Fouille de règles d'association disjonctives à partir ditemsets non fréquents.

Ines Hilali

Université de Tunis El Manar

-Faculté des Sciences de Tunis-

Université de Cergy Pontoise

-Département des Sciences Informatiques UFR des Sciences et Techniques-

Thèse de Doctorat

Pour Obtenir le titre de

Docteur en Informatique

de

l'Université de Cergy Pontoise

École doctorale : École Doctorale Sciences et Ingénierie

et de

la Faculté des Sciences de Tunis

École doctorale : Mathématiques, Informatique, Sciences et Technologie de la Matière

présentée et soutenue publiquement le 18 Décembre 2015 par

Inès HILALI JAGHDAM

Fouille de Règles d'Association Disjonctives à Partir d'Items Non Fréquents

Jury

Rapporteurs :	Arnaud GIACOMETTI,	Professeur Université François-Rabelais de Tours
	Nadia ESSOUSSI,	Maître de conférences F.S.E.G. de Nabeul
Examinateurs :	Nicolas SPYRATOS,	Professeur Université Paris-Sud
	Mohamed-Mohsen GAMMOUDI,	Professeur Université de Manouba
Directeur :	Dominique LAURENT,	Professeur Université de Cergy Pontoise
Co-Directeur :	Sadok BEN YAHIA,	Professeur Faculté des Sciences de Tunis
Co-Encadreur :	Tao-Yuan JEN,	Maître de conférences Université de Cergy Pontoise

Laboratoire IPAH Laboratoire ETIS-UMR CNRS 8051

Remerciements

Je présente toute ma reconnaissance à Monsieur Arnaud Giacometti Professeur à l'Université François-Rabelais de Tours et à Madame Nadia Essoussi Maître de conférences à la Faculté des Sciences Économiques et de Gestion de Nabeul pour avoir accepté d'être rapporteurs de ce travail.

Je tiens également à remercier les deux examinateurs : Monsieur Nicolas Spyratos, Professeur à l'Université Paris-Sud, et Monsieur Mohamed-Mohsen Gammoudi Professeur à l'Université de Manouba de m'avoir accepté de participer au jury de cette thèse.

Mes remerciments sincères vont à mes directeurs de thèse, Monsieur Dominique Laurent, Professeur à l'Université de Cergy Pontoise, et Monsieur Sadok Ben Yahia, Professeur à la Faculté des Sciences de Tunis, pour leur encadrement très efficace dans la conduite de cette thèse.

Il m'est impossible d'exprimer ma gratitude à Monsieur Tao-Yuan Jen, Maître de conférences à l'Université Cergy Pontoise, qui a su co-diriger ce travail de thèse avec beaucoup de patience et de rigueur scientifique.

Merci à Madame Claudia Marinica Maître de conférences à l'Université Cergy Pontoise de m'avoir fait l'honneur de participer à ce jury de thèse.

Je remercie tous les membres du laboratoire Équipes Traitement de l'Information et Systèmes de l'Université de Cergy Pontoise qui m'ont accueillie pendant ces longues années de thèse.

Je tiens également à remercier les membres du laboratoire Informatique en Programmation Algorithmique et Heuristique à la Faculté des Sciences de Tunis et surtout son directeur Professeur Khaled Bsaies pour le financement de mes séjours à Paris.

Je remercie toutes les personnes avec qui j'ai partagé mes études et notamment ces années de thèse.

Je tiens à remercier chaleureusement mon beau frère et sa femme et ma tante pour leur aide précieuse qu'ils m'ont apportée lors de mes séjours à Paris.

Je remercie très chaleureusement mon médecin Hayet Kaaroud Professeur à l'hôpital Charles Nicolle de Tunis au service de médecine interne qui m'a beaucoup encouragée de finir ma thèse malgré la maladie chronique que je souffrais depuis 2013.

L'aboutissement de cette thèse aurait été plus difficile sans le soutien bienveillant et chaleureux de ma famille. Je remercie mes parents, ma sœur, mes deux frères et ma belle

mère pour leur soutien qui ne m'a jamais fait défaut.

Je garde enfin mes remerciements amoureux à mon époux pour son soutien et son encouragement au quotidien durant ces années de thèse.

A mon trésor Ahmed Yassine

Table des matières

Table des figures

Liste des Algorithmes

INTRODUCTION GÉNÉRALE

Le récent développement de la technologie permet de doter les ordinateurs d'une grande capacité de stockage de données, et par conséquent, le volume des données stockées dans les systèmes informatiques ne cesse de croître. Ces données stockées sous forme de banques de données concernent différents domaines : scientifique, commercial, industriel, médical, etc [Dieng, 2011]. Ces masses de données brutes, telles qu'elles sont stockées, sont généralement exploitables mais contiennent néanmoins des informations cachées utiles et non triviales. Il est donc important, voire vital, de les analyser afin d'en extraire des informations qui permettent de répondre aux besoins des utilisateurs en matière de prise de décision. La communauté scientifique s'est intéressée alors à la recherche de ces données implicites que nous appelons *connaissances* ou *motifs intéressants*.

Ainsi, l'extraction de motifs intéressants est une problématique qui a émergé au cours des deux dernières décennies sous le nom générique de *fouille de données* et de la découverte de connaissances.

Selon Han et Kamber, la fouille de données est définie comme étant "la découverte d'informations intéressantes, non triviales, implicites, préalablement inconnues et potentiellement utiles à partir de grandes bases de données" [Han et Kamber, 1992]. Cet axe de recherche est qualifié de multi-disciplinaire car il se situe au confluent des différents domaines de recherche tels que les statistiques, l'algorithmique, l'intelligence artificielle, les bases de données, etc [Salleb, 2003].

La fouille de données est au cœur d'un processus plus général et complexe appelé *Extraction de Connaissances à partir des Bases de Données* (**ECBD**) (ou Knowledge Discovery in Databases en anglais). Ce dernier est défini comme étant un processus interactif, itératif et non trivial d'extraction de connaissances implicites et de nouvelles informations valides, précédemment inconnues, potentiellement utiles, et compréhensibles à partir de données stockées dans les bases de données, [Fayyad et al., 1996] et [Frawley et al., 1992].

L'**ECBD** en tant que processus global d'extraction de connaissances comporte les trois étapes suivantes (cf. figure 1) :

Pré-traitement de données cette étape consiste à récupérer des données issues de plusieurs sources et sous différents formats, de les nettoyer, d'unifier leurs formats, de les intégrer, etc.

Fouille de données cette étape concerne l'ensemble des techniques et des méthodes (i.e., algorithmes) permettant d'extraire de l'information cachée dans les masses de données.

Post-traitement cette étape consiste à caractériser il s'agit d'identifier les motifs intéressants dans le but de les présenter à l'utilisateur.

La fouille de données en tant que l'étape centrale et algorithmique de l'ECBD, constitue le cadre général dans lequel s'inscrit notre travail de thèse.

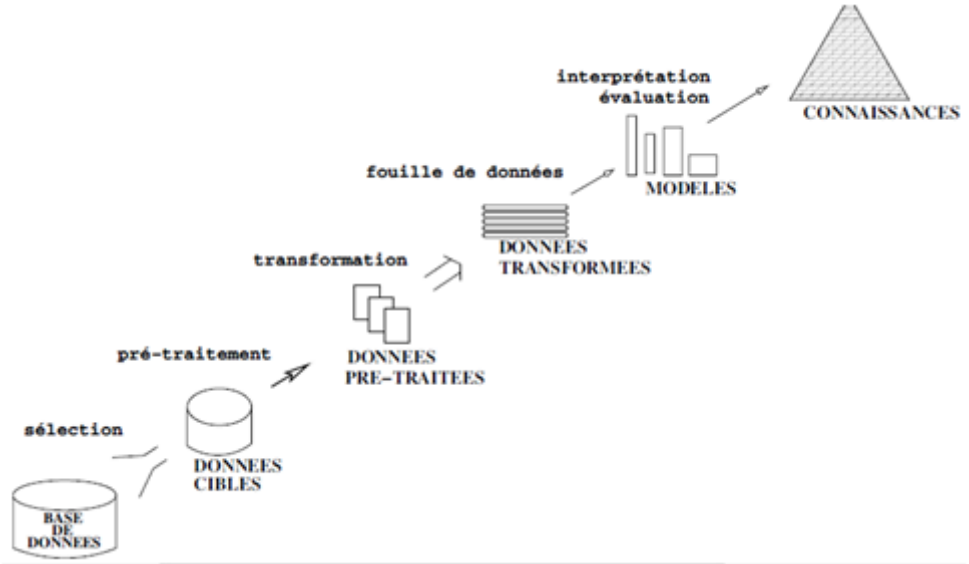


FIGURE 1 – Processus d'Extraction de Connaissances à partir des Bases de Données. [Loubna, 2012]

De manière générale, le problème de la recherche de règles d'association est classé comme étant une méthode d'apprentissage non supervisée permettant de découvrir de relations d'association ou de corrélations intéressantes à partir d'un ensemble de transactions.

