République Algérienne Démocratique et Populaire

وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université des Sciences et de la Technologie d'Oran Mohamed Boudiaf

Faculté des Mathématiques et Informatique
Département d'Informatique

THESE
Présentée par

Mr Salim KHIAT

Pour l'obtention du diplôme de Doctorat en Sciences

Spécialité : Informatique
Option : Systèmes, Réseaux et Bases de données

Thème

LA FOUILLE MULTI-SOURCES DE DONNEES MULTI-NIVEAUX

Soutenue publiquement le :

Devant le jury :

Mr Djebbar Bachir	**Professeur**	**Président**	**USTO-MB**
Mme Belbachir hafida	**Professeur**	**Rapporteur**	**USTO-MB**
Mr Rahal Sid Ahmed	**Maitre de Conférence-A**	**Co-Rapporteur**	**USTO-MB**
Mr Alla Hassane	**Professeur**	**Examinateur**	**Univ. Grenoble**
Mr Beldjilali Bouziane	**Professeur**	**Examinateur**	**Univ. Oran**
Mr Amine Abdelmalek	**Maître de Conférence-A**	**Examinateur**	**Univ. Saïda**

Année universitaire : 2014/2015

Remerciements

Avant tout début et après toute fin je remercie mon dieu le tout puissant qui m'a tout donné la volonté, le courage et surtout la patience pour réaliser ce modeste travail.

Cette thèse s'est déroulée au sein du Laboratoire Signaux, Systèmes et Données (*LSSD*) et au sein de la société *SONATRACH* dans le cadre d'une collaboration universitaire.

Je ne trouve pas de mots assez forts pour exprimer mon sentiment de reconnaissance de profonde gratitude à M^{me} Belbachir Hafida, mon encadreur. Les efforts qu'elle a fait en termes d'encadrement actif, d'encouragement dans mes moments de doute, pour ne citer que ceux là, resteront à jamais gravés dans ma mémoire.

Je tiens à remercier très chaleureusement M. Rahal Sidi Ahmed, mon co-encadreur, dont le guide et l'aide ont été précieuses et essentielles tout au long du chemin qui m'a mené à mon titre de Docteur.

Je remercie également le professeur Djebbar Bachir, pour l'honneur qu'il me fait de présider le jury d'examen.

Un témoignage de ma profonde reconnaissance s'adresse au professeur Beldjilali Bouziane de l'université d'Oran, pour avoir accepté d'examiner ma thèse.

Je remercie vivement le professeur Alla Hassane de l'université de Grenoble, qui me fait l'honneur de faire partie du jury d'examen.

Que monsieur Amine Abdelmalek Maître de Conférence à l'université de Saïda, trouve ici l'expression de toute ma gratitude pour avoir accepté d'être mon examinateur.

Mes remerciements vont aussi à toute l'équipe de la direction de la maintenance *(MNT)* de l'activité *AVAL* de la société *SONATRACH* pour leur contribution dans la partie application de ma thèse.

J'aimerai remercier du fond de mon cœur mes parents pour leur soutien moral et ma femme qui m'a soutenu durant cette thèse.

... A tous ceux qui ont contribué de près ou de loin à l'aboutissement de ce travail par leur confiance et leur soutien.

Cette thèse est dédiée à mes deux enfants Karima et Youcef.

Résumé

L'exploitation des données collectées à partir des différents sites d'une organisation est en passe de devenir un enjeu industriel important. En effet, la prise de décisions fait partie des compétences des dirigeants des entreprises et de ceux qui exercent un pouvoir dans l'entreprise. En permanence de nombreuses décisions sont prises dans les entreprises. Toutes n'ont pas la même incidence mais elles conditionnent le bon fonctionnement de l'entreprise et les performances réalisées.
Cependant dans le monde réel, la structure d'une société multi-branches est habituellement plus complexe dont chaque branche peut également avoir des sous branches…, ce qui va donner naissance à l'organisation multi-niveaux. Pour pouvoir explorer toutes ces données la fouille multi-bases de données multi-niveaux, plus précisément la technique de la fouille de Règles d'Association, s'impose pour extraire des connaissances riches, utiles et potentiellement inconnues qui peuvent aider les décideurs à différents niveaux à prendre des décisions.
Le défi que tente de relever la fouille multi-sources de données dans une organisation multi-niveaux est de pouvoir prendre en compte la totalité des informations recueillies à partir de plusieurs niveaux d'organisations concernant les centres de décisions qui constituent les unités taxinomiques élémentaires. Cela suppose d'être capable d'intégrer des informations à différentes niveaux d'abstraction et de pouvoir ainsi les synthétiser sans perte d'informations.
Dans ce contexte nous abordons deux thèmes principaux qui sont l'intégration des connaissances de l'utilisateur dans le processus de découverte des règles d'association et l'intégration des modèles probabilistes dans le processus de synthétisation des motifs locaux. Le premier problème exige la définition d'un formalisme adapté afin d'exprimer les connaissances de l'utilisateur avec précision et de façon flexible et sans perte de connaissances en utilisant les ontologies et les schémas de règles avec les opérateurs. Les résultats expérimentaux montrent que l'approche proposée permet effectivement d'aider les décideurs des différents niveaux de la maintenance d'une entreprise pétrolière à prendre de bonnes décisions, sans avoir besoin de recourir à une refouille des données. L'application des modèles probabilistes dans l'analyse des motifs locaux permet de réduire au maximum la perte de connaissances. Les expérimentations effectuées sur une base de données synthétique montre l'efficacité de cette approche.

Mots-clefs : La fouille multi-bases de données, Analyse des motifs locaux, Schéma de règles multi-niveaux, Règles d'association, Ontologie.

Abstract

The data exploitation collected from different branches of an organization is on the way to become a significant industrial challenge. The decision making belongs to the leaders of the companies and whose which exert a power in the company. Many decisions are taken in the companies, not all have the same incidence but they ensure the correct operation of the company and the performances carried out. However in real world, the interstate company structure is usually more complex where each branch can also have sub-branches and so on…which gives birth to the multi-levels organization.
The challenge is to be able to take part the totality of information collected from several levels of organization for the decision making. That supposes to be able to integrate information of different level of abstraction and synthesize them without lossless with the same semantic category.

In this context, we address two main issues: the integration of the user knowledge in the discovery process and the integration of the probabilistic model in the synthesizing process. The first issue requires defining an adapted formalism to express user lossless knowledge with accuracy and flexibility and such as ontology and rule schema with operators. Results show that our approach can effectively help decision making of Petroleum Company at different levels to make good decision without lossless knowledge. Second, the integration of the probabilistic model in the local patterns analysis reduces the knowledge lossless. Experiments show the efficiency of our approach.

Keywords: Multi-databases mining, Local pattern analysis, Synthesizing pattern, Rules schema multi-levels, Association rules, Ontology.

Table des matières

Liste des Figures

Liste des tableaux

Abréviations

RAMARO : Règles d'Association Multi-niveaux d'Abstraction en utilisant le schéma de Règles et Ontologie
AMLAME : Analyse des Motifs Locaux Avec Maximum Entropie
SH/AVL : SONATRACH / AVAL
GNL : Division Gaz Naturel Liquéfié
GPL : Division Gaz Propane Liquéfié
LQS : Division Liquéfaction et Séparation de Gaz
RA1Z : Complexe de Raffinerie d'Arzew
RA1A : Complexe de Raffinerie d'Alger
RA1K : Complexe de Raffinerie de Skikda
GLxZ : Complexe n° x pour la production du Gaz Naturel Liquéfié
GPxZ : Complexe n° x pour la production du Gaz Propane Liquéfié
DT : Demande de Travail
GSAO: Gestion du Système du Personnel (S) Assisté par Ordinateur
GLAO : Gestion de la Logistique Assistée par Ordinateur
GMAO : Gestion de la Maintenance Assistée par Ordinateur
GPAO : Gestion de la production Assistée par Ordinateur
UBO : Unité de Base Opérationnelle
ECD : Extraction de Connaissances à partir des Données
KDD : Knowledge Discovery in Databases
FMBD : Fouille Multi-Bases de Données
MDM : Multi-Databases Mining
M.CU : Maintenance curative
M.PV : Maintenance préventive
M.PD : Maintenance prédictive
A.P : Arrêt programmé

Introduction Générale

- Contexte de l'étude
- Motivations
- Problèmes identifiés
- Contributions
- Organisation de la thèse

Introduction générale

1 Contexte de l'étude

De nos jours, nous observons une croissance éclatante de l'électronique, en particulier, le matériel informatique qui occupe de plus en plus une place importante. Toutes les sphères de l'activité de la société utilisent l'outil informatique pour leurs activités quotidiennes. Ceci est motivé par l'évolution très rapide des techniques de génération (telles que les techniques de lecture des codes barres des articles achetés en supermarchés et la naissance des pointeuses utilisées dans les grandes entreprises) et de stockage de données (augmentation de la capacité de stockage et diminution des coûts des disques durs par exemple) qui ont permis la création par de nombreux organisations de bases de données volumineuses. Les avions, automobiles, trains sont gérés par des ordinateurs, les caisses enregistreuses des supermarchés, les factures des restaurants sont produites par des ordinateurs, le suivi d'un médecin de ses patients est enregistré sur ordinateur. Ainsi les industriels d'aujourd'hui sont de plus en plus équipés de systèmes d'acquisition numériques. Ces systèmes, qu'ils agissent au niveau de la conception des produits, de la gestion ou du suivi de la production génèrent des Giga Octets de données chaque jour. Le volume de données est tel qu'il est désormais impossible à un décideur d'avoir une vue d'ensemble sur ces données. Comme le cerveau humain a des capacités limitées à traiter les données, des outils performants automatiques ont vu le jour qui permettant le traitement et l'analyse des données afin d'en extraire automatiquement des informations pertinentes et utiles ayant une valeur ajoutée pour certaines applications. Il s'agit des outils du domaine de l'Extraction de Connaissances à partir des Données (ECD) défini par [Gregory.P et al.1991]. Il est à noter que l'ECD représente tout le processus qui permet de passer des données brutes à des informations exploitables par les analystes humaines. L'étape de la fouille de données est sans doute l'étape la plus importante dans ce processus. En effet, les techniques de la fouille de données (en anglais : « Data Mining ») sont dédiées à la découverte de modèles non triviaux, implicites, non connus, potentiellement utiles et compréhensibles à partir d'un grand ensemble de données.

Nous répertorions plusieurs techniques de la fouille de données telles que :

- Les règles d'association ;
- L'analyse de groupe (« cluster »)
- La classification par des réseaux de neurones, des arbres de décision ou de règles de classification
- Etc…

Dans la présente thèse nous nous intéressons aux techniques de description et prédiction en particulier aux règles d'association, et ce, dans une perspective de la fouille de données. Le choix des règles d'association est motivé par les faits suivants :

- Les règles d'association sont des méthodes prédictives. Ainsi on pourra prévoir le comportement et ceci en fonction des relations entre objets.
- Les règles d'association sont aussi des méthodes descriptives. En effet, les règles décrivent, à travers une vue abstraite, les données brutes en expliquant le pourquoi de la conséquence de chaque objet. En d'autres termes, ces méthodes permettent de comprendre le comportement des données et déterminer les relations entre eux.
- Les règles d'association sont des techniques qui ont connu un succès dans les environnements de la fouille de données car les analystes humains trouvent de la facilité à lire et à comprendre le modèle résultant. De plus les règles d'association sont conceptuellement simples, bien compris de leurs utilisateurs, ce qui en fait souvent un choix privilégié des décideurs en technologie dans les entreprises.

Beaucoup de travaux ont été élaborés pour l'extraction de règles d'association dans une seule source de données. L'algorithme pionnier de cette technique est sans doute l'algorithme *APRIORI* [Agrawal et al. 1994].

2 Motivations

Beaucoup d'entreprises à travers le monde adoptent de façon graduelle la politique de l'économie libérale. Ce qui va générer l'augmentation des compagnies multi-branches. Beaucoup d'entreprises opèrent à partir de différentes branches situées dans différentes régions distribuées géographiquement. En effet, ces branches sont complètement ou partiellement opérationnelles et récoltent des données de façon continue. Prenons toujours l'exemple des magasins de ventes d'une compagnie qui sont ouverts 12h par jour. Toutes les transactions sont stockées localement, ce qui génère plusieurs bases de données de la compagnie. D'où la nécessité de la gestion de toutes ces

bases de données par l'entreprise pour traiter les différents aspects de prise de décision et spécialement si le besoin d'adresser le problème au niveau central.

Beaucoup de décisions importantes se basent sur la distribution des données à travers les branches. Les décisions globales ont besoin de l'analyse de données entières distribuées dans les branches. La validité de ces décisions dépend de manière de manipuler et comprendre les données pertinentes dans différentes branches. L'exploitation de ce grand volume de données reste un défi pour les managers de ces entreprises. Pour cela, quelques issues sont apparues pour bien exploiter ces données de manière intelligente et distribuée que nous présentons dans ce qui suit :

L'agrégation de données : La consolidation des données de plusieurs sites sur un seul site central ensuite la découverte des connaissances dans le site central. Cette approche suppose que chaque site a le même poids, en ignorant la contribution importante des sites de l'ensemble de l'entreprise car les branches ont des poids différents. En plus, dans un contexte multi-sites, s'ajoutent au problème déjà délicat du traitement de l'information, les problèmes liés à la confidentialité de certaines informations stratégiques et à la rétention de ces informations par les entreprises.

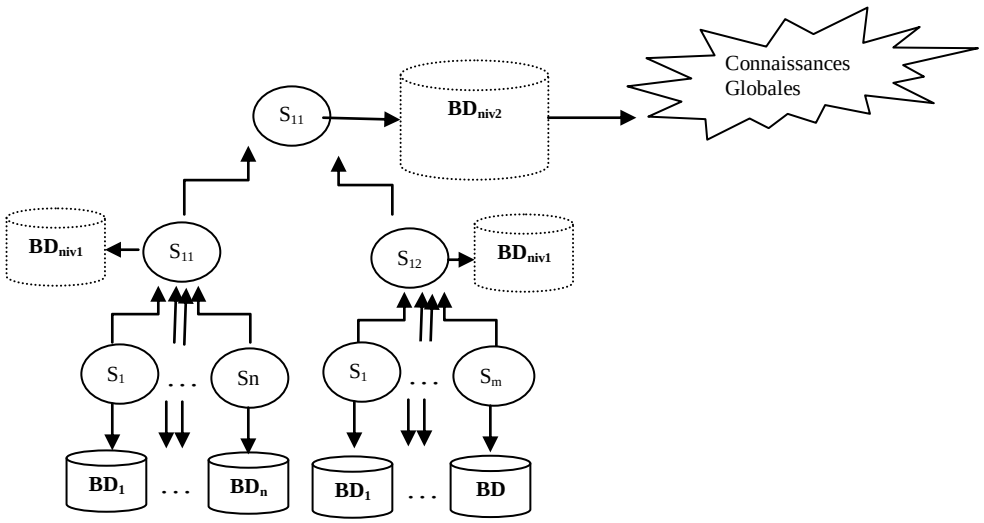


FIG. 1– AGREGATION DE DONNEES

La fouille de données distribuées : Elle fournit un apport important en termes de recherche de connaissances. Cette discipline vise essentiellement à pallier les surcoûts en termes de communication et à augmenter la puissance du traitement en partageant le traitement sur plusieurs sites. Le type de connaissances engendrées par cette approche est global. Les sites coopèrent

entre eux pour fournir des connaissances générales. Cette approche ne permet pas de générer les connaissances locales et de trouver des spécificités des différents sites. Les connaissances locales permettent de décrire et prévoir les relations entre les individus localement. L'apport ou la découverte des connaissances d'un site à un autre ne peut être déduite et interprétée.

La fouille multi-bases de données : Parmi les travaux de recherche en fouille de données, la fouille multi-bases de données ou multi-sources de données est sans doute le domaine qui a attiré le plus l'attention des chercheurs et pour lequel beaucoup de travaux ont été effectués. En effet la fouille multi-bases de données peut être définie par le processus de la fouille de données appliqué sur des multiples sources de données, potentiellement hétérogènes, pour la découverte des connaissances ou motifs nouveaux et utiles. Ce processus consiste à la fouille des données séparément dans chaque site et l'agrégation des connaissances locales pour constituer les connaissances globales en utilisant les algorithmes de synthétisation. En plus de ces connaissances, avec ce processus nous pouvons aussi extraire des connaissances majoritaires/exceptionnelles. Une connaissance majoritaire et/ou exceptionnelle est supportée par plusieurs et/ou quelques sites respectivement.

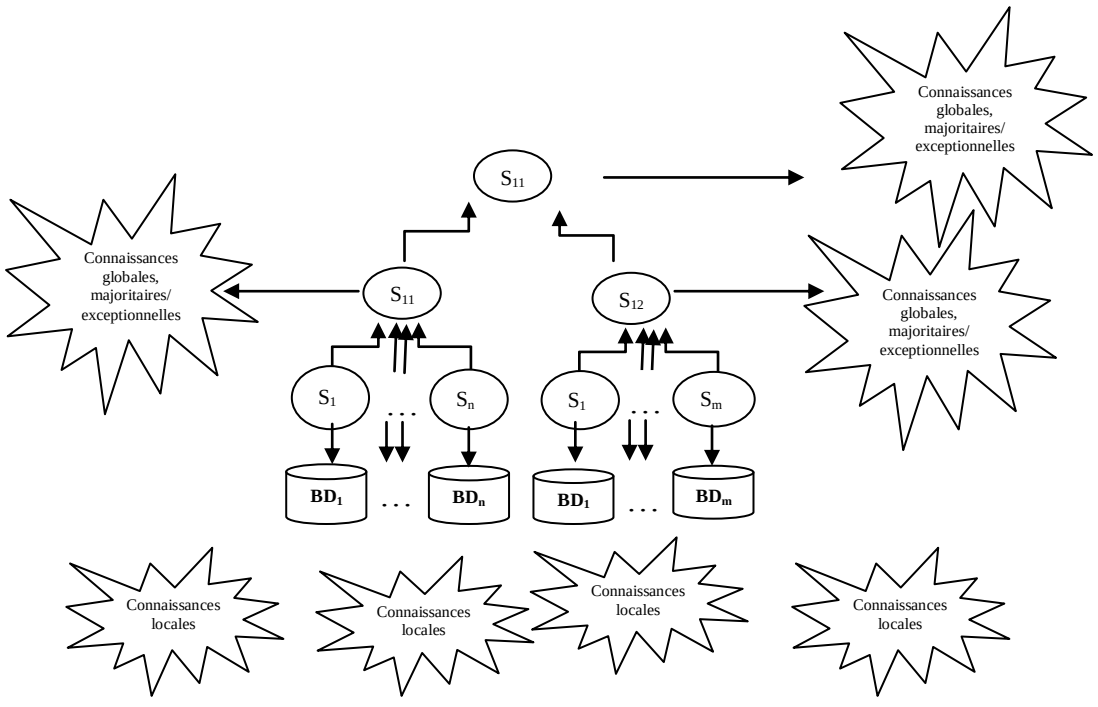


FIG. 2– AGREGATION DE CONNAISSANCES

Les approches d'agrégations de données ne génèrent que des connaissances globales qui ne peuvent expliquer la particularité d'un site ou région par rapport

à d'autres et cache la distribution des connaissances sur les différents sites. Il en est de même pour les approches de fouille de données distribuées dont le souci majeur est la performance. C'est pour cette raison que l'orientation de notre thèse s'est orientée vers la fouille multi-bases de données puisque les connaissances résultantes sont riches et diverses. Ces connaissances se résument à des connaissances globales, majoritaires et/ou exceptionnelles et locales. A travers cette architecture illustrée dans la figure 2, nous pouvons extraire des connaissances locales qui caractérisent les différentes branches et qui peuvent aider les décideurs de ces branches à prendre des décisions locales. En plus de ces connaissances locales s'ajoutent les connaissances globales et majoritaires/ exceptionnelles des différents niveaux qui peuvent être un support solide pour aider les décideurs à prendre des décisions à tous les niveaux de l'entreprise.

Néanmoins, cette issue présente quelques problèmes ouverts et qui sont cités dans le paragraphe suivant.

3 Problèmes identifiés

Le contexte étant posé, nous identifions quelques problèmes ouverts dans la fouille multi-bases de données qui sont : -La perte de connaissances – La bonne connaissance au bon endroit.

Perte de connaissances : Le processus de synthétisation de motifs non localement fréquents est un challenge. Dans plusieurs cas, un motif qui n'est pas localement intéressant est considéré comme absent dans cette base de données. Comme résultat, la synthétisation du motif non localement intéressant devient alors approximative, ce qui va affecter la prise de décision. Peu de travaux on abordé ce problème, le seul travail élaboré dans ce sens est celui de [Ramkumar.T et al. 2009] qui suggère l'introduction d'un facteur de correction dans la synthétisation de règles globales. Vu la nécessité de faire contribuer les motifs locaux non fréquents, notre contribution a consisté à prendre en considération ces motifs dans la décision globale.

La bonne connaissance au bon endroit (Multi-niveaux) : Dans la littérature, le processus de la fouille multi-bases de données [Animesh.A et al. 2010b] ne comprend que deux niveaux de connaissances : niveau local et global. Dans la réalité et avec le développement de moyens de communication et de stockage, les sociétés multi-branches sont organisées en plusieurs niveaux. Chaque niveau possède son centre de décision qui a besoin des connaissances

appropriées pour la prise de décision au niveau adéquat. Nous proposons l'introduction des connaissances du domaine, pour synthétiser des motifs multi-niveaux utiles à la prise de décision pour chaque centre de décision.

4 Contributions

Les contributions de cette thèse concernent plusieurs points dans ce contexte.

Le premier point représente notre principale contribution [Khiat.S et al. 2014a] et qui a consisté à proposer une approche de la fouille multi-bases de données qui se base sur les connaissances des utilisateurs simples ou décideurs afin de ne leur fournir que les règles d'association qui les intéressent. Afin de capter les attentes des utilisateurs, nous proposons un formalisme de représentation basé sur les ontologies dit schéma de règles multi-niveaux, et un ensemble d'opérateurs sur ces schémas de règles, permettant ainsi à l'utilisateur d'exprimer ses besoins. Nous proposons aussi une diversité de connaissances générées à chaque niveau. Ces connaissances qui peuvent êtres locales pour exprimer les particularités des sites, globales pour identifier les connaissances pour l'ensemble de sites, majoritaires et exceptionnelles pour faire ressortir les connaissances en commun et en exceptions respectivement.

Nous avons validé notre approche sur un cas réel, qui est la gestion de la maintenance des équipements industriels de l'entreprise pétrolière *SONATRACH*, plus précisément la division de liquéfaction et séparation de gaz (*LQS*) de la branche *AVAL*. Cette application offre tous les facteurs initiant à l'application de la fouille multi-bases de données. En effet, elle concerne une entreprise organisée en multi-niveaux, distribuée géographiquement, manipulant des données volumineuses, provenant de plusieurs sources. Cette application introduit une nouvelle forme de maintenance qui est la maintenance anticipée qui consiste à découvrir des relations entre pannes des équipements. Cette nouvelle forme de maintenance peut êtres introduite dans la politique de la maintenance de *SONATRACH* qui peut êtres ajoutée à la maintenance curative, prédictive, préventive et arrêt programmé.

La deuxième contribution de cette thèse a consisté à l'intégration du modèle probabiliste « maximum entropie » dans le processus de synthétisation des motifs locaux en motifs globaux [Khiat.S et al. 2014b]. Ce modèle consiste à la restitution des connaissances globales sans perte provenant des différents sites.

5 Organisation de la thèse

Cette thèse est organisée d'une introduction générale, de six chapitres et d'une conclusion générale et perspective.

Le chapitre 1 présente le principe de la fouille de données et la fouille multi-bases de données. Le chapitre 2 décrit les algorithmes de synthétisation des motifs locaux en motifs globaux et les algorithmes de génération des motifs majoritaires et exceptionnels. Ensuite nous exposons les travaux liés à l'introduction des connaissances du domaine dans le processus des règles d'association à partir d'une seule source de données avec une synthèse sur ces travaux dans le chapitre 3. Nous soulignons dans le chapitre 2 et 3, les avantages et les limites des algorithmes proposés. Le chapitre 4, présente notre première et principale contribution méthodologique pour résoudre les deux problèmes identifiés. Le chapitre 5, présente les résultats expérimentaux menés sur une base de données réelle. Le chapitre 6 décrit notre deuxième contribution qui est l'application du modèle probabiliste dans l'estimation des motifs non localement fréquents. Nous validons notre algorithme, en termes de qualité des résultats, sur une base de données synthétique. Nous terminons cette thèse par une synthèse et par un bilan des travaux réalisés et nous citons quelques directions pour des travaux futurs.

Chapitre 1 : Vers la fouille multi-bases de données

- Introduction
- Extraction de Connaissances à partir des Données (ECD)
- Extraction de règles d'association
- La fouille multi-bases de données
- Conclusion

Chapitre 1
Vers la fouille multi-bases de données

1 Introduction

Avec l'augmentation de la capacité de stockage, nous assistons durant ces dernières années à une croissance importante des moyens de génération et de collection des données. Du fait de l'informatisation rapide des administrations, des entreprises, du commerce, des télécommunications, la quantité de données disponibles sous forme numérique augmente très rapidement. Cependant, l'analyse et l'exploitation de ces données restent très difficiles. Il s'est ainsi créé un besoin d'acquisition de nouvelles techniques et méthodes intelligentes de gestion qui permettent d'extraire des données, des informations utiles (également appelées connaissances). Ces masses de données contiennent sûrement des connaissances d'une grande valeur commerciale ou scientifique. C'est ainsi que l'on a commencé à parler de processus d'Extraction de Connaissances à partir des Données (ECD) (en anglais Knowledge Discovery in Databases (KDD)) [Fayyad.U.M et al. 1996]. Le principe de ce processus est décrit dans la section 2.

En effet, l'ECD comporte plusieurs étapes pendant lesquelles l'expert humain joue un rôle important. Cependant, l'étape de la fouille de données est sans doute l'étape la plus importante dans ce processus représentée par un ensemble d'algorithmes et techniques d'extraction de connaissances que nous pouvons classifier en trois classes : les techniques de classification, de clustering et d'extraction de règles d'association. Le choix d'une technique par rapport à l'autre s'effectue selon l'objectif et la tâche de la fouille de données. La technique d'extraction des règles d'association est une branche très active de l'ECD. Il s'agit de trouver des relations ou des corrélations entre différents attributs à partir d'une masse importante de données. Cette technique est présentée dans la section 3 ainsi que l'algorithme pionnier de cette technique dit APRIORI.

Avec l'implantation des entreprises multinationales et le besoin d'extraire des connaissances dans leurs branches et unités, une extension de l'ECD s'est imposée donnant naissance à un nouveau processus appelé Extraction de Connaissances à

partir des Multiples Bases de Données (ECMBD) (en anglais Knowledge Discovery in Multi-Databases KDMD) qui est décrit dans la section 4.

2 Extraction de Connaissances à partir des Données (ECD)

Grâce aux techniques d'extraction de connaissances, les bases de données volumineuses sont devenues des sources riches et fiables pour la génération et la validation de connaissances. La fouille de données n'est qu'une phase du processus d'ECD, et consiste à appliquer les algorithmes d'apprentissage sur les données afin d'en extraire des modèles (ou motifs). L'ECD se situe à l'intersection de nombreuses disciplines comme l'apprentissage automatique, la reconnaissance de formes, les bases de données, les statistiques, la représentation de connaissances, l'intelligence artificielle, les systèmes experts, etc.(figure 1.1).

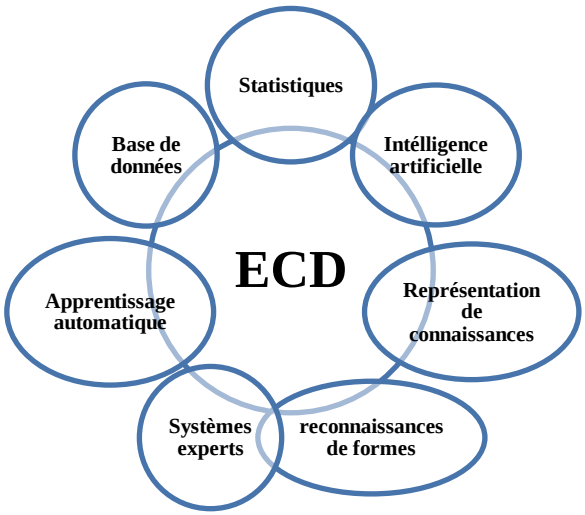


FIG. 1.1 – LES DISCIPLINES DE L'ECD

L'ECD [Nicolas.P. 2000] est un processus semi-automatique et itératif constitué de plusieurs étapes allant de la sélection et la préparation des données jusqu'à l'interprétation des résultats, en passant par la phase d'extraction de connaissances : la fouille de données. La figure 1.2 récapitule ces différentes phases ainsi que les enchainements possibles entre ces phases.

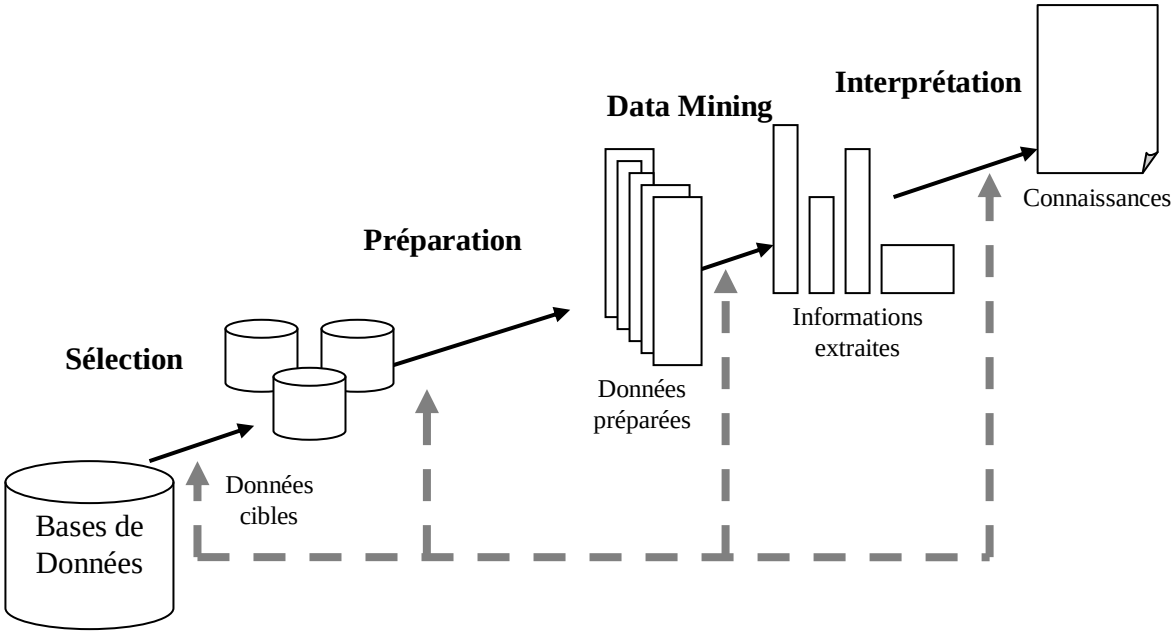


FIG. 1.2 – ETAPES DU PROCESSUS D'ECD

La phase de la fouille de données désigne l'application aux données préparées et transformées, de méthodes et techniques qui fournissent un ensemble d'information sur les données. Le type d'information fournie dépend de la finalité du processus, c'est-à-dire du problème traité. On peut distinguer les trois principales méthodes :

Classification : Le fonctionnement de la classification supervisée se décompose en deux points. Le premier est la phase d'apprentissage. Tout ce qui est appris par l'algorithme est représenté sous forme de règles de classification que l'on appelle le modèle d'apprentissage. Le second point est la phase de classification proprement dite, dans laquelle les données testées vont être utilisées pour estimer la précision des règles de classification générées pendant la première phase. La précision d'un modèle est donnée par le pourcentage d'exemples tests correctement classifiés par le modèle [Nicolas.P. 2000]. Il existe différentes techniques parmi lesquelles les arbres de décision [Quinlan.R. 1993], les réseaux de neurones [Jason.O et al. 1999] et les k-plus proches voisins [Tunkelang.D et al. 2001]. Aucune technique n'est meilleure que l'autre, il faut d'abord analyser le problème ensuite choisir la technique dont les critères sont les mieux adaptés au problème.

Segmentation : La segmentation (ou clustering en anglais), également appelée classification non supervisée, consiste à trouver des groupes homogènes dans une population. Il s'agit de regrouper les données ayant un haut degré de similitude au

sein de classes ou de groupes. Contrairement à la classification supervisée, les classes sont construites en fonction d'un ensemble d'observations mais ces classes ne sont pas connues initialement. Le problème de segmentation a été étudié dans les domaines des statistiques, de l'analyse de données, de l'apprentissage numériques et des bases de données spatiales. Il existe différentes techniques de segmentation comme les méthodes de partitionnement [Hartigan.J.A et al. 1979] [Raymond.T et al. 1994] ou encore les méthodes hiérarchiques [Tian.Z et al. 1996] [Karypis.G et al. 1999].

Règles d'association : Le problème de l'extraction de règles d'association fut introduit par [Agrawal et al. 1993]. Ce problème développé à l'origine pour l'analyse des bases de données transactionnelles des ventes, a pour but de découvrir des relations significatives entres les données de la base de données. Etant donnée une base de données de transactions, chacune constituée d'une liste d'articles achetés par un client, une règle d'association est une relation d'implication $X \rightarrow Y$ entre deux ensembles d'articles X et Y. cette règle indique que les transactions qui contiennent les articles de l'ensemble X ont tendance à contenir les articles de l'ensemble Y. Cette méthode qui est le centre d'intérêt de notre travail, est décrite dans ce qui suit.

3 Extraction de règles d'association

Les règles d'association sont une des méthodes de la fouille de données les plus répandues dans le domaine du marketing et de la distribution. Leur principale application est « l'analyse du panier de la ménagère » qui consiste, comme l'indique son nom, en la recherche d'associations entre produits sur les tickets de caisse et l'étude de ce que les clients achètent. La méthode recherche quels produits tendent à être achetés ensemble. Les règles d'association ont été appliquées avec succès dans de nombreux autres domaines, parmi lesquels l'aide à la planification commerciale, l'aide au diagnostic et en recherche médicale, l'amélioration des processus de télécommunications, de l'organisation et l'accès aux sites internet, l'analyse d'images, de données spatiales, géographiques et statistiques.

Le système génère des règles d'association de la forme "Si action1 ou condition alors action2". Elles peuvent se situer dans le temps : "Si action1 ou condition à l'instant t1 alors action2 à l'instant t2", elles sont dites dans ce cas les règles d'association séquentielles.

Exemples de règles

• Si un client achète du fromage alors il achète du pain avec une certitude de 90%.

• Si un client achète une télévision, il achètera un récepteur satellite dans un an avec une certitude de 50%.
• Si maladie X et traitement Y alors guérison avec une certitude de 95%.
• Si maladie X et traitement Y alors guérison dans Z années avec une certitude de 97%.
• Si présence et travail alors réussite à l'examen avec une certitude de 99%.

Ces règles sont intuitivement faciles à interpréter car elles montrent comment des produits ou des services se situent les uns par rapport aux autres. Elles sont particulièrement utiles en marketing et peuvent être utilisées dans le système d'information de l'entreprise. Le but principal de cette technique est donc descriptif. Dans la mesure où les résultats peuvent être situés dans le temps, cette technique peut être considérée comme prédictive. Cependant, il faut noter que cette méthode, si elle peut produire des règles intéressantes, peut aussi produire des règles triviales ou inutiles.

3.1 Processus d'extraction de règles d'association

L'extraction de règles d'association est un processus itératif et interactif. Ce processus peut se décomposer en quatre phases qui sont représentées dans la figure 1.3. Les connaissances de l'utilisateur concernant le domaine d'application sont nécessaires lors des phases de prétraitement, afin d'assister la sélection et la préparation des données, et de post-traitement, pour l'interprétation et l'évaluation des règles extraites. Avant de présenter ce processus, nous donnons quelques définitions de bases.

Définition 1.1 (Item)
Un item est tout article, attribut, littéral appartenant à un ensemble fini d'éléments distincts $X = \{x_1, x_2, \ldots, x_n\}$.

Exemple 1.1 : Dans les applications de type analyse du panier de la ménagère, les articles en vente dans un magasin sont des items. L'ensemble X peut contenir les items A,B,C et D correspondant aux articles lait, beurre, pain et confiture par exemple.

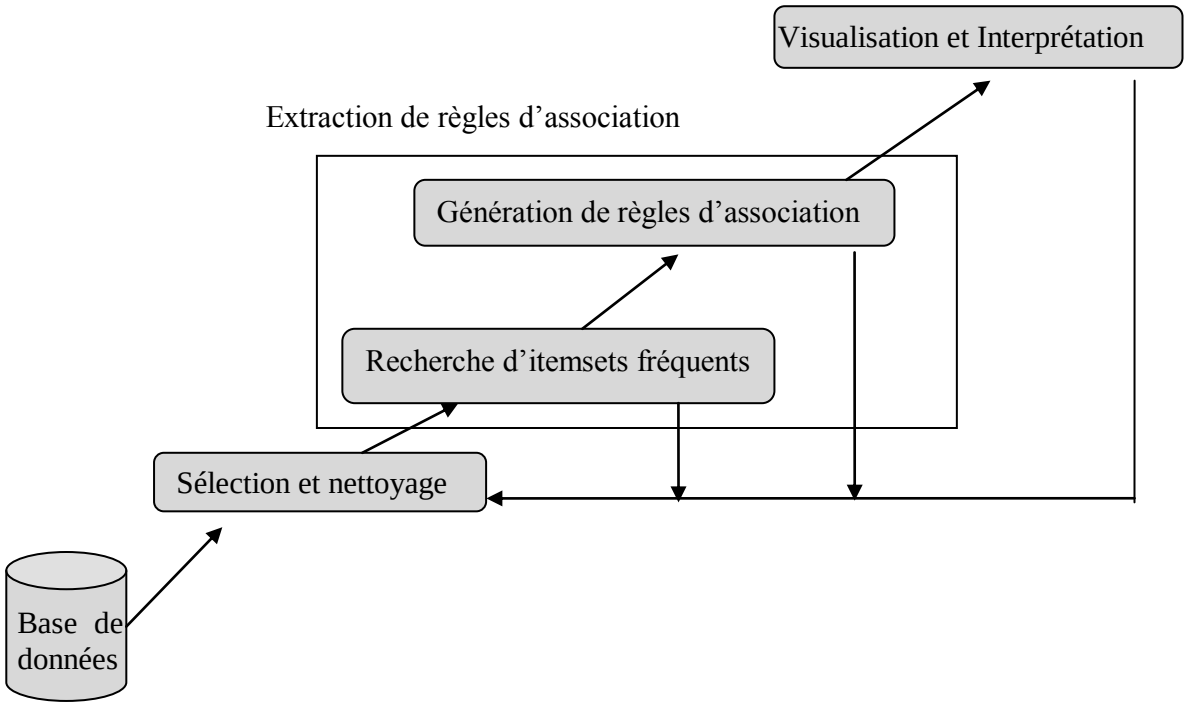


FIG. 1.3 – PROCESSUS D'EXTRACTION DES REGLES D'ASSOCIATION

Définition 1.2 (Itemset)
On appelle itemset tout sous-ensemble d'items de X. Un itemset constitué de k-items sera appelé un k-itemset.

Exemple 1.2 : La conjonction pain $\wedge$ beurre $\wedge$ lait est un itemset composé de trois items pain, beurre et lait. Cette conjonction est 3-itemsets.

Définition 1.3 (Contexte d'extraction de règles d'association)
Un contexte d'extraction de règles d'association est un triplet *D=(O,I,R)* dans lequel *O* et *I* sont respectivement des ensembles finis d'objets et d'items et $R \subseteq O * I$ est une relation binaire entre les objets et les items. Un couple $(o,i) \in R$ dénote le fait que l'objet $o \in O$ est en relation avec l'item $i \in I$. L'ensemble O est appelé aussi transactions et le contexte d'éxtraction est appelé aussi base de données transactionnelle.

Exemple 1.3 (Exemple de contexte d'extraction de règles d'association) : Le tableau 1.1 présente un contexte d'extraction *D* constitué de six objets ou six transactions, chacune représentée par son identifiant (TID) et de cinq items (A,B,C,D,E).

TID	Items
1	A C D
2	B C E
3	A B C E
4	B E
5	A B C E
6	B C E

TAB. 1.1 – Contexte d'extraction de règles d'association D

3.1.1 Sélection et nettoyage des données

Cette phase consiste à sélectionner les données (attributs et objets) de la base de données utiles à l'extraction des règles d'association et transformer ces données en un contexte d'extraction. L'extraction de règles d'association peut être effectuée à partir des bases de données de divers types, comme des données spatiales, temporelles, orientées objets, multimédia, etc. Cette première phase est très importante car à partir de la qualité des données en entrée dépend la qualité des résultats. Cette phase est nécessaire pour pouvoir appliquer les algorithmes d'extraction des règles d'association sur des données de nature différente provenant de sources différentes, de concentrer la recherche sur les données utiles pour l'application et de minimiser le temps d'extraction. Le problème des données incomplètes (valeurs manquantes, etc.), et des données erronées ou incertaines et la taille du jeu de données doivent être pris en considération dans cette phase.

3.1.2 Recherche des itemsets fréquents

Cette phase consiste à extraire à partir du contexte *D* les itemsets fréquents. La recherche des itemsets fréquents d'un contexte quelconque *D* est un problème non trivial car le nombre d'itemsets fréquents potentiels est exponentiel en fonction du nombre d'items du contexte *D*.

Définition 1.4 (Support d'un itemset)
Le support d'un itemset *X* est le pourcentage de transactions de *D* (contexte) dans lequel *X* est un sous-ensemble :

$$supp(X) = \frac{|\{t \in O \ / X \subseteq t\}|}{|t \in O|}$$

Exemple 1.4 : Dans l'exemple du tableau 1.1, le support de l'itemset AC est égale 0.5.

En effet, $supp(AC) = \frac{3}{6} = 0.5$

Définition 1.5 (Itemset fréquent)
Un itemset est dit fréquent, si : $supp(X) \geq minsup$, où $minsup$ est spécifié par l'utilisateur et fixe la borne inférieure du support. L'ensemble des itemsets fréquents est noté par *F*.

Exemple 1.5 : Si on fixe *minsup*=0.4 on peut dire que l'itemset AC est fréquent car :

$$supp(AC) = 0.5 \geq minsup = 0.4$$

3.1.3 Génération des règles d'association

La génération des règles d'association s'effectue à partir des itemsets fréquents générés précédemment. Nous donnons dans ce qui suit quelques définitions :

Définition 1.6 (Règle d'association)
Une règle d'association est une implication de la forme $X_1 \rightarrow X_2$, où X_1 et X_2 sont des itemsets, tels que $X_1, X_2 \subseteq D \; et \; X_1 \cap X_2 \neq \emptyset$. La partie gauche de la règle X_1 est appelée la prémisse ou entête ou antécédent de la règle et la partie droite X_2 est appelée la conclusion ou la conséquence de la règle.

Définition 1.7 (Support d'une règle d'association)
Le support d'une règle d'association r : $X_1 \rightarrow X_2$ est égal au support de l'union des itemsets qui la constituent :

$$supp(r) = supp(X_1 \cup X_2)$$

Exemple 1.6 : Le support de la règle $AC \rightarrow D$ d'après l'exemple 1.1 est :

$$supp(ACD) = \frac{1}{6} = 0.16$$

Définition 1.8 (Confiance)
La confiance de la règle d'association $X_1 \rightarrow X_2$ est la confiance conditionnelle que la transaction contienne X_2 sachant X_1 :

$$conf(r) = \frac{supp(X_1 \cup X_2)}{supp(X_1)}$$

Exemple 1.7 : La confiance de la règle $AC \rightarrow D$ est :

$$conf(ACD) = \frac{supp(ACD)}{supp(AC)} = \frac{1/6}{3/6} = 0.33$$

Définition 1.9 (Règle exacte)
Une règle d'association $X_1 \rightarrow X_2$ est dite exacte, quand son indice de confiance est égal à 1.

Définition 1.10 (Règle approximative)
Une règle d'association $X_1 \rightarrow X_2$ est dite approximative, quand son indice de confiance n'est pas égal à 1.

Définition 1.11 (Règle d'association fréquente)
Une règle d'association r : $X_1 \rightarrow X_2$ est fréquente, si l'itemset $X_1 \cup X_2$ est fréquent.

Définition 1.12 (Règle d'association de confiance (ou valide))
Une règle d'association r : $X_1 \rightarrow X_2$ est dite de confiance, si : *conf(r)* ≥ *minconf*, où *minconf* est spécifié par l'utilisateur et fixe la borne inférieure de la confiance.

