilisent l'annotation dans la phase des prétraitements : les travaux de Hotho et le projet Mining Mart. Dans (Hotho *et al*., 2002) les auteurs formalisent le domaine analysé par une ontologie du domaine associée aux techniques d'extraction d'information dans la fouille de textes et l'interprétation des règles extraites. Les auteurs utilisent une taxonomie pour réduire la dimension des données manipulées pour réduire le coût des prétraitements. Le projet MiningMart (Morik et Scholz, 2003) propose un système de raisonnement à partir de cas (CBR) pour une bonne exploitation et réutilisation du processus. Cette approche consiste à élaborer une base de données des opérateurs pour la phase des prétraitements (discrétisation, agrégation, nettoyage, regroupement des séquences…) et sera décrite par des métadonnées et des ontologies qui serviront dans les prochaines fouilles définies par les utilisateurs. Dans la phase de fouille de données, on trouve le système Damon (Cannatro et Comito, 2003) qui est une ontologie pour le domaine de la fouille de données. Il consiste à simplifier le développement des applications d`ECD, en permettant à l`expert de choisir le modèle convenable. Dans la phase de déploiement, l'approche KAIMAN (Pohle, 2003) consiste à modéliser et à intégrer la connaissance extraite durant le cycle d'extraction. Du fait de la variété des représentations et les modèles de connaissances extraites, KAIMAN propose de les transformer en une représentation générique en termes d'ontologies qui décrit les règles extraites, dans le but de faciliter leur intégration dans le domaine analysé.

On trouve aussi des travaux où l'annotation concerne toutes les phases du processus d'ECD, éventuellement le projet IDA (Intelligent Discovery Assistants) (Bernstein *et al.*, 2002) et l'approche GLS (Global Learning Scheme) (Zhong et al., 2001). Le projet IDA propose une aide aux utilisateurs pour exploiter l'espace des données manipulées. IDA interagit avec les utilisateurs pour obtenir les données, les métadonnées et les desiderata. Ensuite il extrait les plans d'exécutions valides d'un point de vue syntaxique sur l'agrégation des méthodes et la succession des algorithmes utilisés lors d'une analyse issue d'un expert d'ECD. Tandis que l'approche GLS propose un système basé sur la technologie des systèmes multi-agents pour augmenter l'autonomie et l'adaptabilité d'un système d'extraction de la

connaissance par une organisation dynamique du processus d’ECD. GLS associe à chaque agent le contrôle de chaque phase du processus pour augmenter la coopération dans un domaine distribué. Chaque agent est décrit par une ontologie qui spécifie les types des entrées/sorties de chacun des agents les préconditions, les effets de leurs exécutions, leurs fonctionnements, et leurs placements dans une hiérarchie des agents avec qui ils coopèrent.

Tandis que pour les efforts de standardisation dans les langages d’ECD pour la description des opérateurs, on retrouve le langage PMML « Predictive Model Markup Language ». PMML[1] est un standard du DMG (Data Mining Group) qui permet à ses utilisateurs d’intégrer des modèles mathématiques et surtout de faire leurs mises à jour dans leurs applications sans manipulations complexes. Il utilise une taxonomie hiérarchique pour représenter le domaine analysé basé sur les relations Père/Fils. Mais cette représentation reste limitée lorsqu’il s’agit de manipuler des données complexes où les relations d’héritage sont insuffisantes pour les représenter.

Cependant, dans un processus d’ECD, il n’existe aucune approche, à notre connaissance, qui permette d’annoter le processus d’ECD en termes de point de vue et d’objectifs poursuivis par l’analyste i.e. d’annotations orientées processus (voire explication du processus) plutôt qu’orientées données. Aussi nous proposons une nouvelle approche qui revisite le cycle de l’ECD en terme de points de vue.

3 Notre approche

3.1 Travaux sur les points de vue

La notion des points de vue a été introduite depuis les années 70. Depuis, plusieurs travaux et notamment plusieurs séminaires ont été organisés (Spanoudakis, 1996) (OOPSLA, 2003), et bien d’autres qui ont été menés sur leur signification, représentation, interprétation.

Les points de vue ont été utilisés dans différents domaines ; dans les méthodes et langages orientés objets ObjLog (Dugerdil, 1988), VBOOL (Marcaillou, 1995), dans les bases de données objets O2VIEWS (Santos, 1995), (Abiteboul et Bonner, 1995), dans le processus de développement (Finkelstein et al. 1990).