Ce problème a été introduit par Agrawal, [Agrawal et al., 1993], pour l'analyse de tickets de caisse dans un panier de la ménagère, dans le but d'en extraire des règles de la forme de "80 % des clients qui achètent des couches bébé et des lingettes achètent aussi du lait pour bébé". Ensuite, les recherches n'ont pas cessé de se développer pour couvrir différents domaines, e.g., biologique, social, détection de fraudes et des intrusions, etc. En plus, les règles d'association ont été aussi utilisées dans un objectif de classification [Bouzouita et al., 2006] où les auteurs ont proposé une nouvelle méthode de classification associative en utilisant les règles d'association.

Nous nous intéressons dans le cadre de cette thèse au problème de la fouille de *règles disjonctives* à partir d'items *non fréquents*. Dans ce contexte, les items de la base de données sont exprimés à l'aide des concepts dans des *structures hiérarchiques* telles que les *taxonomies*.

Ce travail de thèse porte principalement sur : la considération de l'aspect *disjonctif* dans l'extraction de motifs fréquents, les motifs *rares* (non fréquents) et l'utilisation des *structures hiérarchiques* pour caractériser les informations pertinentes extraites.

La fouille d'itemsets *non fréquents* s'est avérée utile dans certaines applications telles que la prédiction des défaillances dans le secteur de télécommunications, la détection des intrusions et des fraudes, les diagnostiques des maladies, etc. Il est bien connu que, la fouille des motifs non fréquents est un problème difficile étant donné que la majorité des algorithmes de fouille de données extraient des motifs fréquents [Szathmary et al., 2006].

D'autre part, la considération du support disjonctif des itemsets permet l'extraction de règles disjonctives telles que "les gens qui achètent l'article x achètent aussi l'article y *ou* l'article z" [Nanavati et al., 2001]. De telles règles s'avèrent utiles dans de nombreux domaines parmi lesquels nous citons l'analyse des données médicales par exemple les experts en médecine peuvent vouloir trouver des associations entre la consommation de bière indépendamment de leur type et des caractéristiques du patient (telles que la tension artérielle, le niveau de cholestérol, etc) [Ralbovsky et Kuchar, 2007], l'analyse des réseaux sociaux [Vimieiro et Moscato, 2012], etc.

Enfin, l'utilisation des taxonomies dans le domaine de la fouille de règles d'association permet d'apporter une réponse au problème bien connu de la trop grande quantité de motifs extraits. En effet, en considérant la relation de généralisation associée, certains motifs peuvent être agrégés, ce qui permet de diminuer la quantité de motifs extraits.

Nos contributions dans le cadre de cette thèse concernent l'extraction d'itemsets fréquents selon leurs supports disjonctifs et la fouille des règles d'association disjonctives intéressantes à partir d'items non fréquents.
En plus, nos contributions concernent l'incorporation des taxonomies dans le processus de génération de règles d'association disjonctives.

Dans une première contribution, nous abordons le problème d'extraction d'itemsets disjonctifs-fréquents minimaux en se basant sur leurs supports *disjonctifs* et non conjonctifs. Notre approche consiste ainsi à fouiller des itemsets fréquents à partir d'items non fréquents via un algorithme par niveaux nommé Disapriori.

Dans une deuxième contribution, et en liaison étroite avec la première, nous étudions la génération de règles d'association disjonctives comme des implications intéressantes entre les motifs disjonctifs-fréquents minimaux précédemment extraits. Cependant, l'ensemble d'itemsets disjonctifs-fréquents minimaux ne sont pas tous pertinents. Pour filtrer les disjonctions fréquentes non utiles, nous supposons une taxonomie sur l'ensemble des items et une mesure de similarité définie à partir de la taxonomie. Cette mesure permet de définir la notion d'itemset homogène par rapport à un seuil d'homogénéité.

Dans une dernière contribution, nous généralisons ce qui est présenté précédemment en termes de règles disjonctives pour construire des règles d'association généralisées. En effet, partant d'un itemset dont on connaît le support disjonctif et ceux de tous ses sous-ensemble. Grâce aux *règles de De Morgan* et aux *identités d'inclusion-exclusion*, nous

montrons qu'il est possible de trouver le support conjonctif de cet itemset. Par la suite, *quatre* supports différents possibles d'un itemset sont définis (i.e., disjonctif, conjonctif, négation sur disjonction et négation sur conjonction), nous montrons que l'on peut considérer *seize* formes différentes de règles d'association et qu'il est possible de calculer leurs mesures d'intérêt à savoir le support et la confiance.

Dans ce qui suit, nous présentons un exemple illustrant la problématique évoquée dans le cadre de cette thèse et notre façon de la résoudre.

Exemple

TID	$Items$
T_1	l_1 l_2 l_3
T_2	l_2 l_3 l_4
T_3	l_1 l_4
$\vdots$	$\vdots$

FIGURE 2 – Exemple d'une base de transactions.

Considérons la base des transactions de la figure 2, $\{T_1, T_2, T_3, \ldots\}$ est l'ensemble de transactions et $\{l_1, l_2, l_3, l_4 \ldots\}$ l'ensemble des items qui composent ces transactions. Sous entendu, $\{T_1, T_2, T_3, \ldots\}$ contiennent d'autres items à part $\{l_1, l_2, l_3, l_4 \ldots\}$.
Nous supposons de plus que :

— T_1, T_2 et T_3 sont les seules transactions contenant au moins un des items (livres) $l_1, \ldots, l_4$.
— Un item est fréquent s'il apparaît au moins dans *trois* transactions. Seuls les minimaux sont intéressants parmi tous les fréquents.

Par conséquent, l_1, l_2 , l_3 et l_4 sont donc non fréquents. Cependant, (l_1 ou l_2) est *fréquent* puisque l_1 ou l_2 apparaît dans *trois* transactions. L'itemset $\{l_1, l_2\}$ est alors dit d-fréquent (d : disjonctif).
On notera de plus que $\{l_1, l_2, l_i\}$ est d-fréquent pour n'importe quel item l_i, car au moins un des éléments de l'itemset $\{l_1, l_2\}$ seul apparaît dans *trois* transactions.

Cependant, tous les itemsets disjonctifs-fréquents minimaux ne sont pertinents. Par exemple, $\{l_1, l_3\}$ pourrait ne pas être pertinent parce que l_1 est un livre en informatique et l_3 est un roman.

Pour filtrer les disjonctions non pertinentes, nous supposons :

1. Une mesure de similarité sémantique entre les items à partir desquels une mesure d'homogénéité entre ensembles peut être définie.
2. Un seuil d'homogénéité, défini par l'utilisateur permettant de caractériser les ensembles qui sont homogènes.

Le problème est alors de fouiller de règles d'association intéressantes entre des itemsets constitués d'items non fréquents, et tels que :
— ces itemsets sont disjonctifs-fréquents,
— homogènes,
— minimaux nous verrons néanmoins que la conclusion de la règle pourra ne pas être minimale afin de pouvoir extraire des règles pertinentes,
— disjoints,
— la confiance d'une règle est la probabilité de trouver la conclusion sachons qu'on a la prémisse,

Exemple 1. *Nous notons que cet exemple est dans le cadre de l'exemple précédent et sert pour illustrer les items précédents. Ainsi, nous supposons que* (l_1, l_2) *et* (l_3, l_4) *sont deux disjonctions homogènes, alors la règle d'association :* $l_1, l_2 \rightarrow l_3, l_4$ *est intéressante parce que*

1. $\{l_1, l_2\}$ *et* $\{l_3, l_4\}$ *sont d-fréquents, homogènes et disjoints ;*
2. T_1, T_2 *et* T_3 *supportent* (l_1, l_2) *et* (l_3, l_4) *i.e. la règle est fréquente*
3. *Chaque transaction contenant* l_1 *ou* l_2 *contient aussi* l_3 *ou* l_4*, i.e. la confiance de la règle est égale à* 1.

Le reste du mémoire est construit au tour de deux principales parties à savoir l'état de l'art et la contribution. Ainsi, son organisation est comme suit :

Le premier chapitre est dédié à présenter une vue d'ensemble sur le problème d'extraction d'itemsets fréquents et la génération des éventuelles implications entre eux.

Dans le ***deuxième chapitre***, nous présentons un panorama des principales approches de la littérature qui sont connexes à notre sujet de thèse. Nous mentionnons en fait les principaux travaux concernant l'extraction de motifs (qu'ils soient itemsets, règles d'association, représentations condensées, etc). Nous examinons en particulier le formalisme d'extraction d'itemsets fréquents et la génération de règles d'association en discutant les différentes approches qui ont été proposées dans cet axe pour le cas des motifs fréquents.
Pour la fouille des motifs non fréquents, nous sensibilisons le lecteur à l'utilité de ces motifs dans certaines applications du monde réel. Ensuite, nous étudions les principales approches destinées à leur extraction et nous résumons les principales contributions dans cet axe.