En général, la génération des règles d'association est réalisée de manière directe, sans accéder au contexte d'extraction, et le coût de cette phase en temps d'exécution est donc faible comparé au coût de l'extraction des itemsets fréquents. La génération de règles d'association est assez simple. Pour chaque itemset fréquent X_1 dans F, tous les sous-ensembles X_2 de X_1 sont déterminés et la valeur de la confiance (r) est calculée. Si cette valeur est supérieure ou égale au seuil minimal de confiance alors la règle d'association $X_2 \rightarrow (X_1 - X_2)$ est générée.

3.1.4 Visualisation et interprétation

C'est la phase finale du processus d'ECD. Cette phase consiste en la visualisation par l'utilisateur des règles d'association extraites du contexte et leur interprétation afin d'en déduire des connaissances utiles pour l'amélioration de l'activité concernée. Ainsi l'expert du domaine peut juger de leurs pertinences et utilités. Mais le nombre important des règles d'association extraites impose le développement d'outils de classification de règles selon leurs propriétés, de sélection de sous-ensembles de règles selon des critères définis par l'utilisateur, et de visualisation de ces règles sous une forme intelligible. Cette nouvelle problématique est également appelée « Knowledge Mining ». La forme de présentation de règles peut être textuelle, graphique ou bien une combinaison de ces deux formes intelligibles. Ceci va donner naissance à un nouveau domaine de recherche : la fouille visuelle de données « Visual Data Mining » afin d'améliorer

le processus d'extraction de connaissances en proposant des outils de visualisation adaptés à différentes problématiques. Les connaissances de l'utilisateur concernant le domaine d'application sont nécessaires lors des phases de prétraitements afin d'assister la sélection et la préparation des données et de post-traitement, pour l'interprétation et l'évaluation des règles extraites. En fonction de l'évaluation des règles extraites, les paramètres utilisés lors des précédentes phases (critères de sélection et préparation des données et seuils minimaux de support et de confiance) peuvent être modifiés avant d'effectuer à nouveau l'extraction des règles d'association, ceci afin d'améliorer la qualité du résultat.

3.2 Algorithme général de recherche de règles d'association

La découverte de règles d'association (ALG. 1.1) peut être scindée en deux étapes :

- Extraction des itemsets fréquents
- Recherche des règles d'association

ALG. 1.1 Algorithme général de recherche de règles d'association

Entrée : Un Contexte d'extraction *D,minsup et minconf*
Sortie : Une liste de règles d'association F_{ra}
// Recherche des itemsets fréquents
$F_g \leftarrow$ **a) Extraction des itemsets fréquents**;
// Génération de règles d'association
$F_{ra} \leftarrow$ **b) recherche des règles valides;**
Retourner F_{ra}

La performance de tout algorithme basé sur cette approche dépend de la phase d'extraction des itemsets fréquents. C'est une phase non triviale vu son aspect combinatoire pour générer tous les itemsets fréquents. L'espace de recherche pour l'énumération de tous les itemsets fréquents possibles de $|I| = m$ est de 2^m-1, et donc exponentiel en m. Ce problème reste ouvert et constitue la majeure partie des efforts de recherche actuelle.

La seconde étape d'extraction des règles constitue la phase la plus simple, elle est accomplie en considérant tous les sous ensembles des itemsets fréquents pour générer des règles avec des conséquences multiples. Ce deuxième sous-problème est exponentiel dans la taille des itemsets fréquents, car pour un itemset fréquent A, le nombre de règles d'association qui peuvent être générées est de $2^{|A|}-2$.

Cependant les temps de calcul sont faibles puisque aucun balayage de la base de données n'est nécessaire pour la génération des règles. C'est la raison pour laquelle le problème de la recherche de règles d'association se restreint au problème d'optimisation de la découverte des itemsets fréquents. Pour cela, plusieurs algorithmes ont été proposés dans la littérature, dont la majorité se base sur l'algorithme générique *APRIORI*, c'est l'algorithme pionnier pour la recherche des itemsets fréquents. Pour cela on a jugé utile de décrire cet algorithme en détail dans ce qui va suivre.

3.3 L'algorithme APRIORI

Agrawal et al. ont proposé dans [Agrawal et al. 1994] le premier algorithme d'extraction des règles d'association dans les bases de données transactionnelles qui est l'algorithme *APRIORI*. Étant donné une base de données transactionnelles D, le problème consiste à générer toutes les règles d'association valides liant les itemsets fréquents entre eux.

APRIORI se base essentiellement sur la propriété d'antimonotonicité existante entre les itemsets. En effet, cette propriété est utilisée à chaque itération de l'algorithme *APRIORI* afin de diminuer le nombre d'itemsets candidats à considérer ainsi que le calcul de leurs supports.

Propriété 1.1 (Propriété d'antimonotonicité)
Tous les sous-ensembles d'un itemset fréquent sont fréquents.

Cette propriété permet de limiter le nombre de candidats de taille k générés lors de la $k^{ème}$ itération en réalisant une jointure conditionnelle des itemsets fréquents de taille k-1 découverts lors de l'itération précédente.

Propriété 1.2 (Propriété sur les sur-ensembles)
Tous les sur-ensembles d'un itemset non fréquent sont non fréquents.

Cette propriété permet de supprimer un candidat de taille k lorsque au moins un de ses sous-ensembles de taille k-1 ne fait pas partie des itemsets fréquents découverts lors de l'itération précédente.

3.3.1 Extraction des Itemsets fréquents

L'algorithme *APRIORI* utilise une approche itérative par niveaux pour générer les itemsets fréquents. L'algorithme *APRIORI* effectue à chaque itération k, un passage dans la base de données afin de calculer le support de chaque k-itemset. Le pseudo code de cette phase est donné par l'algorithme 1.2. On note dans ce qui

suit, l'ensemble des k-itemsets candidats (i.e. dont on ne connaît pas encore le support dans D) par C_k et l'ensemble des k-itemsets fréquents de taille k par F_k.

$$C_k = \{(c_k, supp(c_k)) \mid \forall X \subseteq c_k, X \neq \emptyset, supp(X) \geq minsup\}$$
$$F_k = \{(l_k, supp(l_k)) \mid l_k \; est \; un \; k-itemset \; \; et \; \; supp(l_k) \geq minsup\}$$

Le contexte d'extraction D est d'abord parcouru afin de trouver F_1, l'ensemble des $1-$itemsets fréquents dans D. Ensuite l'algorithme alterne entre génération et calcul des supports des candidats de taille k.

En effet, à l'itération k, l'ensemble F_{k-1} des *(k-1)−itemsets* fréquents correspondant aux itemsets de niveau *k-1*(calculés à l'étape précédente), est utilisé pour générer l'ensemble C_k des *k−itemsets* candidats. La fonction de génération de candidats appelée *Apriori-Gen* prend en argument F_{k-1} et retourne C_k par l'opération d'auto-jointure entre F_{k-1}. Deux itemsets p et q de F_{k-1} forment un F_k −itemset c si et seulement s'ils ont *k-2* itemsets en commun. Ensuite, la fonction *a-sous-ensemble-infréquent* est appelée à partir de *Apriori-Gen*, après avoir généré un candidat de taille k à partir de deux F_{k-1} −itemsets fréquents, et ceci pour vérifier si le nouveau candidat ne contient pas un sous-ensemble infréquent auquel cas le candidat lui même serait infréquent selon la propriété d'antimonotonicité.

ALG. 1.2 APRIORI : Génération des itemsets fréquents [Couturier.O. 2005]

Entrée : Contexte d'extraction D et un seuil minimal de support *minsup*
Sortie : Un ensemble F_k de k-itemsets fréquents
$F_1 = \{1 - itemsets \; fréquents\}$
Pour $(k = 2; F_{k-1} \neq \emptyset; k++)$
$C_k = Apriori - Gen(F_{k-1})$
  Pour toutes transactions $t \in D$
    $C_t = sous - ensemble\ (C_k, t)$ /$*C_t = \{c \in C_k, c \subseteq t\}*$/
    Pour chaque candidat $c \in C_t$
      ++supp(c)
    fin pour
  fin pour
  $F_k = \{ c \in C_k / supp(c) \geq minsup\}$
fin pour
retourner F_k

Fonction $Apriori - Gen\ (F_{k-1})$

Entrée : Un ensemble F_{k-1} de *(k-1)*-itemsets fréquents;
Sortie : Un ensemble C_k de *k*-itemsets candidats;

Pour chaque itemset $p \in F_{k-1}$
  Pour chaque itemset $q \in F_{k-1}$
    si $p[1] = q[1], \dots, p[k-2] = q[k-2], p[k-1] < q[k-1]$
    alors $c = p \cup q$ //étape de jointure : générer les candidats
      si a-sous-ensemble-infréquent $(c, F_{k-1}) =$ faux alors ajouter c à C_k
      sinon supprimer c de C_k
      fin si
    fin si
  fin pour
fin pour
retourner C_k

Fonction a-sous-ensemble-infréquent $(c, F_{k-1}) =$ faux

Entrée : Un ensemble F_{k-1} de *(k-1)*-itemsets fréquents et un candidat c;
Sortie : Un booléen;
Pour chaque (k-1)-sous ensemble s de c
    si $s \notin F_{k-1}$ alors retourner vrai
    fin si
    retourner faux
fin pour

Ensuite un parcourt du contexte d'extraction est effectué afin de déterminer le support de chaque candidat C_k. La fonction $\text{sous} - \text{ensemble}\,(C_k, t)$ recherche parmi les candidats de C_k ceux qui sont inclus dans la transaction t. Si c'est le cas, alors le support de ces candidats est incrémenté. Parmi les candidats seuls les candidats fréquents notés par F_k sont ajoutés à l'ensemble *F*.

3.3.2 Génération des règles d'association

Pour générer les règles d'association, on considère l'ensemble *F* des itemsets fréquents trouvés en phase précédente. Pour chaque itemset fréquent *l*, on prend tous ses sous-ensembles (tous fréquents d'après la propriété d'antimonotonicité). Á partir de ces sous-ensembles fréquents, on génère toutes les règles valides de la forme générale suivante :

$$(l - C) \rightarrow C$$

Le pseudo code de cette phase est donné par l'algorithme 1.3. Étant donné un itemset fréquent *l*, l'algorithme génère toutes les règles ayant un item en conséquence. Les conséquences de ces règles sont ensuite combinées en réutilisant la fonction *Apriori-Gen*, et ce pour générer toutes les conséquences possibles à

deux items pouvant apparaître dans une règle générée à partir de l et ainsi de suite. L'algorithme récursif utilisant cette idée est donné dans ce qui suit. F représente l'ensemble des itemsets fréquents et H_m l'ensemble des m-itemsets conséquents de règles.

ALG. 1.3 *APRIORI* : Génération des règles d'association [Couturier.O. 2005]

Entrée : Un ensemble des itemsets fréquents F ; Confiance minimum *minconf*
Sortie : Un ensemble des règles d'association R
$R = \emptyset$
Pour chaque $k - itemset\ l_k \in F, k \geq 2$ faire
$H_1 = \{1 - itemsets\ fréquents\ sous - ensemble\ de\ l_k\}$
Pour chaque $h_1 \in H_1$ faire

$$conf(r) = \frac{supp(l_k)}{supp(l_k - h_1)}$$

si *conf (r)* $\geq$ *minconf* alors
r: $(l_k - h_1) \rightarrow h_1$
$R = R \cup r$
sinon supprimer h_1 de H_1
fin si
fin pour
Gen-Règle (l_k, H_1)
fin pour
Retourner R

Procédure Gen-Règle(l_k, H_m)

Entrée : k-itemsets fréquent l_k , ensemble H_m de m-itemsets conséquences de règles valides générées à partir de *l - k* et un seuil minimal *minconf*.
Sortie : Un ensemble de règles d'association R valides augmenté des règles valides générées à partir de l_k, dont la conséquence est (m+1)-itemset.
si $(k > m + 1)$ alors
$H_{m+1} = \text{Apriori} - \text{Gen}(H_m)$
Pour tout $h_{m+1} \in H_{m+1}$ faire

$$conf(r) = \frac{supp(l_k)}{supp(l_k - h_{m+1})}$$

Si *conf (r)* $\geq$ *minconf* alors
r: $(l_k - h_{m+1}) \rightarrow h_{m+1}$
$R = R \cup r$
Sinon supprimer h_{m+1} de H_{m+1}
fin si
fin pour
Gen-Règle (l_k, H_{m+1})
fin si.

3.4 Les limites de la fouille monobase de données

Le processus de l'ECD est intéressant dans l'environnement monobase de données où une seule base de données est à considérer. Durant ce processus et après la phase de prétraitement, une technique de fouille de données est appliquée sur cet ensemble de données pour extraire des connaissances. Ensuite après la phase de visualisation et interprétation, les décideurs peuvent se baser sur ces connaissances pour prendre des décisions. Un manager d'un supermarché, par exemple, peut tirer profit de ce processus pour segmenter ses clients et arranger ainsi ses étages selon leurs comportements. Aussi un dirigeant d'une entreprise, par exemple, peut tirer profit de ce processus pour planifier les actions de formation de son personnel et prévoir leur parcours professionnel. Ce qui n'est pas le cas pour un manager de plusieurs supermarchés ou d'un dirigeant d'entreprise implanté dans plusieurs régions ils ont besoin des informations additionnelles à celles générées par le processus de l'ECD. Dans ce cas, pour prendre en considération l'ensemble des données des différentes bases de données implantées dans différentes régions nous devons construire une seule base de données volumineuse qui est l'union de toutes les bases de données. Cette base de données constitue la source de données du processus de l'ECD. Mais malheureusement ce processus est inadapté dans ce nouveau environnement pour les raisons suivantes :

La taille de la base de données résultante : Les bases de données des différentes régions ou sites peuvent être volumineuses. Cependant l'application des techniques traditionnelles de la fouille de données à ces bases de données volumineuses peut prendre un temps énorme pour générer les connaissances globales. Si nous prenons l'exemple de l'algorithme *APRIORI* qui nécessite plusieurs parcours des bases de données et génère un nombre important de candidats, le recours au parallélisme s'impose pour réduire ce temps d'exécution. Ce qui nécessite un investissement supplémentaire en logiciel et matériel pour les managers des supermarchés et les directeurs d'entreprise, ce qui n'est pas évident.

La confidentialité des données : Mettre l'ensemble des bases de données des différentes régions dans une seule base de données volumineuse nécessite le transfert et l'appropriation des données des différentes régions et sites. Ce qui n'est pas toujours possible dans certains domaines tels que la médecine et les ressources humaines où les données sont strictement confidentielles. A cet effet, le transfert des données brutes est quasiment impossible pour plusieurs considérations telles que la confidentialité, la concurrence et la préservation de la vie privée des gens.

Perte de l'information sur la distribution des données : La consolidation des données des différents sites dans un seul site central détruit les informations sur la particularité des sites ou régions et les caractéristiques d'une région par rapport à d'autres. En effet, l'information sur la distribution des données sera perdue du faite qu'on va prendre en considération les données comme un tout et non pas par site ou par région.

Poids des sites : La consolidation des données de tous les sites dans un seul site permet de considérer l'ensemble des sites de la même manière sans distinction. Un site actif et un autre non actif sont considérés de la même façon du moment qu'on va mélanger les données des deux sites dans un seul site. Ces deux sites auront le même poids de décision au niveau global, ce qui est aberrant.

Besoin d'autres types de connaissances : l'union de toutes les bases de données dans une seule base de données et l'application d'un algorithme de fouille de données traditionnel permet de ne générer que les connaissances globales. Alors que le manager des supermarchés et le directeur de l'entreprise multi-branches par exemple auront besoin aussi d'autres connaissances par site ou par région appelées connaissances locales. Ainsi ils ont besoin de savoir la particularité d'un site par rapport à d'autres exprimée en connaissances exceptionnelles, et aussi de tout ce qui est commun en termes de connaissances exprimé par les connaissances majoritaires.

Bien que le processus de l'ECD a donné des preuves dans un environnement monobase de données il est inefficace dans un environnement multi-bases de données. Ce qui a donné naissance à un nouveau processus appelé la fouille multi-bases de données (*FMBD*) dont le principe est décrit dans ce qui suit.

4 La fouille multi-bases de données

4.1 Définition

On définit la Fouille Multi-Bases de Données *FMBD* (en anglais MDM: Multi-Databases Mining) comme étant le processus d'extraction de connaissances à partir de plusieurs sources ou bases de données souvent hétérogènes pour trouver de nouveaux motifs utiles et significatifs [Ramkumar.T et al. 2010]. Un motif peut être soit un itemset ou une règle d'association. A la différence de la fouille monobase de données, cette approche apporte d'importants avantages tels que :

La richesse du type de connaissances générées : Une des forces de la fouille multi-bases de données est la richesse des connaissances générées. Ce qui n'est pas le cas dans la fouille monobase de données où les connaissances générées sont locales. Dans la fouille multi-bases de données nous trouvons les connaissances locales, globales, majoritaires et exceptionnelles. La définition de ces nouveaux types de connaissances est donnée dans le paragraphe 4.3.2.

La capture de l'individualité des sources de données : Dans une entreprise multinationale et multi-branches les managers ont besoin de trouver les particularités de certaines branches, régions ou pays en termes de connaissances pour pouvoir prendre des décisions adéquates. Chose qui est impossible avec la fouille monobase de données où les données sont consolidées dans une seule base de données ce qui détruit la connaissance sur la distribution des données. La fouille multi-bases de données offre ce type de service pour prendre en compte la spécificité de chaque région.

La confidentialité des données : La fouille multi-bases de données procède par une extraction des connaissances locales ensuite les synthétiser en connaissances globales, majoritaires et exceptionnelles. Les données sources restent chez leurs propriétaires, ce qui préserve la confidentialité des données. Au lieu de transmettre les données il est plus judicieux de transférer les connaissances générées dans chaque site vers des niveaux supérieurs.

4.2 Applications

Chaque organisation distribuée à travers plusieurs régions qui dispose de plusieurs branches trouve son intérêt dans la fouille multi-bases de données. Cette dernière a pour but d'identifier des connaissances locales, globales, majoritaires et exceptionnelles. Les connaissances ainsi identifiées peuvent être utiles pour de nombreux organismes commerciaux, scientifiques, industriels et dans la gestion de l'information, afin d'améliorer leurs résultats dans leurs activités.

4.3 Le processus de la fouille multi-bases de données à deux niveaux

Une organisation multi-branches est composée de plusieurs branches. Les banques nationales par exemple ont plusieurs branches dans différentes locations. Chaque branche dispose de sa propre base de données par conséquent les données de la banque sont largement distribuées.

La figure 1.4 illustre la structure d'une organisation multi-branches à deux niveaux. Le niveau supérieur représente la direction centrale de l'organisation (DC) qui est responsable du développement et de la prise de décision sur l'ensemble de l'organisation. Le niveau du milieu représente les *n* branches de l'organisation $BR_1, BR_2, ..., BR_n$. le niveau bas représente les bases de données des différentes branches $BD_1, BD_2, ..., BD_n$.

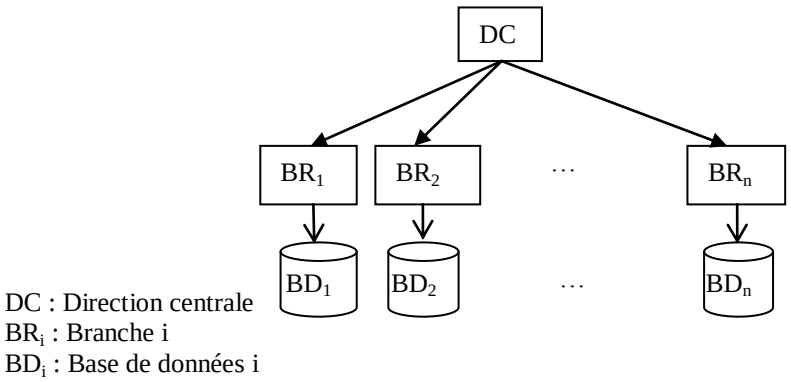


FIG. 1.4 – STRUCTURE D'UNE ORGANISATION A DEUX NIVEAUX

Le processus de la fouille de données appliqué à cette architecture repose sur deux phases : la phase intra-site et la phase inter-site illustrées dans la figure 1.5.

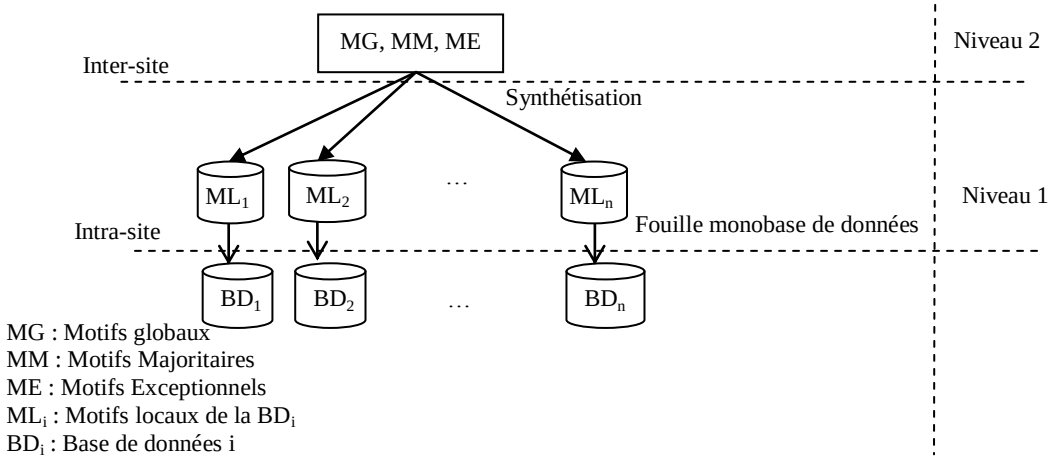


FIG. 1.5 – LA FOUILLE MULTI-BASES DE DONNEES A DEUX

4.3.1 Phase intra-site

Cette phase consiste à l'extraction de connaissances locales à partir de plusieurs bases de données. Un algorithme de la fouille de données est appliqué pour extraire

des motifs locaux. Ces motifs sont intéressants pour les managers des branches pour prendre des décisions locales.

4.3.2 Phase inter-site

Généralement on appelle cette phase : « synthétisation » où les connaissances locales sont synthétisées et transformées en connaissances globales, majoritaires et exceptionnelles. Ces motifs peuvent aider les managers au niveau central pour prendre des décisions au niveau de toute l'organisation.

Les motifs locaux : Les succursales d'une société multi-branches ont besoin de fouiller sur leurs propres données locales pour identifier les motifs locaux afin de prendre des décisions locales. En effet, chaque succursale d'une société multi-branches possède ses fonctions individuelles, son propre plan et sa politique pour le développement et la concurrence. Il convient donc d'analyser uniquement les données disponibles dans leurs bases de données.

Les motifs majoritaires : Ce sont des motifs qui sont pris en charge par la plupart des succursales. Ils sont le reflet des caractéristiques communes entre les succursales et sont généralement utilisés pour la prise des décisions globales.

Les motifs exceptionnels : Ce sont des motifs supportés par quelques succursales. Ils reflètent l'individualité de succursales et sont généralement utilisés pour créer des politiques spéciales spécifiquement pour ces succursales.

Les motifs Globaux : Ce sont des motifs dont le support sur l'ensemble des sites est supérieur ou égal au *minsup*. Ce type de motif prend en compte le poids de chaque site pour déterminer le support. Généralement, le poids du site peut être le nombre de transactions ou la fréquence d'apparition des règles d'association dans les sites.

5 Conclusion

Nous venons de présenter dans ce chapitre le processus de l'ECD et nous avons situé la technique d'extraction des règles d'association à partir d'une seule source de données. Nous avons ensuite élargi les concepts de l'ECD dans un environnement multi-sources de données avec tous ses aspects.

Chapitre 2 : Les Algorithmes de la fouille multi-bases de données

- Introduction
- Les critères de l'étude
- Les algorithmes étudiés
- Etude comparative
- Synthèse
- Conclusion

Chapitre 2

Les algorithmes de la fouille multi-bases de données

1 Introduction

L'analyse de motifs locaux est une méthode utilisée pour synthétiser les motifs locaux issus des algorithmes d'extraction de connaissances à partir de multiples sources de données. En effet elle consiste à fouiller les sources de données individuellement en générant les motifs locaux qui vont être transmis au niveau central afin de les synthétiser en motifs globaux.

De nombreuses méthodes d'analyse de motifs locaux ont été proposées au cours de ces dernières années [Animesh.A et al. 2010b] [Ramkumar.T et al. 2008] [Zhang.S et al. 2003]. Certaines produisent des motifs à deux niveaux, d'autres donnent des motifs à plusieurs niveaux. Nous décrivons dans ce qui suit quelques méthodes de synthétisation de motifs locaux.

Dans la section 2, nous définissons les critères sur lesquels nous nous sommes basés pour notre étude. La section 3 passe en revue les algorithmes de synthétisation des motifs locaux. La section 4 introduit une comparaison de ces algorithmes sur la base de critères que nous avons définis à cet effet, suivie par une synthèse dans la section 5. Enfin nous terminons ce chapitre par une conclusion.

2 Les critères de l'étude

Pour étudier les algorithmes d'analyse de motifs locaux, nous avons défini quelques critères qui nous paraissent importants. En effet, ces critères doivent contenir les paramètres qui permettent d'expliquer les différences entre les algorithmes de synthétisation des motifs locaux. Nous détaillons ci-dessous ces critères :

Les Motifs : Nous nous intéressons aux méthodes qui s'appliquent aux itemsets fréquents ou aux règles d'association. Nous précisons pour chaque méthode étudiée, le type des motifs qui sont synthétisés.

La richesse de génération : Nous identifions pour chaque méthode, les types de connaissances générées.

Le poids de chaque site : Pour définir le poids de chaque site, certaines méthodes utilisent le nombre de transactions d'autres se basent sur le nombre de règles générées de fréquence élevée. Nous définissons pour chaque méthode le poids utilisé.

Les niveaux de synthèse : Dans la littérature les méthodes proposées sont calquées sur une organisation à deux niveaux : -Sites d'exploitations -Site central. Ce qui ramène à définir deux niveaux de synthétisation des motifs. Le premier niveau repose sur l'extraction de motifs locaux pour chaque site. Ensuite ces motifs sont synthétisés dans un deuxième niveau. Nous indiquons pour chaque méthode le nombre de niveau traité.

La régénération : Pour les méthodes qui produisent des motifs globaux, nous décrivons la méthode de régénération proposée pour les motifs.

Contrôle de la taille des ensembles de motifs : La taille des motifs est un facteur important pour les méthodes de synthétisation de motifs locaux. En effet, un ensemble de motifs volumineux est difficilement exploitable alors qu'un ensemble de taille plus réduite est facilement exploitable mais il est généralement sujet à une perte d'information plus importante.

Perte de l'information : Les algorithmes de synthétisation doivent prendre en compte toutes les informations sur les motifs y compris ceux qui sont non localement fréquents pour pouvoir les synthétiser sans perte d'information.

Processus guidé par l'utilisateur : L'analyse du grand nombre de motifs générés peut être difficile par l'utilisateur dans la fouille multi-bases de données. Guider le processus de découverte des motifs fréquents par l'utilisateur est une manière de réduire ce nombre important et de ne présenter à l'utilisateur que ce qui l'intéresse.

3 Les algorithmes étudiés

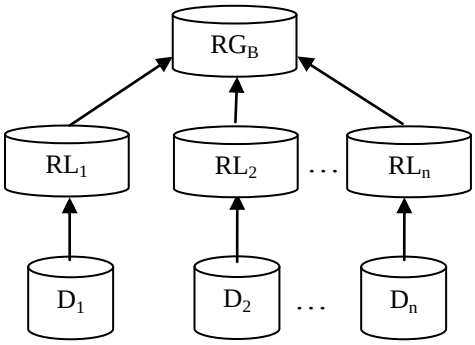
Dans cette section, nous allons présenter les lignes directrices des algorithmes les plus importants parus dans la littérature, en nous focalisant essentiellement sur l'étape de synthétisation des motifs locaux. Nous avons recensé sept algorithmes qui nous semble les plus intéressants et qui feront l'objet d'une étude critique pour dégager les pistes qu'on devra investir pour construire notre approche.

3.1 Synthétisation des règles de fréquence élevée à partir de plusieurs sources de données [Xindong.W et al. 2003]

L'algorithme de synthétisation des règles d'association proposé par [Xindong.W et al. 2003] est sans doute l'un des premiers algorithmes à introduire la notion de synthétisation des règles d'association. Il permet la synthétisation des règles d'association locales à partir de sources de données de même taille par pondération. Le principe de l'algorithme consiste à calculer le poids de chaque source de données et à déterminer le support global suivant l'opération de synthétisation. Le poids de chaque site est déterminé selon le nombre d'apparition des règles d'association dans le site. Selon les auteurs de cet algorithme, un site est important s'il contient un nombre important de règles d'association apparaissant dans plusieurs sites d'où l'affectation d'un poids plus grand que les autres sites. Le calcul du support global repose sur le poids du site et du support local de la règle d'association.

3.1.1 Méthode

Soit *m* sources de données d'une organisation, le problème d'extraction et de synthétisation des règles d'association se décompose en deux étapes illustrées dans la figure 2.1.



D_i : La $i^{ème}$ source de données.
RL_i : Les règles d'association locales de D_i.
RG_B : les règles globales.

FIG. 2.1 – MODELE DE SYNTHESE

Etape 1 : D'abord on applique la fouille locale des règles d'association pour chaque source de données en calculant pour chaque règle son support/confiance

local(e) en utilisant l'algorithme *APRIORI* défini par Agrawal [Agrawal et al. 1994].

Etape 2 : Ensuite, on applique l'opération de synthétisation des règles locales pour trouver des règles d'association globales pour tous les sites. Dans cette étape, avant de faire l'opération de synthétisation, le poids de chaque site est calculé selon la fréquence d'apparition des règles d'association.

3.1.1.1 Définition du poids de chaque source de données

Soit RL_i l'ensemble des règles d'association découvertes dans D_i et $RL = \{RL_1, RL_2, \dots, RL_m\}$ l'ensemble de toutes les règles d'association découvertes à partir de toutes les sources de données. Le poids de chaque source de données D_i est défini comme suit :

$$w_{Di} = \frac{\sum_{R_k \in RL_i} Num(R_k) * w_{R_k}}{\sum_{j=1}^{m} \sum_{R_h \in RL_j} Num(R_h) * w_{R_h}}$$

Où W_{R_k} est le poids de la règle R_k et *Num(R_k)* est le nombre de sources de données supportant R_k ou bien la fréquence d'apparition de R_k dans *RL*.

Le poids de la règle R_i de *RL* est défini comme suit :

$$w_{Ri} = \frac{Num(R_i)}{\sum_{j=1}^{n} Num(R_j)}$$

D'après ce modèle, une règle à fréquence élevée a un poids plus important et la source de données contenant un nombre élevé de règles d'association à fréquence élevée a un poids important par rapport aux autres sources de données.

3.1.1.2 Définition du modèle de synthétisation par pondération

Soient $D_1, D_2, \dots, D_m$, m sources de données de même taille et RL_i l'ensemble des règles d'association extraites à partir de la source de donnée D_i. Etant donnée une règle d'association $X \rightarrow Y$, avec $W_1, W_2, \dots, W_m$ les poids des sources de données respectivement $D_1, D_2, \dots, D_m$, le modèle de synthétisation est défini comme suit :

$$Supp_G(X \cup Y) = \sum_{i=1}^{m} W_i \times Supp_i(X \cup Y),$$
$$Conf_G(X \rightarrow Y) = \sum_{i=1}^{m} W_i \times Conf_i(X \rightarrow Y),$$

Où $\mathrm{Supp_G(R)}$ et $\mathrm{Conf_G(R)}$ représentent le support et la confiance globaux de R et $\mathrm{Supp_i(R)}$ et $\mathrm{Conf_i(R)}$ le support et la confiance de R dans D_i.

3.1.2 Discussion

Les auteurs de cet algorithme se sont basés sur la fréquence d'apparition des règles d'association générées pour chaque site pour déterminer les règles d'association globales. Si un site contient un nombre important de règles d'association à fréquence élevée il doit avoir certainement un poids plus important que les autres sites qui ont moins de règles d'association à fréquence élevée. Ceci se justifie selon ces auteurs par le fait qu'un site qui génère plus de règles d'association doit avoir plus de pouvoir de décision que les autres sites. Et de plus, ces règles d'association ont de forte chance de devenir des règles d'association globales.

3.2 Algorithme modifié pour la Synthétisation des règles de fréquence élevée à partir de sources de données de tailles différentes [Ramkumar.T et al. 2008]

Les auteurs dans [Ramkumar.T et al. 2008] ont proposé une méthode de synthétisation des règles de fréquence élevée à partir de sources de données de tailles différentes. Pour cela, ils se sont basés sur la population de transactions dans les sites c.à.d. le nombre de transactions dans chaque site. En utilisant la population de transactions, les auteurs ont pu démontrer que les résultats obtenus par leur méthode correspondent bien avec ceux obtenus par la fouille monobase de données c.à.d. lorsque toutes les bases de données sont fusionnées en une seule grande base de données et sur laquelle est appliqué l'algorithme *APRIORI*.

3.2.1 Méthode

Soient S_1, S_2,...S_m les sources de données des différentes sites, et *n* le nombre des règles candidates différentes extraites à partir des *m* sources de données.

L'algorithme se résume en 5 étapes :

Etape 1 : Consiste à calculer des règles d'association en se basant sur la taille de la population de transactions :

- Le calcul du poids de chaque source de données se fait comme suit :

$W^{|}_{Si}$ = Le nombre de transactions du site i.

- Le calcul du poids normalisé de chaque source de données se fait comme suit :

$$W_{Si} = \frac{W'_{Si}}{\sum_{j=1}^{m} W'_{Sj}}$$

- Le calcul du poids de chaque règle R_i se fait comme suit :

$$W'_{Ri} = \sum_{j=1}^{m} \delta(Ri, Dj) \times W_{Sj} \text{ tel que :}$$

W_{Sj} est le poids normalisé du site j où se trouve la règle R_i.

$$\begin{cases} \delta(Ri, Sj) = 1 & \text{si la règle } R_i \text{ se trouve dans le site j.} \\ \delta(Ri, Sj) = 0 & \text{sinon} \end{cases}$$

- Le calcul du poids normalisé de la règle R_i se fait comme suit : $W_{Ri} = \frac{W'_{Ri}}{\sum_{j=1}^{n} W'_{Rj}}$

Etape 2 : Consiste à calculer le poids du site suivant le poids des règles d'association. La formule de calcul est la suivante :

$$W_{S_i} = \frac{W_{R_i}}{\sum_{j=1}^{n} W'_{R_j}}$$

Etape 3 : Le calcul du support global des règles synthétisées se fait de la même façon que celui de [Xindong.W et al. 2003].

$$Supp_G(R_i) = \sum_{j=1}^{m} W_{Sj} \times Supp_j(Ri)$$

Tel que $Supp_j(R_i)$ est le support local de la règle R_i dans le site j.

Etape 4 : Le calcul de la confiance globale se fait par la formule suivante :

$$Conf_G(R_i) = \frac{Supp_G(R_i)}{Supp_G(\text{antécédent})}$$

$$= \frac{Supp_G(antécédent \cup Conséquence)}{Supp_G(antécédent)}$$

Tel que le $Supp_G(R_i)$ est le support global de la règle R_i et $Supp_G(\text{antécédent})$ est le support global de l'antécédent de la règle R_i.

3.2.2 Discussion

Les auteurs de cet algorithme ont proposé un modèle de synthétisation des règles d'association locales en règles globales. Pour cela, ils ont adopté le même modèle que celui définie dans [Xindong.W et al. 2003] avec la modification sur la façon de calculer le poids de chaque site. L'algorithme défini dans [Xindong.W et al. 2003] part du principe que toutes les sources de données ont la même taille tandis que dans la réalité le nombre de transactions est différent d'un site à un autre. Pour cela, les auteurs dans [Ramkumar.T et al. 2008] ont modifié la procédure de calcul du poids de telle sorte à prendre en considération la différence du nombre de transactions.

3.3 L'effet du facteur de correction dans la synthétisation des règles globales [Ramkumar.T et al. 2009]

Les auteurs dans [Ramkumar.T et al. 2009] sont les premiers à aborder le problème de la perte de connaissances lors de l'opération de synthétisation des motifs locaux vers des motifs globaux. En effet, ils ont proposé un algorithme de synthétisation des règles d'association à partir de différentes sources de données. Ils se sont basés sur la population de transactions pour calculer le poids des sources de données. Leur contribution majeure réside dans la définition d'un facteur de correction *h* afin de corriger le support des motifs non localement fréquents. Avec une valeur *h* correcte les résultats de la synthétisation des motifs locaux en motifs globaux sont proches de celles de la fouille monobase de données et dépendent fortement de ce facteur *h*. Les auteurs de cet algorithme ont réalisé plusieurs expérimentations pour déterminer la valeur optimale de ce facteur *h*. Ils affirment que pour une valeur *h=0,5* les résultats de la synthétisation des motifs locaux sont proches de ceux de la fouille monobase de données. Dans ce qui suit, nous donnons plus de détail sur cette méthode.

3.3.1 Méthode

Soient m sites S_1, S_2,…, S_m avec $W_1^{'}$, $W_2^{'}$,…, $W_m^{'}$ leurs poids respectifs.

- Le poids normalisé du site Sj : $W_j = \frac{W_j^{'}}{\sum_{j=1}^{m} W_j^{'}}$

- Le support global synthétisé de la règle R_i : $Supp_G(R_i) = \sum_{j=1}^{m} W_j \times Supp_j(R_i)$
- La confiance globale synthétisée de la règle R_i : $Conf_G(R_i) = \frac{Supp_G(R_i)}{Supp\_ante_G(R_i)}$

Si la règle R_i n'est pas fréquente dans le site S_j alors deux scénarios peuvent être envisagés :

Scénario 1 : Le support local de la règle R_i dans le site S_j est égal à 0. Cela veut dire que la règle R_i n'est pas présente dans le site S_j.

Scénario 2 : La règle R_i est présente sur S_j mais avec un support au dessous du *minsup*.

Cependant dans le premier scénario la contribution de la règle n'a aucun impact sur le processus de synthétisation du moment que son support est nul. Par contre, dans le second scénario, le support de la règle R_i doit être pris en compte car il peut avoir un impact sur le processus de synthétisation. Dans les algorithmes classiques de synthétisation tel que [Xindong.W et al. 2003] [Ramkumar.T et al. 2008] la valeur du support de cette règle est considérée comme nul du moment qu'elle n'est pas localement fréquente. Donc, pour ajuster cette valeur un facteur de correction *h* est appliqué. Les auteurs supposent que la valeur du support de la règle non localement fréquente R_i dans un site est certainement entre 0 et *minsup*. Le facteur peut être appliqué de la façon suivante :

$$Supp_j(R_i) = h \times minsup \quad tel\ que:\ 0 \leq h \leq 1$$

Avec ce facteur de correction, les résultats de la synthétisation sont améliorés et convergent vers ceux de la fouille monobase de données. La question qui se pose est comment choisir ce facteur de correction *h* dans les sites où se trouvent des règles non localement fréquentes ?

Pour répondre à cette question, les auteurs de cet algorithme ont procédé à une série d'expérimentations sur deux bases de données synthétique «T10I4D100K»[1] «Mushroom»[1]. Ils ont fait varier la valeur du facteur de correction *h* entre 0 et 1 et ils ont validé le résultat par deux mesures : Erreur moyenne (*Mean-Error*), et erreur sur la racine moyenne (*RMS-Error : Root Mean Square Error*) sur les valeurs du support et de la confiance. Le facteur de correction choisi est celui dont les valeurs des deux mesures sont minimales. À partir des résultats obtenus, ils ont affirmé qu'avec la valeur de h=0.50 et h=0.40 les valeurs des deux mesures sont minimums pour le support et la confiance respectivement. Selon ces auteurs,

[1] http://fimi.cs.helsinki.fi/data

l'expert du domaine pourra choisir un facteur de correction approprié en se basant sur la distribution des données. Dans l'absence de connaissance détaillée sur la distribution des données, choisir un facteur de correction égale à 0.50 est une option possible pour l'expert du domaine.

3.3.2 Discussion

Les méthodes de synthétisations de [Xindong.W et al. 2003] [Ramkumar.T et al. 2008] calculent le support global des motifs sur la base des poids du site et de leurs supports dans ces sites. Cependant, ces méthodes souffrent d'une perte d'informations où certains motifs globaux peuvent être perdus. En effet, si un motif n'est pas localement fréquent il est exclu de la formule qui calcule son support global même si son support est proche de *minsup*, ce qui va cacher certains motifs globaux. Pour cela, pour les capturer, [Ramkumar.T et al. 2009] ont choisi un facteur correcteur *h* pour ajuster le support des motifs non localement fréquents. La qualité des résultats dépend fortement du choix de ce facteur *h*. Le choix de la valeur de *h* est fixe pour l'ensemble des sites ce qui peut réduire la qualité des résultats car la distribution des données peut être différente d'un site à un autre. Et le choix de ce facteur par l'expert du domaine peut donner des résultats de mauvaise qualité. La difficulté de ces deux inconvénients se distingue lorsqu'il faut comparer les résultats de synthétisation à ceux de la fouille monobase de données. En d'autres termes, on procède à la fouille monobase de données sur chaque site indépendamment puis on applique le facteur *h* en se basant sur la distribution des données dans les sites pour enfin comparer les résultats obtenus et déterminer la valeur de *h*. Ce qui est coûteux en termes de temps d'exécution.

3.4 Synthétisation multi-niveaux des règles fréquentes à partir de différentes sources de données [Ramkumar.T et al. 2010]

Récemment, Ramkumar et al dans [Ramkumar.T et al. 2010] ont proposé une approche par niveaux. Ils sont les premiers à introduire la notion de niveaux dans le processus de génération des règles d'association multi-bases de données. Ils ont proposé un algorithme de synthétisation des règles d'association fréquentes à partir de différentes sources de données. Cet algorithme génère non seulement des règles d'association fréquentes globales mais aussi des règles d'association fréquentes par sous groupes de sites qui peuvent être regroupées par région, zones ou par localisation. Les auteurs mettent l'accent sur la nécessitée d'extraire ce type de

règles d'association car il représente l'individualité des sites. Cet algorithme peut aider les décideurs des organisations de plus de deux niveaux à prendre des décisions régionales, par zone ou par localisation. A cette fin, les auteurs ont utilisé deux mesures de sélection de règles nommées : *le taux de vote effectif noté* $\gamma_{effectif}$ *et le taux de vote nominal* noté $\gamma_{nominal}$ pour générer trois groupes de règles d'association fréquentes qui sont : *Globales, sous globales et locales.*

Les règles globales sont destinées à la direction centrale d'une organisation pour la prise des décisions globales sur l'ensemble de ses unités tandis que *les règles sous globales* peuvent aider la direction des branches régionales et enfin *les règles locales* seront utiles pour la prise des décisions au niveau local.

3.4.1 Méthode

Le problème de la synthétisation multi-niveaux se formule comme suit :

Soient S_1, S_2, …, S_m m sites où chaque site dispose d'un ensemble de règles locales de la forme : A → B {*supp, conf*}.

Suivant ces motifs locaux l'algorithme extrait des règles d'association synthétisées à partir des différentes sources de données à des niveaux multiples en utilisant les deux mesures : taux de vote nominal et effectif. Cependant le processus de synthétisation des règles d'association multi-niveaux est composé des étapes suivantes :

3.4.1.1 Calculer les poids de chaque site sur la base des transactions

Le poids de la source de données est calculé à partir de la population des transactions. Soient $W_1^|$,$W_2^|$, …,$W_m^|$ les poids correspondant à la population des transactions relative à chaque site. Le poids normalisé d'un site j, noté W_j, est la proportion entre sa population de transactions et la somme de toutes les populations des transactions des sites participants.

$$W_j = \frac{W_j^|}{\sum_{j=1}^{m} W_j^|}$$

3.4.1.2 Calculer les valeurs des deux mesures : $\gamma_{nominal}$ et $\gamma_{effectif}$

Le taux de vote effectif représente le pourcentage des votes reçus à partir des différentes sources de données pour une règle donnée sur la base de la population des transactions correspondant aux sources de données.

Le taux de vote nominal représente le pourcentage des votes reçus à partir des différentes sources de données pour une règle donnée sur la base d'une égalité de vote par les sites. Pour une règle R_i, les deux mesures $\gamma_{effective}$, $\gamma_{nominal}$ sont calculées comme suit :

$$\gamma_{\text{effective}}(R_i) = \sum_j^m \delta(i,j) \times W_j$$

$$\gamma_{\text{nominal}}(R_i) = \frac{1}{m} \sum_j^m \delta(i,j)$$

Tel que $\delta(i,j) = 1$ si R_i est présente dans le site j sinon $\delta(i,j) = 0$.

