Découverte des Connaissances dans les Bases de Données: une approche centrée objet

Z. Bouzidi (*), H. Kherbachi (*), A. Hocine (**)
(*) Laboratoire Economie-Développement,
Université Abderrahmane Mira,
Targa Ouzemmour 06000 Béjaïa (Algérie)
E-mail : zair_bouzidi@hotmail.com
(**) Laboratoire LICIAP, Université de Pau et des Pays de l'Adour, route de l'Université 64000 Pau (France)
E-mail : Amrane.Hocine@crisv2.univ-pau.fr

Introduction

Les systèmes de gestion de bases de données, conçus à l'origine pour gérer les fichiers, ont vu leurs fonctionnalités évoluer pour aboutir au modèle relationnel qui domine le marché actuel.
Les bases de données relationnelles ont été exploitées le plus souvent en tant que source d'informations mais rarement comme sources de connaissances. En effet, l'on a eu tendance à "oublier" que ces informations sont porteuses de connaissances implicites qui demeurent ainsi inexploitées [3], [5].
Le développement des systèmes à bases de connaissances est plus souvent réalisé par les experts humains. C'est également à ces derniers que l'on fait appel pour interpréter les résultats de sélection d'informations à partir d'une base de données.
La découverte de connaissances est définie comme l'extraction non-triviale, à partir d'une base de données, d'une information potentiellement utile et qui est implicite et inconnue auparavant.
Pour exprimer les connaissances extraites à partir d'une base de données en termes de concepts généralisés de haut de niveau, et non en termes de données initiales, nous avons intégré des connaissances supplémentaires (connaissances du domaine d'application) qui sont données par l'expert du domaine. Cette connaissance du domaine, formalisée en terme de hiérarchie de concepts, permet de généraliser les valeurs initiales d'une base de données [1], [2], [4].

1. Les systèmes de découverte de connaissances dans les bases de données

Les systèmes de gestion de bases de données, conçus à l'origine pour gérer les fichiers, ont vu leurs fonctionnalités évoluer pour aboutir au modèle relationnel qui domine le marché actuel.
Les bases de données relationnelles ont été exploitées le plus souvent en tant que source d'informations mais rarement comme sources de connaissances. En effet, l'on a eu

tendance à “oublier” que ces informations sont porteuses de connaissances implicites qui demeurent ainsi inexploitées [3], [5].
La découverte de connaissances est définie comme l’extraction non triviale, à partir d’une base de données, d’une information potentiellement utile et qui est implicite et inconnue auparavant.
La découverte de connaissances dans les bases de données, plus connue sous le nom générique de KDD : Knowledge Discovery in Database, est un domaine actif ces dernières années. Elle consiste à trouver des connaissances qui n’étaient pas explicites dans sa représentation dans le domaine de connaissance. Elle génère des connaissances implicites qui doivent être exprimées sous une forme compréhensible par l’utilisateur et sans aucune interprétation de sa part. Ce processus doit tenir compte des informations spécifiques de la base de données et de la connaissance du domaine [3], [5].

2. Intégration de la connaissance du domaine par les hiérarchies de concepts

Le processus de généralisation consiste à exprimer les connaissances extraites à partir d’une base de données en termes de concepts généralisés de haut niveau et non en termes de données initiales grâce à l’intégration des connaissances supplémentaires comme la connaissance du domaine par exemple en termes de concepts et qui sont données par l’expert du domaine.
Les connaissances à découvrir sont donc exprimées à l’aide de concepts que définit l’expert du domaine. La connaissance du domaine est formalisée en termes de hiérarchie de concepts. Cette connaissance repose sur un ensemble de concepts permettant de généraliser les valeurs initiales d’une base de données.

Il est plus efficace d’exprimer l’attribut poids en termes de : léger, moyen ou lourd que par des valeurs. L’on peut alors l’exprimer sous forme de hiérarchie de concepts à un niveau, comme ci-dessus :

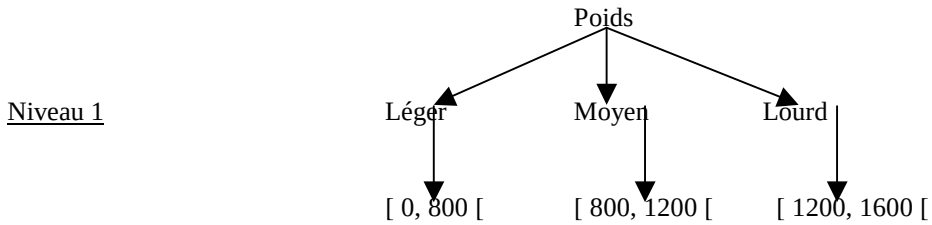

fig. 1 : Hiérarchie de concepts concernant l’attribut Poids.