Le concept de point de vue a été aussi abordé parfois implicitement par une large communauté de l’ingénierie des connaissances ; notamment dans les langages de représentation de la connaissance à objet, KRL (Bobrow et Winograd, 1977), LOOPS (Bobrow et Stefik, 1982), ROME (Carré et Geib, 1990) et TROPES (Marino, 1993), dans l’élaboration des bases de connaissances multi-expertes (Ribiere, 1999) (Dieng *et al*., 2001), en représentation des connaissances en conception de système complexe (Trousse, 1998).

Notre approche s’appuie sur les travaux de (Trousse, 1998), et s’inspire des extensions du langage TROPES, supportant la définition des tâches et la trace du raisonnement lui-même. Elle se place principalement dans l’ingénierie des connaissances, à la différence des travaux évoqués précédemment.

[1] http://www.dmg.org/pmml-v2-1.html

3.2 Définitions

Définition 1 (Point de vue) *Un point de vue du processus d'ECD décrit une vision particulière qu'a l'analyste sur son raisonnement et/ou sur les données manipulées. Il correspond, d'une part, à un savoir-faire selon un point référentiel de connaissance du domaine (connaissance du domaine analysé) et d'autre part, à définir un plan d'exécution établi lors d'une analyse d'ECD (connaissance sur le domaine de l'analyste).*

En effet un point de vue concerne non seulement les attributs qui l'intéressent, mais aussi l'organisation et les opérations définies sur ces attributs en termes d'objectifs poursuivis ; choix des critères, des préférences liées aux objectifs de l'analyste.

Définition 2 (Vue) *Une vue dans un processus d'ECD est l'ensemble des données (voire métadonnées) obtenue par transformation d'une vue à l'étape précédente. Initialement, la vue est composée des données brutes à analyser.*

Une activité d'analyse d'ECD guidée par un point de vue sera considérée comme un processus de génération, transformation et d'enrichissement des vues. Ces dernières seront annotées par des métadonnées pour garder la trace de leur génération.

3.3 Points de vue en ECD

Un processus d'ECD passe par l'utilisation d'un certain nombre d'étapes itératives et interactives pour extraire de la connaissance interprétable par l'être humain. Il est important de capturer les points de vue sous-jacents à ses nombreuses prises de décisions. Ces points de vue sont formalisés par les deux types de connaissances (domaine analysé et domaine de l'analyste) et permettent d'annoter et d'orienter le processus. En effet, le point de vue a une connaissance sur le domaine analysé qui consiste à ne sélectionner que les attributs qui lui sont significatifs. Cette connaissance sera appliquée dans les phases en amont du processus (sélection, nettoyage et transformations). D'un autre coté, la connaissance sur le domaine de l'analyste intervient dans la phase de la fouille de données, pour le choix des méthodes et l'établissement des plans d'exécution pour réaliser la tâche de l'analyste.

On distingue trois niveaux dans notre approche : le niveau *connaissance de l'analyste* qui est composé de quatre modèles dont le modèle du domaine (Cf. Fig 4). Le niveau *vue* (Cf. Fig 2) qui identifie les données générées dans chacune des étapes du processus d'extraction (on retrouve alors la vue de sélection, la vue de transformation, etc.). Le niveau *métadonnée* (Cf. Fig 3) sert à décrire les données générées et la façon dont elles sont générées.

Prenons l'exemple d'une analyse des fichiers log d'un serveur Web. Ils constituent la principale source d'information sur les visiteurs d'un site Web qui listent toutes les requêtes HTTP des clients dans l'ordre de leurs visites. Les formats les plus populaires pour les fichiers logs sont : le format CLF (Common Log File), le format ECLF (Extended Common Log File) et le format LOGML qui est un format standard basé sur XML conçu par le W3C pour décrire les rapport des serveurs Web. Dans notre étude on va se baser sur le format LOGML. Les principales composantes d'un fichier LOGML sont : IP, Nom/login, Date : date et heure, Url , Statut, Taille, Referrer, Agent.

Définissons maintenant le point de vue fiabilité dans une telle analyse : celui-ci concerne l'évaluation de bon fonctionnement du serveur web en se basant sur l'attribut statut des pages consultées, l'attribut date : pour récupérer l'instant de la défaillance et les attributs IP et Agent pour référencer l'utilisateur.