Ensuite, nous consacrons le ***troisième chapitre*** de l'état de l'art aux travaux liés aux ontologies en particulier les taxonomies dans le domaine des règles d'association. Nous étudions ainsi, les taxonomies et les principales mesures de similarité qui ont été

proposées dans la littérature. Ensuite, nous étudions les taxonomies dans le cadre du processus de génération de règles.

La deuxième partie est consacrée à nos contributions dans le cadre de cette thèse.

Tout d'abord, dans le ***quatrième chapitre***, nous abordons l'exploration de l'espace disjonctif d'itemsets. Nous extrayons tous les *itemsets disjonctifs-fréquents minimaux* à partir d'une table transactionnelle et selon un seuil de support minimal *minsup*, à l'aide d'un algorithme par niveau appelé DISAPRIORI. Nous montrons que l'ensemble de ces itemsets extraits constitue une représentation concise approximative de l'ensemble de tous les itemsets disjonctifs, i.e., nous pouvons déduire pour n'importe quel itemset disjonctif son état de fréquence. Notre algorithme a été validé et testé sur des données réelles et synthétiques.

Le ***cinquième chapitre*** est considéré comme étant une poursuite du chapitre précédent pour la génération des règles d'association valides entre les itemsets disjonctifs. Dans ce cadre, et en supposant l'existence d'une *taxonomie* sur tous les items de la base de données, nous utilisons une mesure de similarité *Overall-Relatedness* permettant de qualifier l'homogénéité d'itemsets disjonctifs-fréquents minimaux (extraits dans le ***quatrième chapitre***), qui sont : homogènes (satisfaisant le seuil d'homogénéité) ou hétérogènes (le cas contraire). Ensuite, nous construisons des implications entre des itemsets disjonctifs fréquents homogènes en prémisse et en conclusion et nous calculons les mesures (support et confiance) de ces implications. Nous notons que l'utilisation de la taxonomie à cette étape est très avantageuse dans le sens où elle limiterait la génération des motifs fréquents et des règles d'association à ceux homogènes uniquement.
Pour valider notre proposition, nous l'avons testée sur des données disponibles à cette adresse `http://www.swisspanel.ch`. La base de données Panel Suisse de Ménages (PSM), s'intéresse au changement social dans la population Suisse, notamment la dynamique de l'évolution des ses conditions de vie.

Le ***sixième chapitre*** généralise les formes de règles disjonctives extraites dans le ***cinquième chapitre***. En effet, outre les règles disjonctives extraites dans le chapitre précédent, les différentes formes de calcul des supports *conjonctifs*, *disjonctifs* et *négatifs* montrent *seize* formes possibles de règles dont on étudie la détermination des valeurs de support et la confiance. Ainsi, nous considérons au départ des règles d'association disjonctives avec des prémisses et des conclusions *non fréquentes* et dont la disjonction (Prémisse $\vee$ Conclusion) est *fréquente minimale*. Ensuite, nous calculons les mesures relatives aux différentes formes généralisées qui peuvent en être générées. L'étude de différentes formes des règles généralisées à partir de la forme de base i.e., disjonctive est accomplie grâce aux identités d'inclusion-exclusion. Cette approche a été validée par des expérimentations réalisées sur des données biologiques.

La **dernière partie** résume les principales contributions dans le cadre de cette thèse

et donne des perspectives pour des futurs travaux de recherche.

Première partie

ÉTAT DE L'ART

Chapitre 1

Fondements mathématiques

1.1 Introduction

La génération de règles d'association à partir d'itemsets fréquents est l'une des plus connues techniques de la fouille de données introduite par [Agrawal et al., 1993]. Cette technique est basée sur des fondements mathématiques issus de l'*Analyse de Concepts Formels* .
Ce premier chapitre de thèse rappelle ces fondements dans le cadre de l'extraction d'itemsets fréquents (première section) et de règles d'association (deuxième section).

1.2 Extraction d'itemsets fréquents

Dans cette section, nous introduisons les concepts de base utilisés dans le domaine de l'extraction de motifs fréquents à partir d'une table transactionnelle. Nous généralisons par la suite ces concepts dans les cas d'une table relationnelle et multidimensionnelle, étant donné que la fouille d'itemsets fréquents est basée sur le même principe pour tous ces modèles de base de données à part quelques spécifications que nous verrons par la suite.

1.2.1 Concepts de base pour l'extraction d'itemsets fréquents

L'extraction d'itemsets fréquents et la génération de règles d'association sont basés sur un contexte d'extraction présentant l'ensemble des transactions de la base de données.

Définition 1. ***Contexte d'extraction***
Un contexte d'extraction est un triplet : $\mathcal{K} = (\mathcal{O}, \mathcal{I}, \mathcal{R})$ *tels que* $\mathcal{O}$ *et* $\mathcal{I}$ *sont deux ensembles finis respectivement d'objets (ou transactions) et d'items (ou attributs), et* $\mathcal{R} \subseteq \mathcal{O} \times \mathcal{I}$ *est une relation binaire entre les objets et les items. Un couple* $(\mathrm{o}, \mathrm{i}) \in \mathcal{R}$ *dénote que l'objet* $o \in \mathcal{O}$ *contient l'item* $i \in \mathcal{I}$ *(noté* $o\mathcal{R}i$ *), [Pasquier, 2000].*
Chaque transaction est identifiée par un identificateur appelé TID (Tuple IDentifier).

Exemple 2. *Soit le contexte d'extraction illustré par la figure 1.1.*

Dans ce dernier, nous avons $\mathcal{O} = \{T_1, \ldots, T_6\}$ *et* $\mathcal{I} = \{a, b, c, d, e, f\}$.

TID	a	b	c	d	e	f
T_1	×	×				
T_2	×		×	×	×	
T_3			×	×	×	
T_4				×	×	×
T_5	×	×	×	×	×	
T_6	×	×	×			

FIGURE 1.1 – Contexte d'extraction.

Définition 2. ***Itemset***
Un itemset I *est un sous-ensemble de* $\mathcal{I}$*. Un itemset de taille* k*,* $1 \leq k \leq \mid \mathcal{I} \mid$*, est un* k*-itemset. Par exemple, l'itemset* $\{a, b\}$ *est un 2-itemset.*

Définition 3. ***Supports d'un itemset***
Soit $(\mathcal{O}, \mathcal{I}, \mathcal{R})$ *un contexte d'extraction et* I *un itemset. La fréquence de* I *nommée aussi le support relatif de* I *est le ratio de son support par le nombre total de transactions de* $\mathcal{K}$*. i.e.,*

$$supp(I) = \frac{|\{o \in \mathcal{O} | \forall i \in I, (o,i) \in \mathcal{R}\}|}{|\mathcal{O}|}.$$

Dans le reste de manuscrit, le support d'un itemset I *désignera son support relatif.*

Selon Casali et al. [Casali et al., 2006], nous distinguons principalement quatre types de supports correspondants à un itemset I *:*

$$\begin{aligned}
supp_{\wedge}(I) &= \frac{|\{o \in \mathcal{O} \mid (\forall\ i \in I, (o,i) \in \mathcal{R})\}|}{|\mathcal{O}|} \\
supp_{\vee}(I) &= \frac{|\{o \in \mathcal{O} \mid (\exists\ i \in I, (o,i) \in \mathcal{R})\}|}{|\mathcal{O}|} \\
supp_{\bar{\wedge}}(I) &= \frac{|\{o \in \mathcal{O} \mid (\exists\ i \in I, (o,i) \notin \mathcal{R})\}|}{|\mathcal{O}|} \\
supp_{\bar{\vee}}(I) &= \frac{|\{o \in \mathcal{O} \mid (\forall\ i \in I, (o,i) \notin \mathcal{R})\}|}{|\mathcal{O}|}
\end{aligned}$$

Ainsi,

— *Le support conjonctif, noté* $supp_{\wedge}(I)$*, est le ratio du nombre des transactions qui contiennent tous les items de l'itemset* I *divisé par le nombre total des transactions ;*

— *Le support disjonctif, noté $supp_{\vee}(I)$, est le ratio du nombre des transactions qui contiennent au moins un item de l'itemset I divisé par le nombre total des transactions ;*
— *Le support conjonctif avec négation, noté $supp_{\bar{\wedge}}(I)$, est le ratio du nombre des transactions qui ne contiennent pas au moins un item de l'itemset I divisé par le nombre total des transactions.*
— *Le support disjonctif avec négation, noté $supp_{\bar{\vee}}(I)$, est le ratio du nombre des transactions qui ne contiennent aucun item de l'itemset I divisé par le nombre total des transactions.*

Dans le reste de manuscrit, nous désignons par *supp(I)* le support conjonctif de l'itemset $I \in \mathcal{I}$, i.e., $supp_{\wedge}(I)$. Le support conjonctif avec négation, noté $supp_{\bar{\wedge}}(I)$ est souvent appelé support négatif.