3.4.1.3 Classifier les règles en règles candidates globales, sous globales et règles locales

Soit $S^{|}$ l'ensemble de toutes les règles de la forme A→ B {supp, conf} de tous les sites

$$S^{|} = S_1^{|} \cup S_2^{|} \cup \ldots \cup S_m^{|}$$

Où $S_i^{|}$ représente les règles d'association locales du site S_i.
Étant donné que le nombre de règles d'association découvertes est grand, les auteurs procèdent par classification de ces règles en 3 groupes : *les règles globales candidates*, *les règles sous globales candidates*, et *les règles locales* :

Les règles globales candidates : Ces règles ont un $\gamma_{effectif} >= min\gamma_{effectif}$. La valeur du $min\gamma_{effectif}$ est choisie par l'utilisateur.

Les règles sous globales candidates : Ces règles ont un $\gamma_{effective} < min\gamma_{effectif}$ et un $\gamma_{nominal} >= min\gamma_{nominal}$. La valeur du $min\gamma_{nominal}$ est choisie par l'utilisateur. De plus les règles globales candidates qui ne satisfont pas *minsupp* et *minconf* sont ajoutées à la liste des règles candidates sous globales.

Les règles locales : Les règles restantes sont les règles locales qui existent uniquement dans un seul site ou qui existent dans quelques sites insuffisants pour former un groupe.

3.4.1.4 Calculer le support et la confiance des règles

a) Calcul du support local et de la confiance d'une règle dans un site :

Soit R_i une règle de la forme antécédent → conséquent.

Le support et la confiance locaux d'une règle R_i dans le site j sont notés : $Supp_j(R_i)$ et $Conf_j(R_i)$. Le support de l'antécédent de R_i dans le site j est noté : $Supp\_ante_j(R_i)$.

Les règles intéressantes doivent satisfaire :

$$\text{Support (antécédent} \rightarrow \text{conséquent)} \geq \textit{minsupp}$$

$$\text{Confiance (antécédent} \rightarrow \text{conséquent)} = \frac{\text{Support (antécédent} \rightarrow \text{conséquent)}}{\text{Support(antécédent)}} \geq \textit{minconf}$$

Où *minsupp* et *minconf* sont les seuils définis par l'utilisateur.

b) Calcul du support et de la confiance des règles globales synthétisées

Le support global d'une règle R_i est calculé comme suit :

$$Supp_G(R_i) = \sum_j^m W_j \times Supp_j(R_i)$$

Tel que $Supp_j(R_i)$ est le support local de R_i dans le site j.

La confiance globale synthétisée se calcule de la façon suivante :

$$Conf_G(R_i) = \frac{Supp_G(R_i)}{Supp\_ante_G(R_i)}$$

c) Calcul du support et confiance des règles sous globales synthétisées

Ces expressions sont similaires à celles des expressions pour les valeurs globales mais la sommation est restreinte uniquement pour les sites sous groupés.

$$W_{SG}(R_i) = \sum_{\forall j \; in \; class-list} W_j$$

Où W_j est le poids du site j appartenant à la liste des sites d'une classe (*class-list*) supportant la règle R_i.

$$Supp_{SG}(R_i) = \frac{1}{W_{SG}} \sum_{\forall j \; in \; class-list} W_j \times Supp_j(R_i)$$

$$Conf_{SG}(R_i) = \frac{Supp_{SG}(R_i)}{Supp\_ante_{SG}(R_i)}$$

Les règles sous globales sont classées en 3 classes ou groupes (en anglais : clusters) *(SGR-I, SGR–II et SGR–III)* et le calcul du support et la confiance des règles sous globales s'effectue sur ces groupes. Les trois groupes sont construits de la façon suivante:

Classification selon l'étiquète standard (SGR - I) : Dans cette classification, les caractéristiques des motifs sont capturées en utilisant la classification standard. En effet chaque groupe de sites prend l'étiquette de la classification standard. Les motifs qui correspondent exactement avec l'étiquète de la sous classe des sites où cette règle apparait, vont constituer le groupe de l'ensemble des règles *SGR-I***.**

Classification étiquetée par l'expert du domaine (SGR-II): Les motifs ou les règles qui n'obéissent pas aux caractéristiques générales des groupes standards existants sont synthétisés et mis dans le groupe *SGR–II***.** Autrement dit, la règle R_i apparait dans quelques sites d'une ou plusieurs sous classes différentes. Par conséquent, l'expert du domaine doit attribuer une étiquette appropriée à cette règle.

Les règles sous globales selon des classificateurs standard avec un support réduit (SGR - III) : Dans cette classe, les règles sous globales qui ne correspondent pas exactement aux étiquettes préétablies sont synthétisées sur la base du taux de vote effectif des classificateurs. Les règles dont le taux de vote effectif satisfait $min\gamma_{effectif}$ formeront l'ensemble de règles sous-globale candidates. Après synthétisation de ces règles candidates celles qui ont les valeurs du support et confiance synthétisées satisfaisant respectivement *minsupp* et *minconf* seront mis dans la classe *SGR-III*.

3.4.2 Discussion

Cet algorithme a apporté deux contributions majeures par rapport aux trois premiers algorithmes étudiés : – L'introduction des connaissances du domaine dans l'affectation des règles d'association vers un groupe – L'extraction de nouvelles connaissances (sous-globales) dans une architecture à trois niveaux.

- Nous avons vu dans l'algorithme que l'expert de domaine peut intervenir pour décider d'affecter les règles d'association extraites à un des groupes prédéfinis ou de choisir un autre groupe défini par l'utilisateur.
- Les algorithmes précédents [Xindong.W et al. 2003] [Ramkumar.T et al. 2008] [Ramkumar.T et al. 2009] reposent sur une architecture à deux

niveaux. Le premier niveau qui est le niveau bas représente les motifs locaux extraits à partir de chaque site. Tandis que le deuxième niveau qui est le niveau supérieur représente les motifs globaux calculés à partir des motifs locaux. En réalité, la structure d'une organisation est beaucoup plus complexe et peut contenir plus de deux niveaux. C'est la raison qui a poussé les auteurs de cet algorithme à réaliser cette nouvelle architecture à trois niveaux. En plus des deux niveaux cités, ils ont ajouté un niveau intermédiaire représenté par des motifs sous-globaux. Cette architecture permet l'extraction des connaissances à trois niveaux et chaque utilisateur d'un niveau peut avoir une vue locale de ces connaissances. Cet algorithme peut être appliqué à une organisation composée de trois niveaux : niveau opérationnel (niveau bas), niveau intermédiaire (branches) et niveau stratégique (niveau le plus haut).

3.5 La fouille multi-bases de données en utilisant le modèle de pipeline (Pipelined Feedback Model PFM) [Animesh.A et al. 2010b]

Les auteurs dans [Animesh.A et al. 2010b] ont proposé un modèle de pipeline pour extraire les motifs globaux à partir des entrepôts de données (en anglais : Data warehouse) triés par ordre décroissant selon le nombre de transactions. On note DW_i l'entrepôt de données correspondant à la i^{ème} branche, avec $i=1,2,\ldots,n$. Les motifs locaux de la i^{ème} branche, notés LPB_i, sont extraits à partir de DW_i. La fouille de données dans chaque branche est réalisée avec une simple technique de fouille de données notée *SDMT : Simple Data Mining Technique*.

3.5.1 Méthode

La figure 2.2 illustre ce modèle de pipeline.

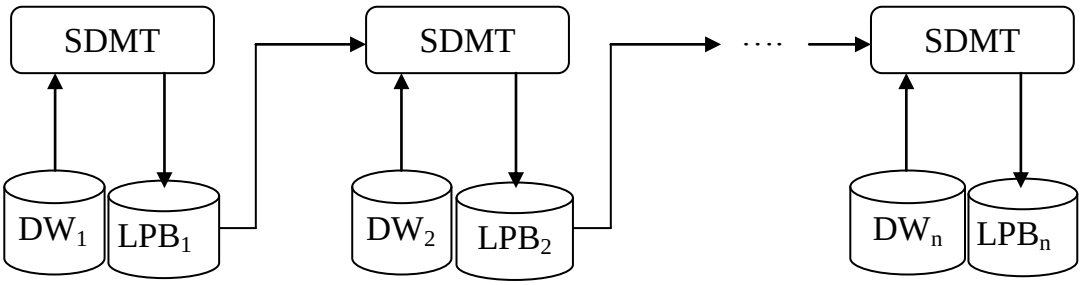


FIG. 2.2 – MODELE PIPELINE DE LA FOUILLE MULTI-BASES DE DONNEES

L'algorithme procède graduellement par application de *SDMT* dans le premier entrepôt de données DW_1 où un ensemble de motifs locaux LPB_1 sont générés. Ensuite, les motifs extraits à partir de DW_2 sont tous les motifs LPB_1 avec les nouveaux motifs extraits par *SDMT* dans DW_2. Les motifs LPB_2 sont donc les motifs globaux des deux entrepôts DW_1 et DW_2. D'après les auteurs de cette approche après application de l'algorithme SDMT dans l'entrepôt DW_n les motifs globaux extraits sont globaux dans toutes les branches.

Soit *DW* l'ensemble des entrepôts de données de toutes les branches. Soit l'itemset *X* extrait à partir des *m* premiers entrepôts de données avec 1≤m≤n alors le support synthétisé de l'itemset *X* dans l'ensemble *DW* est :

$$supp_s(X, DW) = \frac{1}{\sum_{i=1}^{n} |DW_i|} \times \sum_{i=1}^{m} [supp(X, DW_i) \times |DW_i|]$$

Avec $supp(X, DW_i)$ est le support de X dans l'entrepôt DW_i.

Et $supp_s(X, DW)$ est le support synthétisé de X dans l'entrepôt *DW*.

Pour évaluer la qualité de leur résultat, les auteurs de *PFM* ont quantifié l'erreur produite par leur méthode. Premièrement, ils ont procédé par l'extraction des itemsets fréquents dans multiples bases de données avec l'algorithme *PFM*. Ensuite ils ont utilisé un algorithme classique de synthétisation des motifs globaux défini dans [Xindong.W et al. 2003] pour synthétiser les itemsets globaux (SPS : Simple Process Synthesizing).

Ensuite pour valider l'algorithme *PFM* les auteurs ont utilisé 2 types de mesures : erreur moyenne (en anglais : *Average error AE*) et maximum d'erreur (en anglais : *Maximum Error ME*). Avant de présenter les deux mesures, nous définissons l'écart entre les deux valeurs du support d'un itemset dans *SPS* et par l'algorithme *PFM*.

$$E(X|PFM, SPS) = \left| supp(X, D) - \frac{1}{\sum_{j=1}^{n} |D_j|} \times \sum_{j=1}^{n} \left[supp\left(X, D_j\right) \times \left|D_j\right| \right] \right|,$$

$$pour\ X \in LPB_i - LPB_{i-1}\ et\ \ i = 2,3, \dots, n$$

$$et\ E(X|PFM, SPS) = 0, \qquad pour\ X \in LPB_1$$

Quand un itemset est fréquent dans DW_1 alors il est pris en considération dans tous les entrepôts qui suivent DW_1 par l'algorithme PFM, ce qui explique que :

$$(X|PFM, SPS) = 0, pour\ X \in LPB_1.$$

Par contre, si un itemset n'est pas fréquent dans DW_1 alors il n'est pas pris en compte dans tous les entrepôts de données. Son support sera diminué de sa valeur réelle.

Erreur moyenne (AE) :

$$AE(D, \alpha) = \frac{1}{\left|LPB_1 + \sum_{i=2}^{n}(LPB_i - LPB_{i-1})\right|}$$

$$\sum_{X \in \left[LPB_1 \cup \{\cup_{i=2}^{n}(LPB_i - LPB_{i-1})\}\right]} E(X|PFM, SPS)$$

Erreur maximum (ME) :

$$ME(D, \alpha) = maximum\{E(X|PFM, SPS)/\ X\ \in LPB_1 \cup \{\cup_{i=2}^{n}\ (LPB_i - LPB_{i-1})\}\}$$

Ensuite pour évaluer l'algorithme *PFM* un ensemble d'expérimentations a été réalisé sur trois bases de données synthétiques et deux bases de données réelles. Les bases de données synthétiques sont T10I4D100K, random500, random1000. Les bases de données réelles sont : Retail, BMS-Web-Wiew-1. Les résultats obtenus par l'algorithme *PFM* convergent vers *SPS,* car les valeurs de *AE* et *ME* sont soit nul soit autour de 0.

3.5.2 Discussion

La principale contribution de cet algorithme est l'amélioration du processus de synthétisation des motifs locaux vers les motifs globaux. En effet, les résultats expérimentaux réalisés montrent que cet algorithme permet de mieux synthétiser les motifs locaux par rapport aux travaux réalisés par [Xindong.W et al. 2003]. Dans l'algorithme *PFM* et suivant le nombre de transactions, les entrepôts de données sont triés par ordre décroissant ensuite l'algorithme *SDMT* est appliqué sur cet ensemble d'entrepôts de données et les motifs locaux des entrepôts précédents. Et ceci pour générer un maximum de motifs locaux pour les synthétiser de façon graduelle. Mais malheureusement un entrepôt qui a le nombre de

transactions le plus élevé ne génère pas nécessairement un nombre important de motifs locaux. On peut avoir un entrepôt dont le nombre de transactions est minimum mais les motifs générés sont importants par rapport aux autres entrepôts de données. Les résultats expérimentaux montrent l'efficacité de l'algorithme *PFM* par rapport à un algorithme classique de synthétisation *SPS*. Les motifs globaux générés par les deux algorithmes restent toujours approximatifs. Il est plus intéressant de comparer entre les résultats de *PFM*, *SPS* et la fouille monobase de données pour montrer l'apport de l'algorithme *PFM* par rapport à *SPS*.

3.6 L'extraction de motifs à partir d'items sélectionnés dans les bases de données multiples [Animesh.A et al. 2010a]

Le principe de cette méthode repose sur l'utilisation des items sélectionnés dans le processus de la fouille multi-bases de données. En effet, l'analyse des items sélectionnés dans la fouille multi-bases de données est intéressante car elle permet de ne sélectionner que des règles d'association intéressantes. C'est le cas d'une organisation où les managers ont besoin d'informations sur un ensemble d'items pour prendre des décisions. Dans cette partie, on va décrire le modèle de fouille de motifs globaux d'items sélectionnés à partir de multi-bases de données proposé par [Animesh.A et al. 2010a].

3.6.1 Méthode

La figure 2.3, présente l'approche d'extraction de motifs globaux basée sur un ensemble d'items sélectionnés dans de multiples bases de données [Animesh.A et al. 2010a]. L'algorithme se résume par les étapes suivantes :

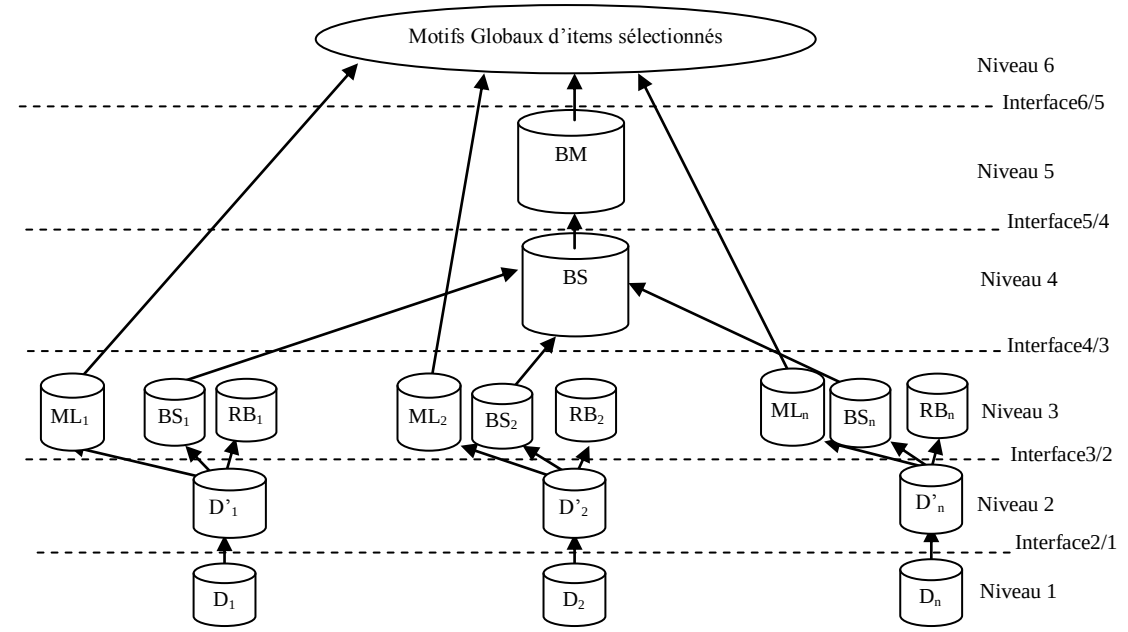
1- Au niveau local, chaque branche extrait les motifs à partir des bases de données locales à l'aide d'un algorithme de la fouille monobase de données (L'algorithme *APRIORI* par exemple).

2- Au niveau central, les bases de données sont consolidées suivant les items sélectionnés dans une seule base de données *BS* (Base de données sélectionnée).

3- Ensuite un algorithme traditionnel d'extraction de règles d'association est appliqué pour extraire les motifs fréquents à partir de *BS*. Le résultat est stocké dans la base des motifs *BM*.

4- Finalement, les motifs globaux d'items sélectionnés peuvent êtres extraits effectivement à partir de motifs locaux (ML_i) et *BM*.

L'algorithme proposé repose sur 5 interfaces et niveaux comme illustré dans la figure 2.3. Chaque interface contient un ensemble d'opérations pour produire d'autres bases de données ou motifs à partir des bases de données D_i. La fonctionnalité de chaque interface est décrite comme suit :



D_i : Base de données i brute.
D'_i : Base de données i nettoyée
ML_i : Motifs Locaux de D'i
BS_i : Base de données i sélectionnée
RB_i : Le reste de la base de données i.
BS : Base de données sélectionnée consolidée.
BM : Base de motifs de BS.

FIG. 2.3 – PROCESSUS DE FOUILLE DE MOTIFS GLOBAUX D'ITEMS SELECTIONNES

-Interface 2/1 est utilisé pour nettoyer/transformer/réduire/intégrer les données des bases de données brut D_i. Les bases de données résultantes D'_i situé au niveau 2 sont prêtes à être exploitées par les techniques de la fouille de données.

-Avant d'appliquer un algorithme d'extraction de règles d'association, un partitionnement de la base de données D'_i en deux bases de données : une base de données sélectionnée qui contient les transactions contenant des items sélectionnés au départ par l'utilisateur notée par BS_i et une autre base de données qui contient le reste des transactions notée par RB_i. Cette opération est effectuée dans l'interface 3/2. En plus de cette opération, un algorithme d'extraction de motifs fréquents

locaux est appliqué sur la base de données D'_i pour extraire les motifs localement fréquents sélectionnés et non sélectionnés.

- Les bases de données contenant des items sélectionnés sont consolidées en une seule base de données BS. Cette dernière est construite avec l'opérateur union de toutes les bases de données BS_i dans l'interface 4/3.

- A partir de la base de données consolidé BS un algorithme d'extraction de règles d'association est appliqué dans l'interface 5/4 pour extraire des motifs fréquents. Ces derniers sont stockés dans la base de motifs *BM*.

- Un algorithme de génération de motifs globaux est appliqué dans l'interface 6/5 à partir de la base des motifs *BM* et des motifs locaux ML_i de toutes les bases de données.

3.6.2 Discussion

Cette méthode est sans doute la seule dans la littérature de la fouille multi-bases de données (*FMBD)* à introduire les connaissances de l'utilisateur dans le processus de la fouille multi-bases de données. Ces attentes sont représentées par les items initialement sélectionnés. Cet algorithme apporte deux contributions majeures au processus de la fouille multi-bases de données : - c'est un processus sans perte de connaissances qui fournit le support exact des motifs fréquents –c'est un processus guidé par l'utilisateur.

Sans perte de connaissances : Cet algorithme résout le problème posé par les algorithmes vus dans ce chapitre [Xindong.W et al. 2003] [Ramkumar.T et al. 2008] [Ramkumar.T et al. 2009] [Animesh.A et al. 2010b] qui est l'estimation du support des motifs non localement fréquents. Cette méthode fournie le support exact du motif global sans avoir recourt à l'estimer. Ceci s'explique par l'utilisation de la base de données sélectionnée *BS* qui récupère les motifs qui ne sont pas dans ML_i pour toutes les bases de données pour comptabiliser le support global.

Processus guidé par l'utilisateur : L'introduction des items sélectionnés dans le processus de la fouille multi-bases de données fournit en quelque sorte l'empreinte de l'utilisateur dans ce processus. Car beaucoup de décisions sont basées sur cet ensemble spécifique d'items. Cependant, le besoin d'utiliser ces items sélectionnés se fait ressentir pour plusieurs considérations :

- Lorsqu'une organisation veut promouvoir certaines catégories de produits (items) de façon indirecte qui consiste à promouvoir les items qui lui sont

associés. L'implication d'items associés entre l'item sélectionné *P* et l'item *Q* est : Si *Q* est acheté par un client alors *P* est aussi acheté par le même client en même temps. Dans ce cas, on peut promouvoir le produit *P*.

- Chacun des items sélectionnés peut être considéré comme standard. Ainsi, ces items seront le bon marché de l'entreprise. Ils aident à promouvoir d'autres items.

3.7. Identification des motifs intéressants dans des multi-bases de données [C.Zhang et al. 2005]

Les auteurs dans [C.Zhang et al. 2005] s'intéressent à d'autres types de connaissances qui sont : connaissances majoritaires et exceptionnelles. Ces motifs identifient la distribution des motifs locaux et reflètent tout ce qui est commun et exception. Les motifs majoritaires et exceptionnels sont utiles dans des applications globales d'une organisation à plusieurs branches. Cette partie présente les techniques pour identifier ce type de motifs à partir des motifs locaux selon [C.Zhang et al. 2005]. En premier lieu nous allons définir l'algorithme de génération des motifs majoritaires ensuite celui des motifs exceptionnels.

3.7.1 Méthode

Soient D_1,D_2,…,D_m des bases de données respectivement de m branches B_1,B_2,…,B_m d'une organisation. Et soit ML_i l'ensemble de motifs locaux de la branche i avec i=1,…,m, tel que :

$$ML = \{r_j | \, r_j \in ML_1 \cup ML_2 \cup \ldots \cup ML_m, 1 \leq j \leq n\}$$

Avec $n = |ML_1 \cup ML_2 \cup \ldots \cup ML_m|$

Le tableau 2.1 montre la fréquence de votes des motifs locaux des branches d'une organisation. Ce vote représente le nombre d'apparition des motifs locaux dans les différentes branches.

	$\mathbf{r_1}$	$\mathbf{r_2}$	**…**	$\mathbf{r_n}$
$\mathbf{B_1}$	$a_{1,1}$	$a_{1,2}$	…	$a_{1,n}$
$\mathbf{B_2}$	$a_{2,1}$	$a_{2,2}$	…	$a_{2,n}$
…	…	…	…	…
$\mathbf{B_m}$	$a_{m,1}$	$a_{m,2}$	…	$a_{m,n}$
Nombre de vote	$Vote_1$	$Vote_2$	…	$Vote_n$

TAB. 2.1 – La fréquence des motifs par le vote des branches d'une organisation

Dans le tableau 2.1, $a_{i,j}$=1 signifie que la branche B_i dispose du motif fréquent r_i. et $a_{i,j}$=0 signifie que le motif r_i n'est pas fréquent dans la branche B_i.

A partir du tableau 2.1 le taux de vote moyen peut être défini comme suit :

$$votemoyen = \frac{vote(r_1) + vote(r_2) + \cdots + vote(r_n)}{n}$$

Avec $vote(r_i) = {}^{vote_1}/_{m}$ qui est le taux de vote de r_i.

Le taux de vote d'un motif est élevé s'il est supérieur au *votemoyen*. On peut classifier les motifs en quatre classes comme décrit dans la figure 2.4.

Dans la figure 2.4 le taux de vote=*votemoyen* est la ligne de référence pour mesurer l'intérêt d'un motif r_i. En effet, si le taux de vote est entre [x_1,1], le motif est référé comme majoritaire et l'espace [x_1,1] est l'espace des motifs majoritaires avec la valeur de x_1 très proche de 1. Si le motif est dans la tranche de [x_2,x_1), il considéré comme motif suggéré avec la valeur de x_2 proche de x_1. Le champs [x_2,x_1) démontre l'espace des motifs suggérés. L'espace (x_3,x_2) décrit les motifs aléatoires qui sont des motifs sans intérêt pour une organisation. Finalement, si le taux de vote est entre [0,x_3], le motif est considéré comme motif exceptionnel et l'espace [0,x_3] ne contient que des motifs exceptionnels avec la valeur de x_3 proche de 0.

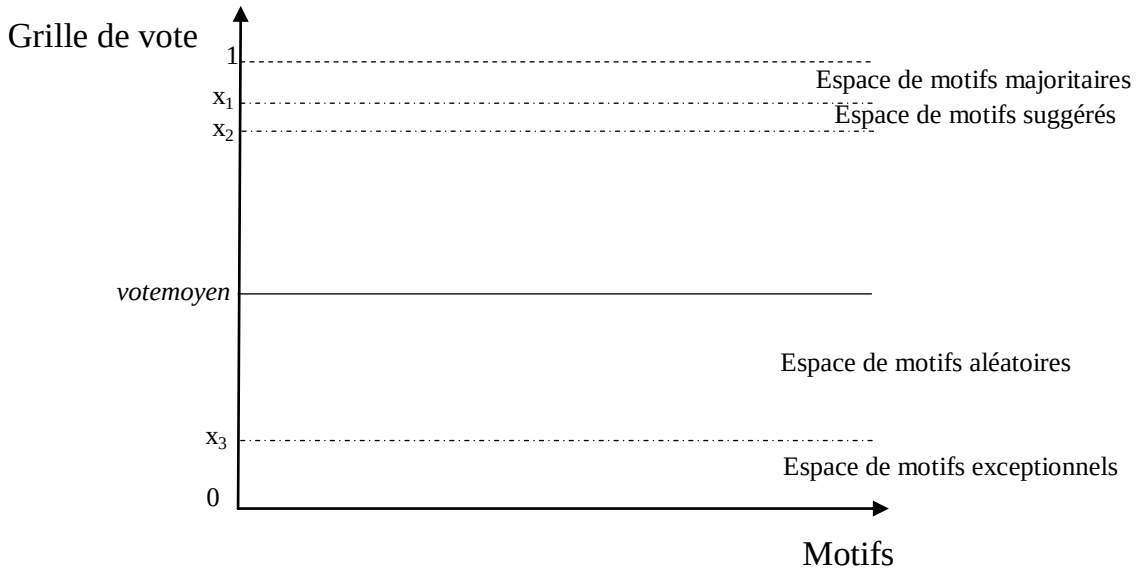


FIG. 2.4 – LES CLASSES DE MOTIFS

Cas motifs majoritaires : [C.Zhang et al. 2005] ont défini une mesure d'intérêt notée $LPI(r_i)$ pour un motif local r_i. Cette mesure est calculée suivant la déviation du taux de vote $vote(r_i)$ pour un motif r_i à partir du vote moyen (*votemoyen*).

$$LPI(r_i) = \frac{vote(r_i) - votemoyen}{1 - votemoyen}$$

Avec : $1 - votemoyen \neq 0$.
Cette mesure d'intérêt $LPI(r_i)$ est positivement reliée avec le taux du vote réel du motif r_i. En effet, $LPI(r_i)$ est maximum avec la valeur du taux de vote réel ($vote(r_i)$)=1. On peut dire qu'un motif est majoritaire si $LPI(r_i)$ est supérieur ou égal à un seuil *minVR* déterminé par un utilisateur. Ce seuil représente le degré minimum d'intérêt d'un utilisateur ou d'expert.

On peur formuler le problème de la génération des motifs majoritaires comme suit:

r_i est un motif majoritaire si $LPI(r_i)$ est supérieur ou égale à *minVR*.

Cas motifs exceptionnels : [C.Zhang et al. 2005] ont défini deux mesures : – Taux déviation de vote noté par $EPI(r_i)$ – L'intérêt du motif r_i noté par $RI_i(r_j)$.

La première mesure $EPI(r_i)$ est calculée suivant la déviation du taux de vote $vote(r_i)$ pour un motif r_i à partir du vote moyen (*votemoyen*).

$$EPI(r_i) = \frac{vote(r_i) - votemoyen}{-votemoyen}$$

Avec $votemoyen \neq 0$.
A partir de cette mesure d'intérêt, $EPI(r_i)$ est négativement relié avec le taux du vote réel du motif r_i. En effet, $EPI(r_i)$ est maximum avec la valeur du taux de vote réel ($vote(r_i)$)=0. On peut dire qu'un motif est exceptionnel si $EPI(r_i)$ est supérieur ou égal à un seuil *minEP* déterminé par un utilisateur. Ce seuil représente le degré minimum d'intérêt d'un utilisateur ou d'expert.

La deuxième mesure $RI_i(r_j)$ introduit la dimension du support du motif r_j. Effectivement, cette mesure valorise les motifs dont le support est élevé. Un motif avec un support plus élevé est intéressant par rapport à d'autres qui ont un support moins élevé.

$$RI_i(r_j) = \frac{supp_{i,j} - minsupp_i}{minsupp_i}$$

où $supp_{i,j}$ est le support de r_j dans la branche B_i.

$RI_i(r_j)$ est positivement relié avec le support de r_j dans la branche B_i. $RI_i(r_j)$ est maximal si le support est égal à 1. On peut dire qu'un motif est intéressant si $RI_i(r_i)$ est supérieur ou égal à un seuil *minEPsup* déterminé par un utilisateur. Ce seuil représente le degré minimum d'intérêt du support d'un utilisateur ou d'expert.

On peut formuler le problème de la génération des motifs exceptionnels comme suit :

r_j est un motif exceptionnel si :

- $EPI(r_j)$ est supérieur ou égal à *minEP*.
- Et $RI_i(r_j)$ est supérieur ou égal à *minEPsup*.

4 Etude comparative

Dans cette partie, nous résumons les algorithmes présentés précédemment sur la base des critères que nous avons énoncés dans la section 2. Le tableau 2.2 montre un récapitulatif des caractéristiques de ces méthodes. Les colonnes représentent les différents critères et les lignes contiennent les références des approches étudiées. Une croix dans une cellule indique que la méthode en ligne possède la caractéristique en colonne. La légende du tableau donne la signification des abréviations et acronymes qui sont utilisés dans le tableau 2.2.

Type de motifs transférés : La majorité des méthodes transfert les motifs fréquents sous formes d'itemsets fréquents ensuite les synthétiser pour générer les itemsets globaux et enfin les règles d'association globales au niveau central. Ceci se justifie par la récolte de maximum d'informations lors de la synthétisation des motifs locaux en motifs globaux. En effet, la procédure de synthétisation repose sur les supports globaux des itemsets constituant la règle d'association et non pas sur les confiances des règles d'association.

La richesse de régénération : Aucune méthode n'est adaptée pour générer toutes les types de motifs, cela se traduit par l'objectif à atteindre. Dans la section précédente nous avons vu une variété de méthodes chacune génère un type de motifs. Sans doute les méthodes de synthétisation des motifs locaux en globaux occupent une place très importante dans la littérature. Ceci s'explique par les problèmes liés à la perte de connaissances et du choix du poids du site approprié lors de l'opération de synthétisation.

Le poids de chaque site : A l'exception de la méthode proposée dans [Xindong.W et al. 2003], tous les autres algorithmes utilisent la population des transactions pour calculer le poids du site. En effet, l'importance d'un site est valorisée par le nombre de transactions. Un site qui dispose d'un nombre important de transactions

contribue plus dans la génération des motifs globaux. Par conséquent les motifs globaux générés seront proches de ceux de la fouille monobase de données. Ce qui n'est pas le cas pour [Xindong.W et al. 2003] où un site est considéré important s'il génère plus de règles d'association de fréquences élevées. Selon l'argument de ces derniers un site dynamique contribue plus à la prise de décision sur l'ensemble des sites de la société. Prenons l'exemple d'un supermarché qui a un chiffre d'affaire d'un million de dinars, ce site aura un poids important pour la prise de décision sur la stratégie globale de la société par rapport à un autre dont le chiffre d'affaire est de quelques dinars. Le choix de ce poids dépend de l'objectif à atteindre, si on veut avoir, par exemple, des motifs globaux proches à celles de la fouille monobase de données, il est préférable d'utilisé la population de transactions comme poids du site.

Le niveau de synthèse : Les méthodes classiques de synthétisation des motifs globaux utilisent généralement deux niveaux. -Le niveau opérationnel et le niveau central. Le premier niveau est représenté par l'ensemble des bases de données alimentées quotidiennement qui vont être transformées en motifs fréquents. Le niveau central contient des motifs globaux calculés à partir de ces motifs fréquents. Le seul algorithme qui utilise la synthétisation à plusieurs niveaux est l'algorithme de [Ramkumar.T et al. 2010]. Effectivement il génère des motifs globaux à trois niveaux : le niveau local, sous global et global. Par conséquent ceci représente une génération à trois niveaux des motifs.

La perte d'information : La majorité des algorithmes de synthétisation ne considère que les motifs localement fréquents pour les synthétiser au niveau central. Ce qui va engendrer une perte d'information du moment que les motifs non localement fréquents peuvent contribuer au processus de la synthétisation de ces motifs. Pour cela, l'algorithme [Ramkumar.T et al. 2009] ajoute un facteur de correction aux motifs non localement fréquents pour les faire contribuer au processus de synthétisation. D'autres algorithmes comme celui de [Animesh.A et al. 2010b] utilisent un ordre décroissant sur la taille des bases données pour calculer le support globale des motifs de façon graduelle. Et finalement un seul travail [Animesh.A et al. 2010a] détecte les motifs fréquents sans perte d'informations basés sur les motifs sélectionnés.

La régénération : La régénération des motifs non localement fréquents est nécessaire pour générer les motifs globaux. Peu de travaux procèdent à la régénération de ces motifs. Certains travaux [Ramkumar.T et al. 2009] procèdent par une régénération approximative et d'autres [Animesh.A et al. 2010a] par une régénération exacte.

Processus guidé par l'utilisateur : A l'exception de la méthode proposée dans [Animesh.A et al. 2010a], aucune méthode n'est adaptée pour générer les motifs intéressants pour l'utilisateur. En effet cet algorithme permet de fournir un moyen pour choisir les motifs qui les intéressent.

Le contrôle de la taille des motifs : A l'exception de la méthode proposée dans [Animesh.A et al. 2010a], aucune méthode n'est adaptée pour contrôler la taille des motifs générés. Ce qui peut générer un nombre important de motifs dont l'analyse et l'exploitation par un utilisateur reste difficile.

5 Synthèse

Rappelons notre problématique initiale qui est l'extraction des règles d'association intéressantes à multi-niveaux où chaque utilisateur à différents niveaux consulte les règles d'association qui l'intéressent. Dans l'étude comparative des différents algorithmes on s'aperçoit qu'un seul algorithme [Animesh.A et al. 2010a] permet de générer les règles d'association suivant les besoins et attentes de l'utilisateur. Et un seul [Ramkumar.T et al. 2010] permet de générer les motifs fréquents multi-niveaux.

Les auteurs de l'algorithme [Animesh.A et al. 2010a] proposent de sélectionner d'abord l'ensemble des itemsets qui intéressent l'utilisateur, ensuite de les synthétiser en motifs globaux selon le processus de synthétisation. Comme résultat chaque utilisateur pourra consulter l'ensemble réduit de règles d'association. Mais cet algorithme présente certaines limites :

La pauvreté de représentation des attentes des utilisateurs : Cet algorithme ne prend en compte que des règles d'association non implicatives de la forme (A,B,C…), c.à.d. que les attentes des utilisateurs sous forme de règles d'association de type A→B ne sont pas prises en considération.

C'est un processus non interactif et non itératif : En effet le choix des attentes des utilisateurs est limité aux items sélectionnés. L'utilisateur ne pourra pas manipuler cet ensemble par des actions pour générer d'autres connaissances.

D'un autre côté, les auteurs de l'algorithme [Ramkumar.T et al. 2010] proposent une architecture multi-niveaux. Bien que cet algorithme peut être une source d'inspiration mais présente aussi certaines limites :

- Ils proposent une architecture à trois niveaux ce qui ne correspond pas au cas général où une organisation peut avoir plusieurs niveaux (unité, branche, division, direction centrale…).

Algorithme	Type de motifs transférés			Richesse de génération			Poids		Niveau de synthèse		Perte d'information			Régénération		Processus guidé par l'utilisateur		Contrôle de la taille des motifs	
	IF	RA	TR	MG	MM	ME	Site	RA	2	+ieurs	Avec	Sans	Partiel	Oui	Non	Oui	Non	Oui	Non
[Animesh.A et al. 2010a]			X	X			X		X			X			X	X		X	
[Animesh.A et al. 2010b]	X			X			X		X				X		X		X		X
[Ramkumar.T et al. 2008]	X			X			X		X		X				X		X		X
[Ramkumar.T et al. 2009]	X			X			X		X				X	X			X		X
[Ramkumar.T et al. 2010]	X			X			X			X			X	X			X		X
[Xindong.W et al. 2003]		X		X				X	X		X				X		X		X
[C.Zhang et al. 2005]	X				X	X			X						X		X		X

- **IF :** Itemsets Fréquents
- **RA :** Règles d'Association
- **TR :** Transactions
- **MG :** Motifs Globaux
- **MM :** Motifs Majoritaires
- **ME :** Motifs Exceptionnels

TAB. 2.2 – Comparaison entre les algorithmes de découverte des règles d'association dans un environnement multi-bases de données

- L’intervention de l’utilisateur est en fin du processus de la fouille de données multi-niveaux et ceci pour affecter l’étiquette de la classe qui ne correspond pas aux groupes déjà définis. L’utilisateur n’est pas présent dans toutes les phases du processus de la fouille de données.

6 Conclusion

Nous venons de présenter dans ce chapitre un état de l’art synthétique des travaux permettant de faire la recherche de règles d’association dans un environnement multi-bases de données. Plusieurs types d’algorithmes on été abordés comme les algorithmes d’extraction des règles d’association majoritaires/exceptionnelles, les algorithmes de synthétisation des motifs globaux à partir des motifs locaux et les algorithmes d’extraction des règles d’association à trois niveaux. Cet état de l’art est non exhaustif mais permet toutefois de réaliser un survol clair et concis de ce qui se fait au niveau de la recherche de règles d’association dans un environnement multi-bases de données. Pour les lecteurs intéressés, une analyse plus approfondie des principaux algorithmes d’analyse de motifs fréquents est proposée dans [Ramkumar.T et al. 2013], notamment la synthétisation des motifs globaux.

Nous avons également présenté dans ce chapitre, une synthèse suivant des critères que nous avons définis pour nous guider sur le choix de la future direction à entreprendre pour résoudre notre problématique. Malheureusement aucun algorithme ne peut extraire des connaissances à multiples niveaux suivant des attentes des utilisateurs ce qui nous a amené à recourir aux services du domaine de la gestion des connaissances dans le processus d’extraction des règles d’association. A cet effet, nous allons dresser, dans le chapitre suivant, les principaux travaux qui intègrent les connaissances de l’utilisateur dans le processus d’extraction des règles d’association dans une seule source de données.

Chapitre 3 : La découverte des règles d'association guidée par les connaissances de l'utilisateur

- Introduction
- Les Différentes approches
- Etude comparative
- Synthèse
- Conclusion

Chapitre 3

La découverte des règles d'association guidée par les connaissances de l'utilisateur

1 Introduction

Dans le chapitre précédent, nous avons vu que la contribution de l'utilisateur dans le processus de la fouille multi-bases de données est très souvent superficielle et le formalisme de représentation de ces connaissances est très pauvre. Ce qui nous a amener à explorer quelques issues dans la fouille monobase de données pour enrichir le formalisme de représentation des connaissances de l'utilisateur dans le processus multi-bases de données. Nous nous sommes intéressés principalement aux travaux de Liu [B.Liu et al. 1999] et de Marinica [Claudia.M et al. 2008] et nous nous sommes inspirés de leurs propositions pour proposer une approche pour la fouille multi-bases de données guidée par les connaissances de l'utilisateur. Le premier travail [B.Liu et al. 1999] dresse les concepts de bases de la représentation des attentes des utilisateurs. Ces concepts ont été élargis par Marinica [Claudia.M et al. 2008] avec les algorithmes *ARIPSO* et *ARLIUS* pour présenter les connaissances intéressantes.

Nous exposons dans la section 2 ces travaux, nous les analysons ensuite selon certains critères dans la section 3. Dans la section 4, nous définissons les aspects issus des travaux auxquels nous nous intéressons pour les généraliser à la fouille multi-bases de données.

2 Les différentes approches

Dans cette partie nous allons présenter quelques travaux de recherche qui font intervenir les connaissances des utilisateurs dans le processus de découverte des règles d'association. Nous nous intéressons aux travaux de Liu [B.Liu et al. 1999] et de Marinica [Claudia.M et al. 2008]. Liu et al sont sans doute les premiers à mettre un cadre formel pour représenter les connaissances des utilisateurs à travers le langage de spécification défini dans leur article [B.Liu et al. 1999]. Ensuite Marinica a utilisé ce langage de spécification avec des mesures objectives et l'a appliqué à un domaine réel qui est celui de l'habitat. Ci-dessous, nous présentons ces deux approches.

2.1. Exploration visuelle des règles d'association intéressantes [B.Liu et al. 1999]

Les auteurs dans [B.Liu et al. 1999] proposent une nouvelle plate forme pour permettre à l'utilisateur d'explorer efficacement les règles d'association découvertes. Cette plate forme est composée de deux composantes : - Analyse des règles d'association – Visualisation de ces règles d'association.

Le premier composant analyse et organise les règles d'association découvertes suivant plusieurs critères avec le respect des attentes des utilisateurs. En effet, un langage de spécification est défini pour représenter ces attentes. Ensuite le deuxième composant permet à l'utilisateur de visualiser ces règles d'association. Dans cette étude nous nous intéressons au premier composant que nous exposons dans ce qui suit, et qui consiste en la définition d'un langage de spécification et l'analyse des règles découvertes en utilisant les connaissances existantes.

2.1.1 La définition du langage de spécification

Le langage de spécification est défini pour permettre à l'utilisateur d'exprimer ses propres connaissances. Ce langage se focalise sur la représentation des connaissances de l'utilisateur existantes sur les règles d'association entre items dans une base de données. La syntaxe classique du langage prend le même format que celui des règles d'association.

Ce langage décrit trois niveaux de spécification. Chacun représente un type de connaissance. Ces niveaux sont : impression générale, concepts raisonnablement précis et connaissances précises. Les deux premiers niveaux représentent les connaissances vagues de l'utilisateur et le dernier niveau représente ses connaissances précises. Cette division est importante car l'utilisateur typiquement dispose des connaissances vagues et précises.

Le langage proposé utilise l'idée d'hiérarchie de classe (taxonomie) comme représentée dans les exemples suivants :

{*Pneumatique Transfert, pompe eau de mer, transfert propane*}$\subset$ {*Pompe*} $\subset$ {*Équipements stratégiques*}

{*Compresseur à air, turbo compresseur, Bog propane*}$\subset$ {*Compresseur*} $\subset$ {*Équipements stratégiques*}

{*Échangeur Principale, Rebut gaz alimentation*}$\subset$ {*Echangeur*} $\subset$ {*Équipements stratégiques*}

Pompe, Compresseur, échangeur et *équipements stratégiques* sont les classes et *Pneumatique Transfert, pompe eau de mer, transfert propane, compresseur à air,*

turbo compresseur, Bog propane, échangeur Principale, Rebut gaz alimentation sont des items. Il faut noter que dans les règles d'association généralisées le nom de la classe peut être aussi traité comme item dans ce cas nous ajoutons le symbole # à côté du nom de la classe.

2.1.1.1 Niveau impression Générale (*IG*)

Ce niveau représente les connaissances vagues de l'utilisateur qui peuvent être quelques associations entre classes d'items, mais il n'est pas sûr comment elles sont associées. Ce qui peut être exprimé comme suit :

$$IG(<S_1, \ldots, S_m>) \text{ [Support, Confiance]}$$

où :

(1) Chaque S_i est un item, classe d'items ou une expression C+ ou C*, où C est une classe. C+ et C* correspondent à une ou plusieurs, et zéro ou plusieurs instances de la classe C, respectivement.

(2) la règle découverte : $a_1 \ldots a_n \rightarrow b_1 \ldots b_k$ est conforme au *IG* si $<a_1, \ldots, a_n, b_1, \ldots, a_k>$ peut être considéré comme une instance de $< S_1, \ldots, S_m>$, sinon la règle est imprévue suivant *IG*.

(3) le support et la confiance sont optionnels. L'utilisateur peut spécifier le support minimum et la confiance minimale des règles qu'il veut extraire.