Evaluer la fiabilité d’un système (Behja, 1999) pour prédire son comportement futur ou pour proposer une restructuration du système consiste à établir des modèles prédictifs de croissance de fiabilité qui se basent, essentiellement, sur les instants de défaillance du système. Ces dernières, dans notre cas, sont identifiées par, à la fois, les statuts d’erreurs de consultation des pages et les dates de leurs manifestations.

La figure 2 illustre les vues générées lors d’une telle analyse, et la figure 3 décrit les métadonnées générées après l’étape de sélection.

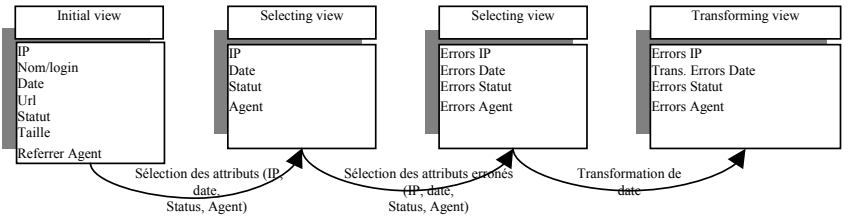

Fig. 2 -*Exemple de génération de vues successives*

```
<GeneralInfo>
   <title>logInria</title>
   <author>semir</author>
   <date>17/05/04</date>
   <vp>reliability</vp>
</GeneralInfo>
```

```
<DataInfo>
 <number>4</number>
   <IP>
     <number>19331</number>
     <type>numeric</type>
     <descri>adresse IP</descri>
   </IP>
   <Date>
     <number>19331</number>
     <type>date</type>
     <descri>la date</descri>
   </Date>
   .....
</DataInfo>
```

```
<MethodeInfo>
<name>filtering</name>
<step>preprocessing</step>
<system>Weka</system>
<methode>
  <package>filter</package>
  <class>filter</class>
    <param1>status=4*</param1>
    <param2>status=5*</param2>
</methode>
</MethodeInfo>
```

Fig. 3 – *Métadonnées de la vue sélectionnée selon des statuts erronés*

4 Modèle conceptuel du processus d’ECD

4.1 Principe

Notre modèle conceptuel de connaissance intégrant la notion de point de vue (cf. Fig. 4.) est composé de quatre modèles structurés selon (Aussenac-Gilles, 1996) :

- Au niveau du *domaine,* un modèle qui décrit, à la fois, la connaissance du domaine analysé en terme d’objets, attributs, données, etc. et la connaissance du domaine de l’analyste qui portera sur les tâches effectuées par l’analyste ; choix de méthodes, variables, etc.
- Au niveau des connaissances stratégiques :
  - Modèle *tâche et méthode* qui décrit la connaissance du domaine de l’analyste d’ECD. Cette fois–ci les objets du domaine sont des méthodes, des algorithmes, des paramètres, etc.
  - Modèle *organisationnel des points de vue* qui décrit la structure, l’organisation et la hiérarchie des points de vue en terme de relation et

liens entre eux. C'est une connaissance du domaine sur les points de vue qui seront les objets manipulés dans ce modèle.

- Quant au modèle du *point de vue*, il s'intéresse à la description des tâches à réaliser donc c'est une connaissance stratégique dont on cherche à identifier la façon avec laquelle les tâches vont être réalisé dans le sens du point de vue.

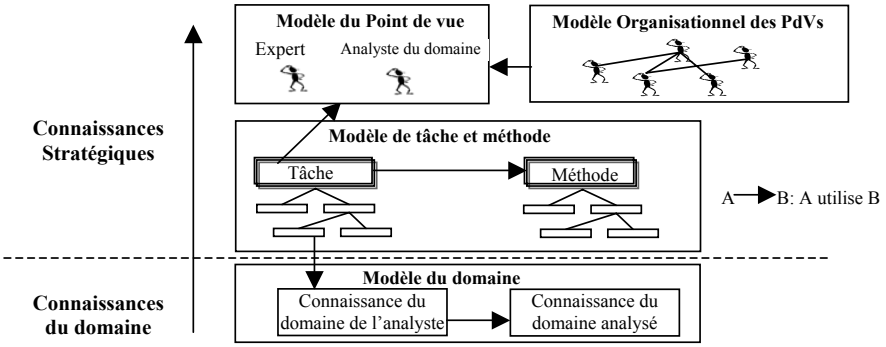

Fig. 4 – *Modèles de connaissance dans un processus d'ECD*

4.2 Modèle du domaine

Le modèle du domaine donne une représentation formelle des concepts du domaine étudié ainsi que des différentes relations qui lient ces derniers ; c'est une ontologie utilisée pour indexer les données et les attributs manipulés. Dans certains cas il est possible de typer les attributs des objets de domaines selon des points de vue définis *a priori* et partagé.