Exemple 3. *En se basant sur le contexte d'extraction présenté dans la figure 1.1, nous calculons les différents supports de l'itemset ABC de la manière suivante :*

— $supp_{\wedge}(abc) = \frac{|\{T_5,T_6\}|}{6} = \frac{2}{6}$.
— $supp_{\vee}(abc) = \frac{|\{T_1,T_2,T_3,T_5,T_6\}|}{6} = \frac{5}{6}$.
— $supp_{\bar{\vee}}(abc) = 1 - supp_{\vee}(abc) = \frac{|\{T_4\}|}{6} = \frac{1}{6}$.
— $supp_{\bar{\wedge}}(abc) = 1 - supp_{\wedge}(abc) = \frac{|\{T_1,T_2,T_3,T_4\}|}{6} = \frac{4}{6}$.

Remarque

Nous remarquons que $supp_{\wedge}(abc)$ et $supp_{\bar{\wedge}}(abc)$ sont complémentaires, de même pour $supp_{\vee}(abc)$ et $supp_{\bar{\vee}}(abc)$. Autrement dit, la somme de $supp_{\wedge}(abc)$ et $supp_{\bar{\wedge}}(abc)$ (respectivement $supp_{\vee}(abc)$ et $supp_{\bar{\vee}}(abc)$) est égale au nombre total des transactions du contexte d'extraction.

Remarque

Un itemset I est dit fréquent *si* supp(I) *dépasse le seuil du support défini par l'utilisateur.*
Également, nous mentionnons que *les identités d'inclusion-exclusion* [Galambos et Simonelli, 2000] fournies par le théorème suivant permettent de dériver le support conjonctif d'un itemset étant donnés les supports disjonctifs de tous ses sous-ensembles. En plus, grâce à la *Règle de De Morgan* [De Morgan, 1847], nous pouvons obtenir le support négatif (respectivement qu'elle soit une négation sur une disjonction ou sur une conjonction) d'un itemset à partir de son support disjonctif respectivement conjonctif.

Théorème 1. *Soit $I \subseteq \mathcal{I}$. Les égalités suivantes sont vérifiées :*

$$supp_{\wedge}(I) = \sum_{\emptyset \subset I_1 \subseteq I} (-1)^{|I_1|-1} supp_{\vee}(I_1)$$

$$supp_{\bar{\wedge}}(I) = \mid \mathcal{O} \mid - supp_{\wedge}(I) \text{(Règle de De Morgan)}$$

$$supp_{\vee}(I) = \mid \mathcal{O} \mid - supp_{\vee}(I) \text{(Règle de De Morgan)}$$

Définissons maintenant la propriété d'idéal d'ordre.

Définition 4. ***Idéal d'ordre*** *[Ganter et Wille, 1999]*

Un ensemble de données S est un idéal d'ordre s'il vérifie les propriétés suivantes :
- Si $x \in S$, alors $\forall\ y \subseteq x$, $y \in S$.
- Si $x \notin S$, alors $\forall\ x \subseteq y$, $y \notin S$.

Un élément x vérifie l'idéal d'ordre d'une propriété P si et seulement si pour tout y tel que $y \subset x$, y vérifie la propriété P.

Il est intéressant également d'étudier la relation de spécialisation / généralisation entre les itemsets.

Définition 5. ***Relation de spécialisation*** *[Mitchell, 1981]*
Une relation de spécialisation $\preceq$ est un ordre partiel défini sur l'ensemble des données de S. Soient x et y deux motifs, on dit que x est plus spécifique que y (respectivement plus général) si $x \preceq y$ (respectivement plus général) si $y \preceq x$.

L'inclusion entre deux motifs définit une relation de spécialisation. Ainsi, si un motif x est inclus dans un motif y, on dit que y est plus spécifique que x ou que x est plus général que y. Par exemple, *ab* est inclus dans *abcd*, donc *ab* est plus général que *abcd*.

Étant donné que nous avons défini le support d'un itemset et la relation de spécialisation entre deux itemsets, nous pouvons déduire que si l'itemset X est plus spécifique que l'itemset Y, alors le support conjonctif de X est plus petit que celui de Y. Par contre, le support disjonctif de X est plus grand que celui de Y.

Exemple 4. *Si nous considérons l'exemple dans la figure 1.1, le support conjonctif de* ab*, noté supp(*ab*) $= 0,2$ et le support conjonctif de* abcd*, noté supp(*abcd*) $= 0$, nous avons bien supp(*abcd*) $\leq$ supp(*ab*).*
Par contre, le support disjonctif de ab*, noté $supp_{\vee}$(*ab*) $= 0,66$ et le support disjonctif de* abcd*, noté $supp_{\vee}$(*abcd*) $= 1$, on a bien $supp_{\vee}$(*abcd*) $\geq$ $supp_{\vee}$(*ab*).*

Par la suite, nous définissons la relation de couverture comme suit :

Définition 6. ***Relation de couverture*** *[Ganter et Wille, 1999]*
Une relation de couverture entre les items de S, notée $\prec$, est définie par $x \prec y$, si $x \preceq y$ et tel qu'il n'existe pas d'élément $z \in S$ tel que $x \preceq z \preceq y$ pour $z \neq x$ et $x \neq y$.
Si $x \prec y$, nous disons que y couvre x ou bien que x est un successeur immédiat de y (et donc x est couvert par y ou y est un prédécesseur immédiat de x).

Propriété 1. ***Propriété de monotonie***
Soit $\mathcal{I}$ un ensemble d'items, une mesure f est dite monotone si :

$$\forall X, Y \in \mathcal{I} : X \subseteq Y \rightarrow f(X) \leq f(Y).$$

Remarque

Tous les sur-ensembles d'un itemset fréquent sont des itemsets fréquents. Par ailleurs, le support disjonctif est une mesure monotone.

Propriété 2. ***Propriété d'anti-monotonie***
Soit $\mathcal{I}$ un ensemble d'items, une mesure f est dite anti-monotone si :

$$\forall X, Y \in \mathcal{I} : X \subseteq Y \rightarrow f(X) \geq f(Y).$$

Remarque

Tous les sur-ensembles d'un itemset non fréquent sont des itemsets non fréquents. Par ailleurs, le support conjonctif est une mesure anti-monotone.

Toute mesure admettant une propriété d'anti-monotonie peut être intégrée dans un algorithme de recherche d'ensembles d'items fréquents.

Définition 7. ***Bordure Positive, Négative*** *[Mitchell, 1981] [Mannila et Toivonen, 1997]*
Soit ($2^{\mathcal{I}}$, $\subseteq$) un ensemble d'éléments partiellement ordonné et $\mathbf{S}$ *un ensemble de données de $2^{\mathcal{I}}$, tel que* $\mathbf{S}$ *est un idéal d'ordre dans ($2^{\mathcal{I}}$, $\subseteq$).* $\mathbf{S}$ *peut être représenté par sa bordure positive, notée $\mathcal{B}d^{+}(\mathbf{S})$, ou bien par sa bordure négative, notée $\mathcal{B}d^{-}(\mathbf{S})$, définies comme suit :*

$$\mathcal{B}d^{+}(\mathbf{S}) = max_{\subseteq}\{I \in \mathbf{S}\}$$

$$\mathcal{B}d^{-}(\mathbf{S}) = min_{\subseteq}\{I \in 2^{\mathcal{I}} \backslash \mathbf{S}\}$$

Nous avons présenté les concepts de base nécessaires à l'extraction d'itemsets fréquents. Nous détaillons, dans ce qui suit, la particularité de cette extraction pour le cas d'une table transactionnelle, table relationnelle et table multidimensionnelle.

1.2.2 Cas d'une table transactionnelle

Une table transactionnelle est l'ensemble des transactions munies de leurs items correspondants. Formellement elle est définie comme suit :

Définition 8. ***Table transactionnelle***
*Une table transactionnelle, est définie dans [Pasquier, 2000] sous la forme d'un triplet $\mathcal{K}$=($\mathcal{O}$, $\mathcal{I}$, $\mathcal{R}$) dans lequel $\mathcal{O}$ et $\mathcal{I}$ sont, respectivement, des ensembles finis d'objets (*les transactions*) et d'attributs (*les items*) et $\mathcal{R} \subseteq \mathcal{O} \times \mathcal{I}$ est une relation binaire entre les transactions et les items. Un couple (o,i) $\in$ R dénote le fait que la transaction o $\in \mathcal{O}$ contient l'item i $\in \mathcal{I}$.*

TID	a	b	c	d	e	f
T_1	1	1	0	0	0	0
T_2	1	0	1	1	1	0
T_3	0	0	1	1	1	0
T_4	0	0	0	1	1	1
T_5	1	1	1	1	1	0
T_6	1	1	1	0	0	0

FIGURE 1.2 – Base de données transactionnelle sous forme binaire.

La table transactionnelle dans la figure 1.1 peut être vue comme une table binaire où "1" indique la présence d'un item dans telle transaction et "0" son absence, comme l'illustre la figure 1.2.