Exemple 3.1 : L'utilisateur peut exprimer qu'il existe quelques associations entre les items {*compresseur à air, turbo compresseur*}, *pompe* et *échangeur principal*. Il peut spécifier sa requête comme suit :

IG(<{compresseur à air, turbo compresseur}, pompe+, échangeur principal >)*

Les exemples suivant sont des règles d'association qui sont conformes à cette spécification :

- *Transfert propane → échangeur principal*
- *Pneumatique Transfert, pompe eau de mer, échangeur principale → pompe à air*

La règle suivante est inattendue :

- *Compresseur à air → échangeur principal*

Car *pompe+* est non satisfaite.

2.1.1.2 Niveau concepts raisonnablement précis (*CRP*)

Ce niveau représente les concepts de l'utilisateur qui peuvent être quelques associations entre les classes d'items, en connaissant la direction de l'association. Ce qui peut être exprimé par :

$$CRP(<S_1, \ldots, S_m \rightarrow V_1, \ldots, V_g>) \text{ [Support, Confiance]}$$

où :

(1) Chaque S_i ou V_i correspond au S_i dans la spécification *IG*.

(2) la règle découverte : $a_1 \ldots a_n \rightarrow b_1 \ldots b_k$ est conforme au *CRP* si la règle peut être considérée comme une instance de *CRP*, sinon la règle est imprévue suivant *CRP*.

(3) le support et la confiance sont aussi optionnels.

Exemple 3.2 : Supposant la requête de l'utilisateur :

CRP(<échangeur, échangeur, #compresseur → { Pneumatique Transfert, transfert propane}+>)

Les expressions suivantes sont des exemples de règles d'association respectant cette spécification :

- *Echangeur principale, Rebut gaz alimentation, compresseur → Pneumatique Transfert*
- *Rebut gaz alimentation, compresseur principale, compresseur → Pneumatique Transfert, transfert propane*

Les deux règles suivantes sont inattendues :

- *(1) Rebut gaz alimentation, compresseur → Pneumatique Transfert*
- *(2) Compresseur à aire, Rebut gaz alimentation → Pneumatique Transfert*

La règle (1) est inattendue car elle ne contient qu'un seul item *échangeur*, alors que deux items *échangeur* sont motionnés dans la spécification.
La règle (2) est inattendue car l'item *compresseur* n'est pas dans la condition de la règle.

2.1.1.3 Niveau connaissances précises (*CP*)

L'utilisateur connait l'association avec précision. Ce qui peut être exprimé par :

$$CP(<S_1, \ldots, S_m \rightarrow V_1, \ldots, V_g>) \text{ [Support, Confiance]}$$

où :

(1) Chaque S_i ou V_i correspond à un item dans I.

(2) la règle découverte : $a_1 \ldots a_n \rightarrow b_1 \ldots b_k$ [Support, Confidence] est égale au *CP* si la règle est la même que $S_1, \ldots, S_m \rightarrow V_1, \ldots, V_g$, qui correspond à *CP*, sinon la règle est imprévue suivant *CP*.

(3) le support et la confiance sont obligatoires.

Exemple 3.3 : Supposons la spécification de l'utilisateur suivante :

CP(<#échangeur, compresseur à air → Transfert propane>)[10%,50%]

La règle suivante découverte est conforme à la spécification car le support et la confiance de la règle sont proches de ceux de la spécification.

- Échangeur, compresseur à air → Transfert propane [8%,53%]

La règle suivante est moins conforme à la spécification *CP* car son support et confiance sont loin de ceux de la spécification *CP*.

- Échangeur, compresseur à air → Transfert propane [4%,30%]

2.1.2 Analyse des règles découvertes

Cette étape consiste à une comparaison syntaxique entre les règles découvertes avec *GI* et *CRP*. Pour *CP* les auteurs utilisent une analyse sémantique basée sur le support et la confiance sur l'ensemble des règles découvertes. Pour plus de détail sur cette analyse se référer à l'article [B.Liu et al.1998].

Soit *U* l'ensemble des spécifications de l'utilisateur représentant ses connaissances. Soit *A* l'ensemble des règles d'association découvertes. La technique d'analyse proposée repose sur la comparaison avec les règles d'association découvertes pour trouver certains types de règles intéressantes. On distingue quatre types : règles conformes, règles antécédent inattendu, règles conséquence inattendue et règles antécédent et conséquence inattendus.

Règles conformes : une règle $A_i \in A$ est conforme aux connaissances de l'utilisateur $U_j \in U$ si l'antécédent et la conséquence de la règle A_i correspondent à U_j. Le degré de conformité de la règle A_i avec U_j est calculé par la mesure $confo_{ij}$:

$$confo_{ij} = L_{ij} * R_{ij}$$

Avec L_{ij} et R_{ij} les degrés de correspondance de l'antécédent et la conclusion de la règle A_i avec U_j respectivement.

Règles en conséquence inattendue : ce sont des règles qui ont la partie conséquence différente de celle exprimée par l'utilisateur. Une règle $A_i \in A$ est en conséquence inattendue aux connaissances de l'utilisateur $U_j \in U$ si l'antécédent de la règle A_i correspond à l'antécédent de U_j et la conséquence de la règle A_i ne correspond pas à la conclusion de U_j. La mesure de degré de conséquence inattendue est donnée par la formule suivante :

$$cons - ina_{ij} = \begin{cases} 0 & L_{ij} - R_{ij} \leq 0 \\ L_{ij} - R_{ij} & L_{ij} - R_{ij} \geq 0 \end{cases}$$

Règles en antécédent inattendu : ce sont des règles qui ont la partie antécédent différente de celle exprimée par l'utilisateur. Une règle $A_i \in A$ est en antécédent inattendu aux connaissances de l'utilisateur $U_j \in U$ si l'antécédent de la règle A_i ne correspond pas à l'antécédent de U_j et la conséquence de la règle A_i correspond à la conclusion de U_j. La mesure de degrés de conséquence inattendue est définie par la formule suivante :

$$ant - ina_{ij} = \begin{cases} 0 & R_{ij} - L_{ij} \leq 0 \\ R_{ij} - L_{ij} & R_{ij} - L_{ij} \geq 0 \end{cases}$$

Règles en antécédent et conséquence inattendus : ce sont des règles qui ont les parties antécédent et conséquence différentes de celles exprimées par l'utilisateur. Une règle $A_i \in A$ est à antécédent et conséquence inattendus aux connaissances de l'utilisateur $U_j \in U$ si l'antécédent et la conséquence de la règle A_i ne correspondent pas à l'antécédent et la conséquence de U_j. La mesure des degrés d'antécédent et de conséquence inattendue est calculée par la formule suivante :

$$ant - cons - ina_{ij} = 1 - \max(conf_{ij}, ant - ina_{ij}, cons - ina_{ij})$$

Les valeurs de $confo_{ij}$,$cons\text{-}ina_{ij}$,$ant\text{-}ina_{ij}$,$ant\text{-}cons\text{-}ina_{ij}$ sont entre 0 et 1. 1 représente la correspondance totale soit le conforme complet ou l'inattendue complet et 0 représente la non correspondance. Pour plus de détail sur le calcul de L_{ij} et R_{ij} se référer à l'article [B.Liu et al.1998].

2.2. Vers la fouille de règles d'association guidée par les ontologies et des schémas de règles [Claudia.M. 2010]

Dans la perspective de réduire le nombre des règles d'association et augmenter la qualité de ces derniers Claudia dans sa thèse [Claudia.M. 2010] aborde deux thèmes principaux: l'intégration des connaissances de l'utilisateur dans le

processus de découverte et l'interactivité avec l'utilisateur. Le premier problème exige la définition d'un formalisme adapté afin d'exprimer les connaissances de l'utilisateur avec précision et flexibilité comme les ontologies dans le Web Sémantique. Deuxièmement, l'interactivité avec l'utilisateur permet la mise en œuvre d'un processus d'exploration plus itératif où l'utilisateur peut tester successivement des hypothèses et des préférences différentes, lui permettant ainsi de se concentrer sur les règles intéressantes. Dans cette optique, l'auteur a proposé un modèle pour représenter les connaissances de l'utilisateur. Premièrement, l'auteur propose un nouveau formalisme de règles, appelé Schéma de Règles, qui permet à l'utilisateur de définir, à travers des concepts ontologiques, ses attentes sur les règles d'association. Deuxièmement, les ontologies permettent à l'utilisateur d'exprimer, à l'aide d'un modèle sémantique de haut niveau, ses connaissances de domaine. Enfin, l'utilisateur peut choisir parmi un ensemble d'opérateurs de traitement interactif celui à appliquer sur chaque schéma de règles (élagage, conforme, inattendu, ...).

Dans sa thèse Marinica a proposé deux approches : la première avec post-traitement et l'autre sans post-traitement. La première nommée *ARIPSO* (Association Rule Interactive Post-processing using rule Schemas and Ontologies) [Claudia.M et al. 2010], permet à l'utilisateur de réduire le volume de règles découvertes et d'améliorer leur qualité. L'algorithme *ARIPSO* est un processus interactif intégrant les connaissances et les attentes de l'utilisateur à l'aide du modèle proposé. La boucle interactive permet à l'utilisateur, à chaque étape d'*ARIPSO*, de modifier les informations fournies et de réitérer la phase de post-traitement qui produit des nouveaux résultats.

La deuxième approche sans post-traitement, nommée *ARLIUS* (Association Rule Local Interactive mining Using rule Schemas) [Andrei Olaru et al. 2009], consiste en un processus d'exploration locale et interactive guidée par l'utilisateur. Elle permet à l'utilisateur de se concentrer sur les règles intéressantes sans qu'il soit nécessaire d'extraire toutes les règles et sans une limite pour le support minimum. De cette façon, l'utilisateur peut explorer l'espace de règles progressivement, une petite quantité à chaque étape, à partir de ses propres attentes et des règles découvertes liées à ces dernières.

2.2.1. ARIPSO

ARIPSO procède dans la phase de post-traitement du processus de l'*ECD*. Dans ce contexte, le processus de l'*ECD* est exécuté en deux étapes : premièrement, les règles d'association sont générées par une technique classique telle que *APRIORI*.

Ensuite *ARIPSO* sélectionne celles qui sont intéressantes dans la phase de post-traitement.

ARIPSO est composé de deux principales phases définies dans la figure 3.1 :

– Définition des connaissances de l'utilisateur –phase de post-traitement.

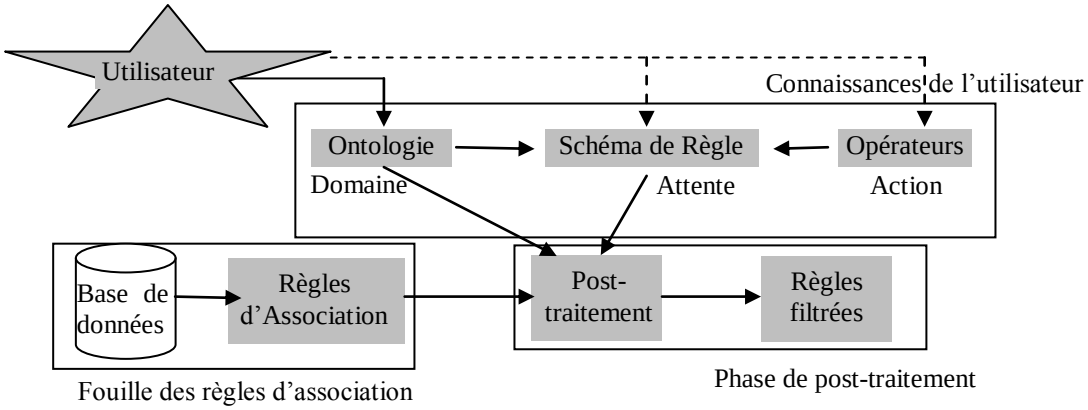


FIG. 3.1 – DESCRIPTION D'ARIPSO

Phase de définition des connaissances de l'utilisateur : Cette étape consiste à définir des connaissances de base permettant de formaliser les connaissances de l'utilisateur. Trois composantes sémantiques sont intégrées dans l'approche:

- **Domaine :** Représente les connaissances de l'utilisateur sur le domaine étudié pour apporter une vue générale subjective sur ce domaine. Cependant, l'utilisateur qui est l'expert du domaine doit décrire complètement le domaine. *ARIPSO* utilise une ontologie composée de trois concepts : –Concept feuille –Concept généralisé créé avec l'utilisation de la relation (≤) –Concept Restriction créé avec l'utilisation des restriction sur les autres concepts et propriétés.

- **Attente :** L'utilisateur a une idée sur les règles d'association découvertes il doit connaître le type de règle qu'il doit ignorer ou garder. Cependant, *ARIPSO* permet d'offrir un moyen d'exprimer ses attentes et ses exceptions sur les règles d'association découvertes. *ARIPSO* utilise un nouveau formalisme appelé schéma de règles qui permet de formaliser les attentes de l'utilisateur.
- **Action :** *ARIPSO* permet de donner la possibilité à des utilisateurs d'agir sur les connaissances en appliquant plusieurs opérateurs sur leurs attentes. *ARIPSO* utilise quatre opérateurs appliqués sur le schéma de règles : –Elagage – Conforme – Inattendu –Exception.

Phase de post-traitement : Cette étape consiste à analyser l'ensemble des règles d'association avec utilisation des différentes mesures objectives dont la perspective de réduire le nombre de ces règles. Deux types de mesures ont été employés dans *ARIPSO* : Schéma de règles créé par application des opérateurs sur le schéma de règles, et des filtres objectifs tirés à partir de la littérature.

Dans ce qui suit nous allons présenter le cadre formel du schéma de règles et les opérateurs définis selon Marinica [Claudia.M. 2010].

2.2.1.1 Schéma de règles

Pour l'amélioration de la sélection des règles d'association, *ARIPSO* défini un model de filtrage de règles d'association, appelé schéma de règles. Ce dernier décrit, sous forme de règles, les souhaits de l'utilisateur en termes de règles intéressantes. Le formalisme de base de schéma de règles est le model de représentation de l'utilisateur introduit par [B.Liu et al. 1999] composé de : *impression générale, concepts raisonnablement précis et connaissances précises*. Le schéma de règles est une extension sémantique du modèle proposé par Liu [B.Liu et al. 1999] du moment qu'il est décrit en utilisant les concepts à partir du domaine ontologie. *ARIPSO* propose de développer deux de ces trois représentations introduites dans [B.Liu et al. 1999] qui sont : Impression Générale et concepts raisonnablement précis.

Définition 3.1. Un schéma de règles est défini comme suit :

$$< X_1, X_2, \ldots, X_n \, (\rightarrow) \; Y_1, Y_2, \ldots, Y_m \; >$$

Tel que X_i, Y_i sont des concepts de l'ontologie et «$\rightarrow$ » est optionnel. En d'autres termes, on peut noter que le formalisme proposé combine les impressions générales, les concepts raisonnablement précis et les concepts précis. Par conséquent, le formalisme avec implication (*schéma de règles implicatives*) définit les connaissances raisonnablement précises et précises. Par contre, si on ignore l'implication (*schéma de règles non implicatives*) ce formalisme définit les connaissances vagues.

2.2.1.2 Opérateurs

Le post-traitement est basé sur les opérateurs appliqués au schéma de règles pour permettre à l'utilisateur de manipuler les règles d'association découvertes. *ARIPSO* se base sur deux classes d'opérateurs : élagage et filtrage des règles

d'association. La classe de l'opérateur de filtrage est composée de trois opérateurs : conforme, inattendu et exception. L'opérateur inattendu comprend : antécédent inattendu, conséquence inattendue et antécédent et conséquence inattendus. Tandis que la classe de l'opérateur d'élagage contient un seul opérateur qui est l'opérateur d'élagage.

Soit un schéma de règles implicatives SR_1 : < X→ Y >, et un schéma de règles non implicatives SR_2 : < U,V >, et une règle d'association AR_1 : A→ B où X,Y,U,V sont des concepts de l'ontologie et A, B sont des itemsets.

L'opérateur **d'élagage** permet à l'utilisateur de supprimer les familles des règles d'association considérées non intéressantes. Dans la base de données il existe dans plusieurs cas des relations triviales et connues entre items. Cependant il n'est pas intéressant de trouver ces relations entre les règles d'association. Cet opérateur élimine toutes les règles d'association en correspondance avec le schéma de règles.

L'opérateur **conforme** appliqué sur le schéma de règles, a pour rôle de confirmer une implication ou de trouver l'implication entre différents concepts. Comme résultat les règles qui correspondent à tous les éléments du schéma de règles non-implicatives sont filtrées. Pour un schéma de règles implicatives, la condition et la conclusion de la règle d'association doivent correspondre à celles du schéma de règles. La règle AR_1 est sélectionnée par l'opérateur conforme si la condition et la conclusion de la règle AR_1 respectivement correspond à la condition et la conclusion de SR_1. De même, la règle AR_1 est filtrée si la condition et/ou la conclusion de la règle AR_1 correspond au schéma SR_2:

L'opérateur **conséquence inattendue** filtre l'ensemble de règles qui ont la partie conséquence différente de celle exprimée par l'utilisateur. Il n'est applicable qu'en cas de schéma de règles implicatives et découvre des règles conformes en respectant le schéma de règles implicatives. Cet opérateur découvre les règles qui sont contradictoires aux connaissances de l'utilisateur.

L'opérateur **antécédent inattendu** filtre l'ensemble de règles qui ont la partie antécédent différente de celle exprimée par l'utilisateur. Il n'est applicable qu'en cas de schéma de règles implicatives et découvre des règles conformes en respectant le schéma de règles implicatives.

L'opérateur **antécédent et conséquence inattendus** filtre l'ensemble de règles qui ont la partie antécédent et conséquence différentes de celles exprimées par l'utilisateur. Il n'est applicable qu'en cas de schéma de règles implicatives et découvre des règles conformes en respectant le schéma de règles implicatives.

Finalement, l'opérateur **exception** appliqué sur SR_1 est défini seulement sur le schéma de règles implicatives et extrait les règles conformes avec respect du schéma de règles implicatives suivant: $X \wedge Z \rightarrow Y$, avec Z est un ensemble d'items.

2.2.2. ARLIUS

Dans cet algorithme, les connaissances de l'utilisateur sont intégrées dans le processus de découverte des règles d'association. Ceci s'explique par l'intérêt de l'utilisateur porté sur les règles d'association intéressantes. L'algorithme *ARLIUS* propose aussi un schéma de règles et opérateurs différents de ceux d'*ARIPSO*.

La différence de *ARLIUS* par rapport à *ARIPSO* réside dans le fait que :

- *ARLIUS* est basé sur la recherche locale tandis que *ARIPSO* est une technique post-traitement.
- Le formalisme du schéma de règles d'*ARLIUS* est différent de celui d'*ARIPSO*.
- L'algorithme de découverte des règles d'association n'est pas le même.
- En plus des deux opérateurs : conforme et exception (k-exception) *ARLIUS* ajoute deux autres opérateurs : k-spécialisation et k-généralisation.

L'approche *ARLIUS* est composée de deux composantes ainsi décrit dans la figure 3.2. Le premier composant est constitué des connaissances des utilisateurs représentées par le nouveau schéma de règles et les opérateurs.

Le second concerne l'algorithme de découverte des règles d'association avec intégration des connaissances de l'utilisateur (schémas de règle et opérateurs) dans le processus de la découverte des connaissances. Par conséquent, l'objectif de ce composant est de présenter à l'utilisateur un ensemble réduit de règles d'association tout en réduisant l'espace de recherche.

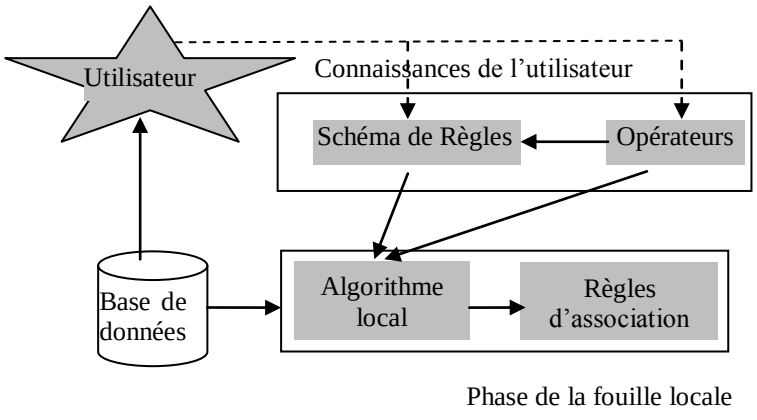


FIG. 3.2 – DESCRIPTION DE ARLIUS

2.2.2.1 Schéma de règles

Ce schéma de règles diffère de celui d'*ARIPSO* dans le sens où il ne peut exprimer certaines attentes des utilisateurs comme par exemple, si un utilisateur veut exprimer une requête sur un item et il ne sait pas sa position (antécédent, conclusion). En effet, les auteurs d'*ARLIUS* justifient ce nouveau schéma de règles par le fait que certains utilisateurs ne savent pas vraiment l'appartenance d'items à quelle partie de la règle. Avec le schéma de règles d'*ARIPSO* il est difficile d'exprimer ce type de requête. *ARLIUS* a augmenté le schéma de règles dans *ARIPSO* en ajoutant deux parties : La première c'est la partie *[Générale]* et la deuxième partie c'est le support et la confiance des règles d'association.

Définition 3.2. Un schéma de règles est défini comme suit :

$$(\text{Antécédent} \rightarrow \text{Conséquence [Générale]}) \quad [s\%,c\%]$$

L'antécédent et la conséquence contiennent des items des attentes de l'utilisateur présents respectivement dans l'antécédent et la conclusion des règles d'association. La partie générale contient des items dont l'utilisateur ne sait pas dans quelles parties les placer.

2.2.2.2 Opérateurs

ARLIUS offre quatre opérateurs, un d'entre eux est équivalent à celui employé dans *ARIPSO* qui est : l'opérateur Conforme. Les trois autres opérateurs sont : les opérateurs k-Spécialisation, k-Généralisation et k-exception.

L'opérateur **k-spécialisation** permet à l'utilisateur de trouver les règles d'association qui ont l'antécédent plus spécifique avec la même conclusion et qui améliore la confiance de la règle initiale. En d'autres termes, cet opérateur permet de spécialiser le schéma de règles en spécialisant l'antécédent, ce qui implique l'ajout d'autres items dans la partie antécédent. La variable k exprime le nombre d'items qui sont ajoutés dans l'antécédent durant la spécialisation. Par exemple, la spécialisation de la règle $X \rightarrow Y$ [s_1 c_1] est la règle X, $Z \rightarrow Y$ [s_2 c_2], si $c_2 > c_1$; si Z contient un item, on peut dire c'est 1-spécialisation.

L'opérateur **k-généralisation** est l'opposé à l'opérateur k-spécialisation. Il recherche des règles d'association qui ont l'antécédent plus général avec la même conclusion. Comme dans k-spécialisation, l'opérateur k-généralisation illustre le nombre d'items supprimés à partir de la partie antécédent.

L'opérateur **k-exception** est un opérateur important, il permet de rechercher les règles avec conclusion inattendue dans le contexte de l'antécédent spécialisé. Par exemple, soit la règle $X \rightarrow y$ [s_1 c_1] alors la règle exception est de la forme $X,Z \rightarrow \neg y$ [s_2 c_2] avec $c_2 \geq c_1$. Par conséquent l'opérateur K-exception est l'opérateur k-spécialisation des règles avec conclusion négative.

3 Etude comparative

Dans cette section, nous résumons les méthodes décrites précédemment en se basant sur les critères que nous avons définis et qui nous semble pertinents pour notre problématique. Nous définissons deux critères : les phases de l'ECD et le contrôle direct sur le nombre des règles d'association générées.

- Les phases de l'ECD : Nous précisons pour chaque méthode étudiée, dans quelle phase de l'ECD s'intègrent les connaissances de l'utilisateur. Pour cela, Nous distinguons deux phases : traitement des données et post-mining.

1. Traitement des données: Motivé par la réduction et le ciblage de l'espace de recherche, ce critère concerne la phase du traitement. La fouille des règles d'association guidée par les connaissances de l'utilisateur permet de découvrir les relations de régularité et d'exception qui associent les données.

2. Post-Traitement : Ce critère concerne l'emploi des connaissances de l'utilisateur dans la phase de post-traitement. En effet, l'usage du modèle des règles d'association en fouille de données est limité par la quantité prohibitive de règles générées. Par conséquent, il requiert la mise en place d'un post-traitement efficace et adapté à la fois aux préférences des utilisateurs et à la structure des données étudiées.

-Contrôle sur le nombre des règles d'association : Il s'agit d'identifier si les méthodes proposées permettent de contrôler le nombre des règles d'association ou pas.

Le tableau 3.1 montre un récapitulatif des caractéristiques de ces méthodes. Les colonnes présentent les différents critères et les lignes contiennent les références et les noms des approches étudiées. Une croix dans une cellule indique que la méthode en ligne possède la caractéristique en colonne.

Excepté l'algorithme *ARLIUS* qui agit au niveau de la phase traitement de l'*ECD,* les deux autres algorithmes agissent au niveau de la phase post-traitement. Les deux algorithmes *ARIPSO* et celui de [B.Liu et al. 1999] proposent de réduire l'ensemble important des règles d'association générées après la génération de

toutes les règles d'association en utilisant les schémas de règles et les opérations sur ces schémas. Par contre l'algorithme *ARLIUS* propose d'abord de réduire l'espace de recherche en utilisant les schémas de règles et les opérateurs ensuite il génère l'ensemble réduit de règles d'association.

Algorithmes	Phases de l'ECD		Contrôle sur le nombre des RA	
	Traitement	Post-Traitement	Oui	Non
[B.Liu et al. 1999]		X		X
ARIPSO		X		X
ARLIUS	X		X	

TAB. 3.1 – Comparaison entre les différentes approches

Le schéma de règles et les opérateurs fournissent un moyen de contrôler la taille des règles d'association. Cependant avec ces schémas de règles et opérateurs ce nombre reste toujours élevé. L'algorithme *ARLIUS* introduit en plus de ces deux moyens un autre moyen qui est le paramètre *k* pour réduire et contrôler le nombre des règles d'association. En effet, cet algorithme permet de contrôler le nombre des règles d'association en variant le paramètre *k*. Toutefois si *k* est plus grand l'algorithme *ARLIUS* génère un nombre important de règles d'association.

4 Synthèse

D'après notre problématique initiale qui est l'extraction des règles d'association sans perte de connaissances à plusieurs niveaux dans un environnement multi-bases de données, les trois algorithmes cités sont candidats à être utilisés dans notre cas avec une adaptation majeure. En effet, l'utilisation de ces approches localement c.à.d. au niveau de chaque site peut être une solution de notre problème. Dans ce cas, chaque utilisateur dans son niveau peut exprimer ses attentes à travers un schéma de règles local et appliquer les opérations sur ce schéma de règles.

Néanmoins pour utiliser ces approches dans un environnement multi-bases de données une adaptation et une généralisation de cette représentation sont nécessaires. En effet, la représentation des schémas de règles ne permet d'exprimer que les attentes d'un seul niveau et qu'un seul type de connaissance (local). Pour cela, nous proposons de généraliser ce schéma de règles afin de prend en considération l'environnement multi-bases de données.

Notre approche est basée sur les trois aspects suivants: – Généraliser le schéma de règles – Définir de nouveaux opérateurs – Associer les schémas de règles et les opérateurs de tous les niveaux supérieurs dans le processus d'extraction des règles d'association locales.

- Généralisation du schéma de règles: L'environnement monobase de données permet de représenter les connaissances des utilisateurs de même niveau (un seul niveau). Ils peuvent être des utilisateurs d'un centre commercial qui récoltent quotidiennement des données et les stockent dans une seule base de données. Les utilisateurs occupent le même poste (niveau) qui est le poste de caissier dont les attentes peuvent être exprimées avec des représentations déjà définies dans l'état de l'art. Ce qui n'est pas le cas avec une organisation multi-niveaux où les centres commerciaux sont éparpillés dans un pays où région. En effet, nous pouvons trouver de simples caissiers, des responsables commerciaux par région et/ou par pays. Par conséquent, il est nécessaire de trouver une représentation qui satisfait ce nouvel environnement en augmentant le schéma de règles déjà défini dans la littérature.

L'environnement monobase de données ne permet de représenter qu'un seul type de connaissances qui est le type local. Par contre, l'environnement multi-bases de données comme déjà mentionné dans le chapitre 1, permet de représenter plusieurs types de connaissances (Locales, Majoritaires/Exceptionnelles et globales). Par conséquent l'enrichissement des schémas de règle déjà proposés dans la littérature est nécessaire pour prendre en compte cet aspect.
L'enrichissement de ces représentations est nécessaire pour prendre en compte l'aspect multi-niveaux et la diversité des types de connaissance. Ce qui va donner naissance à une nouvelle représentation qui est la représentation *du schéma de règles multi-niveaux* dont l'aspect formel de cette représentation est défini dans le chapitre 4.

- Définition de nouveaux opérateurs : Les opérateurs utilisés dans les approches citées ci-dessus sont spécifiques et spécialisés et ne prennent pas en compte la diversité des types de connaissances générées. Pour cela nous proposons quatre types d'opérateurs. Nous utilisons l'idée de *ARLIUS* d'utiliser le paramètre k dans tous les opérateurs proposés pour contrôler le nombre des règles d'association. Nous présentons les opérateurs de façon succincte dans ce qui suit, nous les détaillerons dans le chapitre 4.

-Prenons l'exemple de l'opérateur conforme, ce dernier génère toutes les règles d'association qui sont conformes au schéma de règles avec le mélange des items dans les deux parties de la règle ce qui va engendrer un nombre important de règles

d'association. Aucun contrôle sur la taille et le nombre des règles d'association n'est spécifié. Pour cela, nous proposons un nouvel opérateur *k-conforme* pour parvenir à cette limite.

-Dans une organisation, souvent les utilisateurs s'intéressent à un motif bien particulier dans une des parties de la règle. Prenons l'exemple d'une personne de la maintenance qui s'intéresse à la défaillance d'un équipement stratégique dans la partie conclusion. Hors les opérateurs cités dans la littérature ne permettent pas d'exprimer ce type d'attente. Pour cela, nous proposons un nouvel opérateur *k-objectif* pour prendre en compte ce type de connaissances.

-D'un autre côté, un nombre important des utilisateurs veulent exprimer des attentes sur un motif dans la partie antécédent de la règle d'association. Ils veulent s'avoir par exemple l'effet de la panne d'un joint sur les autres équipements. Vu que les opérateurs cités dans la littérature ne permettent pas de satisfaire cette demande, nous avons proposé un nouvel opérateur *k-non objectif* pour intégrer ce type de connaissance dans le processus de découverte des règles d'association.

- Le dernier opérateur définit l'aspect type de connaissances. En effet, prenons l'exemple d'un utilisateur d'un des niveaux supérieurs (>1) qui veut exprimer une exception sur les connaissances majoritaires, les opérateurs déjà définies dans la littérature ne permettent pas d'exprimer ce type d'attente. Pour cela nous proposons un autre opérateur plus général et qui permet de prendre en compte l'aspect multi-bases de données tel que l'opérateur *type inattendu*.

- Associer les schémas de règles et les opérateurs de tous les niveaux dans le processus d'extraction des règles d'association locale : Cette proposition a un double objectif : -Réduire l'espace de recherche – Avoir l'information sur les motifs non localement fréquents pour les utiliser dans les niveaux supérieurs.

Pour le premier point, l'association des attentes des utilisateurs dans le processus de la fouille de données locale est impérative pour minimiser et réduire l'espace de recherche dans la fouille monobase de données. L'algorithme *ARLIUS* illustre bien cette association dans le processus d'extraction des règles d'association. Sauf que dans notre cas nous allons introduire en plus du schéma de règles locales et opérateurs, les schémas de règles et opérateurs des niveaux supérieurs reliés à la base de données locale (même branche). Dans ce cas nous avons besoin d'un seul parcours de la base de données pour comptabiliser les supports des motifs candidats, ce qui représente un gain énorme par rapport à l'algorithme *APRIORI* par exemple.

Pour le second point, dans un environnement multi-bases de données en plus de la réduction de l'espace de recherche s'ajoute l'association des motifs non localement fréquents dans le processus d'extraction des règles d'association. Ce point va résoudre le problème de la perte de connaissances. En effet, les résultats de ce processus sont : Les règles d'association locales et non localement fréquents qui vont être utilisées dans le processus de synthétisation pour calculer les motifs globaux de façon exacte.

5 Conclusion

Dans ce chapitre, nous avons présenté un état de l'art sur les différents travaux de recherche qui font concourir les connaissances d'expertise dans le processus de découverte des règles d'association. Nous avons procédé à une brève comparaison entre ces approches par des critères que nous avons définis. Ceci nous a permis d'opter pour les ontologies et les schémas de règles afin de les adapter à un environnement multi-bases de données.

Nous nous sommes basés sur le formalisme des schémas de règles défini dans l'algorithme *ARIPSO* avec une adaptation à l'environnement multi-bases de données. Nous avons aussi utilisé l'idée d'introduire les connaissances de l'utilisateur dans la fouille locale des données pour réduire l'espace de recherche et nous avons utilisé le paramètre *k* défini dans l'algorithme *ARLIUS* pour contrôler le nombre de règles d'association générées.

Chapitre 4 : Démarche méthodologique

- Introduction
- Formulation de la problématique
- Présentation de l'algorithme RAMARO
- Approche Intra-Site
- Approche Inter-site
- Conclusion

Chapitre 4
La démarche méthodologique

1 Introduction

Ce chapitre présente l'approche proposée dite *RAMARO* (Règles d'Association Multi-niveaux d'Abstraction en utilisant le schéma de Règles et Ontologie) qui permet l'extraction d'un nombre réduit de règles d'association pour chaque niveau d'abstraction dans un environnement multi-niveaux. Notre principale motivation est justifiée par le besoin de mettre à la disposition des décideurs dans chaque niveau d'abstraction l'ensemble des règles d'association adéquates. A cette fin, nous avons orienté notre approche vers l'utilisateur pour guider le processus de découverte des règles d'association vers les règles pertinentes. Pour cette raison, nous considérons que l'intégration des connaissances de l'utilisateur dans le processus de la *FMBD* est impérative.

Ce chapitre est structuré comme suit : nous présentons dans la section 2 la formulation du problème. Dans la section 3, nous définissons formellement l'approche *RAMARO*. Nous détaillons la phase intra-site et inter-site de l'approche *RAMARO* dans les sections 4 et 5 respectivement. On termine par une conclusion dans la section 6.

2 Formulation de la problématique

La figure 4.1 présente une organisation sur plusieurs niveaux dont le niveau le plus bas est le niveau 1 et le plus haut est le niveau n. Les utilisateurs de niveau 1 peuvent être des responsables d'unité ou de direction dont leur fonction est d'assurer le bon fonctionnement des unités ou directions qu'ils supervisent. Dans le domaine de la maintenance industrielle, ces utilisateurs peuvent s'intéresser par exemple à des types d'équipements tels que les joints, les boulons. Cependant leur vision est différente de celles des utilisateurs des niveaux supérieurs qui ont leurs propres attentes sur les données dans chaque base de données. Les utilisateurs de niveaux supérieurs par exemple de niveau 2 peuvent s'intéresser à d'autres types d'équipement tels que les pompes. Et les utilisateurs d'autres niveaux supérieurs pourront s'intéresser aux compresseurs par exemple et ainsi de suite.

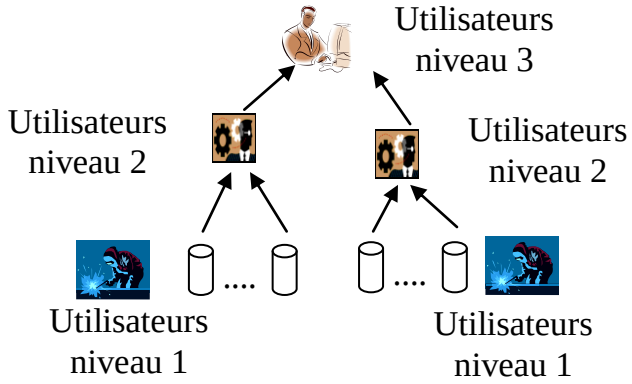


FIG. 4.1 – ORGANISATION MULTI-NIVEAUX

La motivation principale de l'algorithme *RAMARO* est de répondre aux limites du processus d'extraction des règles d'association dans les multiples bases de données comme :

- Règles d'association appropriées pour chaque niveau
- Nombre important de règles d'association
- Perte de connaissance

Dans ce contexte, l'utilisateur joue un rôle important du moment qu'une règle peut être intéressante pour un utilisateur, mais non intéressante pour un autre. Ceci dépend des types d'utilisateurs dans la hiérarchie de l'organisation multi-niveaux. Dans ce contexte les connaissances de l'utilisateur doivent être représentées par un formalisme précis et flexible pour exprimer de façon claire ce que l'utilisateur veut comme connaissances.

Pour cela l'algorithme proposé doit fournir un formalisme de représentation des attentes des utilisateurs et un algorithme d'extraction des règles d'association sans perte de connaissances adéquat à tous les niveaux. Nous décrivons l'algorithme proposé dit *RAMARO* dans les paragraphes qui suivent.

3 Présentation de l'algorithme *RAMARO*

La différence de *RAMARO* avec *ARIPSO* et *ARLIUS* est multiple. Premièrement, comme déjà mentionné, *RAMARO* est basé sur un algorithme de recherche local tandis qu'*ARIPSO* est une approche post-traitement. Par

conséquent dans notre approche, l'espace de recherche va être réduit en ne traitant que les règles qui intéressent l'utilisateur. Deuxièmement le schéma de règles développé dans *RAMARO* est différent de celui d'*ARIPSO* et d'*ARLIUS*. En effet ce nouveau schéma permet de prendre en considération l'environnement multi-bases de données et les différents types de connaissances. Aussi, les opérateurs utilisés dans *RAMARO* sont différents de ceux utilisés dans *ARIPSO* et *ARLIUS*. En effet, *RAMARO* propose de nouveaux opérateurs plus généraux que ceux d'*ARIPSO* et *ARLIUS* et de plus la taille des règles d'association générées est contrôlée à travers ces opérateurs.

L'approche *RAMARO* procède en deux phases : Intra-site et Inter-site. La figure 4.2 décrit le modèle proposé.

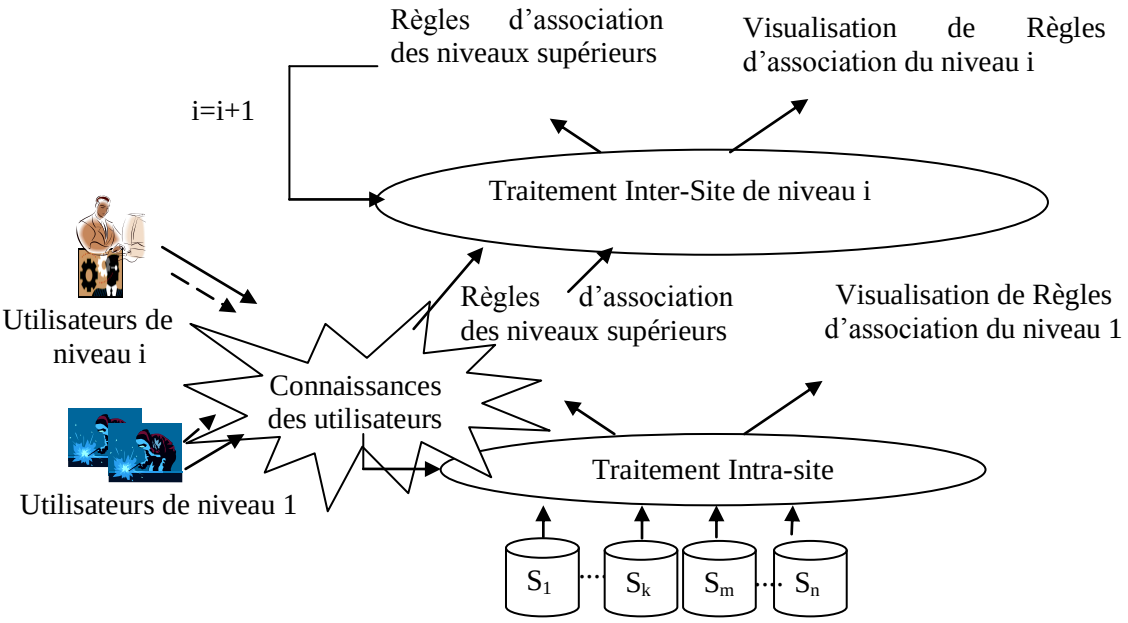


FIG. 4.2 – RAMARO MULTI-NIVEAUX

La première phase se concentre sur l'algorithme d'extraction des règles d'association au niveau local (niveau 1). De ce fait, nous proposons un nouvel algorithme d'extraction de règles d'association avec l'intégration des connaissances de l'utilisateur par l'ontologie, le schéma de règles et les opérateurs. Le formalisme développé dans cette étape introduit un nouvel ensemble d'opérateurs et une nouvelle technique d'extraction de règles locales. L'intérêt principal de cette étape est que le nombre de règles d'association pendant la recherche est réduit et des résultats partiaux peuvent être intégrés dans l'algorithme d'extraction de règles d'association.

La deuxième phase procède dans le post-traitement du processus de la *FMBD*. La nouveauté de notre approche est qu'elle supervise le processus de découverte de règles d'association en utilisant deux structures conceptuelles différentes pour représenter les connaissances de l'utilisateur du domaine par l'Ontologie et les attentes des utilisateurs à différent niveau d'abstraction par les schémas de règles pour chaque niveau. En complément, l'approche intègre aussi un ensemble d'opérateurs sur les schémas de règles dans le but de produire des actions sur les règles d'association.

Avant de détailler les deux phases de *RAMARO*, nous présentons la structure de l'ontologie, le schéma de règles multi-niveaux et les différents opérateurs proposés.

3.1 Ontologie-connaissance de l'expert du domaine

Les connaissances du domaine sont les connaissances de l'utilisateur concernant la base de données sur un domaine spécifié. Nous présentons la structure de l'ontologie et les concepts clés sur lesquels notre approche est basée. Pour cela nous considérons deux types de concepts dans l'ontologie:

- Les concepts feuilles
- Les concepts généralisés créés par l'utilisateur avec la relation (≤);

Soient un ensemble d'items dans une base de données défini par $I=\{i_1,i_2,\ldots,i_n\}$, C un ensemble de concepts et la notation ≤ qui définit les relations entre concepts.

Définition 4.1
Un concept est noté par *concept feuille (CF)* s'il n'englobe pas d'autre concepts dans l'ontologie. Formellement, il est défini comme suit :

$$CF = \{C_0 \in C \mid \nexists C' \in C . C' \leq C_0\}$$

Définition 4.2
Un concept est noté par *concept généralisé (CG)* s'il englobe au moins un concept dans l'ontologie. Formellement, il est défini comme suit :

$$\mathrm{CG} = \{\mathrm{C}_1 \in \mathrm{C} \mid \exists \mathrm{C}' \in \mathrm{C} . \mathrm{C}' \leq \mathrm{C}_1\}$$

La figure 4.3 présente des exemples de *concepts feuilles* et de *concepts généralisés* dans l'ontologie d'une application de maintenance.

Le concept pompe englobe les concepts feuilles : *Pneumatique transfert, pompe eau de mer* et *transfert propane* qu'on peut représenter en utilisant la relation (≤) par : *Pneumatique transfert ≤ pompe, pompe eau de mer ≤ pompe* et *transfert propane ≤ pompe.*

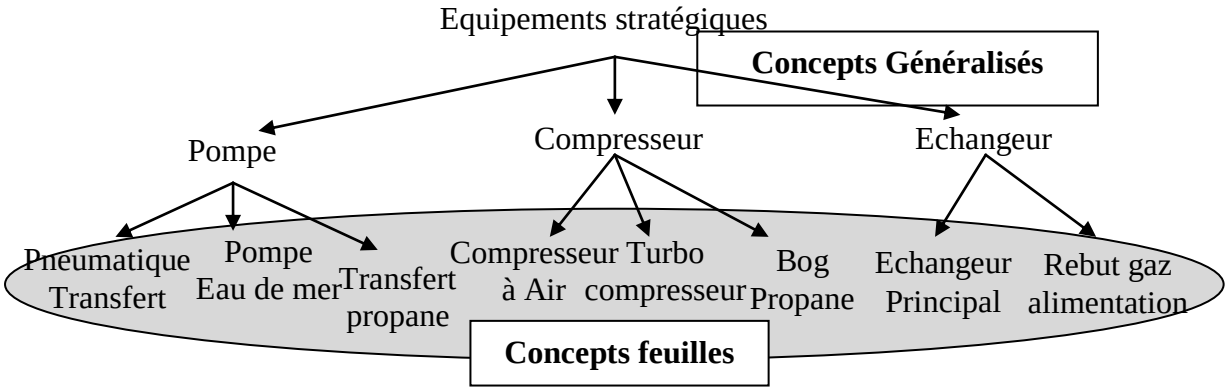


FIG. 4.3 – EXEMPLE DE CONCEPTS FEUILLES ET DE CONCEPTS GENERALISES DANS L'ONTOLOGIE DE LA MAINTENANCE

Une fois que les concepts *C* de l'ontologie définis, il est nécessaire de les connecter avec la base de données. Chaque concept est relié directement à un ou plusieurs items. La connexion directe est réalisée avec les éléments feuilles de l'ontologie.