4.3 Modèle tâche et méthode

Ce modèle consiste en une ontologie générique semi-formelle (Guarino, 1995) qui décrit les méthodes et fonctions indépendamment de la structure des données pour favoriser la réutilisabilité. Les fonctions s'expriment en terme de rôles orientés tâche, c'est-à-dire d'une ontologie des objets manipulés par la tâche, exprimée de manière indépendante du domaine d'application.

Notre ontologie s'inspire de celle utilisée dans le système DAMON (Cannatro et Comito, 2003) relative à la fouille de données. La conception de notre ontologie a nécessité trois étapes comme pour le système DAMON :

1. Définir le domaine.
2. Etablir un dictionnaire qui décrit les concepts et les propriétés manipulés dans un projet d'ECD. Les paramètres importants sont : tâche, étape d'ECD, méthode, algorithme et source de données.
    (a) Tâche : représente le but du processus. On distingue (Fayyad et al., 1996(b)) des tâches de vérification et d'extraction qui est décomposée en deux sous tâches, description et prédiction.

(b) Etape d’ECD : identifie la phase du processus (prétraitement, fouille de donnée ou postraitement).
(c) Méthode : est la méthodologie utilisée dans l’étape courante du processus.
(d) Algorithme : est la façon avec laquelle la méthode est implémentée.
(e) Source de données : correspond aux données que l’algorithme utilise pour l’extraction de nouvelles connaissances.

3. Définir la hiérarchie des concepts à partir des relations taxonomiques. Ces relations sont de deux types :
   (a) Liens de spécialisation/généralisation (« is-a ») : La classification et le « clustering » sont des sous-classes de la tâche de fouille de données.
   (b) Liens d’agrégations « has part » : l’étape d’ECD est composée de trois sous-étapes, prétraitement, fouille de donnée et postraitement.

4.4 Modèle du point de vue

Permet de modéliser une partie de l’expertise nécessaire aux nombreuses prises de décision effectuées dans son analyse. Pour sa conceptualisation, nous avons besoin de modéliser le domaine, les tâches et les méthodes. Par exemple le point de vue fiabilité sera décrit par le schéma RDF (Cf. Fig. 5) : ce schéma simplifié décrit les méthodes de sélection et de transformation nécessaires pour mener à terme une analyse du point de vue fiabilité. L’analyste fiabilité doit pouvoir sélectionner les attributs qui l’intéressent (Statut erroné, dates de manifestation des erreurs et références utilisateurs Ip et UserAgent correspondants).

La spécification du point de vue en RDF permettra de générer, à la fois, la vue correspondante à l’étape ainsi qu’une partie des métadonnées qui annotent la vue.

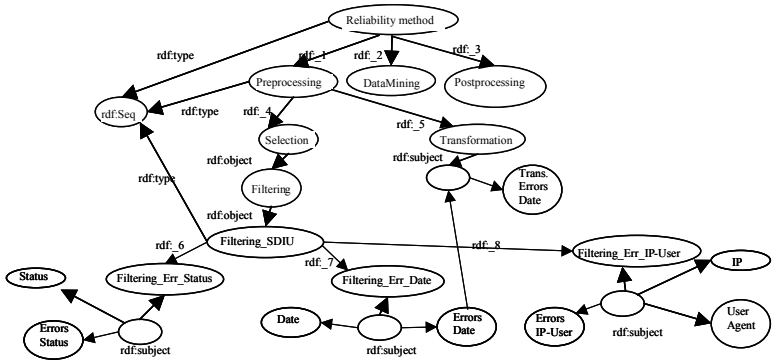

Fig. 5 – *Modèle RDF simplifié des méthodes de prétraitement du point de vue fiabilité*

4.5 Modèle organisationnel des points de vue

Dans une analyse multivues dans un processus d’ECD, il est important de souligner l’interaction et la dépendance entre les différentes analyses selon des points de vue différents. Ce modèle nécessite l’identification de divers types de relation entre points de vue comme : équivalence, subsumption, exclusion, coordination, émergence. Ce modèle est en cours d’élaboration.