Les travaux dans ce contexte sont nombreux et ils ont démarré avec les travaux de Agrawal et al. [Agrawal et al., 1993] aussi bien pour l'extraction d'itemsets fréquents que pour la génération des implications entre eux connues sous le nom des *règles d'association.*

L'algorithme APRIORI [Agrawal et al., 1993], dont le pseudo-code est décrit dans algorithme 1 est utilisé pour (i) l'extraction d'itemsets fréquents, qui est une étape coûteuse en temps d'exécution et en espace mémoire car elle consiste à parcourir un espace de recherche de taille exponentielle à la taille de l'ensemble des items ; et (ii) la génération des règles intéressantes à partir des itemsets fréquents pré-extraits : seules les règles ayant un support et une confiance supérieurs respectivement aux seuils du support et de la confiance sont retenues. APRIORI est un algorithme par niveau, i.e., il teste les itemsets fréquents par ordre croissant de leurs tailles. L'algorithme commence par calculer la fréquence des singletons, puis celles des paires. Les paires non fréquentes par rapport au seuil du support sont éliminées et leurs sur-ensembles sont élagués. Les motifs de taille trois sont obtenus par fusion des paires fréquentes et ainsi de suite.
L'algorithme 1 fait appel à la fonction 2 pour la génération des nouveaux candidats à partir des non fréquents du niveau précédent. Cette dernière fonction fait appel à son tour à la fonction 3 afin de vérifier si une auto-jointure entre deux itemsets non fréquents du niveau précédent est un bon candidat ou non.
Pendant la phase d'extraction d'itemsets fréquents, APRIORI utilise deux propriétés d'élagage très efficaces : celle d'anti-monotonocité (Propriété 2) et celle d'ordre lexicographique.

Algorithme 1 : L'algorithme Apriori

Données : Base de transaction Δ et le seuil du support σ.
Résultat : L'ensemble *Freq* de tous les itemsets fréquents.

```
1  F1= {1-item fréquent};
2  pour k=2 ; F_{k-1} ≠ ∅ ; k++ faire
3  |  C_k= Apriori-Gen(F_{k-1});
4  |  pour chaque transaction t ∈ Δ faire
5  |  |  c_t = sous-ensembles (C_k, t) ;
   |  |  // C_t = { c ∈ C_k, c ⊆ t}
6  |  |  pour chaque candidat c ∈ C_t faire
7  |  |  |  supp(c) + +;
8  |  |  fin
9  |  fin
10 |  F_k={c ∈ C_k/supp(c)≥ σ };
11 fin
12 retourner Freq=∪_k F_k ;
```

Algorithme 2 : La fonction Apriori-Gen(F_{k-1})

```
1  C_k= ∅;
2  pour chaque itemset p ∈ F_{k-1} faire
3  |  pour chaque itemset q ∈ F_{k-1} faire
4  |  |  si (p[1]=q[1] ∧ ... ∧ p[k-2]=q[k-2]∧p[k-1]<q[k-1]) alors
5  |  |  |  c=p⋈q;
6  |  |  fin
7  |  |  si a-sous-ensemble-infréquent(c, F_{k-1})=faux alors
8  |  |  |  ajouter c to C_k;
9  |  |  fin
10 |  fin
11 fin
12 retourner C_k ;
```

Algorithme 3 : La fonction a-sous-ensemble-infréquent(c, F_{k-1})

```
1  pour chaque sous-ensemble s de niveau (k-1) de c faire
2  |  si s ∉ F_{k-1} alors
3  |  |  vrai;
4  |  sinon
5  |  |  faux;
6  |  fin
7  fin
```

1.2.3 Cas d'une table relationnelle

Le modèle relationnel a été proposé par Codd au début des années 1970 [Codd, 1970, Codd, 1972]. Ce modèle est devenu le plus utilisé à nos jours grâce à son efficacité et sa simplicité d'usage.

Il est donc intéressant d'étudier l'extraction d'itemsets intéressants à partir d'une table relationnelle. Cette dernière peut être vue comme une relation classique inspirée du modèle relationnel, formellement nous la définissons comme suit :

Définition 9. ***Table relationnelle*** *Une table relationnelle est la représentation d'une relation en deux dimensions (lignes et colonnes). Chaque ligne (n-uplet) représente une entité. Chaque colonne correspond à un attribut.*

Définition 10. ***Attribut*** *On appelle attribut les noms des colonnes qui représentent les constituants de l'entité. Un attribut est repéré par un nom et un domaine de définition, c'est-à-dire l'ensemble des valeurs qu'il peut prendre.*

Définition 11. ***Clé primaire*** *Un ou plusieurs attributs permettent de désigner de façon unique un tuple, ils sont donc renommés clé primaire.*

Par analogie à la base transactionnelle, les tuples dans une table relationnelle correspondent aux transactions et les items sont alors des paires (attribut, valeur).
Une base transactionnelle peut être vue comme étant une table relationnelle si on considère chaque relation binaire associant une transaction avec son item comme une entité indépendante, (cf figure 1.3) [Diop, 2003].

TID	Liste d'items
T_1	a d f
T_2	a b d e
T_3	b c
T_4	f
T_5	d e

TID	item
T_1	a
T_1	d
T_1	f
T_2	a
...	...

FIGURE 1.3 – Base de données transactionnelle (à gauche) et Base de données transactionnelle vue comme base relationnelle (à droite).

De même, une table relationnelle peut être transformée par discrétisation de ses attributs en une table transactionnelle. En effet, dans une table relationnelle, nous pouvons rencontrer différents types d'attributs : des attributs qualitatifs, des attributs quantitatifs, etc. Au contraire, les items dans une base transactionnelle sont des attributs qualitatifs ou catégoriques prenant leurs valeurs dans un ensemble fini $\{0, 1\}$.

Dans une table relationnelle, si un attribut est qualitatif ou quantitatif avec peu de valeurs numériques alors il s'agit d'énumérer ses différentes modalités et de former pour chacune une paire (attribut, valeur). Cependant, si l'attribut est quantitatif, alors nous

procédons à sa discrétisation, en découpant son domaine en certains intervalles et nous attribuons à chacun un identifiant différent [Zighed et Rakotomalala, 2000]. Ainsi, nous obtenons un attribut qualitatif facile à interpréter via les algorithmes classiques d'extraction de motifs fréquents à partir des bases transactionnelles.

TID	âge	marié
T_1	23	non
T_2	25	oui
T_3	29	non
T_4	34	oui
T_5	38	oui

TID	âge	marié
T_1	1	2
T_2	2	1
T_3	2	2
T_4	3	1
T_5	4	1

FIGURE 1.4 – Base de données relationnelle (à gauche) et Base de données relationnelle après discrétisation (à droite).

L'attribut *marié* est un attribut catégorique. Ainsi, nous avons considéré les deux paires suivantes (marié, oui) et (marié, non). Puisque l'attribut *âge* est quantitatif, alors nous avons procédé à sa discrétisation en quatre intervalles disjoints (cf figure 1.4).
Les motifs à considérer sont alors des conjonctions des propriétés (par exemple âge = 1, marié = 1...), et les implications entre elles (âge = 1 $\Rightarrow$ marié = 1), qui est en réalité (âge = 20...24 $\Rightarrow$ marié = oui). Le calcul des propriétés fréquentes et des règles d'association se fait de la même manière que dans le cas transactionnel i.e., selon des seuils du support et de confiance définis par l'utilisateur.

Conclusion : l'extraction de règles d'association quantitatives ne peut pas être une extension de celle de règles catégoriques par des discrétisations répétées et fusionnées, et ceci à cause du grand nombre des modalités pour le cas numérique impossible à gérer. La discrétisation optimisée pendant la phase de génération des règles d'association a fait perdre à ces dernières leur caractère non supervisé vers plutôt un autre guidé.

1.2.4 Cas d'une table multi-dimensionnelles

Nombreux sont les travaux qui se sont intéressés à l'extraction des motifs intéressants et l'établissement des corrélations entre les valeurs d'un ou plusieurs attributs [Agrawal et al., 1993] [Agrawal et Srikant, 1994] [Zighed et Rakotomalala, 2000]. Néanmoins, combiner plusieurs dimensions ou attributs peut pourtant permettre d'obtenir des motifs décrivant mieux les données et offrant une meilleure compréhension des données source. Une table multi-dimensionnelle est définie par un certain nombre d'attributs catégoriques ou quantitatifs appelés *dimensions* et par un seul attribut numérique appelé *mesure*.

Remarque

En plus des composants $\mathcal{O}$, $\mathcal{I}$ et $\mathcal{R}$ du contexte d'extraction $\mathcal{K}$, nous ajoutons dans

TID	Produit	Ville	Année	Prix
T_1	Écran	Paris	2015	1180
T_2	Souris	Lyon	2006	1200
T_3	Clavier	Paris	2007	750
T_4	Imprimante	Nantes	2008	1680
T_5	Écran	Nice	2015	1000

FIGURE 1.5 – Base de données multi-dimensionnelle.

ce cas m qui est la mesure associée.