3.2 Schéma de règles multi-niveaux

Nous décrivons le schéma de règles multi-niveaux par un ensemble de concepts d'ontologie que l'utilisateur estime que leur présence est nécessaire dans les règles d'association découvertes. Formellement, un schéma de règles est défini par :

$$SR\ (\ <X_1,X_2,\ \ldots\ X_n\ (\rightarrow)\ Y_1,\ldots\ Y_m><T><N>)$$

Avec X_i, $Y_i \in C$ et T={L,G,M,E} exprime le type de règle d'association qui peut être Local, Global, Majoritaire et Exceptionnel et N >1 représente le numéro du niveau. L'implication optionnelle « → » combine le formalisme impression générale et concepts raisonnablement précis. Nous définissons un ensemble d'opérateurs sur les éléments du schéma de règles comme suit :

- L'opérateur * pour exprimer qu'un élément peut être présent ou non dans le schéma de règles;

- L'opérateur + pour exprimer qu'un élément doit être présent dans le schéma de règles.

Exemple 4.1 : En utilisant les concepts de l'ontologie de la figure 4.3, on peut définir les schémas de règles du niveau 2 et 3 suivants :

$$SR_1 : SR(<pompe, Compresseur> <M> <2>)$$

$$SR_2 : SR(<pompe+ \rightarrow Compresseur> <E> <3>)$$

Dans le premier schéma de règles, l'utilisateur de niveau 2 sait qu'il y a une relation entre les deux concepts pompe et compresseur et que ce type de règle est majoritaire, par contre dans le deuxième schéma de règles l'utilisateur de niveau 3 attend à trouver des règles d'association dont la partie antécédent contient une ou plusieurs instances de la pompe et une instance du compresseur dans la partie conséquence et que le type de règle est exceptionnel.

Le but principal d'un schéma de règles est de filtrer et d'élaguer des règles d'association découvertes. En conclusion, le schéma de règles multi-niveaux décrit les attentes de l'utilisateur en termes de règles intéressantes à n'importe quel niveau.

3.3 Operateurs sur le schéma de règles

Les opérateurs que nous proposons dans notre approche sont inspirés des travaux de [B.Liu et al. 1999] avec diverses différences que nous allons présenter dans cette section. Il faut noter que le schéma de règles exprime les attentes de l'utilisateur dans chaque niveau sous forme de règles. Par contre les opérateurs sont des actions appliquées sur ce schéma de règles. Les opérateurs proposés sur le schéma de règles doivent tenir compte de l'architecture multi-bases de données en introduisant les types de règles. Ils doivent tenir compte aussi des attentes de l'utilisateur en terme des composantes de la règle d'association c.à.d. la partie antécédent et conclusion, et de la nature de la règle d'association qui peut être locale, majoritaire, exceptionnelle ou globale.

Ci-dessous nous allons définir les quatre types d'opérateurs utilisés dans notre approche. Nous proposons deux importants ensembles d'opérateurs :

- Opérateurs de sélection : k-conforme, k-Objectif, k-non objectif, avec $k \geq 0$.
- Opérateurs de Typage : Type inattendu.

L'opérateur de sélection k-conforme est une généralisation de l'opérateur conforme proposé par [B.Liu et al. 1999]. Les opérateurs de sélection k-objectif et k-non objectif sont de nouveaux opérateurs que nous avons proposés dans notre approche.
Pour le deuxième ensemble d'opérateurs constitué d'un opérateur de typage appelé type inattendu pour ne sélectionner que les règles d'association dont le type n'est pas conforme à celui du schéma de règles.

L'opérateur k-conforme : L'opérateur conforme proposé par [B.Liu et al. 1999] permet de filtrer parmi l'ensemble des règles d'association celles qui sont conformes à un schéma de règles en combinaison avec d'autres items. En d'autres termes si le nombre des items est important le résultat de cet opérateur est un ensemble important de règles d'association dont l'exploitation reste difficile par l'utilisateur. Pour cette raison nous proposons l'opérateur k-confirme pour contrôler la taille des items dans les règles d'association. Cet opérateur permet d'extraire parmi l'ensemble des règles d'association celles qui satisfont le schéma de règles en combinaison avec k items. Par conséquent, l'utilisateur aura la possibilité de contrôler la taille des items et le nombre de règles d'association, ce qui facilitera leur analyse par les utilisateurs.

L'opérateur k-objectif : permet de ne filtrer que les règles d'association qui ont la partie conclusion conforme au schéma de règles en combinaison au maximum avec k-items distincts. Dans cet opérateur l'utilisateur s'intéresse beaucoup plus à la partie conclusion qu'à la partie entête.

L'opérateur k-non objectif : permet de ne filtrer que les règles d'association qui ont la partie entête conforme au schéma de règles en combinaison au maximum avec k-items distincts. Dans cet opérateur l'utilisateur s'intéresse beaucoup plus à la partie entête qu'à la partie conclusion.

L'opérateur type inattendu : permet d'extraire parmi l'ensemble des règles d'association celles qui ne sont pas conformes du point de vu type au schéma de règles.

En résumé, la technique de typage des règles d'association consiste à comparer les types de règles d'association avec le schéma de règles. Par contre, la technique de sélection des règles d'association est basée sur l'idée de comparer les règles d'association avec le schéma de règles.

Définition 4.3

Soit un concept d'ontologie C associé à un ensemble d'items I de la base de données :

$$F(C)=\{y_1,\ldots,y_n\}$$

Avec $\{y_1,\ldots,y_n\} \in I$ et un itemset

$$X=\{x_1,\ldots,x_n\}$$

Nous pouvons dire que l'itemset X est conforme au concept C, si Conf (X,C)=VRAI, avec :

$$conf(X,C) = \begin{cases} VRAI & si\ \exists yi, yi \in X\ et\ yi \in F(C) \\ FAUX & Sinon \end{cases}$$

En d'autres termes, un itemset est conforme à un concept d'ontologie si ce dernier est associé avec au moins un item de l'itemset. Dans ce qui suit nous dérivons de façon formelle les quatre opérateurs.

3.3.1 L'opérateur k-Conforme

Appliqué au schéma de règles (SR), l'opérateur k-conforme noté k-C(SR), confirme une implication ou découvre une implication entre différents concepts. Pour une règle d'association sélectionnée par l'opérateur k-conforme sur un schéma de règles, l'ensemble suivant de conditions doit être satisfait en fonction des différents types de schéma de règles.

- Dans le cas du schéma de règles non implicatif, une règle d'association est k-conforme si l'itemset créé avec l'union des itemsets antécédent et conclusion de la règle d'association est conforme à chaque concept qui compose la règle en combinaison avec un maximum de k autres items.

  Soit la règle d'association globale suivante :

  $$A \rightarrow B$$

  Avec A et B des itemsets et le schéma de règles multi-niveaux suivant :

  $$SR(< M > <T><N>)$$

  Avec

  $$M = \{C_1, \ldots, C_k\}\ et\ T = \{L, G, M, E\}\ \text{et N} > 1$$

  La règle d'association globale est sélectionnée avec l'opérateur de conformité, ou en d'autres termes, la règle d'association globale est conforme au schéma de règles si :

$$\forall C_i \in M, conf(A \cup B, C_i) = VRAI$$
et
$$|M| + k \geq |A \cup B| \ et \ T = G$$

- Dans le cas du schéma de règles implicatif, une règle d'association est k-conforme au schéma de règles si les itemsets antécédent et conclusion de la règle d'association sont conformes à chaque concept antécédent et conclusion du schéma de règles respectivement avec la combinaison de k autres items.

  Soit la règle d'association globale suivante :

  $$A \rightarrow B$$

  Avec A et B des itemsets et le schéma de règles :

  $$SR(< M_A \rightarrow M_B > <\text{T}><\text{N}>)$$

  Avec

  $$M_A = \{C_1, \dots, C_k\} \ et \ M_B = \{C'_1, \dots, C'_{k'}\} \ et \ T = \{L, G, M, E\} \ et \ N > 1$$

  La règle d'association globale est sélectionnée avec l'opérateur k-conforme, ou en d'autres termes, la règle d'association globale est conforme au schéma de règles si :

  $$\forall C_i \in M_A, conf(A, C_i) = VRAI$$
  et
  $$\forall C'_{i'} \in M_B, conf(B, C'_{i'}) = VRAI$$
  et
  $$|M_A + M_B| + k \geq |A \cup B| \text{ et T=G}$$

3.3.2 L'opérateur k-Objectif

Cet opérateur noté par k-O(SR), filtre l'ensemble des règles conforme avec la partie conclusion du schéma de règles. Etant donné un schéma de règles, une règle d'association est k-Objectif si l'itemset conclusion de la règle d'association est conforme aux concepts conclusion du schéma de règles

Pour formaliser cette définition, soit la règle d'association Exceptionnelle suivante :

$$A \rightarrow B$$

Avec A et B des itemsets et le schéma de règles :

$$RS(< M_A \rightarrow M_B >><T><2>)$$

Avec

$$M_A = \{C_1, \dots, C_k\}\ et\ M_B = \{C'_1, \dots, C'_{k'}\}\ et\ \ T = \{L, G, M, E\}\ et\ N > 1$$

La règle d'association est sélectionnée avec l'opérateur k-Objectif, ou en d'autres termes, la règle d'association est conforme aux concepts conclusion du schéma de règles en combinaison avec k-items si :

$$\forall C'_{i'} \in M_B, conf(B, C'_{i'}) = VRAI$$
et
$$|M_B| + k \geq |B| \text{ et } T = E$$

3.3.3 L'opérateur k-Non Objectif

Cet opérateur noté par k-NO(SR), filtre l'ensemble de règles qui ont la partie entête conforme au schéma de règles. Ce type de règles intéresse l'utilisateur plus que celles de type k-conforme. Généralement, les décideurs cherchent à découvrir de nouvelles connaissances selon leurs connaissances à priori.

Pour formaliser cette définition, soit la règle d'association majoritaire suivante :

$$A \rightarrow B$$

Avec A et B des itemsets et le schéma de règles suivant :

$$SR(< M_A \rightarrow M_B > <T><N>)$$

Avec

$$M_A = \{C_1, \dots, C_k\}\ et\ M_B = \{C'_1, \dots, C'_{k'}\}\ et\ T = \{L, G, M, E\}\ et\ N > 1.$$

La règle d'association est sélectionnée avec l'opérateur k-Non Objectif, ou en d'autres termes, la règle d'association est conforme aux concepts antécédent du schéma de la règle en combinaison avec k-items si :

$$\forall C_i \in M_A, conf(A, C_i) = VRAI$$
et
$$|M_A| + k \geq |A|\ \ et\ T = M$$

3.3.4 L'opérateur Type Inattendu

Etant donné un schéma de règles, une règle d'association est de type inattendu noté par *TI(SR)*, si la règle d'association est conforme au schéma de règles et le type du schéma de règles est différent de celui de la règle. En d'autres termes, les règles d'association générées par ce type d'opérateurs sont des règles d'association de types différents à celui du schéma de règles.

Pour formaliser cette définition, soit la règle d'association Exceptionnelle suivante :

$$A \rightarrow B$$

Avec A et B des itemsets et le schéma de règles suivant:

$$SR(< M_A \rightarrow M_B >><T><N>)$$

Avec

$$M_A = \{C_1, \dots, C_k\}\ et\ M_B = \{C'_1, \dots, C'_{k'}\}, et\ T = \{L, G, M, E\}\ et\ N > 1$$

La règle d'association est sélectionnée avec l'opérateur type inattendu, ou en d'autres termes, la règle d'association est conforme aux concepts antécédent et conclusion du schéma de règles et de type différent par rapport au schéma de règles si :

$$\forall C_i \in M_A, conf(A, C_i) = VRAI$$
$$et$$
$$\forall C'_{i'} \in M_B, conf(B, C'_{i'}) = VRAI$$
$$et$$
$$T \neq E$$

4 *RAMARO* Intra-Site

Dans cette phase, nous proposons d'intégrer les connaissances de l'utilisateur dans le processus de la fouille de données locale. L'intérêt principal est de ne se concentrer que sur les règles qui intéressent l'utilisateur sans la nécessité d'extraire toutes les règles d'association.

Par exemple, soit deux utilisateurs de niveaux 1 et 2 qui veulent des informations sur les règles d'association qui contiennent l'item *Pompe* et *compresseur* respectivement. Au niveau local c.à.d. niveau 1 l'algorithme d'extraction des règles d'association doit prendre en considération l'item *pompe* et aussi l'item *compresseur*. Car l'item *compresseur* est intéressant pour l'utilisateur

de niveau supérieur c.à.d. niveau 2. Donc il est nécessaire de construire un ensemble d'items sélectionnés à partir des différents schémas de règles des niveaux supérieurs.

RAMARO intra-site est composé de deux principales phases comme mentionné dans la figure 4.4.

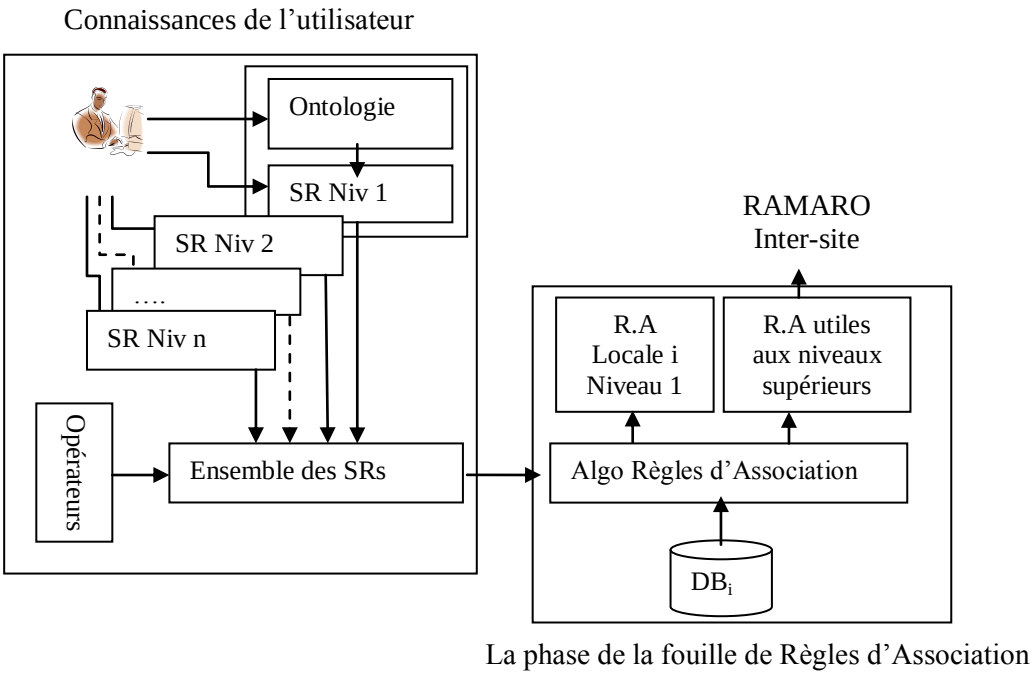


FIG. 4.4 – TRAITEMENT INTRA-SITE DU SITE I

La première partie consolide les connaissances des utilisateurs à divers niveaux. Dans cette partie, l'ensemble des schémas de règles des niveaux supérieurs sont prises en considération pour la fouille de données locale. D'un autre côté, nous considérons que les utilisateurs à divers niveaux disposent des connaissances concernant les règles d'association découvertes.

La seconde partie procède dans la découverte des règles d'association. Son objectif principal est de proposer à l'utilisateur de niveau local un ensemble réduit de règles d'association et de préparer les autres règles d'association aux niveaux supérieurs. Le processus *RAMARO* intra-site est présenté dans la figure 4.5 dont l'objectif est de :

Guider l'utilisateur dans la phase d'extraction des règles d'association : L'utilisateur pourra réviser ses attentes et ses actions.

Restreindre l'espace de recherche de l'algorithme de découvertes de règles d'association : La découverte des règles d'association est autour des items sélectionnés.

Minimiser le nombre des itemsets candidats : Les itemsets candidats dont l'utilisateur juge non intéressant sont retirés du processus de découverte des règles d'association.

Déterminer le support exact de règles non localement fréquentes : c'est sans doute l'objectif principal de ce processus pour résoudre le problème de la perte de connaissances évoqué dans l'état de l'art. En effet, les supports des règles non localement fréquentes sont calculés de façon précise pour les synthétiser aux niveaux supérieurs, ce qui va résoudre le problème de la perte de connaissances évoqué dans l'état de l'art.

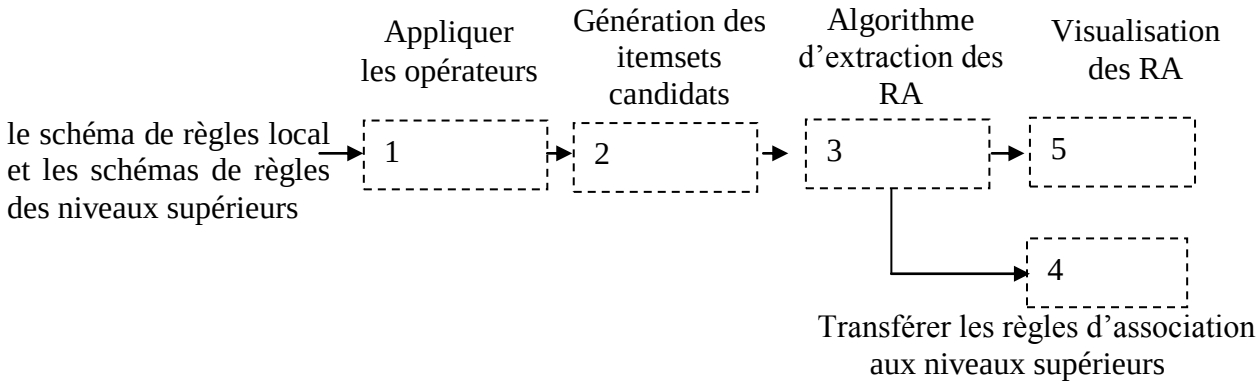


FIG. 4.5 – PROCESSUS INTRA-SITE

Les différentes étapes du processus *RAMARO* intra-site sont détaillées dans ce qui suit :

1. Appliquer les opérateurs : Dans cette phase un ensemble d'opérateurs est appliqué sur tous les schémas de règles de niveaux supérieurs y compris celui du niveau local. Dans *RAMARO* intra-site, nous n'utilisons que les opérateurs de sélection.

2. Génération des itemsets candidats : Dans cette étape, l'ensemble des itemsets candidats est généré. Cet ensemble contient les itemsets locaux et ceux qui vont être utilisés dans des niveaux supérieurs.

3. Algorithme d'extraction des RA : Dans cette étape, les supports des itemsets candidats vont être calculés à partir de la base de données.

4. Transférer les règles d'association aux niveaux supérieurs : Cette étape est très importante car elle prépare la phase *RAMARO* inter-site. Effectivement ces règles d'association seront utilisées pour classifier les différents types de règles d'association.

5. Visualiser les Règles d'association : Cette phase est très importante car elle propose à l'utilisateur le résultat de la découverte des règles d'association.

Dans ce qui suit nous allons détailler les phases de génération des itemsets candidats et l'extraction des règles d'association.

4.1 Génération des itemsets candidats et extraction des règles d'association

Dans cette approche, la recherche des règles d'association est effectuée localement, guidée par les schémas de règles et les opérateurs des différents niveaux. Au lieu de générer l'ensemble des règles d'association ensuite les filtrer selon la conformité des règles d'association de l'utilisateur, notre approche procède en premier lieu par génération localement des itemsets candidats en se basant sur le schéma de règles ensuite vérification de leur support et confiance à partir de la base de données. Les itemsets candidats sont tous les itemsets possibles conformes au schéma de règles et opérateurs. Après leur régénération, un parcours vers la base de données est effectué pour calculer leur support et confiance. Les règles d'association satisfaisant les seuils du support et de confiance et le schéma de règles local seront présentées à l'utilisateur et celles qui satisfont les schémas de règles des niveaux supérieurs seront transférées aux niveaux supérieurs. En effet, la génération des itemsets candidats s'effectue selon les schémas de règles et les opérateurs appliqués à ces schémas de règles. On distingue un algorithme par opérateur appliqué au schéma de règles. Ci-dessous les trois algorithmes de génération des itemsets candidats et les règles d'association locales selon le type d'opérateur utilisé (k-Conforme, k-Objectif et K-non Objectif).

4.1.1 L'opérateur k-conforme

Cet opérateur peut être appliqué à un schéma de règles implicatif et non implicatif. Dans le cas d'un schéma de règles non implicatif, la génération des règles d'association candidates consiste en toutes les possibilités qui existent entre les items des concepts présents dans le schéma de règles. Par contre dans le cas d'un schéma de règles implicatif la génération des règles d'association candidates consiste en toutes les possibilités qui existent entre les items des concepts présents dans le schéma de règles en respectant la partie antécédent et conséquence.

4.1.1.1 Schéma de règles non implicatif

L'application de l'opérateur k-conforme dans un schéma de règles non implicatif consiste à générer toutes les possibilités qui existent entre les items des concepts présents dans le schéma de règles dont la taille maximale des items est k. L'utilisateur sait qu'il existe ces itemsets quelque part dans la règle d'association soit dans la partie antécédent, soit dans la partie conséquence.

4.1.1.2 Schéma de règles implicatif

La génération des règles d'association k-conforme au schéma de règles et aux opérateurs se fait en un seul parcourt de la base de données. Les itemsets candidats sont générés d'abord suivant les schémas de règles et les opérateurs. Ensuite, un parcourt de la base de données est effectué pour calculer les supports et les confiances. Seules les règles d'association dont la confiance satisfait le seuil de confiance sont visualisées à l'utilisateur et les autres qui sont intéressantes pour les utilisateurs de niveau supérieur seront transférées vers les niveaux supérieurs.

Le pseudo code de l'opérateur k-confirme sur un schéma de règles implicatif est présenté dans l'algorithme 4.1. Premièrement, les entrées de l'algorithme sont : Les schémas de règles de tous les niveaux représentés par *SR[N]* décrit par la partie Antécédent *Ant* et la partie conséquence *Con*, le degré de généralisation *k*, l'ensemble des items *I* et le support et la confiance minimum *minsup, minconf*. Les sorties sont la liste des règles d'association conformes au schéma de règles local *rulelist* et la liste des règles d'association conformes aux schémas de règles des niveaux supérieurs *rulelistsup*.

L'idée de l'algorithme est de prendre des sous ensembles de la liste du reste des items sans les items Antécédent et Conséquence du schéma de règles et de les ajouter dans les deux cotés de la règle d'association pour créer la liste des itemsets candidats (lignes 2 au 7) et les règles d'association candidates (lignes 2,3,4,5,8), de vérifier ensuite si ces règles d'association candidates sont intéressantes sur la base des deux mesures *minsup, minconf* (lignes 11,12), de calculer enfin les supports et les confiances des règles de la liste des règles qui seront utilisées dans les niveaux supérieurs (lignes 10).

ALG. 4.1 Algorithme de k-confirme d'un schéma de règles implicatif

Entrées : l'ensemble des schémas de règles *SR* par niveaux décrit par la partie *Ant* antécédent, *Cons* conséquence, *N* niveau, k degrés de généralisation, *minsup*, *minconf* et *I* l'ensemble des items,.

Sorties : *rulelistsup* la liste des règles avec leurs supports et confiances par niveaux qui seront utilisés dans les niveaux supérieurs

rulelist la liste des règles d'association locales conformes au schéma de règles local.

1. *rulelist* = ϕ, *pour i allant de 2.. N faire rulelistsup [i]*=ϕ
2. Soit l'ensemble des items qui ne sont pas dans la partie antécédent et conséquence *RQ* = {*I-Ant-Cons*} et |*RQ*|=*k*
3. Pour chaque schéma de règles *SR[N]*
4. Pour chaque partie *Pd* dans *SR[N]* {*Ant*, *Cons*}
5. Pour chaque sous ensemble *SE* de *RQ* faire
6. *Si* N≠1 *Alors /*les règles des niveaux supérieurs*/*
7. Ajouter à *rulelistsup* une nouvelle règle candidate *RCsup[N]* (*Ant* U *SE* → *Cons* U (*RQ-SE*))
8. *Sinon /*Règles d'association du niveau local*/* Ajouter à *rulelist* une nouvelle règle candidate *RC* (*Ant* U *SE* → *Cons* U (*RQ-SE*))
9. Pour chaque règle candidate *RCsup*$\in$ *rulelistsup* et la règle candidate *RC*$\in$ *rulelist*
10. Calculer les supports *s* et les confiances *c* des règles dans *RCsup*
11. Vérifier le support *s* et la confiance *c* pour *RC*
12. Supprimer à partir de la liste *rulelist* toutes les règles dont *s*<*minsup*
13. Return *rulelist* et *rulelistsup*

4.1.2 L'opérateur k-Non Objectif

Le pseudo code de l'opérateur k-non Objectif sur un schéma de règles implicatif est présenté dans l'algorithme 4.2. Premièrement, les entrées de l'algorithme sont : Les schémas de règles de tous les niveaux représentés par *SR[N]* décrit par la partie Antécédent *Ant* et la partie conséquence *Con*, le degré de généralisation *k*, l'ensemble des items *I* et le support et la confiance minimum *minsup, minconf*. Les sorties sont la liste des règles d'association conformes au schéma de règles locale *rulelist* et la liste des règles conformes aux schémas de règles des niveaux supérieurs *rulelistsup*. L'idée de l'algorithme est de prendre tous les sous ensembles de la liste des items de taille *k* sans les items Antécédent du schéma de règles et de les ajouter dans le côté antécédent du schéma de règle pour créer la liste des itemsets candidats (lignes 2 au 7) et les règles d'association candidates (lignes 2,3,4,5,8). Ensuite vérifier si ces règles d'association candidates sont intéressantes sur la base des deux mesures *minsup, minconf* (lignes 11,12), et

calculer les supports et les confiances des règles de la liste règles qui seront utilisés dans les niveaux supérieurs (lignes 10).

ALG. 4.2 Algorithme k-Non Objectif d'un schéma de règles implicatif

Entrées : l'ensemble des schémas de règles *SR* par niveaux décrit par la partie *Ant* antécédent, *Cons* conséquence, *N* niveau, k degrés de généralisation, *minsup, minconf* et *I* l'ensemble des items

Sorties : *rulelistsup* la liste des règles avec leurs supports et confiances par niveaux qui seront utilisés dans les niveaux supérieurs

rulelist la liste des règles d'association locales dont l'antécédent est conforme au schéma de règles local.

1. *rulelist = ϕ, pour i allant de 2.. N faire rulelistsup [i]=ϕ*
2. Soit l'ensemble des items qui ne sont pas dans la partie antécédent *RQ* = {*I-Ant*} et |*RQ*|=k
3. Pour chaque schéma de règles *SR[N]*
4. Pour chaque partie *Pd* dans *SR[N]* {*Ant*}
5. Pour chaque sous ensemble *SE* de *I-RQ-Ant* faire
6. *Si* N≠1 *Alors /*les règles des niveaux supérieurs*/*
   Ajouter à *rulelistsup* une nouvelle règle candidate
   RCsup[N] (*Ant U RQ* → *SE*)
7. *Sinon /*Règles d'association du niveau local*/*
   Ajouter à *rulelist* une nouvelle règle candidate
   RC (*Ant U RQ* → *SE*)
8. Pour chaque règle candidate *RCsup* ∈ *rulelistsup* et la règle candidate *RC* ∈ *rulelist*
9. Calculer les supports *s et les confiances c* des règles dans *RCsup*
10. Vérifier le support *s* et la confiance *c* pour *RC*
11. Supprimer à partir de la liste *rulelist* toutes les règles dont *s*<*minsup*
12. Return *rulelist* et *rulelistsup*

4.1.3 L'opérateur k-Objectif

Le pseudo code de l'opérateur k-Objectif sur un schéma de règles implicatif est présenté dans l'algorithme 4.3. Premièrement, les entrées de l'algorithme sont : Les schémas de règles de tous les niveaux représentés par *SR[N]* décrit par la partie Antécédent *Ant* et la partie conséquence *Cons*, le degré de généralisation *k*, l'ensemble des items *I* et le support et la confiance minimum *minsup, minconf*. Les sorties sont la liste des règles d'association conformes au schéma de règles locale *rulelist* et la liste des règles conformes aux schémas de règles des niveaux supérieurs *rulelistsup*. L'idée de l'algorithme est de prendre des sous ensembles de la liste des items de taille *k* sans les items conséquence du schéma de règles et de les ajouter dans la partie conséquence du schéma de règle pour créer la liste des itemsets candidats (lignes 2 au 7) et les règles d'association candidates (lignes

2,3,4,5,8). Ensuite vérifier si ces règles d'association candidates sont intéressantes sur la base des deux mesures *minsup, minconf* (lignes 11,12), et calculer les supports et les confiances des règles de la liste des règles qui seront utilisées dans les niveaux supérieurs (lignes 10).

ALG. 4.3 Algorithme k-Objectif d'un schéma de règles implicatif

Entrées : l'ensemble des schémas de règles *SR* par niveaux décrit par la partie *Ant* antécédent, *Cons* conséquence, *N* niveau, k degrés de généralisation, *minsup*, *minconf* et *I* l'ensemble des items

Sorties : *rulelistsup* la liste des règles avec leurs supports et confiances par niveaux qui seront utilisés dans les niveaux supérieurs

rulelist la liste des règles d'association locales dont la conclusion est conforme au schéma de règles local.

1. *rulelist* = ϕ, *pour i allant de 2.. N faire rulelistsup[i]=*ϕ
2. Soit l'ensemble des items qui ne sont pas dans la partie conséquence *RQ* = {*I-Cons*} et |*RQ*|=k
3. Pour chaque schéma de règles *SR[N]*
4. Pour chaque partie *Pd* dans *SR[N]* {*Cons*}
5. Pour chaque sous ensemble *SE* de *I-RQ-Cons* faire
6. *Si* N≠1 *Alors /*les règles des niveaux supérieurs*/*
7. Ajouter à *rulelistsup* une nouvelle règle candidate *Rcsup[N]* (*SE* → (*Cons U RQ*))
8. *Sinon /*Règles d'association du niveau local*/* Ajouter à *rulelist* une nouvelle règle candidate *RC* (*SE* → (*Cons U RQ*))
9. Pour chaque règle candidate *RCsup* ∈ *rulelist* et la règle candidate *RC* ∈ *rulelist*
10. Calculer les supports *s* et les confiances *c* des règles dans *RCsup*
11. Vérifier le support *s* et la confiance *c* pour *RC*
12. Supprimer à partir de la liste *rulelist* toutes les règles dont *s<minsup*
13. Return *rulelist* et *rulelistsup*

5 *RAMARO* Inter-site

RAMARO inter-site procède dans la phase inter-site dans le processus de la *FMBD*. Dans ce contexte, il est composé de deux principales parties comme mentionné dans la figure 4.6. Dans la première partie nous nous concentrons sur les connaissances de l'utilisateur. Cette partie concerne les connaissances de base qui permettent de formaliser les connaissances de l'utilisateur qui sont les connaissances du domaine, les attentes de l'utilisateur et les actions sur les attentes de l'utilisateur.

Premièrement nous avons intégré les connaissances de l'expert du domaine, à travers les ontologies, pour donner une description complète du domaine.

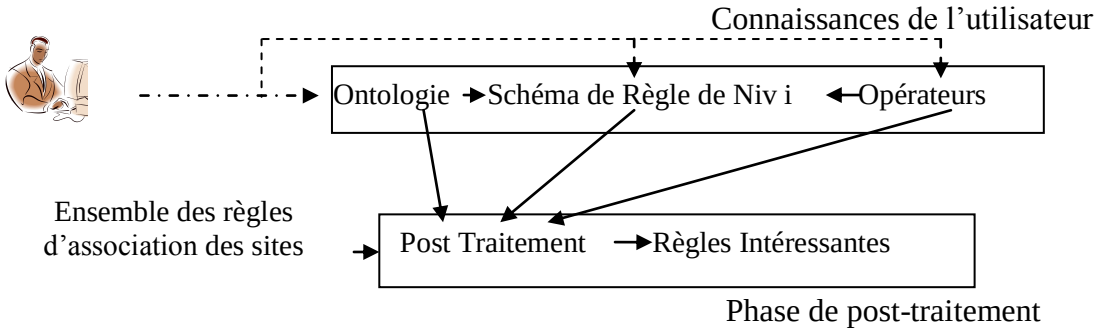


FIG. 4.6 – DESCRIPTION DU PROCESSUS INTER-SITE DE NIVEAU I

Ensuite nous intégrons les attentes des utilisateurs à différentes niveaux d'abstraction. En effet l'utilisateur doit disposer de quelques informations concernant les règles découvertes. Finalement nous offrons à l'utilisateur la possibilité d'appliquer différents opérateurs sur ses attentes.

La seconde partie de notre approche procède dans la tâche du post-traitement. Elle consiste à analyser l'ensemble des règles d'association avec l'utilisation de différentes mesures intéressantes basées sur les schémas de règles avec les opérateurs. Dans cette partie, les règles d'association sont issues de divers sites, d'où leur très grand nombre. L'application d'opérateurs dans cet ensemble doit être réalisée sur des classes de type de règle d'association. Pour cela, nous proposons de classifier d'abord les règles d'association ensuite appliquer les opérateurs sur ces classes.

La figure 4.7 illustre le processus inter-site composé de 3 étapes :

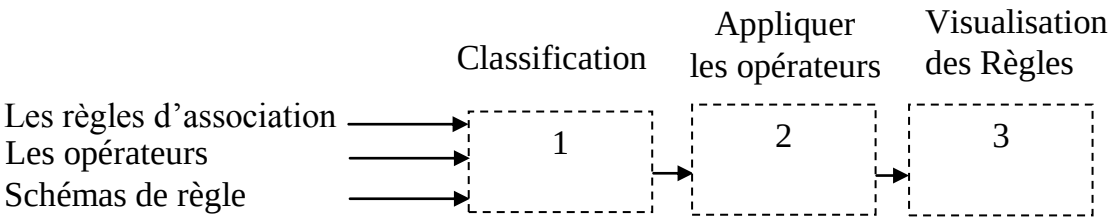


FIG. 4.7 – PROCESSUS INTER-SITE

1. **Classification :** Cette phase se base sur le schéma de règles et les opérateurs pour classifier à partir de l'ensemble des règles d'association de l'ensemble des sites, les règles d'association globales, majoritaires et exceptionnelles.
2. **Application des opérateurs :** Dans cette phase l'ensemble d'opérateurs définis est appliqué sur les classes de règles d'association pour ne faire ressortir que les règles d'association intéressantes.

3. **Visualisation des règles d'association :** C'est la phase de restitution visuelle des règles d'association de façon souple et flexible.

L'opération de classification est guidée aussi par les schémas de règles et les opérateurs pour réduire l'espace de recherche. En effet, si les utilisateurs des différents niveaux ne s'intéressent qu'aux règles d'association globales alors la construction des autres ensembles majoritaires et exceptionnels est inutile. Dans ce qui suit et avant de définir les optimisations sur la construction des classes, nous définissons les algorithmes de construction de ces trois classes : – Majoritaires – Exceptionnelles et Globales.

5.1. Classification des règles d'association

La figure 4.8 présente le processus de classification des règles d'association. Les règles générées à partir des différents sites sont regroupées dans un niveau de la hiérarchie pour être segmentées selon le schéma de règles et les opérateurs. Comme finalité de ce processus nous avons en sortie au plus trois ensembles : un ensemble pour les règles globales, un deuxième ensemble pour les règles majoritaires et le dernier ensemble pour les règles exceptionnelles. Les algorithmes qui permettent de construire ces ensembles sont définis dans les sections suivantes.

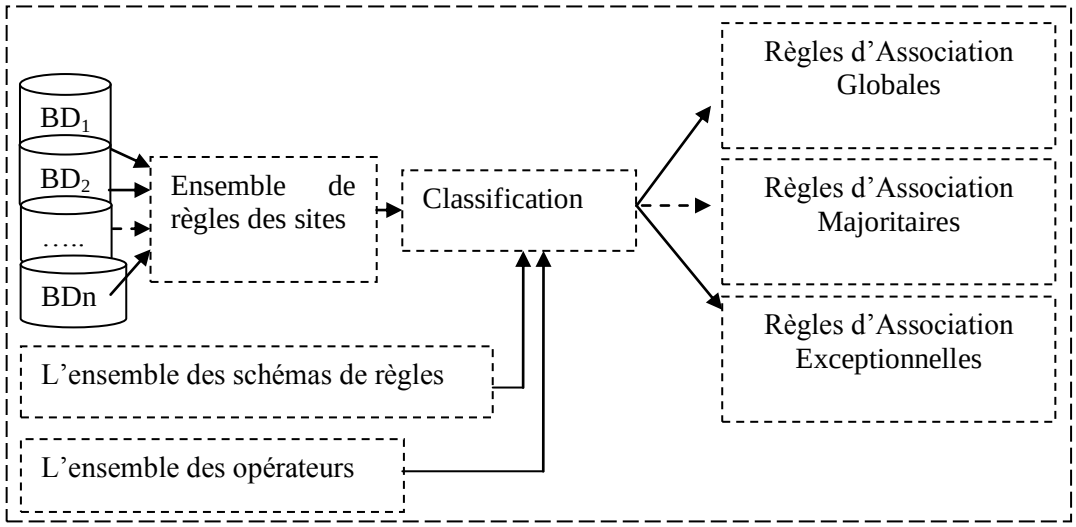


FIG. 4.8 – DESCRIPTION DU PROCESSUS DE CLASSIFICATION

5.1.1 Ensemble des motifs Majoritaires

Les motifs majoritaires décrivent la distribution des motifs locaux dans les différentes branches de la compagnie et reflètent les ressemblances des branches.

Dans cette section, nous allons présenter la technique inspirée des travaux de [C.Zhang et al. 2005] pour construire et identifier ce type de motifs à partir des motifs locaux.

Soient D_1, D_2,…,D_m des bases de données de m sites et RL_i l'ensemble des règles d'association de D_i avec i=1…m et :

$$RL = \{r_j \,/\, r_j \in RL_1 \cup RL_2 \cup \ldots \cup RL_m \; 1 \leq j \leq n\}$$

Avec $n = |RL_1 \cup RL_2 \cup \ldots \cup RL_m|$

$$\text{Soit } a_{ij} = \begin{cases} 1 & si\ r_i\ est\ presente\ dans\ le\ site\ j \\ 0 & sinon \end{cases}$$

Et soit $vote_i$ le nombre de sites qui votent pour la règle r_i.

Et $VoteMoyen\ (VM) = \frac{vote(r_1)+vote(r_2)+\cdots+vote(r_n)}{n}$

Avec $vote(r_i) = vote_i/m$, qui est le taux de vote de la règle r_i.

$Si\ vote(r_i) > VM$ r_i peut être considérée comme règle d'association majoritaire.

Cependant $vote(r_i) - VM$ satisfait :

$$0 < vote(r_i) - VM \leq 1 - VM$$

En particulier

$$0 < \frac{vote(r_i) - VM}{1 - VM} \leq 1$$

Cependant la mesure d'intérêt $LPI(r_i)$ de la règle d'association r_i est définie suivant la déviation du taux de vote $vote(r_i)$ à partir du vote moyen VM et :

$$LPI(r_i) = \frac{vote(r_i) - VM}{1 - VM}$$

Le taux de vote est important si $LPI(r_i) = 1$.

Une règle d'association est intéressante et constitue un vote élevé si $LPI(r_i)$ est supérieur ou égal à un paramètre qui est le degré minimum de vote $minVR$ donné par l'utilisateur ou un expert.

- Si $vote(r_i) = VM$, r_i est non intéressante et la mesure d'intérêt de la règle r_i est :

$$LPI(r_i) = 0$$

- Si $vote(r_i) - VM > 0$, r_i est très intéressante et la mesure d'intérêt de la règle r_i est :

$$LPI(r_i) = 1$$

- Aussi$si\ vote(r_i) - VM < 0$, r_i est à vote minime. Si $vote(r_i) = 0$ qui est le pire des cas : r_i à un vote minimum.

Le problème de la recherche des règles d'association majoritaires revient à rechercher toutes les règles d'association dont la mesure d'intérêt $LPI(r_i)$ est supérieure ou égale à $minVR$.

Soit le pseudo code suivant :

ALG. 4.4 Algorithme de recherche des règles d'association Majoritaires

Entrées : $D_1,\dots,D_m$, *m* bases de données, l'ensemble de règle locales LR_i ($1 \le i \le m$) de D_i, *minVR* taux de vote minimum

Sorties : *MR* la liste des règles d'association majoritaires.

1. $MR = \emptyset,\ \ LR = \emptyset$
2. Pour i=1 …m
3. $LR = LR \cup LR_i$
4. $n = |LR|$
5. Pour chaque règle d'association r_i de *LR* faire
6. $\text{vote}_i = 0$
7. Pour chaque base de données D_j faire
8. Si $(r_i) \in D_j$ Alors
9. $\text{vote}_i = \text{vote}_i + 1$
10. Pour chaque règle d'association r_i de LR_i faire
11. $\text{vote}(r_i) = \text{vote}_i/m$
12. $VM = \frac{\text{vote}(r_1)+\text{vote}(r_2)+\cdots+\text{vote}(r_n)}{n}$
13. Pour chaque règle d'association r_i de LR_i faire
14. $LPI(r_i) = \frac{\text{vote}(r_i)-VM}{1-VM}$
15. Si $LPI(r_i) \ge minVR$ alors
16. $MR \leftarrow MR \cup \{r_i\}$
17. Retourner MR

5.1.2 Ensemble des motifs Exceptionnels

Les motifs exceptionnels décrivent la distribution des motifs locaux dans les différentes branches et reflètent les divergences des branches.

Dans cette section nous allons présenter la technique inspirée des travaux de [C.Zhang et al. 2005] pour identifier ce type de motifs à partir des motifs locaux.

Soit D_1, D_2,…,D_m des bases de données de m sites d'une compagnie et RL_i l'ensemble des motifs locaux (Règles) de D_i avec i=1…m et :

$$RL = \{r_j \ /\ r_j \in RL_1 \cup RL_2 \cup \dots \cup RL_m\ 1 \le j \le n\}$$

Avec $n = |RL_1 \cup RL_2 \cup \dots \cup RL_m|$

$$\text{Soit } a_{ij} = \begin{cases} 1 & si\ r_i\ est\ presente\ dans\ le\ site\ j \\ 0 & sinon \end{cases}$$

Et soit $vote_i$ le nombre de sites qui votent pour la règle r_i.

Et $VoteMoyen\ (VM) = \frac{vote(r_1)+vote(r_2)+\cdots+vote(r_n)}{n}$

Avec $vote(r_i) = vote_i/m$, qui est le taux de vote de la règle r_i.

$Si\ vote(r_i) < VM$ r_i peut être considérée comme règle d'association exceptionnelle.

Cependant $vote(r_i) - VM$ satisfait :

$$-VM < vote(r_i) - VM < 0$$

En particulier

$$0 < \frac{vote(r_i) - VM}{-VM} \leq 1$$

Plus ce facteur est grand et plus la règle d'association est plus intéressante.

Cependant la mesure d'intérêt $EPI(r_i)$ de la règle d'association r_i est définie suivant la déviation du taux de vote $vote(r_i)$ à partir du vote moyen VM et :

$$EPI(r_i) = \frac{vote(r_i) - VM}{-VM}$$

Avec $VM \neq 0$

Le taux de vote est important si $EPI(r_i) = 1$.

Une règle d'association est intéressante et constitue un vote élevé si $EPI(r_i)$ est supérieur ou égal à un paramètre qui est le degré minimum de vote $minER$ donné par l'utilisateur ou un expert. Le problème de la recherche des règles d'association exceptionnelles revient à rechercher toutes les règles d'association dont la mesure d'intérêt $EPI(r_i)$ est supérieure ou égale à $minER$.

Soit le pseudo code suivant :

ALG. 4.5 Algorithme de recherche des règles d'association Exceptionnelles

Entrées : $D_1,\ldots, D_m$, *m* bases de données, l'ensemble de règle locales LR_i ($1 \leq i \leq m$) de D_i, *minER* taux de vote minimum

Sorties : *ER* la liste des règles d'association exceptionnelles.

1. $ER = \emptyset,\ LR = \emptyset$
2. Pour i=1 …*m*
3. $LR = LR \cup LR_i$
4. $n = |LR|$
5. Pour chaque règle d'association r_i de *LR* faire
6. $vote_i = 0$
7. Pour chaque base de données D_j faire
8. Si (r_i) $\in D_j$ Alors

9. $\text{vote}_i = \text{vote}_i + 1$
10. Pour chaque règle d'association r_i de LR_i faire
11. $\text{vote}(r_i) = \text{vote}_i / m$
12. $VM = \frac{\text{vote}(r_1)+\text{vote}(r_2)+\cdots+\text{vote}(r_n)}{n}$
13. Pour chaque règle d'association r_i de LR_i faire
14. $EPI(r_i) = \frac{\text{vote}(r_i)-VM}{-VM}$
15. Si $EPI(r_i) \geq minER$ alors
16. $ER \leftarrow ER \cup \{r_i\}$
17. Retourner ER

5.1.3 Ensemble des motifs globaux

Les motifs globaux décrivent les motifs fréquents dans l'union des sites.