Cette hiérarchie de concepts est exprimée dans le langage centré objet RECOS [1], [2], [4]. Considérons une hiérarchie de concepts concernant l’attribut “ Modèle ” :

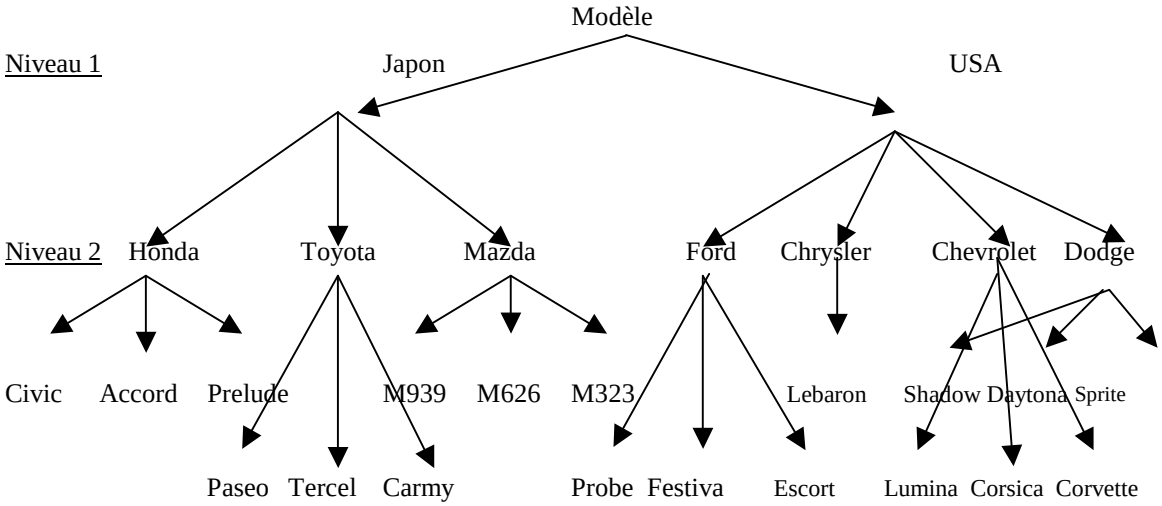

Fig. 2 : Hiérarchie de concepts concernant l'attribut " Modèle ".

3. Extraction automatique de règles caractéristiques et de règles décisionnelles

Le processus de généralisation consiste à substituer les valeurs initiales des données par un niveau désiré de la hiérarchie de concepts. Et la réduction permet de ne regarder que les tuples distincts de la table généralisée.
Avant d'introduire la notion de règles découvertes, il est important de préciser le type de problème auquel nous nous intéressons. On considère une table contenant la réponse en extension à une requête SQL, modélisant le domaine d'intérêt de l'utilisateur.

Exemple :
Considérons une table de données d'ophtalmologie dont nous donnons une partie ci-dessus :

Code	*Modèle*	*Carburant*	*Emplacement*	*Poids*	*Cylindre*	*Puissance*	*Turbo*	*Compression*	*transmission*	*Kilométrage*
D3	Lumina	Efi	Petit	1000	4	***Moyenne***	Non	***Elevée***	***Manu***	***Elevé***
D5	Corvette	Dbbl	Petit	1557	6	***Moyenne***	Non	***Moyenne***	***Auto***	***Moyen***
D7	Corsica	Dbbl	Petit	1287	6	***Moyenne***	Non	***Moyenne***	***Auto***	***Moyen***
D8	Escort	Efi	Moyen	867	6	***Elevée***	Oui	***Elevée***	***Manu***	***Elevé***
D11	Paseo	Dbbl	Petit	1023	4	***Basse***	Non	***Moyenne***	***Auto***	***Moyen***
D13	M323	Efi	Moyen	698	4	***Moyenne***	Non	***Moyenne***	***Manu***	***Elevé***
D14	Daytona	Efi	Moyen	798	6	***Elevée***	Oui	***Elevée***	***Auto***	***Elevé***
D16	Prelude	Efi	Petit	800	4	***Elevée***	Oui	***Elevée***	***Manu***	***Elevé***
D18	Civic	Dbbl	Petit	796	4	***Basse***	Non	***Elevée***	***Manu***	***Elevé***
D20	M626	Efi	Petit	980	4	***Moyenne***	Non	***Elevée***	***Manu***	***Moyen***
D22	Accord	Efi	Petit	989	4	***Basse***	Non	***Moyenne***	***Auto***	***Elevé***
D24	Shadow	Efi	Moyen	658	6	***Elevée***	Oui	***Elevée***	***Auto***	***Elevé***