Les motifs à considérer sont les mêmes que dans le cas d'une base relationnelle à savoir les conjonctions des propriétés et les règles d'association sont de même des implications entre ces conjonctions des propriétés. Par exemple, nous pouvons considérer les motifs fréquents suivants (Produit = Écran et Ville = Paris, Produit = Écran et Année = 2015). Un exemple de règles sera (Produit = Écran, Ville = Paris) $\Rightarrow$ (Année = 2015).

L'extraction d'itemsets fréquents diffère dans le cas d'une table multi-dimensionnelle par rapport aux cas des tables transactionnelles et relationnelles.
En effet, dans ces derniers cas, un motif était intéressant par rapport à sa fréquence d'apparition dans la table. Alors que dans le cas d'une table multi-dimensionnelle, un motif sera qualifié intéressant par rapport à un critère de sélection qui est lié à la mesure m et n'aura vraiment du sens qu'avec celle là.

Définition 12. ***Critère de sélection***, *[Diop, 2003]*
On note agg (m, x)op α le critère de sélection q défini pour tout $x \subseteq 2^P$ par :

$$q(x) = vrai, si\ agg(\{m(o) \mid o \in f(x)\})op\ \alpha \tag{1.1}$$

Avec :
- x est un motif, i.e., une conjonction des propriétés,
- $f(x)$ est l'ensemble des objets contenant le motif x,
- $m(o)$ est la mesure correspondante à l'objet o,
- agg est la fonction d'agrégation par exemple count, avg, sum, etc.
- op est un opérateur qui peut être $\{\leq,\geq\}$ et α est un seuil défini par l'utilisateur.

Ensuite, pour décider si un motif est intéressant ou non, nous comparons sa mesure obtenue par rapport à une valeur seuil en tenant compte de l'opérateur α.

Exemple 5. *En s'appuyant sur l'exemple de la figure 1.5, et supposons que nous nous intéressons à l'extraction des motifs fréquents dont le total des ventes dépasse* 2000 *unités pour une année quelconque. Ainsi, le motif X = {Produit = Écran et Année = 2015} désigne le produit Écrans vendus à l'année 2015. Nous calculons d'abord sa fréquence*

f(X) = |{T_1, T_5}|, ensuite pour déterminer le total des ventes de ce produit, nous utilisons la fonction d'agrégation sum.

agg (m, X) = sum (m, (Produit = Écran , Année = 2015))
= sum ({ m (o) | o ∈ f ((Produit = Écran , Année = 2015))})
= sum ({ m (o) | o ∈ { 1, 5 } })
= 2280

Par conséquent le motif (Produit = Écran, Année = 2015) est intéressant.

Le cas multi-dimensionnel est plus riche que celui du transactionnel et du relationnel grâce à la variété des critères de sélection engendrés par la fonction d'agrégation.

Pareillement, nous étudions la génération des règles d'association à partir des bases multi-dimensionnelles. Dans leur travail, [Kamber et al., 1997] ont été les premiers qui ont approché l'extraction des règles d'association à partir des bases multi-dimensionnelles. Ils ont proposé un modèle général *méta-règles*, qui définit le contenu de la règle recherchée, i.e., dire la prémisse et la conclusion de la règle sont des conjonctions des prédicats, où chaque prédicat impose une condition sur une dimension bien définie.

Conclusion : l'extraction des motifs intéressants et la génération des corrélations entre eux à partir d'une table multi-dimensionnelle sont d'une grande richesse sémantique suite au développement des technologies *OLAP* permettant des analyses multi-dimensionnelle très assistées.

1.3 Extraction de règles d'association

L'extraction de règles d'association, à partir d'itemsets fréquents, a pour but de découvrir de manière automatique des corrélations significatives entre les attributs de la base de données.

Définition 13. ***Règle d'association***
Une règle d'association est une implication de la forme $X \Rightarrow Y$, où $X \subseteq \mathcal{I}$ est appelé prémisse, $Y \subseteq \mathcal{I}$ est appelé conclusion, et $X \cap Y = \emptyset$.

Définition 14. ***Support et confiance d'une règle d'association***
Soit $\mathcal{K} = (\mathcal{O}, \mathcal{I}, \mathcal{R})$ un contexte d'extraction et $X \Rightarrow Y$ une règle d'association.

Le support de $X \Rightarrow Y$ est : $supp(X \cup Y) = \frac{|f(X \cup Y)|}{|O|}$.

La confiance de $X \Rightarrow Y$ est : $conf(X \Rightarrow Y) = \frac{supp(X \cup Y)}{supp(X)} = \frac{|f(X \cup Y)|}{|f(X)|}$.

La confiance mesure la proportion de transactions contenant X et qui contiennent aussi Y.

Une règle $X \Rightarrow Y$ est dite *valide* si son support et sa confiance soient respectivement supérieurs ou égaux à deux seuils minimaux de support $minsup$ et de confiance $minconf$ définis par l'utilisateur.

À part les mesures du support et de la confiance présentées ci-dessus, certaines autres mesures ont été étudiées telles que la couverture, la conviction, l'intérêt, etc. Si une règle d'association possède une confiance égale à 1, elle est dite *exacte* sinon elle est dite *approximative*. Dans la littérature, il y a certains travaux concernant la fouille de règles d'association tels que [Gasmi et al., 2007] [Bouker et al., 2012] [Ayouni et al., 2011] [Ben yahia and Nguifo, [Ben yahia and Nguifo, 2004].

Exemple 6. *Le support de l'itemset ab selon la table de la figure 1.1 (page* 14*) est égal à* $0,5$*. Il est dit fréquent pour un seuil du support minimal inférieur ou égal à* $0,5$*. La confiance de la règle* $ab \Rightarrow d$ *est égale à* $0,33$*, elle est dite confiante pour un seuil de confiance minimal inférieur ou égal à* 0.33*.*

La fouille de règles d'association a touché à plusieurs domaines intéressants, nous citons par exemple le domaine des réseaux sociaux [Jelassi et al., 2014] et réseaux sociaux en ligne [Hamdi et al., 2013] et la fouille de données text ou connu sous le nom du text mining [Ferjani et al., 2012].

1.4 Conclusion

La fouille d'itemsets fréquents, que ce soit dans le cadre de l'exploration conjonctive respectivement disjonctive (où un itemset est une suite des conjonctions respectivement des disjonctions des items), ainsi que la génération des implications entre eux, sont basées et depuis leur apparition sur des fondements purement mathématiques issus de l'analyse formelle des concepts.
Dans le chapitre suivant, nous détaillons le thème de nos travaux de thèse qui est la fouille de motifs (fréquents ou non fréquents). Dans ce dernier, nous examinons certains types de motifs et les différents critères pour les fouiller.

Chapitre 2

Fouille de motifs

2.1 Introduction

Nous présentons dans ce chapitre une synthèse des principaux travaux de la littérature qui sont en relation avec notre thématique de thèse. En premier lieu, nous passons en revue les principales approches présentées dans la littérature pour l'extraction des motifs fréquents (itemsets, règles d'association, représentations condensées, etc). Ensuite, nous nous intéressons aux contributions relatives à la fouille de motifs rares (non-fréquents) vue l'importance de ces derniers dans certaines applications du monde réel.

2.2 Fouille de motifs fréquents

Nous introduisons, dans cette section, les principales méthodes relatives à l'extraction des motifs fréquents. En particulier, nous nous intéressons aux deux thématiques qui sont en liaison étroite avec nos travaux de thèse, à savoir (i) les représentations concises relatives au support disjonctif ; et (ii) les règles d'association généralisées intéressantes.

2.2.1 Représentations condensées relatives au support disjonctif

L'extraction de motifs intéressants à partir des grands volumes de données est une tâche primordiale en fouille de données grâce à sa capacité à déchiffrer des informations de forte utilité pour l'utilisateur.

Néanmoins, cette extraction est très coûteuse en espace mémoire et en temps de calcul à cause du nombre des candidats générés et de la taille d'un seul candidat surtout dans le cas où les bases sont fortement corrélées.

En outre, un nombre beaucoup plus important de règles d'association peut être généré. Ainsi, nous pouvons, et au pire des cas, générer 2^k-1 règles valides, parfois redondantes, à partir d'un *k-itemset* fréquent de taille supérieure à 1.
Pour pallier ce problème d'abondance d'itemsets fréquents, les recherches sont orientées

vers la localisation de nouveaux ensembles des motifs de taille plus réduite capables de générer de manière exacte ou approximative les itemsets fréquents. Cet ensemble de motifs est connu sous le nom de *représentation condensée*, concept introduit par [Mannila et Toivonen, 1996].

Une représentation condensée ou concise de l'ensemble d'itemsets fréquents est un ensemble représentatif de l'ensemble total permettant de le caractériser d'une manière exacte ou approximative [Hamrouni, 2007].