Soient m sites $D_1, D_2, \ldots, D_m$ avec $W_1', W_2', \ldots, W_m'$ leurs poids respectifs.

- Le poids normalisé d'une base de données D_j : $W_j = \frac{W_j'}{\sum_{j=1}^{m} W_j'}$

- La confiance globale synthétisée de la règle r_i : $Conf_G(r_i) = \sum_{j=1}^{m} W_j \times conf_j(r_i)$

Le problème de la recherche des règles d'association globales revient à rechercher toutes les règles d'association dont la confiance $Conf_G(r_i)$ est supérieure ou égale à $minconf$.

Soit le pseudo code suivant :

ALG. 4.6 Algorithme de recherche des règles d'association Globales

Entrées : $D_1, \ldots, D_m$, m bases de données, l'ensemble des règles locaux LR_i ($1 \leq i \leq m$) de D_i, et W_i' le poid du site i, $conf_j(r_i)$ la confiance de la règle r_i dans le site D_j, *minconf* la confiance minimale.

Sorties : GR_i la liste des règles d'association globales.

1. $GR_i = \emptyset$
2. Pour chaque base de données D_j faire
3. $W_j = \frac{W_j'}{\sum_{j=1}^{m} W_j'}$
4. Pour $i = 1 \ldots m$
5. $LR = LR \cup LR_i$
6. Pour chaque règle r_i de LR faire
7. $Conf_G(r_i) = \sum_{j=1}^{m} W_j \times conf_j(r_i)$
8. Si $Conf_G(r_i) \geq minconf$ alors
9. $GR \leftarrow GR \cup \{r_i\}$
10. Retourner GR

5.2. Optimisations

La construction de ces trois classes est guidée par les schémas de règles et les opérateurs des différents niveaux. Dans cette section, nous proposons deux optimisations pour réduire l'espace de recherche et le temps d'exécution.

5.2.1 Optimisation 1

Soit un ensemble de schémas de règles suivant : SR_1:<$M_a \rightarrow M_b$> <T_1> <N_1>, … , SR_n:<$M_{a'} \rightarrow M_{b'}$> <$T_n$><$N_1$> et un mélange d'ensemble d'opérateurs de k-conforme et k-Objectif et k-Non Objectif choisi par l'utilisateur sur le schéma de règles, l'opérateur type inattendue n'est pas inclus dans cet ensemble, les classes générées par le processus de classification sont de type T_1 ,…, T_n.

Exemple 4.1 : Soit l'ensemble des schémas de règles d'un utilisateur de niveau 2 suivant :

$$SR_1 : SR(< Pompe \rightarrow Compresseur > <M> < 2 >)$$

$$SR_2 : SR(< Pompe \rightarrow Echangeur > <G>) < 2 >$$

Avec l'opérateur 0-conforme sur les deux schémas de règles SR_1 et SR_2. En appliquant l'optimisation 1, les classes générées sont les classes des règles d'association majoritaires et globales. Une simple comparaison entre la partie antécédente *Pompe* et la partie conséquence *Compresseur* du schéma de règles SR_1 avec les règles d'association de l'ensemble majoritaire est effectuée pour confirmer ou rejeter la règle d'association. Il en est de même pour le schéma de règles SR_2, la partie antécédente *Pompe* et la partie conséquence *Echangeur* du schéma de règles SR_2 avec les règles d'association de l'ensemble globale.

5.2.2 Optimisation 2

Soit un ensemble de schémas de règles : SR_1:<$M_a \rightarrow M_b$> <T_1><N_1> , , … , SR_n:<$M_{a'} \rightarrow M_{b'}$> <$T_n$><$N_m$> et un ensemble d'opérateurs de k-conformité, k-Objectif et k-Non Objectif et type-inattendu choisi par l'utilisateur sur le schéma de règles les classes générées par le processus de classification sont les trois types de classes : –majoritaires –exceptionnels – globales.

Exemple 4.2 : Soit l'ensemble de schémas de règles suivant pour l'utilisateur de niveau 2:

$$SR_1 : SR(< Pompe \rightarrow Compresseur > <M> < 2 >)$$

$$SR_2 : SR(< Pompe \rightarrow Echangeur > <G> < 2 >)$$

L'utilisateur s'intéresse sur les règles d'association dont le type est inattendu sur les deux schémas de règles SR_1 et SR_2. Pour cela, nous aurons besoin de construire les trois ensembles : ensembles des règles d'association majoritaires, globales et exceptionnelles. Une simple comparaison entre la partie antécédente *Pompe* et la partie conséquence *Compresseur* du schéma de règles SR_1 avec les règles d'association exceptionnelles et globales est effectuée. L'utilisateur pourra avoir comme résultat que la règle exceptionnelle $Pompe \rightarrow Compresseur$ et non pas majoritaire ce qui constitue une connaissance nouvelle pour lui.

Une comparaison de la partie antécédent *Pompe* et la partie conséquence *Echangeur* du schéma de règles SR_2 avec les règles d'association majoritaires et exceptionnelles est aussi effectuée. L'utilisateur pourra avoir comme résultat que la règle $Pompe \rightarrow Echangeur$ qui est exceptionnelle et non pas globale ce qui constitue une connaissance nouvelle pour lui.

Effectivement, les ensembles générés sont tributaires du contenu du schéma de règles spécialement de la partie type et du contenu des opérateurs. Lorsque les opérateurs choisis ne contiennent pas des types inattendus alors l'ensemble de règles générées sont de type qui existe dans le schéma de règles. Par exemple, l'utilisateur pourra s'intéresser à la conformité des règles majoritaires dont le schéma de règles est :

$$RS(< Pompe \rightarrow Comprésseur > <M> < 2 >)$$

Le groupe construit par le processus de segmentation est l'ensemble de règles majoritaires dont l'opérateur de conformité sera appliqué sur cet sous ensemble. Il n'est pas nécessaire de générer d'autres ensembles de type global ou exceptionnel car ces derniers n'intéressent pas l'utilisateur. Par contre, si l'un des opérateurs choisi est de type inattendu tous les ensembles sont générés car l'utilisateur s'intéresse sur le type de règles d'association ce qui nécessite leur génération. Par exemple, l'utilisateur pourra s'intéressé au type inattendue des règles dont le schéma de règles est :

$$SR(< Pompe \rightarrow Comprésseur > <M> < 2 >)$$

Tous les ensembles sont générés par le processus de classification et l'opérateur de type inattendu sera appliqué sur tous les ensembles. L'utilisateur aura comme

résultat par exemple la règle d'association exceptionnelle $Pompe \rightarrow Comprésseur$.

6 Conclusion

Dans ce chapitre, nous avons présenté notre approche pour extraire les règles d'association intéressantes pour chaque utilisateur suivant ses attentes et besoins. Cette approche procède en deux phases : – La phase de la fouille de données locale – et la phase de la synthétisation des motifs globaux.

L'originalité de cette approche réside dans la supervision du processus d'extraction des règles d'association, dans les deux phases, en utilisant deux modèles conceptuels pour représenter les connaissances des utilisateurs : les ontologies et les schémas de règles multi-niveaux. Ensuite, nous avons appliqué sur ces schémas de règles un ensemble d'opérateurs pour enrichir la représentation des attentes des utilisateurs.

Dans le chapitre suivant nous allons montrer l'utilité et l'apport de *RAMARO* dans le processus de la fouille multi-bases de données multi-niveaux. Et pour illustrer notre contribution, une étude de cas réel est appliquée au domaine de la maintenance de l'entreprise *SH/AVAL*.

Chapitre 5 : Expérimentations et Résultats

- Introduction
- Le système d'information SH/AVL
- Organisation de SH/AVL
- La maintenance Anticipée
- Les demandes de travaux (DT) dans la base de données Maintenance
- Application de RAMARO
- Expérimentations
- Conclusion

Chapitre 5
Expérimentations et résultats

1 Introduction

Ce chapitre est centré sur les expérimentations que nous avons menées pour tester notre approche dans un domaine d'application qui est le domaine industriel. Ces expérimentations sont élaborées sur des bases de données réelles de la maintenance industrielle de *SH/AVL*. Pour l'approche *RAMARO*, les expérimentations se sont effectuées avec une coopération complète avec des experts de la maintenance. Dans la première étape, nous avons établi les besoins et les objectifs des experts du domaine que nous avons suivi. Ensuite, nous avons établi des séances de travail avec des experts de la maintenance pour concevoir la structure de l'ontologie et le développement des schémas de règles et des opérateurs. Ensuite, nous avons décrit un ensemble de cas d'études sur les caractéristiques de l'approche qui sont la réduction du nombre des règles d'association et la qualité des règles d'association générées.

Ce chapitre est structuré comme suit : Les sections 2 et 3 définissent le système d'information *SH/AVL* avec son organisation. Ensuite nous nous sommes intéressés au sous-système d'information qui est la maintenance dans la section 4. La section 5 décrit la structure de la base de données utilisée. La section 6 est une préparation pour les deux études qui vont être menées sur l'algorithme *RAMARO*. La section 7 décrit les résultats de ces deux études dans le domaine de la maintenance.

2 Le système d'information *SH/AVL*

L'étude est basée sur les bases de données de *SH/AVL*, et spécialement les bases de données de la maintenance industrielle.

Le système d'information *SH/AVL* englobe tous les domaines d'activités de l'entreprise (Figure 5.1), allant de la gestion du personnel *GSAO*, de la finance *GFAO*, de la production *GPAO*, de la maintenance *GMAO* jusqu'au logistique *GLAO*. A chaque activité correspond une base de données dont les contenus sont décrits dans ce qui suit :

GSAO : Contient les données relatives au personnel, leurs pointages, leurs formations et séminaires qu'ils ont suivis, leurs mobilités, leurs prestations sociales et paies...

GMAO : Contient des données relatives aux fonctions de Maintenance, Approvisionnements, Inspection, Travaux Neufs et Suivi des Appels d'Offres….

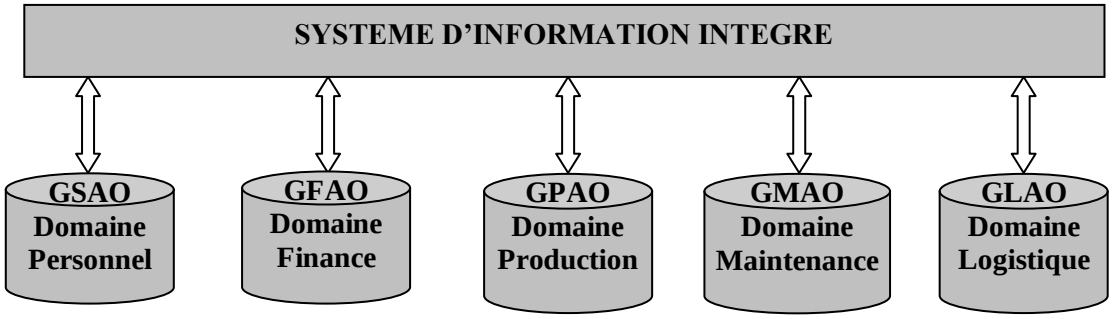


FIG. 5.1 – ORIGINES MULTIPLES DES DONNEES DANS LE SYSTEME D'INFORMATION SH/AVL PAR UNITE.

GPAO : Cette base de données contient tout ce qui concerne la production, les laboratoires, les études, la sécurité et l'intervention pour les complexes de l'activité *SH/AVL*….

GFAO : Dispose de l'ensemble d'informations concernant l'engagement pour la gestion des contrats et des bons de commandes, la facturation, la comptabilité générale et analytique, le stock/investissement, la trésorerie, le budget….

GLAO : Contient des données relatives à la gestion du transport, les moyens généraux (stock et achat), la restauration et la gestion des relations externes (ordre de mission,…)….

Ces bases de données constituent pour *SH/AVL* une plate forme idéale pour la fouille multi-bases de données et par conséquent l'application des techniques de la fouille multi-bases de données pour extraire des connaissances cachées, non triviales, utiles, intéressantes qui peuvent être exploitées à des fins décisionnelles à différents niveaux. La figure 5.2 montre quelques connaissances qu'on peut extraire à partir de ces bases de données.

En résumé, peu de travaux ont été réalisés pour la fouille multi-bases de données. L'intervention de l'expert du domaine pendant toutes les phases du processus de découverte de connaissances est nécessaire afin d'extraire des connaissances intéressantes.

Nous limitons notre étude dans cette thèse au domaine de la maintenance où nous montrons l'intérêt d'utiliser les techniques de la fouille multi-bases de données pour extraire des connaissances utiles afin d'optimiser le coût de la maintenance.

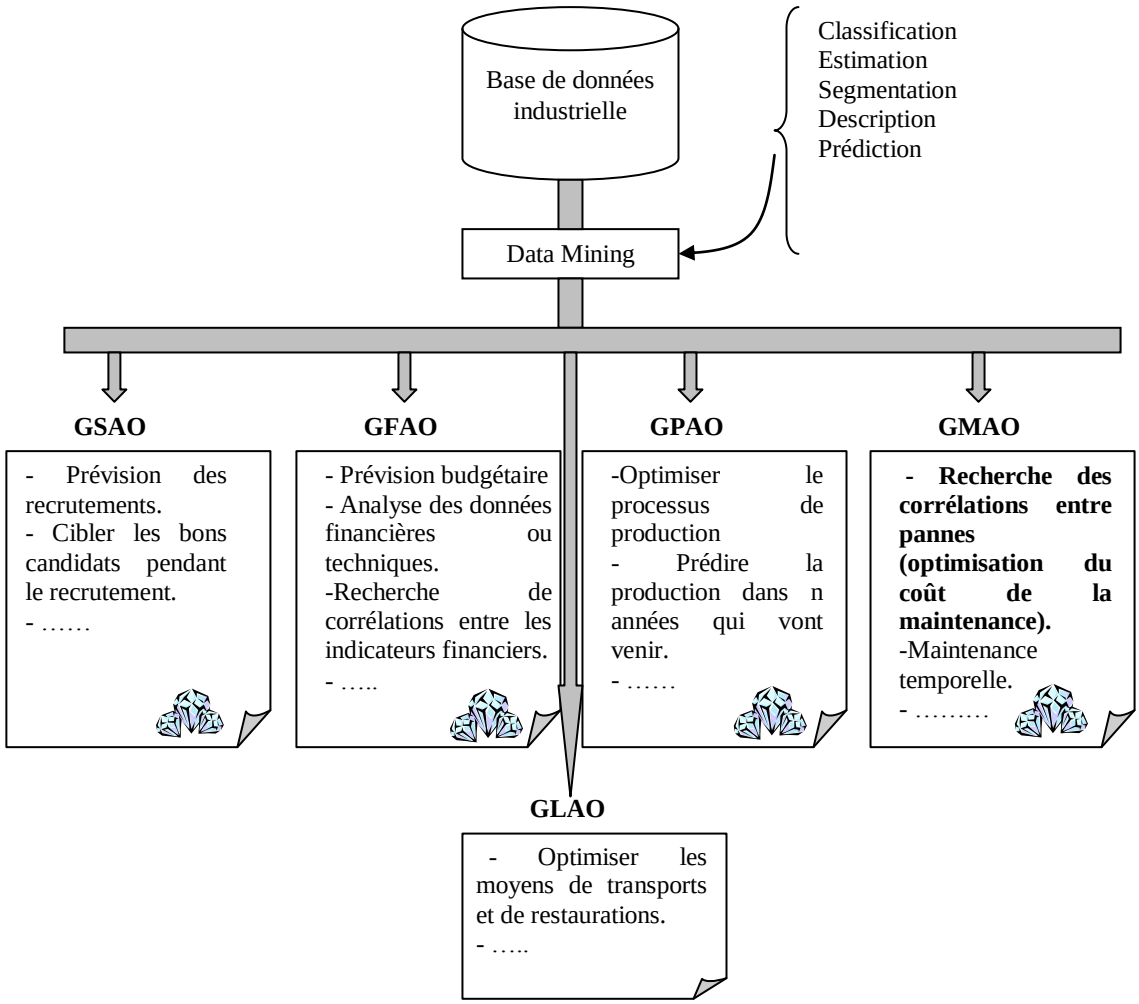


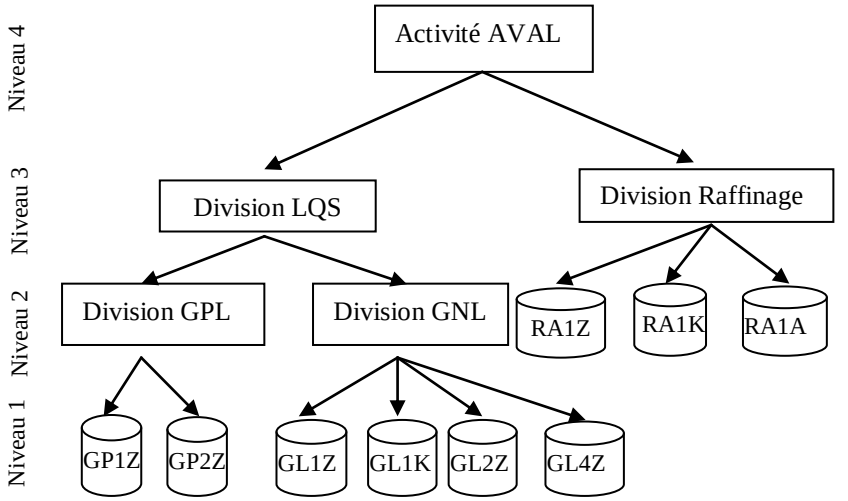
FIG. 5.2 – CHAMPS D'APPLICATION POSSIBLE DANS LE SYSTEME D'INFORMATION *SH/AVL*

3 Organisation de *SH/AVL*

La figure 5.3 décrit la structure de l'organisation de l'activité *AVAL* de La société *SONATRACH*. En effet, *SH/AVL* est une organisation multi-niveaux dont le niveau le plus bas le niveau 1 correspond aux unités opérationnelles contenant des bases de données transactionnelles. Chaque unité dispose de son propre centre de décision qui est la direction de l'unité dont le rôle est d'assurer le bon fonctionnement du complexe pour atteindre les objectifs tracés chaque année. Chaque unité opérationnelle de *SH/AVL*, dispose d'une base de données de chaque

domaine. L'alimentation de ces dernières a commencé depuis 1998 d'où un énorme nuage de données, la taille de chaque base de données peut dépasser les gigas octets.

Le niveau le plus haut qui est le niveau 4 correspond au Vice Président de l'Activité *SH/AVL* qui doit avoir une vue globale sur l'ensemble des installations pétrolières. Ce centre de décision centrale doit veiller au respect des objectifs stratégiques de l'entreprise tracés chaque dizaine d'années. Les niveaux intermédiaires de l'entreprise correspondent aux différentes divisions. Les divisions Liquéfaction et Séparation de Gaz *(LQS)* et Raffinage constituent le niveau 3 et les divisions Gaz Propane Liquéfié (GPL) et Gaz Naturel Liquéfié (GNL), par exemple, constituent le niveau 2 dont l'objectif est d'assurer le bon fonctionnement des unités qui leur sont rattachées. Ces centres de décision intermédiaires doivent veiller au respect des objectifs tactiques de l'entreprise tracés au moyen terme. Tous ces niveaux ont besoin d'un outil d'aide à la décision pour prendre des décisions basées sur la réalité des bases de données. Ils devront avoir une vue globale sur les différentes unités opérationnelles.



LQS : Liquéfaction et Séparation de Gaz
GPL : Gaz Propane Liquéfié
GNL : Gaz Naturel Liquéfié
RA1Z : Raffinerie d'Arzew
RA1K : Raffinerie de Skikda
RA1A : Raffinerie d'Alger

GP1Z : Gaz propane Numéro 1 de Marsa El Hadjaj
GP2Z : Gaz propane Numéro 2 d'Arzew
GL1Z : Gaz Liquéfié Numéro 1 de Bethioua
GL2Z : Gaz Liquéfié Numéro 2 de Bethioua
GL4Z : Gaz Liquéfié Numéro 4 d'Arzew
GL1K : Gaz Liquéfié Numéro 1 de Skikda

FIG. 5.3 – ORGANISATION DE L'ACTIVITE AVAL

4 La maintenance Anticipée

Au niveau de *SH/AVL*, la maintenance des équipements a été de tout temps une préoccupation majeure pour le management. Cette préoccupation s'explique par la complexité des installations en place et les difficultés inhérentes aux opérations de

réapprovisionnement des pièces de rechange commandées aux différents fournisseurs étrangers. Pour cela, la société *SONATRACH* a adopté quatre stratégies de maintenance qui sont : [PM. 1993]

Maintenance accidentelle ou curative (M.CU) : Elle consiste à l'intervention après l'apparition de pannes ou d'anomalies. C'est la forme de maintenance la plus primitive et la plus spontanée.

Maintenance préventive (M.PV): Elle vise à diminuer la probabilité de défaillance d'un système. Pour cela elle s'appuie sur des opérations de remplacement systématique qui consistent à changer suivant un échéancier établi à l'avance de pièces jugées proches de l'usure. Elle est définie lors de la conception par le constructeur.

Maintenance prédictive (M.PD) : Elle consiste à l'application des techniques de mesure sur les équipements (corrosion, fuite, fissures, ancrage, température, pression, débit, des huiles de lubrification et d'étanchéité) en service. Ces techniques permettent de diagnostiquer l'état d'un organe ou d'un équipement afin de juger l'opportunité de lancer l'opération de maintenance préventive ou de la reporter sur des bases rationnelles.

Arrêts programmés (A.P) : Correspond à des formes de maintenance les plus utilisées dans l'industrie des hydrocarbures. Elles sont intimement liées aux exigences réglementaires techniques et sécurité. Elles concernent dans la plupart du temps des équipements dont la maintenance est impossible durant leur fonctionnement.

Bien que ces stratégies ont fait leur preuve dans plusieurs situations, elles ont échoué dans d'autres situations. Ceci est dû à l'inexploitation de la masse de données dans chaque unité opérationnelle. Cette dernière peut contribuer à donner un plus à la politique de la maintenance. Plusieurs questions peuvent avoir une réponse en exploitant les bases de données comme :

- Que peut-on tirer comme connaissance dans ces bases de données ?
- Comment peut-on extraire ces connaissances à partir de ces bases de données ?
- Peut-on trouver des relations entre les pannes des pièces du même équipement, dans le même jour, de la même semaine, …. ?
- Peut-on prévoir les défaillances des pièces, des équipements ?
- Peut-on classifier les pièces ou les équipements par la fréquence de leurs pannes ?
- Peut-on déterminer les pièces et équipements à risque ?

En résumé, à partir de ces bases de données de la maintenance, on peut tirer plusieurs connaissances nouvelles, cachées et utiles pour la maintenance afin d'améliorer le processus de la maintenance, de réduire le coût de la maintenance et d'ajouter d'autres politiques de maintenance à partir de l'historique des données de la maintenance. Dans notre application, on s'intéresse à la découverte des relations entre les pannes des équipements, ce qui va donner naissance à un nouveau concept dans la maintenance, qui est la ***maintenance anticipée***. Ce nouveau concept, peut être intégré comme une nouvelle stratégie de *SH/AVL* dans la maintenance.

Les opérateurs de la maintenance ont besoin d'un outil pour décrire et comprendre les relations entre les données de la maintenance et spécialement les pannes et les défaillances des équipements. Car les pannes des équipements coûtent très chères, le coût d'arrêt non planifié d'un train, par exemple (équipement stratégique pour le procédé) peut dépasser des millions de dollars par mois. La maintenance des installations pétrolières, les systèmes d'armes et des avions sont des exemples dont la défaillance d'un équipement peut être très coûteuse en argent et en vie. La maintenance anticipée est la combinaison entre les bases de données de la maintenance et les règles d'association. Suivant l'historique des pannes des pièces, ou équipements, on peut extraire des relations entre les pannes qui vont ensemble. Ceci permet de réduire énormément le coût de la maintenance. Car si par exemple, chaque fois que le joint X tombe en panne, le compresseur C tombe en panne. Dans ce cas une panne de quelques dinars engendre des pannes de millions de dinars. Pour cela, les opérateurs de la maintenance devront donner l'importance à la pièce X tout en la changeant régulièrement même avant la maintenance préventive.

On peut situer la maintenance anticipée au même niveau que la maintenance curative c.à.d. on doit agir comme s'il y avait une défaillance dans l'organe et même avant la terminaison de sa durée de vie (maintenance préventive).

En résumé, nous proposons une nouvelle politique de maintenance qui est la maintenance anticipée qui s'ajoute aux quatre politiques de maintenance. La maintenance anticipée complète la maintenance traditionnelle en employant les techniques de fouilles de données, pour prévoir les pièces qui sont susceptibles de tomber en pannes. Il est primordial d'utiliser un outil d'extraction de données pour trouver des relations (règles d'association) entre les réparations ou des relations entre les rapports des pannes (Figure 5.4.).

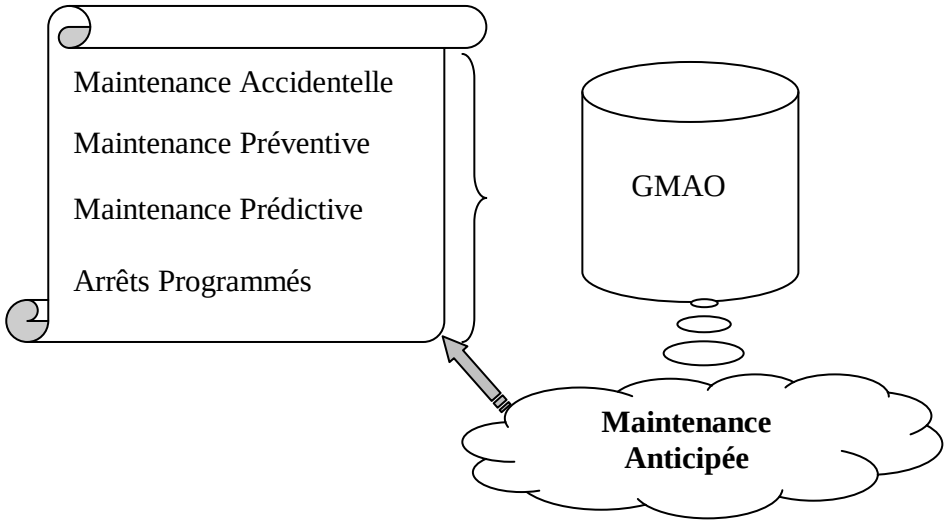


FIG. 5.4 – LA MAINTENANCE ANTICIPEE

Dans l'organisation multi-niveaux et dans le niveau le plus bas qui correspond aux unités de base opérationnelles (GP1Z, GP2Z, GL1Z, GL2Z, GL4Z, …) chacune d'elles contient un historique sur les équipements et leurs manipulations. Ces bases de données alimentées depuis une vingtaine d'années constituent une plate forme idéale pour la fouille multi-bases de données. L'application des techniques de la fouille de données dans chaque Unité de Base Opérationnelle UBO peut générer des connaissances nouvelles qui peuvent aider le centre de décision locale pour l'optimisation du coût de la maintenance. Faisant suite aux connaissances locales le centre de décision local engage des actions correctives ou évolutives pour garantir la chaine de production au niveau local.

Ensuite, ces connaissances locales de différentes UBO seront transmises aux niveaux supérieurs par exemple à l'unité GNL ou GPL pour les analyser et les synthétiser. Le centre de décision de niveau 2 dispose d'une vue globale sur les UBOs qu'il supervise. Il sélectionne les connaissances qui influent sur les UBOs et ceci pour faire ressortir le plan d'action sur l'ensemble des UBOs. Dans ce niveau le centre de décision utilise l'expérience de certains UBOs pour les faire bénéficier aux autres UBOs. Il anticipe les autres UBOs sur les futures pannes par exemple, qui peuvent se présenter et le processus continu aux niveaux supérieurs.

Dans cette optique, notre travail consiste à élaborer un processus basé sur la technologie de la fouille multi-bases de données afin de rechercher les connaissances locales avec agrégation pour chaque niveau. Notre proposition consiste à intégrer les connaissances de l'utilisateur de différents niveaux dans le processus de la fouille multi-bases de données en appliquant la plate forme *RAMARO*.

5 Les demandes de travaux (DT) dans la base de données maintenance

La récolte des pannes des équipements est effectuée principalement à partir de la vue de la demande de travail du système *GMAO*. Nous considérons que chaque demande de travail implique une défaillance de l'équipement. Nous avons construit une vue sur l'ensemble des demandes de travaux dont la structure est présentée dans le tableau 5.1. Celui-ci introduit une simple demande de travail avec la description de chaque élément.

N_DEMANDE	Le numéro de la DT
DATE_ETA	La date d'établissement de la DT
EQUIPEMENT	Le code équipement concerné par la DT
DESC_EQUI	Description de l'équipement
CODE_ARTI	Le code article concerné par la DT
CLASSE_ARTI	La classe d'article concerné par la DT
DESI_ARTI	Description de l'article

TAB. 5.1 – Structure de la vue utilisée

L'exemple dans le tableau 5.2 illustre une demande de travail sur le ventilateur de l'équipement Chaudière de procède type 30-VP-12W-R, le 20 Juillet 1993. Ce qui signifie que le ventilateur avait besoin d'une maintenance.

N_DEMANDE	0000002971
DATE_ETA	20-juil-93
EQUIPEMENT	061-D-321
DESC_EQUI	CHAUDIERE DE PROCEDE TYPE 30-VP-12W-R
CODE_ARTI	121031
CLASSE_ARTI	602
DESI_ARTI	VENTILATEUR

TAB. 5.2 – Exemple d'une demande de DT

Nous limitons notre étude à la division *LQS* pour cause de non disponibilité des données des autres unités opérationnelles. Mais le principe est générique c'est-à-dire que le traitement est le même sur n'importe quel niveau. La figure 5.5 expose l'organisation multi-niveaux utilisée dans les expérimentations.

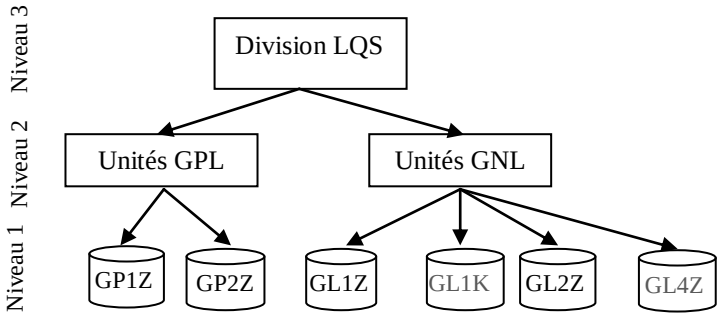


FIG. 5.5 – ORGANISATION DE L'ACTIVITE AVAL DIVISION LQS (SH/AVL/LQS)

Les caractéristiques de ces bases de données sont résumées par le tableau 5.3 qui définit pour chaque base, le nom de la base, le nombre des items et le nombre des enregistrements.

Site	Nombre d'items	Nombre d'enregistrements
GP1Z	932	110500
GP2Z	521	80250
GL1Z	850	95050
GL2Z	920	108600

TAB. 5.3 – Description des bases de données utilisées

6 Application de *RAMARO*

RAMARO est une approche intégrée utilisée au niveau local et par niveaux en fonction des intérêts de l'expert. Avant la description des deux études menées dans le cadre de cette thèse, nous définissons les concepts et la structure de l'ontologie et le schéma de règles multi-niveaux utilisés.

6.1 Concepts des ontologies

Avant de présenter la structure de notre ontologie, nous donnons quelques notions de base sur les ontologies :

Les ontologies sont des outils de représentation de connaissances et de raisonnements sur ces dernières. Elles permettent d'organiser l'ensemble des concepts et des relations entre concepts dans un domaine spécifique.

Formellement, Soit C un ensemble de concepts, T un ensemble de termes, R_c un ensemble de relations entre concepts, R_t un ensemble de relations entre termes.

L'ontologie O est définie par le tuple O = {C, T ,R_c, R_t } tel que :

R_c : C x C est la relation d'ordre partiel sur C définissant la hiérarchie entre les concepts,

R_c (c_1, c_2) signifie c_1 est plus général que c_2. Avec $c_1 \in C$ et $c_2 \in C$

R_t: C$\rightarrow$ T est la fonction d'association d'un terme préféré à un concept.

Pour désigner un concept de l'ontologie, on peut utiliser l'un de ses termes associés. Ce terme sera alors le terme préféré de ce concept.

Les auteurs [Heijst et al. 1997] distinguent quatre types d'ontologies :

Ontologies d'application : Elles contiennent toutes les informations nécessaires pour modéliser les connaissances pour une application particulière.

Ontologies de domaine : Elles fournissent un ensemble de concepts et de relations décrivant les connaissances d'un domaine spécifique.

Ontologies générique (dites aussi de haut niveau) : Elles sont similaires aux ontologies de domaine, mais les concepts qui y sont définis sont plus génériques et décrivent des connaissances telles que l'état, l'action, l'espace et les composants. Généralement, les concepts d'une ontologie de domaine sont des spécialisations des concepts d'une ontologie de haut niveau.

Ontologies de représentation (Ou méta-ontologies) : Elles fournissent des primitives de formalisation pour la représentation des connaissances. Elles sont généralement utilisées pour écrire les ontologies de domaine et les ontologies de haut niveau. Exemples : Frame Ontology [Thomas.G. 1993] et RDF Schema Ontology [McBride. 2004].

Dans notre application, nous nous intéressons aux ontologies du domaine. Nous avons aussi utilisé le langage OWL qui est le standard développé pour représenter les ontologies. La sémantique formelle d'OWL pour une telle ontologie, spécifie comment se déroule la logique de ses conséquences.

6.2 Structure conceptuelle de l'ontologie dans *RAMARO*

L'ontologie est un élément principal dans la plate forme *RAMARO*. Les résultats dépendent fortement de la représentation de l'ontologie. Une représentation qui n'est pas assez développée peut influencer sur la qualité des connaissances générées. L'objectif principal de la construction de l'ontologie est de capturer toutes les connaissances de l'expert du domaine. La structure de l'ontologie finale est composée de deux grandes parties comme illustré dans la figure 5.6.

Pour établir la hiérarchie des classes, nous avons procédé de haut en bas en commençant par les concepts les plus généraux et en terminant par la spécialisation des concepts. Nous avons, donc, commencé par les classes les plus générales, à savoir *: Pétrolier, Equipements_Stratégiques, Equipements_Non_Stratégiques*, etc… Ensuite, nous avons affiné chacune de ces classes. Par exemple, la classe *Pétrolier* a été affinée en deux concepts : *Equipements_Stratégiques, Equipements_Non_Stratégiques* ainsi que la classe *Equipements_Stratégiques* a été spécialisée en sous-concepts : *Condenseur, Turbo-Générateur, Echangeur, Pompe*, … et ainsi de suite pour les autres concepts. « owl:Thing » est une classe prédéfinie. Toute classe OWL est une sous-classe d'*owl:Thing*. Les figures 5.7 et 5.8 sont une représentation graphique (des captures d'écran) de la hiérarchie des classes de notre ontologie, produite à l'aide de l'outil *OWLViz*[2] Protégé.

FIG. 5.6 – LA STRUCTURE DE L'ONTOLOGIE PROPOSEE

[2] OWLViz disponible dans http://mvnrepository.com/artifact/edu.stanford.protege/org.coode.owlviz/4.1.4. Il est conçu pour être utilisé avec l'éditeur Protégé OWL plugin. Cet outil permet de visualiser la hiérarchie des classes dans une ontologie OWL.

Cette représentation de connaissance a été élaborée avec les utilisateurs de tous les niveaux de la maintenance en commençant par les opérateurs de la maintenance en passant par des chefs de sections, chefs de groupes et services et départements jusqu'au directeurs et chefs de divisions.

FIG. 5.7 – LES EQUIPEMENTS NON STRATEGIQUES (OU NON CRITIQUES)

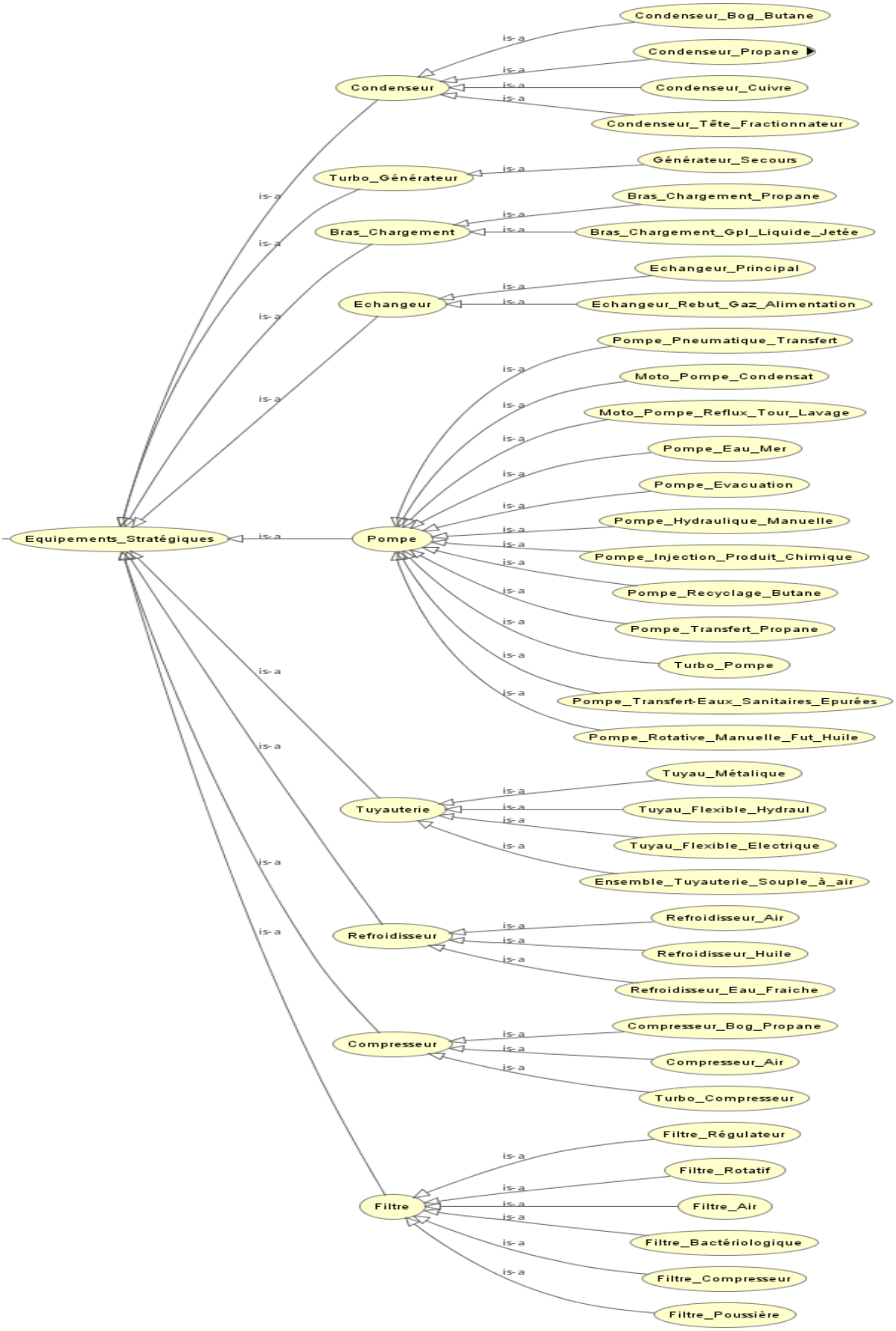


FIG. 5.8 – LES EQUIPEMENTS STRATEGIQUES (OU CRITIQUES)

6.3 Connexion avec la base de données

La connexion avec la base de données est réalisée manuellement par un expert du domaine. Pour une large base de données, la connexion manuelle consomme beaucoup de temps et peut devenir impossible. Afin de bénéficier des connaissances décrites dans l'ontologie, elles doivent être connectées à la base de données afin que tout concept de l'ontologie soit instancié par un sous ensemble d'enregistrements de la base de données.

Concepts	Attributs
Refroidisseur_Air	Tube de refroidisseur inter 3ème étage 40268 ITEM 36, Tube de refroidisseur inter /après 40268 ITEM 38,…
Joint_Torique	Joint torique Dia.Int.:1-3/4" Dia.Tore:1/8", Joint torique Dia.Tore:1/4" Mat.:Viton, …
Ruban_Adhésif	Ruban Adhésif (Scotch électricien)
Compresseur D'air	Compresseur d'air instrument, compresseur 1er étage pour MCR,…
Filtre Rotatif	Filtres25 microns, filtre secondaire, filtre primaire type A1-128…
Pompe à eau	Pompe à eau alim. Chaudière HP(Part of 720-D-323), …
Tuyauterie_souple_à_air	Ensemble de tuyauterie souple à air shop part n° 668,…
….	….

TAB. 5.4 – Un extrait de la connexion directe entre les concepts de l'ontologie et les attributs de la base de données

Plusieurs types de connexions entre l'ontologie et la base de données sont possibles. La plus simple consiste à connecter directement un concept de l'ontologie à un attribut de la base (le plus proche du point de vue sémantique). Plus généralement, les concepts bénéficiant de ce type de connexion font partie de l'ontologie *Equipements stratégiques* et *Equipements non stratégiques*. Un fragment du tableau des connexions est donné dans le tableau 5.4.

6.4 Développement des Schémas de Règles

Le schéma de règles multi-niveaux et les opérateurs permettent à l'utilisateur d'exprimer ses attentes et de superviser le processus de règles d'association dans les deux étapes : Intra-site et Inter-site.

L'expert du domaine utilise le schéma de règles pour chaque niveau pour extraire des connaissances utiles et valides. Dans ce qui suit nous avons élaboré avec l'expert du domaine un ensemble de schémas de règles et d'opérateurs définis dans le tableau 5.5. Ce dernier, présente trois exemples chacun contient un ensemble de schémas de règles et opérateurs des utilisateurs de niveaux différents.

La première colonne représente le numéro de l'exemple. La deuxième représente le numéro du schéma de règles et la troisième colonne les sites et les deux dernières colonnes représentent le schéma de règles et l'opérateur appliqué à ce schéma de règles.

Exemples	**Numéro de schémas de règles**	**Sites**	**Schema de règles**	**Opérateurs**
1	SR_1 Intra-Site	GP1Z GP2Z GL1Z GL2Z	<Equipement_Lavage><L><1>	1-C(SR_1)
	SR_2 Inter-Site	GNL GPL	<Prechauffeurs> <G><2>	2-C(SR_2)
	SR_3 Inter-Site	Division LQS	<Pompe,Joint><M><3>	3-C(SR_3)
2	SR_4 Intra-Site	GP1Z GP2Z GL1Z GL2Z	<Equipements_non_stratégiques→ Echangeur><L><1>	0-C(SR_4)
	SR_5 Inter-Site	GNL	<Joint→Pompe><E><2>	TI(SR_5)
	SR_6 Inter-Site	GPL	<Prechauffeurs> <G><2>	1-C(SR_6)
	SR_7 Inter-Site	Division LQS	<Joint→Equipements_stratégiques><E><3>	1-NO(SR_7)
3	SR_8 Intra-Site	GP1Z GP2Z GL1Z GL2Z	<Joint→ compresseur><G><1>	1-O(SR_8)
	SR_9 Inter-Site	GNL	<Joint→Pompe><E><2>	TI(SR_9)
	SR_{10}	GPL	<Relais_termique→Equipe	1-C(SR_{10})

	Inter-Site		ments_stratégique> <G><2>	
	SR_{11} Inter-Site	Division LQS	<Equipements_non_strategique→ Equipements_strategique > <G><3>	3-C(SR_{11})

TAB. 5.5 – Les exemples de schéma de règles et opérateurs

6.5 Interprétation des schémas de règles

Nous interprétons quelques schémas de règles et opérateurs définis dans le tableau 5.5.

SR_4: Ce schéma de règles exprimé par des utilisateurs des différentes *UBO* exprime les relations entre les pannes des équipements "*Equipments_non_strategiques*" et l'équipement "*Echangeur*". On peut interpréter cette règle par : une panne d'un equipement non couteux comme "*joint","Scotch_Gomme", "Ruban_isolant"*…peut engendrer la panne d'un équipement très couteux comme "*Echangeur*". Pour cela le décideur doit donner de l'importance à la maintenance des "*Equipments_non_strategiques*" régulièrement.