5 Vers une plateforme d'annotation multivues en ECD

Nous présentons ici quelques éléments de spécification de notre maquette de plateforme pour l'annotation multivues en ECD, en cours de développement.

5.1 Génération de métadonnées

L'architecture de notre plateforme est basée sur l'utilisation du système Weka (Witten et Frank, 2000) qui est une bibliothèque d'algorithmes pour la préparation, la classification, le classement des données et la recherche d'associations. Il permet de pré-traiter les données, de les analyser à l'aide d'une méthode d'apprentissage et d'afficher le modèle résultant et ses performances. Weka est une plateforme ouverte, entièrement développée en Java, ce qui nous permettra d'y intégrer, via des transformations, nos spécifications de point de vue.

Pour la mise en œuvre nous avons besoin de quelques outils (en plus de Weka):

- XML4J: pour transformer les fichiers XML des spécifications des utilisateurs en des classes Java
- Des feuilles de style XSLT pour des transformations des documents XML vers d'autres documents XML restreint qui spécialisent les anciens.

Dans l'architecture proposée, on part d'une spécification du point de vue décrite dans un langage de représentation de connaissances (RDF/RDFS dans notre cas).

Les méthodes utilisées dans les étapes d'ECD serviront pour générer les vues de chacune d'entre elles ainsi que leur métadonnées correspondantes. Cependant, il faut extraire pour chaque étape, la méthode à utiliser ainsi que ses paramètres. Or, pour établir cette transformation, nous avons besoin de définir des feuilles de style.

Cette spécification du point de vue à l'étape i servira à :

- *Générer une partie des métadonnées d'une étape i* : les métadonnées de l'étape i-1 serviront surtout à garder une trace d'exécution des étapes précédentes. Celles-ci seront générées et auront une structure incrémentale, se mettannt à jour à chaque étape du processus.
- *Générer la vue correspondante à l'étape i* : la spécification du point de vue à la ième étape doit être transformée en une méthode supportée par Weka à l'aide d'un outil de transformation de documents XML vers Java (XML4Java). La classe java correspondante doit agir sur les vues de l'étape i-1 pour procéder aux éventuels exécutions. Enfin, on met à jour les métadonnées correspondantes de l'étape i.

5.2 Interface Utilisateur

La Figure 6 présente la version actuelle de l'interface utilisateur de notre plateforme contenant: le niveau connaissance (ontologie *tâche/méthode*, point de vue), les niveaux vue (données) et métadonnées générées (Fig. 6.c) pour une étape donnée, la tâche du point de vue fiabilité (Fig. 6.b) et le schéma RDF du point de vue fiabilité (Fig. 6.d).

6 Conclusions et Perspectives

Dans cet article nous avons proposé une plate-forme d’aide à l’annotation du processus en terme de *point de vue* qui intègre le savoir-faire de l’expert dans le cycle de l’ECD, ainsi qu’une capitalisation de deux types de connaissances manipulées dans une activité d’ECD. Cette nouvelle vision du processus d’ECD permet :