Table 1: Table de données ophtalmologiques.

Si l'objectif de l'utilisateur est de découvrir des connaissances permettant d'identifier les voitures qui sont résistantes dont le kilométrage est élevé, il l'exprimera sous la forme d'une requête SQL suivante :

Select modèle,carburant, emplacement, poids, cylindre, puissance, turbo, compression, transmission, kilométrage

From Voiture

Where (kilométrage = 'élevé') **or** (kilométrage = 'élevé').

Le résultat de cette requête constitue l'ensemble des informations positives (noté TI+) et les tuples qui ne vérifient pas cette requête constituent l'ensemble des informations négatives (noté TI-).

Après la généralisation sur les attributs “ Poids ” et “ “Modèle ” de niveau 1, l’on obtient la table suivante :

Code	*Modèle*	*Carburant*	*Emplacement*	*Poids*	*Cylindre*	*Puissance*	*Turbo*	*Compression*	*Transmission*	*Kilométrage*
D3	Chevrolet	Efi	Petit	Moyen	4	***Moyenne***	Non	***Elevée***	***Manu***	***Elevé***
D5	Chevrolet	Dbbl	Petit	Lourd	6	***Moyenne***	Non	***Moyenne***	***Auto***	***Moyen***
D8	Ford	Efi	Moyen	Moyen	6	***Elevée***	Oui	***Elevée***	***Manu***	***Elevé***
D11	Honda	Dbbl	Petit	Moyen	4	***Basse***	Non	***Moyenne***	***Auto***	***Moyen***
D13	Mazda	Efi	Moyen	Léger	4	***Moyenne***	Non	***Moyenne***	***Manu***	***Elevé***
D14	Dodge	Efi	Moyen	Léger	6	***Elevée***	Oui	***Elevée***	***Auto***	***Elevé***
D16	Honda	Efi	Petit	Moyen	4	***Elevée***	Oui	***Elevée***	***Manu***	***Elevé***
D20	Mazda	Efi	Petit	Moyen	4	***Moyenne***	Non	***Elevée***	***Manu***	***Moyen***
D22	Honda	Efi	Petit	Moyen	4	***Basse***	Non	***Moyenne***	***Auto***	***Elevé***

Table 2 : La table Voiture après la généralisation.

L’extraction est réalisée après le processus de généralisation et de réduction du paragraphe précédent.

Après le processus de généralisation, et de réduction, vient l’extraction des règles caractéristiques et décisionnelles. Les premières caractérisent un concept qui est satisfait par l’ensemble des données de la base de données. Elles sont destinées principalement pour une prise de décision. Les secondes sont des descriptions discriminantes des concepts exprimés par les décisions.

Une approche ensembliste basée sur la notion de caractérisation logique d’une partition (méthode d’apprentissage) [7] est étendue et adaptée au contexte d’une base de données afin de découvrir les règles décisionnelles.

Les connaissances découvertes sont formalisées en termes de règles de production, forme symbolique facilement compréhensible par l’utilisateur, sans aucune nouvelle interprétation. Elles peuvent être utilisées pour compléter la construction de base de connaissances d’un système expert par exemple, ou aider à la prise de décision lors du processus d’une analyse symbolique numérique de données.

Une règle caractéristique est une assertion qui caractérise un concept qui est satisfait par l'ensemble des données de la base de données.