SR_{11}: Ce schéma de règles exprimé par des utilisateurs de la division LQS exprime les relations entre des pannes des équipements "*Equipments_non_strategiques*" avec d'autres équipements "*Equipements_Strategiques*". On peut interpréter cette règle par : une panne d'un équipement non couteux comme "*joint","Scotch_Gomme", "Ruban_isolant"*…peut engendrer la panne d'un équipement très couteux comme "*Echangeur*", "*Compresseur*", "*Pompe*", "*Condenseur*"…..... Pour cela le décideur doit donner de l'importance à la maintenance des "*Equipments_non_strategiques*" de façon régulière.

7 Expérimentations

7.1 Etude 1

La première étude propose de prouver l'efficacité de notre approche en terme de réduction du nombre des règles d'association en utilisant le schéma de règles multi-niveaux. Pour cela, nous proposons à l'expert du domaine de tester les schémas de règles et opérateurs définis dans le tableau 5.5. Nous avons choisi trois

valeurs du support minimum, de la confiance, de $minVR$ et de $minER$ dans les expérimentions suivantes.

- La première valeur est V_1: $minsup_1$=1%, $minconf_1$=10%, $minVR_1$=0.55 et $minER_1$=0.58.
- La seconde est V_2: $minsup_2$=5%, $minconf_2$=40%, $minVR_2$=0.60 et $minER_2$=0.62.
- Et la troisième est V_3: $minsup_3$=10%, $minconf_3$=60%, $minVR_3$=0.65 et $minER_3$=0.7.

Au début l'expert du domaine est face à un nombre important de règles d'association. Par exemple dans GP2Z l'expert doit parcourir et analyser environ deux cent règles d'association comme décrit dans le tableau 5.6. Quelques exemples de ces règles d'association sont présentés dans l'annexe.

Etapes	**Sites**	V_1	V_2	V_3
Intra-Site Niveau 1	GL1Z	120	90	28
	GL2Z	130	80	18
	GP1Z	150	70	35
	GP2Z	200	101	80
Inter-site Niveau 2	GPL (Global)	75	35	30
	GNL(Global)	80	25	15
Inter-site Niveau 3	Division LQS (Global)	65	32	20

TAB. 5.6 – Nombre de règle d'association sans application des schémas de règles

Le tableau 5.7 reporte le nombre des règles d'association après application des schémas de règles et opérateurs. Soit le schéma de règles SR_1 appliqué à un nombre initial de 120 règles d'association (GL1Z) avec l'opérateur de confirmation et la mesure V_1, nous obtenons 15 règles d'association à analyser au lieu de 120 règles d'association. Un autre exemple avec SR_{10} appliqué au nombre initial de 75 règles d'association (GPL) avec l'opérateur type inattendue et la mesure V_1, nous obtenons 13 règles d'association. Il est clair que l'utilisateur peut analyser les 13 règles d'association plus facilement que les 75 règles d'association.

Etapes	**Schemas de Règles**		***Nombre de règles***			***Règles élaguées***		
			V_1	***V_2***	***V_3***	***V_1***	***V_2***	***V_3***
Intra-site Niveau 1	SR_1	GL1Z	15	13	0	105	77	28
		GL2Z	10	0	0	120	80	18
		GP1Z	40	10	5	110	60	30
		GP2Z	100	35	20	100	66	60
Inter-site Niveau 2	SR_2	GNL	40	25	4	35	10	26
		GPL	36	24	4	44	1	11
Inter-site Niveau 3	SR_3	Division LQS	42	30	10	23	2	10
Intra-site Niveau 1	SR_4	GL1Z	32	19	0	88	71	28
		GL2Z	19	0	3	111	80	15
		GP1Z	47	21	11	103	49	24
		GP2Z	190	70	39	10	31	41
Inter-site Niveau 2	SR_5	GNL	12	5	4	63	30	26
	SR_6	GPL	13	6	4	67	19	11
Inter-site Niveau 3	SR_7	Division LQS	14	5	7	51	27	13
Intra-site Niveau 1	SR_8	GL1Z	31	20	25	89	70	3
		GL2Z	15	11	14	115	69	4
		GP1Z	57	17	31	93	53	4
		GP2Z	77	10	17	123	91	63
Inter-site Niveau 2	SR_9	GNL	13	5	3	62	30	27
	SR_{10}	GPL	13	6	2	67	19	13
Inter-site Niveau 3	SR_{11}	Division LQS	36	21	1	29	11	19

TAB. 5.7 – Nombre de règles d'association élaguées

Les figures 5.9., 5.10 et 5.11 montrent le nombre de règles d'association avec et sans utilisation des filtres pour les exemples 1, 2 et 3 respectivement. Nous pouvons observer que le nombre des règles d'association avec utilisation des

schémas de règles est réduit par rapport au nombre de règles d'association sans utilisation des schémas de règles.

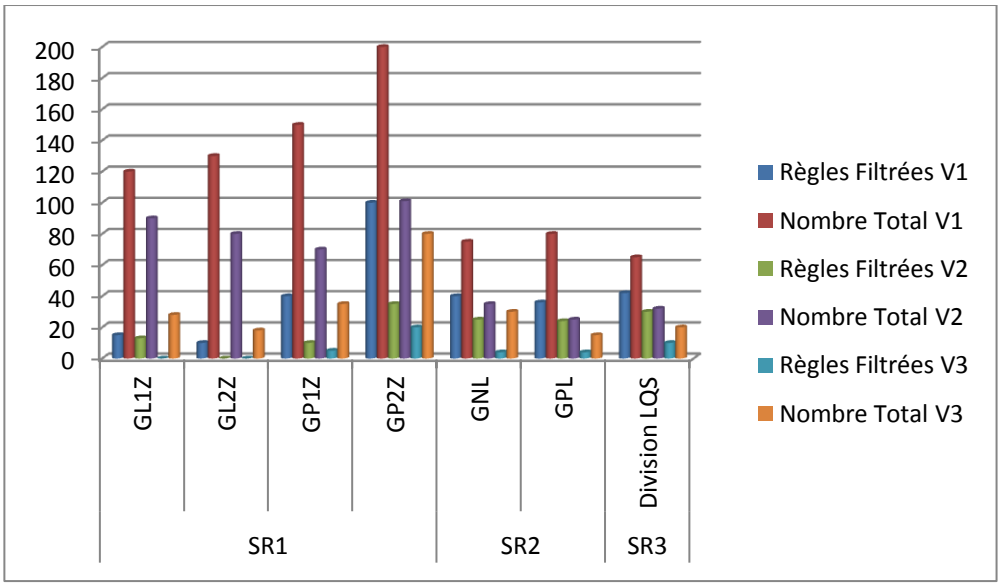


FIG. 5.9 – NOMBRE DE REGLES FILTREES DE L'EXEMPLE 1

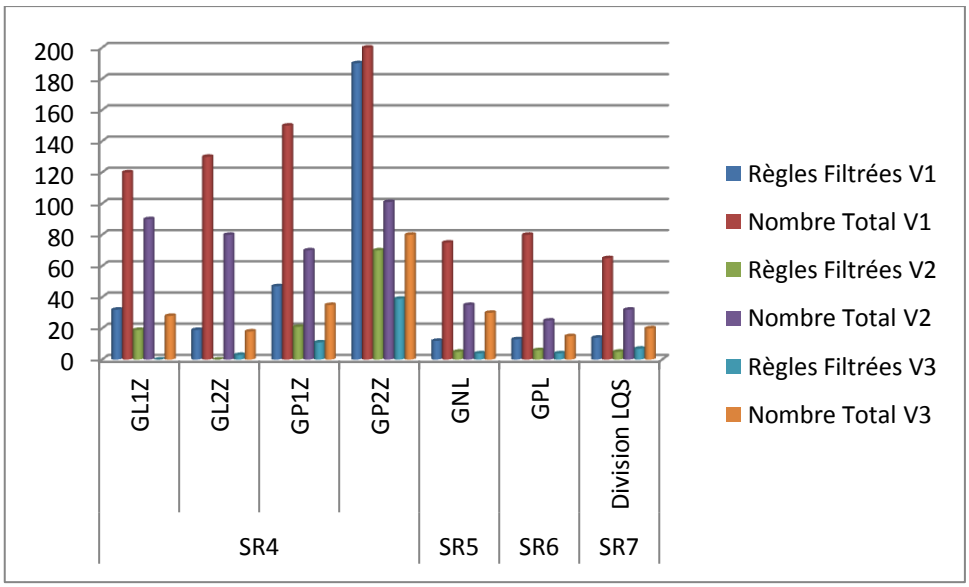


FIG. 5.10 – NOMBRE DE REGLES FILTREES DE L'EXEMPLE 2

On peut conclure que le nombre des règles d'association présentées à l'expert du domaine est acceptable car seules les règles d'association qui l'intéressent lui sont présentées.

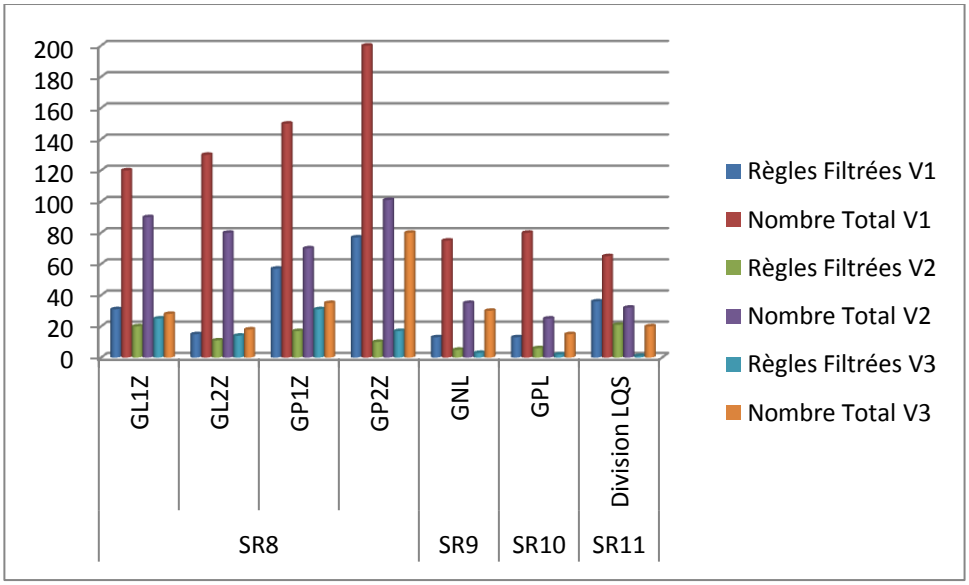


FIG. 5.11 – NOMBRE DE REGLES FILTREES DE L'EXEMPLE 3

7.2 Etude 2

Dans cette étude, nous démontrons l'efficacité de transférer les règles d'association non localement fréquents vers les niveaux supérieurs du processus de synthétisation.

La première colonne du tableau 5.8 expose le numéro de l'exemple et la seconde contient les sites du niveau 2 et 3. La troisième, quatrième et la cinquième colonne représentent le nombre de règles extraites selon les trois valeurs des mesures V_1, V_2 et V_3 avec l'utilisation du processus de synthétisation traditionnel c.à.d sans transfert des règles d'association non localement fréquentes. Tandis que la sixième, septième et huitième colonnes représentent le nombre de règles extraites avec notre approche c.à.d avec transfert des règles d'association non localement fréquentes, selon les trois valeurs des mesures. Finalement, les trois dernières colonnes contiennent le nombre de règles d'association globales perdues suivant les trois mesures.

Exemples		**Sans Transfert**			**Avec Transfert**			**Règles Perdues**		
		V_3	**V_2**	**V_1**	**V_3**	**V_2**	**V_1**	**V_3**	**V_2**	**V_1**
1	GNL	0	12	36	4	25	40	4	13	4
	GPL	0	12	16	4	24	36	4	12	20
	Division LQS	0	18	22	10	30	42	10	12	20
2	GNL	2	3	6	4	5	12	2	2	6
	GPL	2	3	6	4	6	13	2	3	7

	Division LQS	4	3	6	7	5	14	3	2	8
3	GNL	0	3	6	3	5	13	3	2	7
	GPL	0	3	6	2	6	13	2	3	7
	Division LQS	0	10	19	1	21	36	1	11	17

TAB. 5.8 – Le nombre de règles perdues

La figure 5.12 présente le nombre de règles d'association perdues avec et sans transfert des règles d'association non localement fréquentes. Nous constatons que plusieurs règles d'association sont perdues si nous utilisons le processus traditionnel de la synthétisation des règles globales c.à.d sans transfert des règles d'association non localement fréquentes pour toutes les valeurs de *minsup* et *minconf*. Ces règles perdues peuvent être utiles pour les décideurs pour prendre des bonnes décisions.

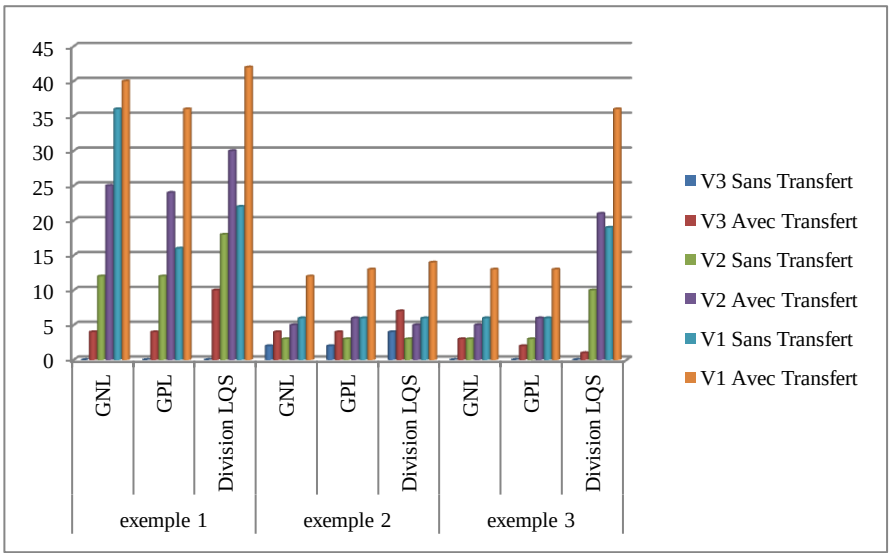


FIG. 5.12 – NOMBRE DE REGLES D'ASSOCIATION SANS ET AVEC TRANSFERT DES REGLES D'ASSOCIATION

8 Conclusion

Ce chapitre montre l'efficacité de *RAMARO* et la qualité des règles d'association découvertes. En effet, Il présente les expérimentations que nous avons menées pour tester notre approche dans un domaine d'application. Les études ont été menées sur des bases de données réelles provenant de l'industrie pétrolière. D'après l'étude 1, l'algorithme *RAMARO* permet de diminuer le volume important des règles d'association en présentant que celles qui sont intéressantes, et nous avons obtenu un nombre exact des règles d'association globales selon l'étude 2.

Chapitre 6 : Les Modèles probabilistes dans l'analyse de motifs locaux

- Introduction
- Travaux connexes
- L'algorithme *AMLAME*
- Expérimentations
- Conclusion

Chapitre 6

Les Modèles probabilistes dans l'analyse des motifs locaux

1 Introduction

L'analyse des motifs locaux constitue une approche intéressante pour faire face aux problèmes liés à la fouille multi-bases de données tels que la voluminosité et la confidentialité des données. Cependant dans cette approche beaucoup de points restent à améliorer notamment l'estimation du support des motifs non localement fréquents pour le calcul des motifs globaux, ce qui est l'objet de ce chapitre.

Ce chapitre est structuré comme suit. Nous présentons dans la section 2, un état de l'art sur l'application des modèles probabilistes dans le processus de découverte des règles d'association avec une synthèse. Ensuite dans la section 3, nous décrivons notre approche de synthétisation des motifs locaux enfin une série d'expérimentations est présentée pour valider les résultats sur un jeu de données synthétiques.

2 Travaux connexes

Dans la littérature l'utilisation des modèles probabilistes dans le processus de découverte des règles d'association se justifie par la réduction du temps d'exécution et de l'espace de recherche. En effet, l'extraction des règles d'association se ramène à une extraction d'un sous ensemble informatif et approximatif de ces règles d'association : – Informatif car à partir de ce sous ensemble nous pouvons déduire le reste des règles d'association – Approximatif car nous pouvons déduire l'ensemble des règles d'association de façon approximative.

Nous pouvons classifier l'ensemble des travaux dans la littérature en deux classes : – Modèles probabilistes appliqués directement sur le contexte de données – Et modèles probabilistes appliqués sur les motifs.

Dans la première classe, les modèles probabilistes se basent sur la réduction des sources de données en un ensemble plus réduit. L'exécution d'un algorithme de découverte des règles d'association se fait sur cet ensemble réduit.

Dans la deuxième classe la découverte des règles d'association se fait de façon approximative basée sur l'ensemble des motifs déjà générés. Le modèle

indépendant [Poosala.V et al. 1997], le modèle inclusion-exclusion [Szymon.J et al. 2002b] [Toon.C et al. 2006], le modèle de l'inégalité de Bonferroni [Szymon.J et al. 2002a] et la méthode maximum entropie [Dmitry.P et al. 2000] [Dmitry.P et al. 2003] appartiennent à cette classe. Suivant notre problématique, nous nous intéressons dans cette section aux travaux appartenant à la deuxième classe qui se basent sur les ensembles des motifs.

Le modèle indépendant se base sur l'indépendance entre les évènements représentés par les itemsets. En effet le support estimé d'un itemset est calculé sur la base de ses sous ensembles. Il est le résultat du produit de tous les supports de ses sous ensembles.

Soit R un ensemble de données binaires et $\varphi_i = P(A_i) = 1$, avec A_i le i^{ème} itemset, φ_i est obtenue directement à partir des données par la formule suivante : $\varphi_i = \frac{f_i}{n}$ avec f_i le nombre de transactions où l'itemset i est présent et n est le nombre de toutes les transactions. L'estimation des supports des itemsets de taille supérieure à deux est calculée par le produit des probabilités pour chaque item présent dans l'itemset. L'estimation des supports d'un itemset ABC se fait par le modèle indépendant de la façon suivante :

$$supp(ABC) = supp(A) * \sup(B) * \text{supp}\ (C)$$

La complexité en temps du modèle indépendant est de $O(n_Q)$, et en espace elle est de $O(k)$ où *k* est le nombre des itemsets dans l'ensemble de transactions et n_Q est la taille de la requête (itemset). Cependant, la qualité d'estimation produite par le modèle indépendant peut être pauvre.

Néanmoins, en raison de sa simplicité, le modèle indépendant est employé dans les systèmes commerciaux car les bases de données correspondantes sont généralement éparses avec un haut degré d'indépendance. En effet, les résultats de ce modèle sont satisfaisants si les données sont éparses ce qui n'est pas le cas dans des contextes complexes où les données sont corrélées comme les bases de données denses où la qualité de l'estimation est médiocre. Par conséquent, on ne peut l'utiliser dans la fouille multi-bases de données car les itemsets peuvent être dépendants, notamment dans les sites locaux.

Les auteurs dans [Toon.C et al. 2006] ont implémenté le modèle inclusion-exclusion pour l'estimation du support des itemsets. A cause de la complexité en temps de ce modèle les auteurs ont utilisé la structure ADTree introduite par [Anderson.B.S et al. 1998]. En effet, cette structure a pour objectif d'organiser et de stocker les itemsets. En premier lieu tous les itemsets fréquents sont stockés et

indexés dans la structure ADTree ensuite une procédure récursive qui implémente le modèle inclusion-exclusion est appliquée [Toon.C et al. 2006]. Dans la structure ADTree les itemsets fréquents sont stockés de façon optimale dans une structure d'arbre. Ce modèle déduit les supports des itemsets à partir des supports de ses sous ensembles.

Ce modèle est simple à implémenter et supporte les itemsets indépendants et dépendants. En plus, il peut être utilisé pour construire la représentation condensée des itemsets non-dérivables [Mannila.H et al. 1996]. Ce modèle calcul le support estimé en peu de temps mais nécessite un espace important pour stocker le nombre important des itemsets dans la structure ADTree. Cependant, l'application de ce modèle nécessite la connaissance du support de tous les sous-ensembles de l'itemset à estimer ce qui est impossible dans notre cas où nous ne disposons que d'un sous ensemble des itemsets fréquents des différentes sources de données.

Le modèle des inégalités de Bonferroni est une extension du modèle inclusion-exclusion. Rappelons que dans le modèle inclusion-exclusion nous avons besoin du support de tous les sous-ensembles de l'itemset à estimer. Les auteurs dans [Szymon.J et al. 2002a] traitent ce problème par l'utilisation des inégalités de Bonferroni de façon récursive pour estimer le support des itemsets dont le support est inconnu. L'estimation des supports d'un itemset ABC se fait par les inégalités de Bonferroni de la façon suivante :

$$supp(ABC) \geq 1 - supp(\bar{A}) - supp(\bar{B}) - supp(\bar{C})$$
$$supp(ABC) \leq 1 - supp(\bar{A}) - supp(\bar{B}) - supp(\bar{C}) + supp(\bar{A}\bar{B}) + supp(\bar{A}\bar{C}) + supp(\bar{B}\bar{C})$$

Avec :

$$supp(\bar{A})=1\text{-}supp(A)$$

Par conséquent, les résultats des expérimentations réalisées dans [Szymon.J et al. 2002a] montrent qu'un nombre important d'itemsets ont des supports estimés triviaux et varient entre *0* et *minsup*.

Les auteurs dans [Dmitry.P et al. 2003] utilisent la méthode maximum entropie pour estimer le support des itemsets. Cette méthode est utilisée pour sélectionner la distribution probabiliste unique de $P_M(x_Q)$ à partir de l'ensemble des distributions satisfaisant les contraintes. $P_M(x_Q)$ est la distribution probabiliste de la requête Q (support estimé de l'itemset) contenant l'itemset x_Q. Si la contrainte est satisfaisante alors la distribution cible est unique et peut être déterminée par un algorithme itératif [Darroch.J et al. 1976].

La méthode maximum entropie est définie comme suit :

(1) Les itemsets fréquents sont générés de façon classique avec un simple algorithme tel que *APRIORI*.

(2) une requête est définie pour estimer l'itemset.

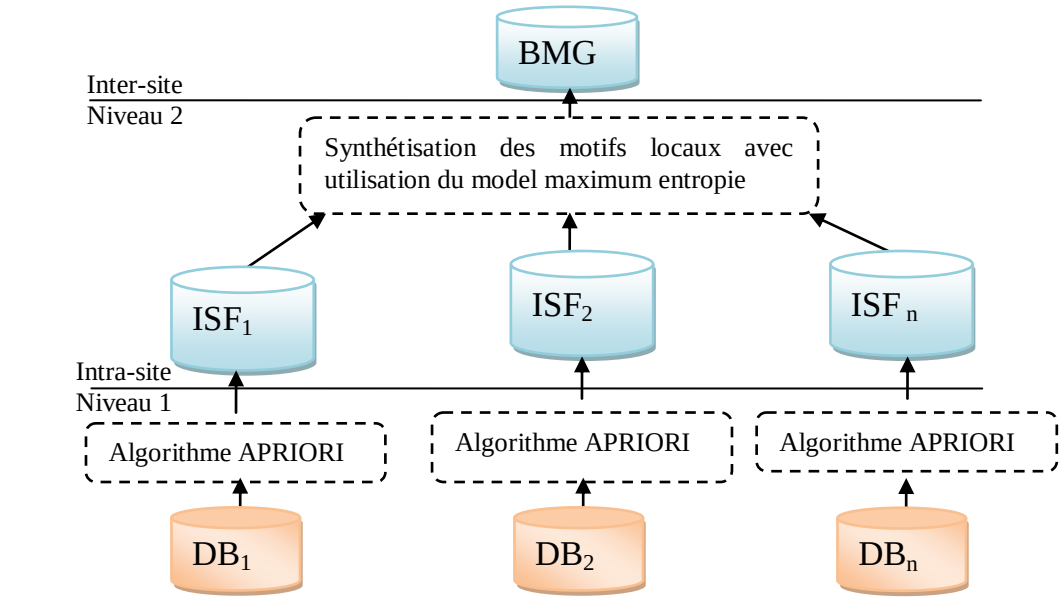
(3) une combinaison des distributions probabilistes sur les variable de la requête est calculée sur ses ensembles fréquents.

La méthode maximum entropie est une méthode de mesure de la probabilité de la distribution. En effet, cette méthode permet d'estimer le support d'un itemset dont on ne connaît pas son support en combinant le support de ses sous ensembles. Toutefois et selon les expérimentations effectuées dans [Dmitry.P et al. 2003], l'utilisation de la méthode maximum entropie pour estimer le support d'un itemset est plus précise et flexible que les autres méthodes citées dans cette section. Mais cette méthode présente une complexité en temps très élevée et qui est exponentielle par rapport au nombre des variables dans la requête. Ce qui ramène les auteurs dans [Dmitry.P et al. 2000] à proposer l'utilisation de la technique de bucket élimination et de la méthode arbre clique pour réduire cette complexité en temps.

Dans cette optique et suivant les résultats satisfaisants en terme de qualité d'estimation, des travaux de [Dmitry.P et al. 2003] effectués sur deux bases de données (données anonymes Microsoft Web et données commerciales), nous proposons d'utiliser la méthode maximum entropie dans notre processus de synthétisation des motifs locaux en motifs globaux. Nous proposons aussi deux optimisations en termes de complexité de temps d'exécution de l'algorithme de graduation itérative en utilisant les buckets élimination dans les arbres cliques. Ce nouveau modèle est détaillé dans la section suivante.

3 L'algorithme *AMLAME*

L'objectif de *AMLAME* (Analyse des Motifs Locaux Avec Maximum Entropie) est de déterminer les supports des motifs globaux en se basant sur les motifs locaux en utilisant la méthode de maximum entropie. Pour cela, nous nous sommes basés sur une organisation à deux niveaux : le niveau 1 représente les unités et le niveau 2 le niveau central. Chaque unité dispose de sa propre base de données. L'algorithme *AMLAME* se déroule en deux phases comme défini dans la figure 6.1. Phase 1 : Intra-site et phase 2 : Inter-site.



DB_i : L'ième source de données.
ISF_i: Les itemsets fréquents du site i.
BM_G : Base des motifs globaux.

FIG. 6.1 – L'ALGORITHME AMLAME

3.1 AMLAME intra-site

Cette étape consiste à appliquer un algorithme (Algorithme 6.1) d'extraction de connaissances au niveau des sites locaux (Niveau1). Dans ce présent travail nous allons utiliser l'algorithme *APRIORI* afin d'extraire pour chaque source de données les itemsets fréquents.

ALG. 6.1 Traitement intra-site

Entrée : Tableau de transactions binaire R, seuil *minsup*.
Sortie : Ensemble de tous les Itemsets fréquents dans R.

Algorithme : l'algorithme *APRIORI*.

3.2 AMLAME inter-site

Cette étape consiste à synthétiser les itemsets fréquents locaux vers des itemsets fréquents globaux. Pour cela, nous procédons comme suit :

Soient m sites $D_1, D_2, \ldots, D_m$ avec $W_1', W_2', \ldots, W_m'$ leurs poids respectifs.

- Le poids normalisé d'une base de données D_j : $W_j = \frac{W_j'}{\sum_{j=1}^{m} W_j'}$

- Le support global synthétisé d'itemset I_i : $Supp_G(I_i) = \sum_{j=1}^{m} W_j \times Supp_j(I_i)$

Le problème de la recherche des itemsets globaux revient à rechercher tous les itemsets dont le support $Supp_G(I_i)$ est supérieur ou égal à $minsup$.

Pour les itemsets non localement fréquents dans quelques sites et au moins fréquents dans un autre site nous appliquons récursivement un algorithme (Algorithme 6.2) d'estimation de ces motifs non localement fréquents au niveau central (Niveau2) afin de les synthétiser en motifs globaux. Nous proposons d'utiliser la méthode maximum entropie dans notre processus de synthétisation des motifs locaux en motifs globaux. Et nous utilisons l'algorithme de graduation itérative pour converger vers le maximum entropie. Les entrées de l'algorithme de graduation itérative sont l'itemset à estimer et tous ses sous ensembles fréquents. La sortie est le support de l'itemset à estimer.

ALG. 6.2 Traitement inter-site

Entrée : Ensemble de tous les Itemsets fréquents et la requête conjonctive x_Q.
Sortie : $P_M(x_Q)$.

Algorithme :
a) Choisir les itemsets dont les variables sont toutes mentionnées dans x_Q.
b) Appliquer l'algorithme graduation itérative pour estimer $P_M(x_Q)$.
c) Renvoyer: $P_M(x_Q)$.

L'algorithme de graduation itérative est détaillé dans ce qui suit :

3.2.1 Graduation itérative comme une convergence de maximum entropie

L'algorithme (ALG. 6.3) de graduation itérative est bien connu dans la littérature statistique comme une technique qui converge à la solution de maximum entropie [Darroch.J et al. 1976]. Le calcul de la probabilité $P_M(x_Q)$ s'effectue comme suit :

Soit la requête Q, et soit ses sous ensemble d'itemsets représentés par les variables x_Q notés par v_Q. La distribution maximum d'entropie est donnée par la formule suivante [Dmitry.P et al. 2000]:

$$P_M(x_Q) = \arg\max_{p \in P} H(p) \qquad (6.1)$$

où $$H(p) = -\sum_x p(x) \log p(x)$$

$$P_M(x_Q) = \mu_0 \prod_{j:V_j \in \nu_Q} \mu_j^{I(x_Q\ satisfies\ V_j)} \quad (6.2)$$

Où les constantes $\mu_0, \dots, \mu_j, \dots$ sont estimées à partir des données. Une fois que les constantes μ_j estimées, l'équation 6.2 peut être employée pour estimer n'importe quel itemset x_Q. Le facteur μ_0 est une constante de normalisation dont la valeur est calculée par la condition suivante :

$$\sum_{x_Q \in \{0,1\}^{n_Q}} P_M(x_Q) = 1 \quad (6.3)$$

Les coefficients μ_j sont calculés en imposant itérativement chaque contrainte. Un itemset V_j, $V_j \in \nu_Q$, définit une contrainte sur l'estimation $P_M(x_Q)$. L'exécution de la j[ème] contrainte peut être effectuée en additionnant $P_M(x_Q)$ de toutes les variables qui n'appartiennent pas à V_j et que le résultat de cette somme soit égale au support exacte : f_j de l'itemset V_j (supp(V_j)) dans la table *R*. La contrainte correspondant au j[ème] itemset est calculée selon l'équation 6.4.

$$f_j = \sum_{x_Q \in \{0,1\}^{n_Q}} P_M(x_Q) I(A_1^j = 1, \dots, A_{n_j}^j = 1) \quad (6.4)$$

Où I (.) est la fonction d'indicateur. L'algorithme de graduation itérative [Dmitry.P et al. 2000] est définit comme suit :

ALG. 6.3 Graduation itérative

Entrée : L'ensemble de tous les T-itemsets fréquents dans R, la requête Q.

Sortie : La distribution $P_M(x_Q)$.

Algorithme :

1. Choisir une approximation initiale de $P_M(x_Q)$
2. Tant que toutes les contraintes ne sont pas satisfaisantes
   Faire varier j sur toutes les contraintes
   Mettre à jour μ_0;
   Mettre à jour μ_j;
   Fin de faire;
   Fin de tant que;
   Renvoyer les constantes μ_j

Les règles modifiées pour le paramètre μ_j^t correspondant à la $j^{ème}$ contrainte à l'itération t sont :

$$\mu_0^{t+1} = \mu_0^t \frac{1-f_i}{1-s_j^t} \quad (6.5)$$

$$\mu_j^{t+1} = \mu_j^t \frac{f_i(1-S_j^t)}{S_j^t(1-f_i)} \qquad (6.6)$$

$$S_j^t = \sum_{x_Q\ satisfait\ j} P_M^t(x_Q) \qquad (6.7)$$

L'équation 6.7 calcule les contraintes comme l'équation 6.4, mais avec l'utilisation de l'estimation courante P_M^t qui est définie (alternativement) par les estimations de ses facteurs μ_j^t à travers l'équation 6.2. L'addition est effectuée sur toutes les valeurs des variables de requête satisfaisant le $j^{ème}$ itemset. Puisque le P_M^t ne traite pas la $j^{ème}$ contrainte, l'équation 6.5 et 6.6 mettent à jour les facteurs μ_0 et μ_j. L'algorithme procède avec des boucles pour chaque contrainte à chaque itération. Cette procédure itérative est une convergence à la solution unique de maximum d'entropie et fourni les contraintes sur la distribution conformée.

La convergence de l'algorithme est déterminée par l'équation 6.8 qui est la condition d'arrêt :

$$|P_M^t\ (\mathrm{Q}) - P_M^{t-1}\ (\mathrm{Q})| < \ \varepsilon\ P_M^{t-1}(Q) \qquad (6.8)$$

Où $\varepsilon = 10^{-4}$ est l'un des paramètres libres de l'algorithme.

Bien que l'algorithme de graduation itérative converge vers la solution de maximum entropie il présente une complexité en temps assez importante. Pour cela nous proposons deux optimisations pour réduire cette complexité.

L'optimisation 1 permet de faire une sommation plus optimale dans l'équation 6.7 et l'optimisation 2 permet de décomposer le problème original (équation 6.2) en problèmes plus petits.

L'idée de base est de représenter la distribution des probabilités communes des itemsets fréquents sur un arbre clique ensuite d'appliquer l'algorithme de graduation itérative sur cet arbre avec l'utilisation de bucket élimination.

3.2.2 Optimisations de l'algorithme de graduation itérative

Pour Réduire la complexité en temps dans la méthode maximum entropie nous proposons :

- D'appliquer la technique de Bucket élimination [Dechter.R 1999] pour accélérer l'estimation des facteurs dans l'équation 6.7 (optimisation 1).
- D'utiliser la structure de graphe arbre clique [Malvestuto.F.M 1992] pour rassembler les distributions communes dans l'équation 6.2 (optimisation 2).

3.2.2.1 Optimisation 1

L'équation 6.7 de l'algorithme de graduation itérative procède par une sommation de la distribution $P_M^t(x_Q)$. Le nombre total de la sommation dans l'équation 6.7 est de $2^{n_Q - n_j}$ où n_Q est la taille de l'itemset à estimer et n_j est la taille du j^{ème} itemset. Chaque partie de la sommation est sous forme d'un produit et contient au plus N facteurs où N est le nombre des itemsets. Dans cette optique nous avons utilisé une optimisation pour réduire le nombre de sommation dans l'équation 6.7. La technique de bucket élimination [Dechter.R 1999] permet d'accélérer l'algorithme de graduation itérative et d'estimer les facteurs en employant la loi de distribution dans l'équation 6.2. Pour comprendre le principe de bucket élimination l'article [Dechter.R 1999] contient une description détaillée de cette méthode. Nous montrons à travers l'exemple 6.1 l'avantage de l'utilisation de la technique bucket élimination.

Exemple 6.1 : Soit une requête composée de six attributs $A_1 \ldots A_6$. On suppose qu'il y a des itemsets fréquents correspondant à chaque item et les itemsets fréquents de taille 2 suivants : $\{A_1,A_2\}$,$\{A_2, A_3\}$, $\{A_3, A_4\}$, $\{A_4, A_6\}$, $\{A_3, A_5\}$ et $\{A_5, A_6\}$. Nous obtenons le graphe H représenté dans la figure 6.2 qui montre les interactions entre les items.

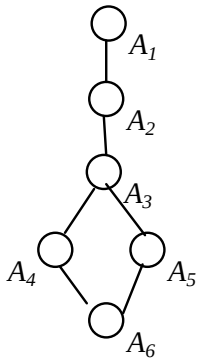


FIG. 6.2 – LE GRAPHE H

Selon l'équation 6.2, la distribution maximum d'entropie se calcule de la façon suivante :

$$P_M(A_1..A_6) = \mu_0 \prod_{i=1}^{6} {\mu_i}^{I(A_i=1)} \prod_{i,j} {\mu_{ij}}^{I(A_i=A_j=1)} \quad I(\exists edge(i,j) \in H)$$

$$= \mu_0 {\mu_1}^{I(A_1=1)} {\mu_2}^{I(A_2=1)} {\mu_3}^{I(A_3=1)} {\mu_4}^{I(A_4=1)} {\mu_5}^{I(A_5=1)} {\mu_6}^{I(A_6=1)}$$

$* \mu_{12}{}^{I(A_1=A_2=1)} \mu_{23}{}^{I(A_2=A_3=1)} \mu_{34}{}^{I(A_3=A_4=1)} \mu_{46}{}^{I(A_4=A_6=1)} \mu_{35}{}^{I(A_3=A_5=1)} \mu_{56}{}^{I(A_5=A_6=1)}$

On suppose que sur l'itération courante, nous mettons à jour le coefficient μ_{56} correspondant à l'itemset $\{A_5, A_6\}$. Selon la règle de mise à jour dans l'équation 6.7, nous devons fixer les attributs A_5 et A_6 et additionner le reste des attributs $P_M^t(A_1..A_4)$. Nous partitionnons les μ facteurs sur les buckets. Chaque attribut ν définit un bucket séparé (le bucket est une structure de données qui contient tous les facteurs μ sur l'attribut ν). Le partitionnement des facteurs sur les buckets est donné comme suit :

Bucket (A_1)=$\{ \mu_1, \mu_{12}\}$; *Bucket* (A_2)=$\{ \mu_2, \mu_{23}\}$;

Bucket (A_3)=$\{ \mu_3, \mu_{34}, \mu_{35}\}$; *Bucket* (A_4)=$\{ \mu_4, \mu_{46}\}$;

Après le partitionnement de chaque bucket dans l'ordre, on utilise les trois opérations de base suivantes qui sont appliquées consécutivement :

(a) Multiplier toutes les fonctions de chaque bucket,

(b) Sommer le reste de la variable du bucket,

(c) Placer la fonction résultat f dans le bucket le plus élevé contenant certaines variables de f.

L'utilisation de la technique de bucket élimination permet de réduire le nombre de termes par un facteur de 2 comparée à la méthode classique:

$$\sum_{A_1..A_4} P_M(A_1..A_4, A_5 = 1, A_6 = 1) =$$
$$\mu_0\mu_4\mu_5\mu_{56} \cdot \sum_{A_4}(\mu_4\mu_{46})^{I(A_4=1)} \cdot (\sum_{A_3}(\mu_3\mu_{35})^{I(A_3=1)} \ \mu_{34}{}^{I(A_3=A_4=1)}$$
$$\cdot(\sum_{A_2}(\mu_2)^{I(A_2=1)} \ \mu_{23}{}^{I(A_2=A_3=1)} \ (\sum_{A_1}(\mu_1)^{I(A_1=1)} \ \mu_{12}{}^{I(A_1=A_2=1)})))$$

3.2.2.2 Optimisation 2

Dans cette optimisation, nous allons procéder à une décomposition de la distribution de la probabilité selon un model de graphe. Il est possible de représenter la distribution de maximum entropie comme un produit exponentiel correspondant aux cliques du modèle graphique H. Pour plus de détail sur ce modèle de graphe se référer à l'article [Malvestuto.F.M 1992].

Une clique est un sous-graphe complet, c-à-d., un sous-graphe pour lequel tous les nœuds sont interconnectés. En effet, deux sommets quelconques de la clique sont toujours adjacents. Elle est dite maximale lorsqu'elle est le plus grand sous graphe complet contenant tous ses nœuds. La construction de l'arbre clique d'un ensemble d'itemsets fréquents se fait comme suit :

Soit l'itemset à estimer représenté sous forme de requête conjonctive Q et soit l'ensemble ν_Q de tous ses sous ensembles. Ces derniers représentent un modèle graphique non dirigé H sur les variables de la requête, avec des nœuds correspondants aux variables de requête x_Q. Un lien relie les deux nœuds si et seulement si les variables correspondantes sont inclues dans l'itemset. Nous avons d'abord sélectionné un ordre lexicographique des variables dans le graphe H et imposé la liaison dans cet ordre, c.-à-d., nous relions deux parents déconnectés quelconques de n'importe quel nœud [Pearl.J. 1988]. La figure 6.3 présente le graphe triangulé H' de l'exemple 6.1.

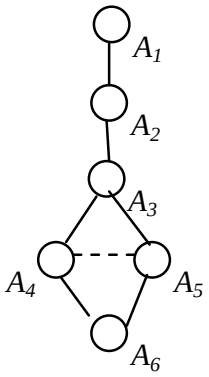


FIG. 6.3 – LE GRAPHE TRIANGULE H'

Les cliques maximales du graphe sont placées dans un arbre de jointure (une forêt). L'arbre de jointure ou de clique représente l'ensemble des cliques avec leurs interactions. La figure 6.4 présent la forêt de clique du problème de la figure 6.2.

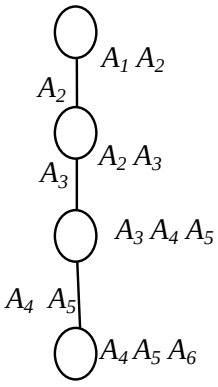


FIG. 6.4 – LA FORET DE CLIQUE DU PROBLEME DE LA FIGURE 6.2.

Ainsi, le problème initial peut être décomposé en plus petits problèmes correspondant aux cliques du graphe triangulé H'. Chaque petit problème peut être résolu séparément en utilisant l'algorithme de graduation itérative, alors que les

distributions correspondantes aux intersections peuvent être trouvées en additionnant le reste des distributions correspondantes sur les cliques [Dmitry.P et al. 2003].

La technique bucket élimination utilise les informations sur la structure des fonctions pour faire la sommation plus efficacement, et le reste de l'algorithme de graduation itérative reste inchangé. Par contre la méthode arbre clique essentiellement part du principe que la distribution maximum entropie équivaut à estimer le produit des fonctions correspondantes au graphe clique modèle [Dmitry.P et al. 2003]. Nous utilisons l'algorithme de graduation itérative avec la technique de bucket élimination pour estimer les fonctions correspondantes aux cliques et retourner le produit de la clique distribution comme résultat.

4 Expérimentations

Pour valider notre approche nous avons mené quelques expérimentations sur une base de données synthétiques. En effet, nous avons choisi la base de données « Mushroom » disponible dans http://fimi.ua.ac.be/data/. Cette base de données dense contient 8124 transactions et 22 items différents. Nous l'avons partitionné horizontalement en 4 bases de données S_1, S_2, S_3 et S_4 contenant chacune 2500, 2500, 1624 et 1500 transactions respectivement.

Les itemsets sont générés par l'algorithme *APRIORI* appliqué à :

a) L'ensemble de toutes les bases de données (Monobase de données).
b) Chaque base de données avec des *minsup* différents 0.5, 0.7, 0.85 et de *minconf*= 0.30, selon les cas :
   b.1) Sans estimation
   b.2) Avec utilisation de l'estimation par la méthode de facteur de correction [Ramkumar.T et al. 2009]
   b.3) Avec l'algorithme *AMLAME*.

Les critères que nous avons définis pour réaliser les expérimentations portent sur :

- Le taux moyen des motifs globaux –l'erreur moyenne totale des supports globaux.

Le taux moyen des motifs globaux (TMMG): Ce critère permet de mesurer le taux des motifs fréquents globaux (*TMG*) généré par un algorithme pour les trois valeurs de support minimum. En d'autres termes, il permet de mesurer le taux de perte de motifs fréquents globaux. Le taux moyen de motifs générés ($TMMG$) est

la moyenne des taux de motifs globaux (TMG_{minsup}) générés pour les trois valeurs de support minimum.

$$TMMG = \sum_{minsup=0.5,0.7,0.85} \frac{TMG_{minsup}}{3}$$

Où TMG_{minsup} se calcule par le rapport entre le nombre des motifs globaux synthétisés par les deux algorithmes (facteur de correction et *AMLAME*) et le nombre exact des motifs globaux pour une valeur de support minimum *minsup*.

$$TMG_{minsup} = \frac{nombre\ des\ motifs\ globaux\ générés\ par\ un\ algorithme}{nombre\ des\ motifs\ globaux\ exact}$$

L'erreur moyenne totale des supports (EMTS) : Représente la différence entre les valeurs exactes des supports globaux avec les valeurs des supports synthétisés sur les trois valeurs de *minsup*. En effet, c'est la moyenne des erreurs moyennes (EM) de chaque valeur de *minsup*.

$$EMTS = \frac{EM_{minsup=0.5} + EM_{minsup=0.7} + EM_{minsup=0.85}}{3}$$

où :

EM_{minsup} : est la moyenne des erreurs des supports (ES) de tous les itemsets estimés pour une valeur de *minsup*.

$$EM_{minsup} = \frac{\sum_{i=1}^{n} \mathrm{ES}_i}{n}$$

Où n est le nombre des itemsets frequents estimés pour une valeur de *minsup*.