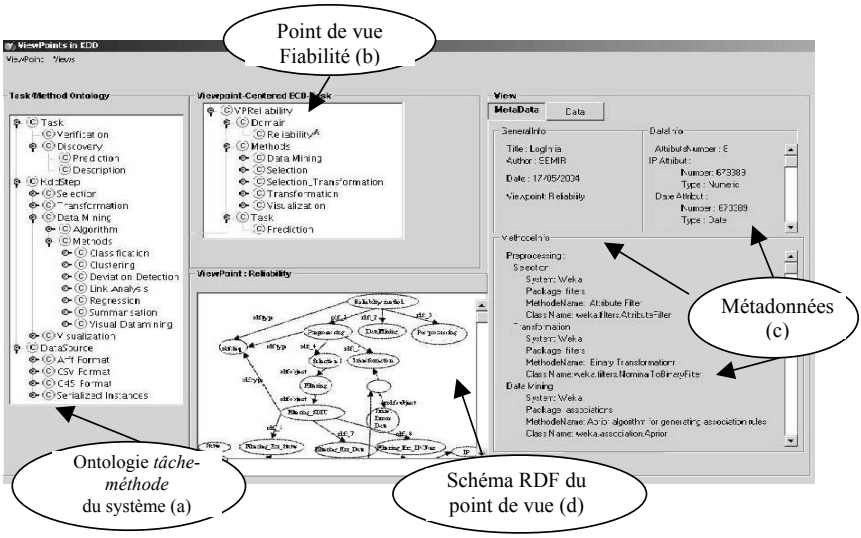

Fig. 6 – *Interface des différentes composantes des points de vue en ECD*

1. d’orienter et limiter l’espace du domaine analysé,
2. de planifier les tâches et de ne prendre que celles qui sont « bonnes » du point de vue de l’expert,
3. de bien gérer les conflits lors d’une analyse multi points de vue lorsqu’on veut favoriser les résultats d’un point de vue par rapport à un autre, ceci passe via l’intégration non seulement de la connaissance extraite mais aussi en gardant une trace de la sémantique de sa génération (contexte, point de vue, etc.),
4. et enfin de réutiliser les analyses antécédentes grâces aux métadonnées générées.

Nous avons développé un prototype de notre plate-forme à base du système Weka (du fait de sa disponibilité et de son ouverture comme une bibliothèque des méthodes d’ECD). Le modèle proposé n’est pas définitif : son développement et l’extension de ses fonctionnalités sont actuellement poursuivis en recherche. Le prototype établi ne répond pas à la totalité des attentes. Il reste à être amélioré notamment au niveau des spécifications des préférences des analystes et des relations entre points de vue. L’ontologie *tâche et méthode* proposée semble bien maîtrisée mais souffre de se restreindre sur un seul outil (Weka dans notre cas).

Références

Abiteboul S. et Bonner A.(1991), Objects and Views. In Proc. of the ACM SIGMOD Conf. on Management of Data, pages 238--247. ACM Press, 1991.

Sarabjot S. Anand, 1995. The role of Domain Knowledge in data mining in Proc. of the 4th inter. Conf. on Information and Knowledge management pp: 37 – 43.

Aussenac-Gilles (1996), Acquisition et ingénierie des connaissances : tendances actuelles Éd. : N. Aussenac-Gilles, P. Laublet et C. Reynaud. Cépaduès Éditions, Toulouse, 1996.

Behja. H (1999), Conception et réalisation d'un outil informatisé pour l'estimation et la prédiction des paramètres de la fiabilité des logiciels (cas de la correction différée) thèse de spécialité de 3eme cycle, université de rabat maroc 1999.

Bernstein A., Hill S., Provost F. (2002) An Intelligent Assistant for the Knowledge Discovery Process. Working Paper, No. IS-02-02, New York University, Center for Digital Economy Research, 2002.

Bobrow D.G., and Winograd T., (1977) An Overview of KRL, A Knowledge Representation language, Cognitive Science, V. 1, No. 1, 1977.

Bobrow, D. G. et Stefik, M. J. (1982) LOOPS: Data and Object Oriented Programming for Interlisp in European AI Conference, Orsay, France. 1982.

Cannatro M., Comito C. (2003) A data Mining Ontology for Grid Programming. in Proceedings of SemPGRID' 03, Budapest, Hungary, pp 115-134.

Carre B. and Geib J.M., (1990) The Point of View notion for Multiple Inheritance, in Proc. of ECOOP/OOPSLA'90, ACM SIGPLAN, pp. 312-321, 1990.

Devedzic V., (2001) Knowledge Discovery and Data Mining in Databases, in Handbook of Software Engineering and Knowledge Engineering Vol.1 - Fundamentals, World Scientific Publishing Co., Singapore, , pp. 615-637.

Dieng R., Corby O., Giboin A., Golebiowska J., Matta N., Ribière M., Gandon F., (2001) Méthodes et outils pour la gestion des connaissances 2nd ed Editeur(s) : Dunod , 2001.

Dugerdil, P. (1988) Contribution à l'étude de la représentation des connaissances fondées sur les objets. Le langage OBJLOG, PhD thesis, Univ. d'Aix-Marseille, 1988.

Duineveld J., Stoter R., Weiden M. R., Kenepa B., Benjamins V. R., (1999) WonderTools? A comparative study of ontological engineering tools, Proc. of the Twelfth Workshop on Knowledge Acquisition, Modelling and Management, 1999.

Fayyad U. M., Piatetsky-Shapiro G., Smyth P. (1996a): The KDD Process for Extracting Useful Knowledge from Volumes of Data, Communications of the ACM 39(11): 27-34.