Dans la littérature, plusieurs sont les représentations condensées qui ont été introduites, nous citons à titre d'exemple [Hamrouni et al., 2008] [Hamrouni et al., 2008]. Ces dernières considèrent le support *conjonctif* ou le support *disjonctif* d'itemsets. Nous nous intéressons, dans ce qui suit, seulement aux représentations condensées considérant le support *disjonctif* puisqu'elles sont en liaison avec nos contributions dans le cadre de cette thèse. En plus, les représentations condensées peuvent être classées en deux catégories : les représentations condensées *exactes* et les représentations condensées *approximatives*.

Définition 15. ***Représentation Condensée exacte*** *[Diop, 2003]*
Étant donné un ensemble de données S, soit $X_1,\ldots,X_n$ et Y des sous-ensembles de S. $R = \{X_1,\ldots,X_n\}$ est une représentation condensée exacte de Y si :

1. $X_1 \cup \ldots \cup X_n \subseteq Y$.
2. *Il existe une fonction F indépendante de S permettant de calculer Y à partir de R.*

Le terme 1 de la définition indique que R est un sous-ensemble de Y (d'où le terme de représentation concise). Le terme 2 indique que F permet de calculer Y à partir de R sans accéder de nouveau aux données.

Définition 16. ***Représentation condensée approximative*** *[Diop, 2003]*
Étant donné un ensemble de données S, soit $X_1,\ldots,X_n$ et Y des sous-ensembles de S. $R = \{X_1,\ldots,X_n\}$ est une représentation condensée approximative de Y si :

1. $X_1 \cup \ldots \cup X_1 \subseteq Y$.
2. *Il existe une fonction F indépendante de S permettant de calculer Y à partir de R à ε-près, c'est à dire de calculer sans accéder aux données les motifs de Y et une approximation de leurs supports à ε-près.*

Nous étudions, dans ce manuscrit, les principales représentations condensées *exactes* relatives au support disjonctif. Pour les représentations *approximatives* considérant le support disjonctif, nous n'avons pas découvert des représentations de ce genre. Ceci constituera l'une de nos premières contributions dans cette thèse.

Revenons aux représentations condensées exactes, nous pouvons distinguer deux types de ces dernières en se basant sur la présence ou non des supports disjonctifs d'itemsets fermés.

Le premier type est celui de représentations condensées impliquant le support disjonctif uniquement dans la procédure du calcul du support conjonctif des ses fermés. Ces représentations fournissant en output seulement les supports *conjonctifs* exacts d'itemsets fermés fréquents. Le second type est celui des représentations qui fournissent absolument les supports *disjonctifs* exacts d'itemsets disjonctifs-fréquents fermés.

Représentation basée sur les Ensembles libres disjonctifs

Introduits par [Bykowski et Rigotti, 2001], comme des extensions d'itemsets libres [Bastide et al., 1981], donc des clés [Boulicaut et al., 2000 (a)], les itemsets libres disjonctifs se basent principalement sur le concept de règles disjonctives simples.

Définition 17. *Soit X un itemset de $\mathcal{I}$ et a et b deux items de X. Une règle disjonctive simple basée sur X est une expression de la forme $Y \Rightarrow a \vee b$, où $Y \subset X$ et $a, b \in X \backslash Y$.*

Les items a et b ne sont pas forcément différents. C'est ainsi que la règle $Y \Rightarrow a \vee a$ est un cas particulier de la règle simple disjonctive.

Selon [Bykowski et Rigotti, 2001], une règle simple disjonctive $Y \Rightarrow a \vee b$ est valide si sa confiance est égale à 1, i.e., f(Y)= f(Y $\cup$ {a}) $\cup$ f(Y $\cup$ {b}).
À partir de la définition précédente, les auteurs proposent le lemme fondamental suivant.

Lemme 1. *Soit X un itemset, et a et b deux items dans X. Alors, il existe $Y \in X$ tel que $Y \Rightarrow a \vee b$ soit une règle valide si et seulement si supp(X) = supp(X\ {a}) + supp(X\ {b}) − supp(X\ {a, b}).*

Définition 18. *Un itemset X est un ensemble-libre-disjonctif dans $\mathcal{I}$, s'il n'existe pas de règle disjonctive simple valide basée sur X dans $\mathcal{I}$. L'ensemble d'itemsets libres disjonctifs est noté* DFree($\mathcal{I}$).

TID	a	b	c	d
T_1	1	1	1	0
T_2	0	1	1	1
T_3	1	0	0	1
T_4	0	0	0	1
T_5	0	0	1	0
T_6	1	0	0	1
T_7	0	1	1	0
T_8	1	1	1	0

FIGURE 2.1 – Panier de clients [Bykowski et Rigotti, 2001].

Exemple 7. *Nous considérons la base de transactions de la figure 2.1, l'itemset {a, b, c, d} n'est pas un ensemble-libre-disjonctif puisque la règle $a \wedge b \Rightarrow c \vee d$ existe dans la*

base de transactions.
En effet, le support de $\{a, b, c, d\}$ *peut être déterminé à partir des supports des* $\{a, b\}$, $\{a, b, c\}$ *et* $\{a, b, d\}$.
Ainsi, supp($\{a, b\}$*) + supp(*$\{a, b, c, d\}$*) = supp(*$\{a, b, c\}$*) + supp(*$\{a, b, d\}$*), d'où* $\frac{2}{8}+\frac{0}{8} = \frac{2}{8}+\frac{0}{8}$.
Pour un seuil de minsup égal à $\frac{2}{8}$, *l'itemset* $\{a, b, c\}$ *est un ensemble-libre-disjonctif puisque la règle* $a \Rightarrow b \vee c$ *n'existe pas dans la base de transactions (i.e., a apparaît dans les transactions* T_3 *et* T_6 *sans que ni le b ni le c apparaissent).*
Nous avons supp($\{a, b, c\}$*)* $= \frac{2}{8}$, *donc* $\{a, b, c\}$ *est un ensemble-libre-disjonctif fréquent.*

Remarque

La propriété d'anti-monotonie est vérifiée comme suit : pour tout $Y \subseteq X$, *si* $X \in$ DFree($\mathcal{I}$), *alors* $Y \in$ DFree($\mathcal{I}$).

Notons par FreqDFree($\mathcal{I}$) l'ensemble d'itemsets libres disjonctifs fréquents, FreqDFreeSupp($\mathcal{I}$) l'ensemble d'itemsets libres disjonctifs fréquents avec leurs supports et Bd^-(FreqDFree($\mathcal{I}$)) la frontière négative des ensembles libres disjonctifs fréquents.
Bd^-(FreqDFree($\mathcal{I}$)) = { X $\subseteq \mathcal{I}$ | X $\notin$ FreqDFree($\mathcal{I}$) $\wedge$ $\forall$ Y $\subset$ X : Y $\in$ FreqDFree($\mathcal{I}$)}.

L'ensemble d'itemsets libres disjonctifs munis de leurs supports ne constitue pas seuls une représentation condensée de l'ensemble des itemset fréquents. En effet, il faut lui ajouter sa frontière négative pour qu'il constitue une représentation condensée exacte de motifs fréquents.
{FreqDFreeSupp($\mathcal{I}$), $BdSup^-$(FreqDFree($\mathcal{I}$))} est une représentation condensée exacte de (FreqSupp($\mathcal{I}$) $\cup$ $BdSup^-$(FreqDFree($\mathcal{I}$))).

L'algorithme de reconstitution des fréquents est le suivant : soit X un itemset quelconque, s'il existe un sous-ensemble $Y \subseteq X$ dans Bd^-(FreqDFree($\mathcal{I}$), alors X n'est pas fréquent. Sinon X est un itemset fréquent mais qui n'est pas un ensemble libre disjonctif, et alors son support peut être calculé par le lemme 1.
C'est ainsi que cette représentation basée sur les itemsets libres disjonctives permet de déduire le support *conjonctif* exact d'un itemset dans le cas où il est montré fréquent et n'est pas un ensemble libre disjonctif.

Conclusion : il a été prouvé que les itemsets libres disjonctifs ne sont qu'un cas particulier d'itemsets clés. Les itemsets libres disjonctifs ont été étudiés aussi par [Boulicaut et al., 2000 (a)] sous le nom des ensembles libres et par [Kryszkiewicz et Gajek, 2002] sous le nom des ensembles libres disjonctifs généralisés.
Les expérimentations ont montré que la représentation condensée baseé sur les itemsets libres disjonctifs est plus efficace que celle basée sur les ensembles libres.

Représentation basée sur les Itemsets essentiels fréquents

La représentation concise basée sur les itemsets essentiels fréquents a été introduite par [Casali et al., 2006]. La notion d'itemsets essentiels est basée sur le principe d'inclusion-exclusion, [Galambos et Simonelli, 2000].