Les données se présentent toujours sous la forme d'une relation :

Tiplus(Att_clé , Att_c1 , …. , Att_ci , … , Att_cp , Att_c)

A chaque attribut est associé un domaine D qui décrit l'ensemble des valeurs qu'il peut prendre, où :

- Dd : composé de valeurs {d1 , … dk}, désigne le domaine de l'attribut décisionnel Att_d ;
- Dci : désigne le domaine d'un attribut conditionnel Att_ci.

L'attribut Att_d représente l'attribut décisionnel que l'on retrouvera dans les règles caractéristiques à extraire. Une règle décisionnelle exprime une relation entre les attributs conditionnels et l'attribut décisionnel tel que :

SI Att_c1 = V1 **et** Att_c2 = V2 **et …… et** Att_cp = Vp **et** **Alors** Att_d = Vd

Qui s'interprète comme suit : " Si la conjonction des couples (Att_ci , Vi) est vérifiée Alors on peut prendre la décision Att_d = Vd " où :

- Att_ci : désigne un attribut conditionnel ;
- Att_d : est l'attribut décisionnel ;
- Vd : désigne une valeur du domaine de ces attributs.

Les règles décisionnelles sont des descriptions discriminantes des concepts exprimés par les décisions (le couple Att_d = Vd).

L'approche KDD sous-entend l'exploitation de données. L'algorithme de caractérisation logique d'une partition est un algorithme d'apprentissage qui traite des données codées. Un système de découverte de connaissances ne peut pas passer par une étape de codage. Cette méthode doit travailler avec des données brutes telles qu'elles sont stockées dans les bases de données. Ainsi, l'algorithme de caractérisation logique d'une partition est étendu pour permettre l'exploration d'une base de données relationnelles.

Les connaissances découvertes sont formalisées en termes de règles de production, forme symbolique facilement compréhensible par l'utilisateur, sans aucune nouvelle interprétation. Elles peuvent être utilisées pour compléter la construction de base de connaissances d'un système expert par exemple, ou aider à la prise de décision lors du processus d'une analyse symbolique numérique de données [6].

4. Conclusion

La découverte de connaissances, qui est l'objet de nos recherches actuellement, est définie comme l'extraction non triviale, à partir d'une base de données, d'une information potentiellement utile et qui est implicite et inconnue auparavant. Plus connue sous le nom générique de KDD : Knowledge Discovery in Database, elle est un domaine actif ces dernières années. Elle consiste à trouver des connaissances qui n'étaient pas explicites dans sa représentation dans le domaine de connaissance. Elle génère des connaissances implicites qui doivent être exprimées sous une forme compréhensible par l'utilisateur et sans aucune interprétation de sa part. Ce processus doit tenir compte des informations spécifiques de la base de données et de la connaissance du domaine.

Références bibliographiques

[1] Z. Bouzidi, A. Hocine, "RECOS un langage de représentation et d'exploitation de connaissances par les schémas ", Rapport interne ICOG 3/95, Dpt. d'Informatique, Université de Pau, Août 1995.

[2] Z. Bouzidi, A. Hocine, H. Kherbachi, "Vers un Environnement de Développement Centré objet ", IV èmes Rencontres de Recherche Opérationnelle, 6-8 Octobre 1996, USTHB, Alger.

[3] Z. Bouzidi, A. Hocine, S. Smadhi, " Extraction automatique des connaissances: une approche fondée sur la caractérisation logique d'une partition ", Proceeding II èmes Rencontres Francophones de Recherche Opérationnelle (FRANCORO II), Sousse (Tunisie), pp. 14-15, 6-8 Avril 1998.

[4] Z. Bouzidi, H. Kherbachi, A. Hocine, " Procedural Knowledge Modelling : a centered object approach ", in Applied Informatics, Proceeding of the Fourtheenth IASTED International Conference, Applied Informatics, Austria, pp. 330-332, February 20-22, 1996.

[5] A. Hocine, S. Smadhi, " Extracting rules from database ", in Applied Informatics, Proceeding of the Fourtheenth IASTED International Conference, Applied Informatics, Austria, pp. 250-253, February 20-22, 1996.

[6] Y. Kodratoff, E. Diday, " Induction symbolique Numérique à partir des données", Volume 1, Cépaduès-Editions, Mai 1991.

[7] G. Levy, " Un algorithme efficace pour une caractérisation logique d'une partition ", Actes des III èmes Journées " Symbolique Numérique ", Paris 14-15 Mai, pp. 247-257, 1992.