Fayyad U., Piatetsky-Shapiro G. and Smyth P., (1996b) From data mining to knowledge discovery in databases, in AI Magazine, volume 17, 37--54.

Finkelstein A., Kramer J., and Goedicke M. (1990), ViewPoint Oriented Software Development, Proceedings of International Workshop on Software Engineering and its Applications, Toulouse, France, December 1990.

Piatetsky-Shapiro, G., and Frawley, W., (1991), Knowledge Discovery in Databases, AAAI/MIT Press, pp. 1—27, 1991

Gançarski P. et Trousse B., (2004) editors. 1er atelier sur la Fouille de données complexes dans un processus d'extraction des connaissances, ECG'04, 2004.

Guarino et Poli, (1995) Guarino N., Poli R. (1995) The role of formal ontology in the information technology. Int. J. Hum.-Comput. Stud. 43(5-6): 623-624 (1995).

Hotho A., Staab S., Stumme. G. (2003) Text Clustering Based on Background Knowledge. Technical Report 425, University of Karlsruhe. 2003.

Marcaillou-Ebersold S., (1995) Intégration de la notion de points de vue dans la modélisation par objets: le langage VBOOL. Thèse de doctorat, Toulouse. 1995.
Marino O., (1993) Raisonnement classificatoire dans une représentation objets multi-points de vue, thèse de doctorat de l'Université Joseph Fourier-Grenoble 1, 1993.
Morik K. and Scholz. M., (2003) The MiningMart Approach to Knowledge Discovery in Databases. In Intell. Technologies for Information Analysis. Springer 2003.
OOPSLA, (2003) Multiple Viewpoints for System Modelling in OOPSLA, C.Casanave, W.Frank, S.Hendryx, S.Mellor, http://oopsla.acm.org/oopsla2003/files/ws-8.html
Pohle C., (2003) Integrating and Updating Domain Knowledge with Knowledge Discovery. At the Doctoral Consortium held in conjunction with the 6th International Conference for Business Informatics 2003 (WI-2003), pp15-17, 2003.
Ribiere M. (1999) Représentation et gestion de multiples points de vue dans le formalisme des graphes conceptuels Phd Univ. de Nice-Sophia Antipolis 1999.
Santos E.L. (1993) Shood : un modèle meta-circulaire de représentation de connaissances. Thèse de doctorat 1993 INPG de Grenoble.
Souza C.S., (1995) Design and Implementation of Object-Oriented Views DEXA 1995: 91-102.
Spanoudakis G., Finkelstein A. et Emmerich W., (1996) The proceedings of the workshop Viewpoints 96: International Workshop on Multiple Perspectives in Software Development, ACM press 1996.
Tanasa D., Trousse B. (2004) Advanced Data Preprocessing for Intersites Web Usage Mining. In IEEE Intelligent Systems (Vol. 19, No. 2) pp. 59-65. 2004.
Trousse B. (1998) Viewpoint Management for Cooperative design . In Proceedings of the the IEEE Computational Engineering in Systems Applications (CESA'98), UCIS - Ecole Centrale de Lille - CD-Rom, M. K. P. Borne, A. E. Kamel, editors, 1998.
Witten I. H. et Frank E., (2000) Data mining : practical machine learning tools and techniques with Java implementations / editeur(s) : Morgan Kaufmann , 2000.
Yoon S-C., Henschen L.J., E. K. Park et Makki S., (1999) Using domain Knowledge in Knowledge discovery. in Proceedings of the eighth international conference on Information and knowledge management. Missouri, United States pp: 243 – 250.
Zhong N., Liu C. et Ohsuga S. (2001) Dynamically organizing KDD. International Journal of Pattern Recognition and Artificial Intelligence Vol. 15, No. 3, 451-473. World Scientific Publishing Company. 2001.

Summary

We propose in this paper a new approach for applying the viewpoint notion in the Knowledge Discovery from Data Base (KDD) multiviews analysis. We define viewpoint as a perception of an expert on a KDD process, perception referred by its own knowledge. In order to facilitate both reusability and adaptability of a KDD process, and to reduce his complexity with maintaining the trace of the past analysis viewpoints. The KDD process will be considered as a generating and transformation views annotated by metadata to store the discovery knowledge. Our approach position relatively to the methodological works in KDD process will be given. The elements of modelling viewpoint-based KDD process will be described at the ontological level. At the end, we illustrate our approach in web usage mining according to the reliability viewpoint.