Et ES_i : est l'erreur entre le support de l'itemset$_i$ synthétisé et de son support exact donnée par la formule suivante :

$$\mathrm{ES}_i = |\text{support de l'itemset global estimé}_i - \text{support global exacte de itemset}_i|$$

Algorithmes	**Motifs Fréquent monobase de données**	**Motifs globaux synthétisés (sans estimation)**	**Motifs globaux synthétisés (Avec estimation)**	**Erreur Moyenne**
***minsup* 50%**				
Facteur de correction	6030	211	1370	0,0846
AMLAME			4898	0,0557
***minsup* 70%**				
Facteur de correction	1156	175	194	0,0314
AMLAME			760	0,0214
***minsup* 85%**				
Facteur de correction	478	50	90	0,0401
AMLAME			334	0,0479

TAB. 6.1 – Le nombre de motifs globaux estimé avec l'erreur moyenne

Critère 1 : Le taux moyen des motifs globaux :

Le tableau 6.1 montre le nombre des itemsets globaux estimé selon *AMLAME*. Nous observons que ce nombre est important et proche de la fouille monobase de données. Par contre, la méthode de facteur de correction est loin de l'être. Prenons l'exemple, avec *minsup* =50%, nous avons exactement 6030 itemsets fréquents globaux. Par contre dans le processus de multi-bases de données sans estimation nous avons 211 itemsets fréquents globaux d'où 5819 itemsets perdus. Avec la méthode de facteur de correction [Ramkumar.T et al. 2009] ce nombre est légèrement amélioré par 1370 itemsets fréquents globaux, d'où 1159 nouveaux itemsets fréquents globaux. Mais nous avons toujours une perte d'information de l'ordre de 4660 itemsets fréquents globaux. Ce qui représente une perte d'environ 77% de l'ensemble des itemsets fréquents globaux. Avec l'algorithme *AMLAME*, nous avons pu estimer environs 4898 itemsets fréquents globaux. Ce qui représente 81% des itemsets fréquents globaux avec une perte seulement de 19% des itemsets fréquents globaux. On peut affirmer que notre approche a nettement amélioré le nombre des itemsets fréquents globaux. Et ce résultat est proche de celui de la fouille monobase de données.

Critère 2 : L'erreur moyenne totale des supports

L'erreur moyenne totale représente la différence entre les valeurs des supports globaux des itemsets découverts par les deux méthodes : Facteur de correction et *AMLAME* pour les trois valeurs des supports minimum. Cette erreur est (0.0846+0.0314+ 0.0401)/3=0.052 dans l'algorithme de facteur de correction. Ce qui est supérieur à l'erreur de l'algorithme *AMLAME* qui est de (0.0557+

0.0214+0.0479)/3=0.041. Donc on peut conclure que la qualité dans notre méthode est meilleure que celle de la méthode facteur de correction. A partir des résultants précédents, nous pouvons noter que (tableau 6.2) :

- Les algorithmes de synthétisation classiques sans estimation génèrent environ (211+175+50)/(6030+1156+478)=6% des itemsets fréquents globaux parmi l'ensemble total avec une perte de 94%, ce qui est énorme.
- L'estimation par l'approche de facteur de correction donne des estimations acceptables et donne (1370+194+90)/(6030+1156+478)=21% des itemsets fréquents globaux parmi l'ensemble total avec une perte de 79%.
- Dans l'algorithme *AMLAME*, les résultats se rapprochent de celles de la fouille monobase de données et donne environ (4898+760+334)/(6030+1156+478)=78% des itemsets fréquents globaux soit une perte que de 22%.

Algorithme	Qualitée de l'estimation	Taux moyen du nombre des motifs globaux
Synthetisation (Sans estimation)	Mauvaise Qualité	6%
Facteur de Correction	Qualité Moyenne	21%
AMLAME	Meilleur Qualité	78%

TAB. 6.2 – Comparaison entre les différentes approches

5 Conclusion

Nous avons présenté dans ce chapitre, en premier lieu un état de l'art synthétique des travaux permettant de faire la recherche de règles d'association avec des méthodes probabilistes. Plusieurs types d'algorithmes ont été abordés comme les algorithmes reposant sur le modèle indépendant, inclusion-exclusion, les inégalités de Bonferroni et la méthode maximum entropie. Nous nous sommes intéressés à cette dernière car elle génère des règles d'association de qualité mais pose le problème de la complexité en temps. Afin de pallier à cette limite, nous avons proposé une optimisation de la méthode en utilisant la technique de bucket élimination et le concept d'arbre clique. Nous avons validé l'approche proposée sur une base de données synthétiques. Les résultats obtenus montrent que notre approche réduit le nombre de règles d'association globales perdues par rapport à l'algorithme facteur de correction.

Conclusion générale et Perspectives

- Conclusion générale
- Perspectives

Conclusion générale et Perspectives

1 Conclusion générale

Cette thèse adresse deux issues principales qui sont résumées en deux contributions : La première se focalise sur l'intégration des connaissances de l'utilisateur dans le processus des règles d'association dans un environnement multi-bases de données. Tandis que la deuxième contribution traite le problème de l'estimation des supports des motifs non localement fréquents.

Pour la première contribution, nous avons intégré les connaissances des utilisateurs dans les deux phases de la fouille multi-bases de données pour ne conserver que les règles d'association intéressantes. Pour cela, nous avons proposé un nouveau formalisme de représentation des attentes des utilisateurs appelé schéma de règles multi-niveaux. Nous avons proposé aussi de nouveaux opérateurs applicables sur ces schémas de règles multi-niveaux. Dans la phase intra-site, nous avons réduit le parcours de la base de données à un seul parcourt pour n'extraire que des règles d'association locales intéressantes et les règles d'association globales candidates. Dans la phase inter-site, ces règles locales sont synthétisées en règles globales, majoritaires et exceptionnelles.

Dans la deuxième contribution nous avons utilisé le modèle probabiliste maximum entropie pour synthétiser des motifs locaux en motifs globaux.

Les contributions majeures dans cette thèse sont décrites dans ce qui suit selon les deux approches :

1.1 1ère Contribution

Nos principales contributions sont résumées en 3 points : Modèle de représentation des connaissances, l'algorithme *RAMARO* et la maintenance anticipée.

-Modèle de représentation des connaissances de l'utilisateur

Nous avons proposé un nouveau modèle de représentation des connaissances de l'utilisateur dans le processus d'extraction des règles d'association dans un environnement multi-bases de données. Il est composé de trois formalismes : Ontologies, schéma de règles multi-niveaux et opérateurs.

L'ontologie permet à l'expert du domaine de représenter les connaissances du domaine à travers un modèle sémantique.

Le schéma de règles multi-niveaux permet à l'utilisateur d'exprimer ses attentes et besoins en terme de règles d'association. Ce schéma de règles est issu des concepts de l'ontologie, ce qui permet d'avoir une flexibilité à l'utilisateur.

Les opérateurs permettent à l'utilisateur de faire des actions sur les schémas de règles. Nous avons proposé quatre opérateurs : K-Conforme, K-Objectif, K-Non Objectif et type inattendue.

- RAMARO

La contribution principale de cette thèse réside dans la proposition innovatrice appelée *RAMARO*, qui aide l'utilisateur à réduire le volume des règles découvertes et à améliorer leur qualité. En effet le nouveau modèle de représentation proposé prend en compte les attentes des utilisateurs de tous les niveaux permettant ainsi de ne régénérer que les règles qui les intéressent. De plus, grâce au transfert des règles d'association non localement fréquentes aux niveaux supérieurs, toutes les règles globales intéressantes sont générées.

-Maintenance Anticipée

La contribution de *RAMARO* dans la maintenance peut être résumée par l'ajout d'une nouvelle politique de maintenance qui est la maintenance anticipée. Cette dernière utilise les données des différentes bases de données pour prévoir et mettre l'accent sur la possibilité de futures pannes des équipements.

1.2 2ème contribution

La deuxième contribution consiste à estimer les motifs non localement fréquents pour les faire contribuer dans la synthétisation des motifs globaux. Nous avons utilisé la méthode maximum entropie pour cette fin. Les expérimentations effectuées confirment la réduction de la perte des motifs globaux.

2 Perspectives

Les travaux futurs qu'on peut envisager sont décrits dans ce qui suit selon la contribution utilisée :

2.1 1ère contribution

La validation de *RAMARO* sur d'autres contextes

La validation de notre approche peut être effectuée suivant deux axes. Premièrement, elle peut être testée sur différentes données avec la collaboration de l'expert du domaine. Ces expérimentations peuvent être validées suivant de nouvelles données, et testées par différents utilisateurs avec différentes attentes.

Deuxièmement, il est possible de comparer notre approche avec celles existantes. Comparer notre approche par les approches guidées par l'utilisateur est une tâche difficile dû à la pauvreté du benchmark. Néanmoins, la proposition à l'expert du domaine de tester les différentes approches suivant des scénarios prédéfinis peut être une bonne initiative.

La représentation visuelle des règles d'association

Les méthodes d'intégration des connaissances du domaine dans le processus de la fouille multi-bases de données et la représentation visuelle d'ensembles de règles d'association convergent vers le même objectif qui est de faciliter l'exploration de grands ensembles de règles d'association dans différents niveaux. Différents utilisateurs auront besoin de visualiser les règles générées sur différents axes qui peuvent êtres par sites, par type de connaissances. La représentation visuelle doit offrir à l'utilisateur une vue complète sur les règles d'association découvertes. Les utilisateurs de niveaux supérieurs peuvent visualiser les règles d'association de ceux des niveaux inférieurs pour les aider à prendre des décisions globales.

Le Filtrage des règles d'association par des mesures objectives

Les schémas de règles filtrent un nombre important des règles d'association suivant les attentes des utilisateurs. Mais ce nombre reste comme même un peu élevé et contient des règles d'association plus générales et redondantes. Pour cela, l'utilisation des filtres objectifs tels que CRg et TRg [Inhaúma.N et al. 2013] peuvent réduire le nombre de règles d'association sans affecter la sémantique.

Le Mapping automatique des concepts de l'ontologie vers les bases de données

Une des difficultés rencontrées lors de l'implémentation de l'ontologie réside dans le mapping entre les attributs de la base de données et les concepts de l'ontologie. Cependant le nombre des attributs est important ce qui rend cette tâche difficile pour affecter et associer les concepts de l'ontologie avec les attributs de la base de données. Le recours à des outils automatiques s'avère nécessaire pour réduire cette complexité de correspondance attributs/concepts.

Utilisation de structure compacte pour représenter les schémas de règles

Nous avons vu dans le chapitre 4 exactement dans la section 4.1, que la génération des motifs candidats s'effectue de façon classique. En revanche, certains utilisateurs de mêmes niveaux ou de niveaux différents, peuvent exprimer des schémas de règles identiques ou presque par conséquent leur représentation reste redondante. D'où la nécessite de trouver une structure adéquate pour absorber cette redondance et minimiser l'espace de recherche ce qui va certainement minimiser le temps d'exécution.

2.2 2ème contribution

-La validation de *AMLAME* sur d'autres jeux de données

La validation de *AMLAME* peut être effectuée sur d'autres contextes tels que des contextes denses et éparses pour confirmer la qualité des résultats de notre approche. La littérature est assez riche en termes de jeux de données sous forme de base de données benchmark.

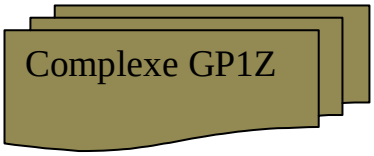
-La validation de *AMLAME* avec les autres modèles probabilistes

Une étude comparative pratique avec les autres modèles probabilistes est nécessaire pour montrer l'efficacité de *AMLAME*. La comparaison peut être effectuée avec le modèle indépendant, le modèle inclusion-exclusion et les inégalités de Bonferroni.

Annexe

Nous donnons dans cette annexe quelques exemples de règles d'association obtenues pour l'application de la maintenance. Nous signalons si la règle est sélectionnée en respectant les attentes des utilisateurs exprimées à travers les schémas de règles et les opérateurs.

I- Règles d'association locales :



------ **Règle 1 :** 319300105 → 319300132

Si

- **319300105:** SOUPAPE D'ASPIRATION 2eme ETAGE HOERBIGER TYPE 154R1.

Alors avec (une confiance = 87.8 %)

- **319300132**: JOINT ALUMINIUM P/SIEGE CLAPET 2eme ETAGE DI=175mm x DE=184mm x EP= 1.5mm.

La règle 1 explique que le remplacement de la « Soupape d'aspiration » dans l'équipement « Compresseur d'air » implique le remplacement du « Joint Aluminium » dans le même équipement avec une confiance de 87,8%.

--------**Règle 2:** 319300132 → 319300105

Si

- **319300132**: JOINT ALUMINIUM P/SIEGE CLAPET 2eme ETAGE DI=175mm x DE=184mm x EP= 1.5mm.

Alors avec (une confiance = 64.4 %)

- **319300105:** SOUPAPE D'ASPIRATION 2eme ETAGE HOERBIGER TYPE 154R1.

La règle 2 explique que le remplacement du « Joint Aluminium » implique le remplacement de la « Soupape D'aspiration » avec une confiance de 64,4.

--------Règle 3 : 319300131 → 319300132

Si

- **319300131:** JOINT TORIQUE "ROUGE" DI = 179.3mm x TORE = 5.7mmMAT: SI (CAOUT. SILICONE).

Alors avec (une confiance = 84.8 %)

- **319300132:** JOINT ALUMINIUM P/SIEGE CLAPET 2eme ETAGE DI=175mm x DE=184mm x EP= 1.5mm.

La règle 3 explique que le remplacement du « Joint Torique Rouge» du « Compresseur d'air » implique avec une confiance de 84,8% le remplacement du « Joint Aluminium » dans le même équipement.

--------Règle 4 : 319300132→ 319300131

Si

- **319300132:** JOINT ALUMINIUM P/SIEGE CLAPET 2eme ETAGE DI=175mm x DE=184mm x EP= 1.5mm.

Alors avec (une confiance = 62.2 %)

- **319300131:** JOINT TORIQUE "ROUGE" DI = 179.3mm x TORE = 5.7mmMAT: SI (CAOUT. SILICONE).

La règle 4 explique que le remplacement du « Joint Aluminium » du « Compresseur d'air » implique le remplacement du « Joint Torique Rouge» avec une confiance de 62,2%.

--------Règle 5 : 315182299 → 315182213

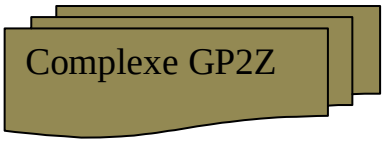
Si

- **315182299:** ELEMENT DE FILTRE.

Alors avec (une confiance = 62.2 %)

- **315182213:** CARTOUCHE DE FILTRE.

La règle 5 explique que le remplacement d'un élément de filtre d'une turbine à gaz implique le remplacement de la cartouche de filtre avec une confiance de 62,2%. **Cette règle est conforme au schéma de règles SR_1.**



-------- **Règle 6 :** ZZ0007→ ZZ0006

Si

- **ZZ0007:** ELECTRODE DE SOUDURE - BASIQUE - 3,2mm - OK 48.00 ou AWS E.7018.

Alors avec (une confiance = 76.9 %)

- **ZZ0006:** FLEXIBLE "3/8" SAE 100R1LG 2 M RPR 120 PL.24019930366 01 HYDR SR BRS 12"X65" FP RCMA FMC 245644000120.

La règle 6 signifie que à chaque fois que « Electrode de Soudure » est remplacé, alors avec une confiance de 76.9% le « FLEXIBLE 3/8 » sera aussi remplacé.

-------- **Règle 7 :** ZZ1122 → ZZ2545

Si

- **ZZ1122:** VIS SPECIALE - REP: 05 - REF: V.4621 - P/REGARD TAI/TAII.

Alors avec (une confiance = 97.5 %)

- **ZZ2545:** PALIER COMPLET INFERIEUR REF: HUDSTON PRODUCTS CORP BOTTOM FIN.FAN BEARING INVNR 50080 POUR AEROS.

--------**Règle 8 :** ZZ2545→ ZZ1122

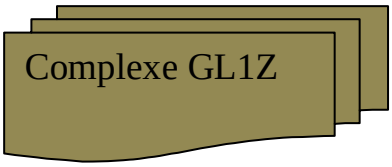
Si

- **ZZ2545:** PALIER COMPLET INFERIEUR REF: HUDSTON PRODUCTS CORP BOTTOM FIN.FAN BEARING INVNR 50080 POUR AEROS.

Alors avec (une confiance = 100 %)

- **ZZ1122:** VIS SPECIALE - REP:05 - REF: V.4621 - P/REGARD TAI/TAII.

Les règles 7 et 8 montrent que si on remplace le «Vis Spéciale » du « Compresseur C de la Réfrigération Principal» on aura une confiance de 97,5% de remplacer le « Palier Complet Inferieur». Par contre (règle 8) le remplacement du « Palier Complet Inferieur» implique avec certitude le remplacement du «Vis Spéciale ».



------Règle 9 : 060728→ 110001

Si

- **060728:** JOINT SPIRALE AVEC ANNEAU DE CENTRAGE Dia. Nom:12", Série:900 #, ASME-B16.20, Mat: .316/Graphité.

Alors avec (une confiance = 66.2 %)

- **110001:** OXYGENE GAZEUX INDUSTRIEL.

La règle 9 explique que le remplacement du « Joint Spirale Avec Anneau » de la « Chaudière de Procède TYPE 34-VP-18W-R» implique avec une confiance de 66,2% l'utilisation de « Oxygène Gazeux Industriel ».

------Règle 10 : 508809 → 000501

Si

- **508809:** RUBAN ISOLANT ELECTRIQUE, ROUGE Ep.: 18/100.

Alors avec (une confiance = 98.3 %)

- **000501:** TOLE PLATE LISSE Ep.:1/4", Long.:96", Larg.:48", Mat.:A283 GrD.CHAUDIERE PROCEDE, 400 T/ TUBES DE CHAUDIERE

La règle 10 exprime que le remplacement du « Ruban Isolant Electrique » de la « Chaudière Procède, 400 T/ TUBES DE CHAUDIERE » alors avec une confiance de 98,3% la « Tôle Plate Lisse » sera remplacée.

------Règle 11 : 000501 → 060806 508809

Si

- **000501 :** TOLE PLATE LISSE Ep.:1/4", Long.:96", Larg.:48", Mat.:A283 GrD ;

Alors avec (une confiance = 98.3 %)

- **060806:** JOINT SPIRALE AVEC ANNEAU DE CENTRAGE Dia. Nom:3", Serie: 1500#, ASME-B16.20, Mat:316/Graph ;
- **508809 :** RUBAN ISOLANT ELECTRIQUE, ROUGE Ep.: 18/100 ;

La règle 11 montre que le remplacement de la « Tole Plate Lisse » de la «Chaudière Procède » implique le remplacement du « Joint Spirale Avec Anneau de Centrage » et du « Ruban Isolant Electrique » avec une confiance de 98,3%.

--------**Règle 12 :** 000501 060806 → 508809

Si

- **000501:** TOLE PLATE LISSE Ep.:1/4", Long.:96", Larg. : 48", Mat.:A283 GrD.
- **060806:** JOINT SPIRALE AVEC ANNEAU DE CENTRAGE Dia. Nom:3", Serie: 1500#, ASME-B16.20, Mat:316/Graph.

Alors avec (une confiance = 100.0 %)

- **508809:** RUBAN ISOLANT ELECTRIQUE, ROUGE Ep.: 18/100.

La règle 12 exprime que le remplacement de la « Tole Plate Lisse » et du « Joint Spirale » et « Ruban Isolant Electrique » sont ensemble avec certitude.

------**Règle 13 :** 00020804→ 00000608

Si

- **00020804:** LAMPE A REFLECTEUR INTERN 100 E 27 220-230 SPOT EXTENSIF 3 STD.

Alors avec (une confiance = 78.1 %)

- **00000608**: JOINT PLAT DE BRIDE 4"150# RF (175x114x1,6mm) SVT.ANSI B16-5 1AA .

La règle 13 exprime que le remplacement de la « Lampe à Réflecteur Interne » de la CHAUDIERE BP implique avec une confiance de 78,10% le remplacement du « JOINT PLAT DE BRIDE ».

------**Règle 14 :** 23014103 → 00000404 60001504

Si

- **23014103:** ANODE SORTIE EVAPORAT. FER PUR SASAKURA 250x150x24 26 B15021075001 (2010L).

Alors avec (une confiance = 98.2 %)

- **00000404:** JOINT PLAT 2" RF 150 20 GRAPHITEE.COMP. NOIR 105. 60. 1.6 SVT.ANSI B16-51AA.
- **60001504:** VAN.SOUPAPE 1" 0/ 0 150 FF SEA WAT+ 93 BRONZE? BRONZE PTFE 25 79 16 4 108 04 FQM10S.

La règle 14 exprime que le remplacement de «Anode Sortie Evaporat » de « Dessaleur A » implique aussi le remplacement de «Joint Plat 2 » et «Van Soupape » avec une confiance de 98,2%.

------Règle 15 : 060766 $\rightarrow$ 151204

Si

- **060766:** JOINT SPIRALE AVEC ANNEAU DECENTRAGE.

Alors avec (une confiance = 98.2 %)

- **151204:** ECHANGEUR COMPLET

La règle 15 exprime que le remplacement d'un joint spiral d'une Chaudière haute pression implique aussi le remplacement d'un échangeur complet avec une confiance de 98,2%. **Cette règle est conforme au schéma de règles SR_4.**

II. Règles d'association au niveau GPL (niveau 2)

Les règles d'association obtenues lors de la phase inter-site au niveau GPL (GP1Z et GP2Z) :

------Règle 16(Globale) : ZZ0013 → 972300

Si

- **ZZ0013**: COURROIE REF: 3850.14. M 85 POUR AERO REFRIGERANTS.

  Alors avec (une confiance = 68.4 %)

  - **972300**: DEGRIPPANT - (En bombe aerosol).

La règle 16 exprime que le remplacement de «Courroie » de « Aérocondenseur de Propane Réfrigérant » implique avec 68,4% l'utilisation de « Dégrippant ».

------Règle 17 (Globale):122010 ZZ0013 →972300

Si

- **122010**: BOULON AVEC ECROUS - 8 x 40 m/m - P/ DIVERS TRAVAUX MECANIQUE -.
- **ZZ0013**: COURROIE REF: 3850.14. M 85 POUR AERO REFRIGERANTS.

  Alors avec (une confiance = 100 %)

  - **972300**: DEGRIPPANT - (En bombe aerosol).

La règle 17 expose que le remplacement de «Boulon-Ecrous » avec «Courroie » de « AEROREFRIGERANT » implique avec 100% l'utilisation de « Dégrippant ».

------Règle 18 (Globale): 319300105 319300132 → 010230

Si

- **319300105:** SOUPAPE DASPIRATION 2eme ETAGE HOERBIGER TYPE 154R1.
- **319300132:** JOINT ALUMINIUM P/SIEGE CLAPET 2eme ETAGE DI=175mm x DE=184mm x EP= 1.5mm (voir commentaire).

  Alors avec (une confiance = 100.0 %)

  - **010230:** RHODORSIL CAF 4 TUBE DE 100g.

La règle 18 explique que le remplacement de « Soupape d'aspiration » avec « Joint Aluminium » de l'équipement «Compresseur d'air » implique avec certitude de 100% l'utilisation de « Rhodorsil ».

------**Règle 19 (Globale):** ZZ2545→122010 ZZ0013 972300

Si

- **ZZ2545:** PALIER COMPLET INFERIEUR REF:HUDSTON PRODUCTS CORP BOTTOM FIN.FAN BEARING INVNR 50080 POUR AEROS.

Alors avec (une confiance = 80 %)

- **122010:** BOULON AVEC ECROUS - 8 x 40 m/m - P/ DIVERS TRAVAUX MECANIQUE.
- **ZZ0013:** COURROIE REF: 3850.14. M 85 POUR AERO REFRIGERANT.
- **972300:** DEGRIPPANT - (En bombe aérosol).

La règle 19 exprime que le remplacement de « Palier complet inferieur» de l'équipement «Aérocondenseur de Spliter B» implique avec confiance de 80% le remplacement de «Boulon avec écrous » et «Courroie » ainsi que l'utilisation de « Dégrippant».

------**Règle 20 (Globale):** 717600 → 240688

Si

- **717600:** RELAIS THERMIQUE, 600VAC, 18/25 AMPS, TYPE RA1-DB

Alors avec (une confiance = 60 %)

- **240688:** DISQUE DE RUPTURE POUR TURBINE DE COMPRESSEUR

La règle 20 exprime que le remplacement d'un relais thermique implique le remplacement de disque pour turbine dans l'ensemble des unités GPL. **Cette règle d'association est conforme au schéma de règles SR_{10}.**

------**Règle 21 (Majoritaire) :** 491230180241→ 491230180231

Si

- **491230180241** : ENSEMBLE PORTES PALIERS REF.=C-UCF 315

Alors avec (*LPI*=0.79)

- **491230180231** : ENSEMBLE PORTES PALIERS REF.=C-UCF 215

La règle 21 exprime que le remplacement de l'ensemble des portes paliers supérieurs du condenseur de propane implique le remplacer l'ensemble des portes paliers inférieurs dans la majorité des sites GPL.

------Règle 22 (Exceptionnelle) : 493210220009 → 493210220010

Si

- **493210220009** : JOINT METALLO-PLASTIQUE A DOUBLE ENVELOPPE FORME = « O » (894 x 926 x3)

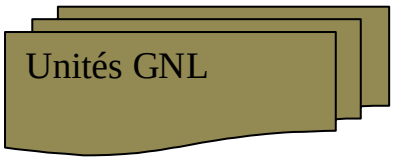
Alors avec (*EPI*=0.85)

-**493210220010** : JOINT METALLO-PLASTIQUE A DOUBLE ENVELOPPE FORME = « L » (894 x 926 x 3)

La règle 22 explique que le remplacement du joint de la forme « O » du prechauffeur implique le remplacement du joint forme « L » de façon exceptionnel.

III. Règles d'association au niveau GNL (niveau 2)

Les règles d'association obtenues lors de la phase inter-site au niveau de GNL (GL1Z et GL2Z) :



------Règle 23 (Globale) : 60001504→ 00000404 20054001

Si

- **60001504**: VAN.SOUPAPE 1" 0/ 0 150 FF SEA WAT+ 93 BRONZE? BRONZE PTFE 25 79 16 4 108 04 FQM10S.

  Alors avec (une confiance = 98.3 %)

  - **00000404**: JOINT PLAT2 "RF 150 20 GRAPHITEE.COMP. NOIR 105. 60. 1.6 SVT.ANSI B16-51AA.
  - **20054001**: ROULEAU DE TEFLON PTFE 1/2" x 50' x 0.003" TROUVAY & C RUBAN PTFE.

La règle 23 expose que le remplacement de «Van. Soupape » de « Chaudière BP » implique avec 98,3% l'utilisation de « Joint Plat » et «Rouleau de Téflon».

------Règle 24 (Globale): 110001→990509

Si

- **110001:** OXYGENE GAZEUX INDUSTRIEL.

  Alors avec (une confiance = 61.6 %)

  - **990509:** SILICONE RTV, "ETANCHEITE HERMETIQUE", BLANC.

La règle 24 expose que le remplacement de l' « Oxygène Gazeux Industriel» du «Compresseur 1er Etage Pour MCR » implique avec 61,6% le remplacement de l'«Etanchéité Hermétique».

------Règle 25 (Globale) : 060806→000501

Si

- **060806:** JOINT SPIRALE AVEC ANNEAU DE CENTRAGE Dia. Nom:3, Série:1500#, ASME-B16.20, Mat:316/Graph.

  Alors avec (une confiance = 95,1 %)

- **000501:** TOLE PLATE LISSE Ep.:1/4", Long.:96", Larg.:48", Mat.:A283 GrD.

La règle 25 explique que le remplacement du «Joint Spirale» de l'équipement «Chaudière de Procède TYPE 34-VP-18W-R» implique avec 95,1% le remplacement de «Tôle Plate».

------Règle 26 (Globale) : 000501 →252703 110001

Si

- **000501:** JOINT SPIRALE AVEC ANNEAU DE CENTRAGE Dia. Nom:3, Série:1500#, ASME-B16.20, Mat:316/Graph.

Alors avec (une confiance = 85 %)

- **252703:** JOINT ECRAN, Plan N°920349, Rep : G-8Dim.:340mm x 34mm x, Ep.:0.45mm, Mat.: Mica.
- **110001:** OXYGENE GAZEUX INDUSTRIEL.

La règle 26 explique que le remplacement du «Joint Spirale» de l'équipement «Chaudière Procède, 400 T/ Tubes de Chaudière» implique avec 85% le remplacement de «Joint Ecran» et l'utilisation de l'«Oxygène Gazeux Industriel ».

-----Règle 27 (Majoritaire) : 498383370002 777383372221 → 777383372222

Si

- **498383370002** : GARNITURE AD-3REF .=3123227
- **777383372221** : JOINT TORIQUE DI=430mmxTORE=5,7mmMAT :NBR

Alors avec (*LPI*=0.80)

- **777383372222** : JOINT TORIQUE DI=448mmx TORE=3,1mmMAT :NBR

------ **Règle 28 (Majoritaire) :** 777383372222 777383372221 → 498383370002

Si

-**777383372221** : JOINT TORIQUE DI=430mmxTORE=5,7mmMAT :NBR

-**777383372222** : JOINT TORIQUE DI=448mmx TORE=3,1mmMAT :NBR

Alors avec (*LPI*=0.75)

- **498383370002** : GARNITURE AD-3REF .=3123227

Les règles d'association 27 et 28 expriment que le remplacement des deux joints et la garniture du bras de chargement de gpl liquide jetee sont ensemble sur la majorité des sites GPL.

------Règle 29 (Exceptionnelle) : 493210220010 → 493210220009

Si

-493210220010 : JOINT METALLO-PLASTIQUE A DOUBLE ENVELOPPE FORME = « L » (894 x 926 x 3)

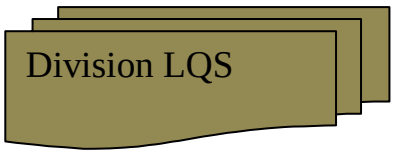
Alors avec (*EPI*=0.74)

-493210220009 : JOINT METALLO-PLASTIQUE A DOUBLE ENVELOPPE FORME = « O » (894 x 926 x3)

La règle 29 explique que le remplacement du joint de la forme « L » du préchauffeur implique le remplacement du joint forme « O » de façon exceptionnel.

IV. Règles d'association au niveau LQS (niveau 3)

Les règles d'association obtenues lors de la synthèse globale au niveau LQS (GLZ et GPZ) :



------Règle 30 (Globale) : 60001504 → 23014103

Si

- **60001504** : VAN.SOUPAPE 1" 0/ 0 150 FF SEA WAT+ 93 BRONZE? BRONZE PTFE 25 79 16 4 108 04 FQM10S.

  Alors avec (une confiance = 94.89 %)

  - **23014103 :** ANODE SORTIE EVAPORAT. FER PUR SASAKURA 250x150x24 26 B15021075001 (2010L).

La règle 30 signifie que le remplacement de «Van. Soupape » de « Chaudière BP » implique avec 94,89% le remplacement de l' «Anode Sortie Evaporat» et ceci dans l'ensemble des complexes.

------Règle 31 (Globale) : ZZ0023→ZZ0029

Si

- **ZZ0023**: VIS REF: F/900024/3/001 PLAN:F/12477 12463.01 F/12463.02 REPERE:27 MAT INOX 316.

  Alors avec (une confiance = 71.39 %)

  - **ZZ0029:** TE EGAL EN INOX - 1/2" -P/DIVERS ASSEMBLAGE DE RACCORDERIE ET TUBE EN INOX.

La règle 31 indique que le remplacement du «Van. Soupape » de «CHROMATOGRAPHE EN LIGNE SEPARATION A» implique avec 71,39 % le remplacement de l' «Divers Assemblage de Raccorderiez et Tube en Inox» dans l'ensemble des complexes.

------Règle 32 (Exceptionnelle) : 23014103→60001504

Si

- **23014103**: ANODE SORTIE EVAPORAT. FER PUR SASAKURA 250x150x24 26 B15021075001 (2010L).

  Alors avec (*EPI*=0.79)

  - **60001504:** VAN.SOUPAPE 1 0 / 0 150 FF SEA WAT+ 93 BRONZE? BRONZE PTFE 25 79 16 4 108 04 FQM10S.

La règle 32 montre que le remplacement de l'«Anode Sortie Evaporat» de «Chaudière BP » implique le remplacement de la «VAN.SOUPAPE» de façon exceptionnelle.

------ Règle 33 (Majoritaire): 122010 → ZZ0013

Si

- **122010**: BOULON AVEC ECROUS - 8 x 40 m/m - P/ DIVERS TRAVAUX MECANIQUE.

  Alors avec (*LPI*=0.72)

  - **ZZ0013:** COURROIE REF: 3850.14. M 85 POUR AERO REFRIGERANT.

La règle 33 signifie que le remplacement de «Boulon Avec Ecrous» de «Aéroréfrigérant » implique le remplacement de «Courroie» dans la majorité des complexes.

------ Règle 34 (Majoritaire) : 00024708→12141305

Si

- **00024708**: JOINT SPIRALE "1/2 RF 300 4 INOX 54.1 14.7 4.5 SVT.API 601 300/600.

  Alors avec (*LPI*=0.70)

  - **12141305:** ARBRE DE POMPE PN:R104-733 2229 MAT:316SS ITEM:122.

La règle 34 signifie que le remplacement d'un joint d'un alternateur implique le remplacement d'un arbre de pompe dans la majorité des complexes. **Cette règle d'association est conforme au schéma de règles SR_3.**

Bibliographie

[Agrawal et al. 1993] Rakesh Agrawal, T.Imielinski and A.Swami. Mining Association rules between sets of items in large databases. *In proceeding of the 1993 ACM SIGMOD international conference on Management of Data (SIG-MOD'93)*, pages 207-216. ACM Press, May 1993.

[Agrawal et al. 1994] Rakesh Agrawal and Ramakrishnan Srijant. Fast Algorithms for association Rules. *In Proceeding of 20th VLDB Conference* Santiago, Chile 1994.

[Anderson.B.S et al. 1998] B.S. Anderson and A.W. Moore. ADtrees for fast counting and for fast learning of association rules. *In proceedings Fourth International Conference on Knowledge Discovery and Data Mining*, 1998.

[Animesh.A et al. 2010a] Animesh Adhikari, Pralhad Ramachandrarao and Witold Pedrycz. Study of select items in different data sources by grouping. *Knowl Inf Syst, Springer-Verlag* London Limited 2010.

[Animesh.A et al. 2010b] A. Adhikari, Pralhad Ramachandrarao and Witold pedrycz , Developing Multi-database Mining Applications. *Advanced Information and Knowledge Processing*, DOI 10.1007/978-1, 2010.

[Andrei Olaru et al. 2009] Andrei Olaru, Claudia Marinica and Fabrice Guillet. Local Mining of association Rules with Rule Schemas. *In Proceedings of the IEEE Symposium on Computational Intelligence and Data Mining*, Nashville, TN, USA, March 30, 2009 - April 2, 2009. IEEE 2009.

[B.Liu et al. 1998] Bing Liu, W.Hsu and K.Wang. Helping user identifying interesting association rules. *Technical Report*, 1998

[B.Liu et al. 1999] Bing Liu, Wynne Hsu, Lai-Fun Mun and Hing-Yan Lee. Finding interesting patterns using user expectations. *IEEE Transactions on Knowledge and Data Engineering*, pages 817–832, 1999.

[Claudia.M et al. 2010] Claudia Marinica and Fabrice Guillet. Knowledge-Based Interactive Postmining of Association Rules Using ontologies. *IEEE Transactons on Knowledge and Data Engeneering*, Vol 22 No 6. June 2010.

[Claudia.M. 2010] Claudia Marinica. Association Rule Interactive post-processing using rules schemas and ontologies- ARIPSO. *Thèse de doctorat* Octobre 2010.

[Claudia.M et al. 2008] Claudia Marinica, Fabrice Guillet and Henri Briand. Vers la fouille de règles d'association guidée par des ontologies et des schémas de règles. *LINA-equipe COD,* Article 2008. Ecole polytechnique de l'université de Nantes.

[Couturier.O. 2005] Couturier Olivier. Contribution à la fouille de données : règles d'association et interactivité au sein d'un processus d'extraction de connaissances dans les données. Thèse de doctorat de l'université d'Artois, 12 décembre 2005.

[C.Zhang et al. 2005] Chengqi Zhang, Jeffrey Xu Yu and Shichao Zhang. Identifying Interesting Patterns in Multi-databases. *In Studies in Computational Intelligence (SCI) 91-112 (2005) Springer-Verglag* Berlin Heidelberg 2005.

[Darroch.J et al. 1976] Darroch.J.N and Ratcliff.D. Generalized iterative scaling for log-linear models. *Annals of Mathematical Statistics*, 43:1470– 1480, 1972.

[Dechter.R 1999] Dechter.R. Bucket elimination: A unifying framework for structure-driven inference. *In Artificial Intelligence, Volume 113, pages 41–85, 1999.*

[Dmitry.P et al. 2000] Dmitry Pavlov, Heikki Mannila and Padhraic Smyth. Probabilistic Models for Query approximation with large Sparse Binary Data Sets. *Proc. Uncertainty in AI Conf. (UAI ',00)*, pp. 465-472, 2000.

[Dmitry.P et al. 2003] Dmitry Pavlov, Heikki Mannila and Padhraic Smyth. Beyond Independence: Probabilistic Models for Query approximation on Binary Transaction Data. *Knowledge and Data Engineering, IEEE Transactions* on (volume : 15, issue : 6), page : 1409-1421, 2003.

[Fayyad.U.M et al. 1996] Fayyad U.M, Piatetsky-Shapiro and P.Smyth. From Data Mining to Knowledge Discovery: an Overview. *In Advances in Knowledge Discovery and Data Mining*, pages 1-34, 1996.

[Gregory.P et al.1991] Gregory Piatetsky-Shapiro and William Frawley. Knowledge Discovery in Databases. *AAAI Press publications* 1991.

[Hartigan.J.A et al. 1979] Hartigan.J.A and Wong.M.A. A K-Means Clustering Algorithm. *Journal of the Royal Statistical Society, Series C*, 100–108, 1979.

[Heijst et al. 1997] Heijst.G .Van , Chreiber.A.T H.S and Ielinga.B.J.W. Using explicit ontologies in KBS development. *Int. J. Human– Computer Studies* (1997) 45,183–292.

[Inhaúma.N et al. 2013] Inhaúma Neves Ferraz and Ana Cristina Bicharra Garcia. Ontology in association rules. *SpringerPlus* 2013.

[Jason.O et al. 1999] Jason Ong and Syed Sibte Raza Abidi. Data Mining using Self-Organizing Kohonen maps : A Technique for Effective Data Clustering & Visualisation. *In International Conference on Artificial Intelligence (IC-AI'99),* 1999.

[Karypis.G et al. 1999] Karypis George, Eui-Hong Han and Vipin Kumer. CHAMELEON: A Hierarchical Clustering Algorithm Using Dynamic Modeling. *IEEE Computer Society Press p-68-75,* 1999.

[Khiat.S et al. 2014a] Khiat Salim, Belbachir Hafida and Rahal Sid Ahmed. MAROR: Multi-level Abstraction of Association Rule using Ontology and Rule schema. *In International Journal of Information Technology and Computer Science (IJITCS)*. Vol.6 N12-4, November 2014.

[Khiat.S et al. 2014b] Khiat Salim, Belbachir Hafida and Rahal Sid Ahmed. A Probabilistic Models for Local Pattern Analysis. *In Journal of Information and processing System (JIPS)*. Vol.10, No.1, pp.145-161, March 2014.

[Malvestuto.F.M 1992] Malvestuto.F.M. A unique formal system for binary decompositions of database relations, probability distributions and graphs. *Information sciences, 59:21-52,* 1992.

[Mannila.H et al. 1996] Mannila.H and Toivonen.H. Multiple uses of frequent sets and condensed representations. *In proceeding International Conference in Knowledge Discovery In Databases (KDD)*, 1996.

[McBride. 2004] Brian McBride. The Resource Description Framework (RDF) and its Vocabulary Description Language RDFS. *Handbook on Ontologies*, 2004

[Nicolas.P. 2000] Nicolas Pasquier. Data Mining : Algorithmes d'extraction et réduction des règles d'association dans les bases de données. *Thèse de doctorat Université Clermont-ferrand II*, Janvier 2000.

[Pearl.J. 1988] Pearl.J. Probabilistic Reasoning in Intelligent Systems Networks of Plausible Inference. *Morgan Kaufmann Publishers Inc*., 1988.

[PM. 1993] Approche des politiques maintenances des complexes LTG. Document fourni par l'entreprise *Sonatrach* 1993.

[Poosala.V et al. 1997] Poosala.V and Ioannidis.Y. Selectivity estimation without the attribute value independence assumption. *In proceedings of the 23rd international conference on Very Large Data Bases (VLDB'97), pages 486-495*. San Francisco, 1997.

[Quinlan.R 1993] Quinlan.R. C4.5: Programs for Machine Learning. *Morgan Kaufmann Publishers Inc, 1993*

[Ramkumar.T et al. 2008] Ramkumar Thirunavukkarasu and Rengaramanujam Srinivasan. Modified algorithms for synthesizing high-frequency rules from different data sources. *Knowl Inf Syst 17:313–334, Springer-Verlag* London Limited 2008

[Ramkumar.T et al. 2009]Ramkumar Thirunavukkarasu and Rengaramanujam Srinivasan. The Effect of Correction Factor in Synthesizing Global Rules in a Multi-Database Mining Scenario. *Journal of Applied Computer Science, no. 6 (3)*, Suceava, pp.33, 2009.

[Ramkumar.T et al. 2010] Ramkumar Thirunavukkarasu and Rengaramanujam Srinivasan. Multi-Level Synthesis of Frequent Rules from Different Data-Sources. *International Journal of Computer Theory and Engineering*, Vol. 2:195-204, 2010.

[Ramkumar.T et al. 2013] Ramkumar Thirunavukkarasu, Hariharan.S and Selvamuthukumaran.S. A survey on mining multiple data sources. *WIREs Data Mining Knowl Discov*, Vol. 3, 2013.

[Raymond.T et al. 1994] Raymond.T and Jiawei Han. CLARANS : A Method for Clustering Objects for Spatial Data Mining. *IEEE Computer Society*, 1994.

[Szymon.J et al. 2002a] Szymon Jaroszewicz and Dan A.Simovici. Support Approximations Using Bonferroni-type Inequalities. *Lecture Notes in Computer Science Volume 2431, 2002, pp 212-224, Springerlink.*

[Szymon.J et al. 2002b] Szymon Jaroszewicz, Dan A.Simovici and Ivo Rosenberg. An inclusion-exclusion result for Boolean polynomial and its applications in data mining. *In proceedings of the Discrete Mathematics in DM Workshop SIAM DM Conference*, Washington .D.C. 2002.

[Thomas.G. 1993] Thomas Gruber R. A Translation Approach to Portable Ontology specifications. *In Knowledge Acquisition*,5 (2):199-220, 1993.

[Tian.Z et al. 1996] Tian Zhang, Raghu Ramakrishnan and Mirou Livy. BIRCH: An Efficient Data Clustering Method for Very Large Databases. *In Proc. of the ACM SIGMOD Intl. Conference on Management of Data (SIGMOD),* 1996.

[Toon.C et al. 2006] Toon Calders and Bart Goethals. Quick Inclusion-Exclusion. *University of Antwerp, Belgium.Springer Verlag Berlin Heidelberg* 2006.

[Tunkelang.D et al. 2001] Tunkelang Daniel and Endeca. Making the Nearest Neighbor Meaningful. *Citeseer*, 2001.

[Xindong.W et al. 2003]Xindong Wu and Shichao Zhang. Synthesizing High-Frequency Rules from Different Data Sources. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 15, No. 2, 2003.

[Zhang.S et al. 2003] Zhang Shichao, Xindong Wu and Chengqi Zhang. Multi-Database Mining. *In Proc. of IEEE Computational Intelligence* Bulletin, 2003.

Publications scientifiques

Publications parues dans les journaux scientifiques :

1

« A Probabilistic Models for Local Pattern Analysis». Journal of Information and processing System (JIPS). ISSN: 1976-913X(Print), ISSN: 2092-805X(Online). Vol.10, No.1, pp.145-161, Mars 2014.
Lien: www. http://jips-k.org/q.jips?cp=pp&pn=305

2

- **« MAROR: Multi-level Abstraction of Association Rule using Ontology and Rule schema » IJITCS: International Journal of Information Technology and Computer Science. ISSN: 2074-9007 (Print), ISSN: 2074-9015 (Online). Vol –6 N12-4, Novembre 2014.**

Publications parues dans les proceedings des conférences :

1

- **« Multi-Level Synthesis of Frequent Rules from Different Databases Using A Clustering Approach ». The 10th International Conference on Data Mining July 2014 DMIN'14 in Las Vegas Nevada USA.**

2

- **« Clomaint: a New Data Mining Algorithm in Maintenance Petroleum Plants » SETIT 2009, 5th International Conference: Sciences of Electronic,Technologies of Information and Telecommunications March 22-26, 2009 – TUNISIA IEEE proceeding conference**
Lien:http://www.setit.rnu.tn/last_edition/setit2009/Information%20Processing/ 179.pdf