Les *identités d'inclusion-exclusion* [Galambos et Simonelli, 2000] (page 13) permettent de dériver le support conjonctif d'un itemset étant donnés les supports disjonctifs de tous ses sous-ensembles. En plus, grâce à la *Règle de De Morgan*, nous pouvons obtenir le support négatif (respectivement qu'elle soit une négation sur une disjonction ou sur une conjonction) d'un itemset à partir de son support disjonctif ou conjonctif.

De ce théorème, dérive l'intérêt du support disjonctif pour la détermination des autres supports conjonctif et négatif (e.g., négation sur conjonction ou négation sur disjonction).

Une caractéristique intéressante de cette représentation concise est la possibilité de la dérivation directe des supports disjonctifs et négatifs d'itemsets fréquents à partir de leurs supports conjonctifs. Chose qui est très utile dans le calcul des mesures d'évaluation de la pertinence des règles d'association généralisées, à savoir le support et la confiance [Hamrouni et al., 2007].
Cette représentation est fondée alors sur la notion suivante d'itemsets essentiels fréquents et elle est considérée la seule qui s'appuie sur le support disjonctif, outre celui conjonctif.

Définition 19. ***Itemsets essentiels fréquents*** *[Casali et al., 2006]*
Soit $\mathcal{K} = (\mathcal{O}, \mathcal{I}, \mathcal{R})$ un contexte d'extraction et $I \subseteq \mathcal{I}$ un itemset. I est dit essentiel si et seulement si $supp_{\vee}(I) > max\ \{\ supp_{\vee}(I \backslash i),\ i \in I\}$. Un itemset I est dit fréquent essentiel s'il est simultanément fréquent et essentiel. Dans ce qui suit, l'ensemble d'itemsets essentiels fréquents est noté par $\mathcal{IEF}$

Le lemme suivant montre comment on peut obtenir le support disjonctif d'un itemset fréquent, étant donné l'ensemble d'itemsets essentiels fréquents.

Lemme 2. *Soit* I *un itemset fréquent. $Supp_{\vee}$(I)= $max\{supp_{\vee}(I_1) \mid I_1 \subseteq I$ et $I_1 \in \mathcal{IEF}\}$, [Casali et al., 2005].*

Par la suite, nous présentons la définition de l'ensemble *Argmax* associé à un itemset fréquent *I*. Cet ensemble contient les itemsets essentiels fréquents contenus dans *I* et ayant le support disjonctif maximal parmi les sous-ensembles de *I*.

Définition 20. ***Argmax*** *[Casali et al., 2005]*
Soit I *un itemset fréquent. $J \in$* Argmax(I) *si $J \subseteq$* I*, $J \in \mathcal{IEF}$ et $supp_{\vee}(J) = max\{$ $supp_{\vee}(I_1) \mid I_1 \subseteq I\ \}$*

Le théorème suivant indique comment dériver le support conjonctif d'un itemset fréquent, une fois que l'ensemble d'itemsets fréquents essentiels est extrait, [Casali et al., 2005].

Théorème 2. *Soit $I \in \mathcal{IF} \setminus \mathcal{IEF}$ et $J \in$* Argmax(I)*. Alors, nous avons :*

$$supp(I) = \sum_{\substack{\emptyset \subset I_1 \subseteq I \\ J \not\subseteq I_1}} (-1)^{|I_1|-1} supp_{\vee}(I_1)$$

L'information fournie par l'ensemble $\mathcal{IEF}$ seule ne permet pas de décider si un itemset est fréquent ou non. Pour surmonter cette limite, Casali et al. ont augmenté l'ensemble $\mathcal{IEF}$ avec l'ensemble d'itemsets fréquents maximaux $Bd^+(\mathcal{IF})$.

Théorème 3. *L'ensemble d'itemsets essentiels fréquent $\mathcal{IEF}$, associés à leurs supports disjonctifs respectifs, augmentés de la bordure positive $Bd^+(\mathcal{IF})$ de l'ensemble d'itemsets fréquents maximaux est une représentation condensée exacte de l'ensemble d'itemsets fréquents $\mathcal{IF}$, [Casali et al., 2005].*

Conclusion : la représentation concise exacte basée sur les itemsets essentiels permet de dériver directement les supports disjonctifs d'itemsets fréquents et d'offrir un mécanisme sain et correct de dérivation de leurs supports conjonctifs et négatifs.
Cette représentation est la première dans la littérature qui a pu permettre la dérivation des supports disjonctifs d'itemsets fréquents. Par conséquent, toutes les représentations qui viennent par la suite permettent la dérivation du support disjonctif exact d'un itemset fréquent.

Représentation basée sur les Itemsets disjonctifs fermés

Hamrouni et al. ont proposé dans [Hamrouni et al., 2007b] une nouvelle représentation concise exacte basée sur les disjonctifs fermés. Cette représentation se base sur la définition d'une nouvelle fermeture dite *fermeture disjonctive* appliquée aux itemsets disjonctifs afin de regrouper ceux qui caractérisent le même ensemble de transactions. Cette représentation hérite de la représentation précédente la possibilité de dérivation directe des supports disjonctifs. Nous présentons la fermeture disjonctive dans la définition suivante :

Définition 21. ***Fermeture disjonctive*** *[Hamrouni et al., 2007b]*
Soit $\mathcal{K} = (\mathcal{O}, \mathcal{I}, \mathcal{R})$ un contexte d'extraction. L'opérateur de fermeture disjonctive, noté h_d, est défini comme suit :

$$\begin{aligned} h_d : \mathcal{P}(\mathcal{I}) &\rightarrow \mathcal{P}(\mathcal{I}) \\ I &\mapsto h_d(I) = \{i \in \mathcal{I} \mid [(\exists o \in \mathcal{O})((o,i) \in \mathcal{R})] \wedge \\ &\quad [(\forall o_1 \in \mathcal{O})((o_1,i) \in \mathcal{R}) \Rightarrow ((\exists i_1 \in \mathcal{I})((i_1 \in I) \wedge \\ &\quad ((o_1,i_1) \in \mathcal{R})))]\} \end{aligned}$$

La fermeture disjonctive $h_d(I)$ d'un itemset I est égale à l'ensemble maximal des items qui apparaissent seulement dans les transactions qui contiennent au moins un item $i \in I$.

Exemple 8. *Soit le contexte d'extraction illustré par la figure 2.2.*
Alors : $h_d(\{a,c\}) = \emptyset$, $h_d(\{a,b\}) = \{\ a,\ b\ \}$ *et* $h_d(\{c,d\}) = \{\ c,\ d,\ e\ \}$.

TID	a	b	c	d	e	f
T_1	×	×				
T_2	×		×	×		
T_3			×	×	×	
T_4				×	×	×
T_5	×	×	×	×	×	
T_6	×	×	×			

FIGURE 2.2 – Contexte d'extraction.

Définition 22. ***Itemset fermé disjonctif***
Un itemset I est dit fermé disjonctif, si h_d *(I)= I. Si I est fermé disjonctif, alors* $supp_\vee(I) < min\ \{\ supp_\vee(I \cup \{i\}) \mid i \in \mathcal{I} \setminus I\}$.

Tout d'abord, nous débutons par établir la relation entre le plus petit fermé disjonctif contenant un itemset I et $h_d(I)$.

Proposition 1. *Soit* $I \subseteq \mathcal{I}$. $h_d(I)$ *est le plus petit fermé disjonctif contenant* I *et ayant le plus petit support disjonctif parmi les fermés disjonctifs contenant* I.

La proposition qui suit établit la relation entre le support disjonctif d'un itemset et sa fermeture.

Proposition 2. *Soit I un itemset, alors* $supp_\vee(I) = supp_\vee(h_d(I))$.

La proposition qui suit établit la relation entre le support disjonctif d'un itemset et celui du plus petit fermé disjonctif qui le contient.

Proposition 3. *Le support disjonctif d'un itemset quelconque I est celui du plus petit fermé disjonctif qui le contient.*

La proposition suivante permet d'éviter, dans plusieurs cas, le calcul des fermetures déjà extraites à partir d'un contexte d'extraction. Ainsi, elle permettra d'améliorer les performances des algorithmes d'extraction des fermés disjonctifs.

Proposition 4. *Soit* X *et* Y *deux itemsets tels que* $X \subseteq Y$ *et* $Y \subseteq h_d(X)$, *alors* $h_d(X) = h_d(Y)$ *et* $supp_\vee(X) = supp_\vee(Y)$.

L'ensemble de tous les itemsets fermés disjonctifs qui peuvent être tirés à partir d'un contexte d'extraction donné, est noté $\mathcal{IFD}$, acronyme de (Itemsets Fermés Disjonctifs). L'identification d'un nouvel opérateur de fermeture disjonctive a permis l'extraction de tous les $\mathcal{IFD}$ présents dans un contexte $\mathcal{K}$, ainsi la construction d'un treillis pour ces

derniers sous forme des classes d'équivalences disjonctives. Une classe d'équivalence disjonctive regroupe l'ensemble d'itemsets ayant le même support disjonctif.

Étant donné, que nous avons l'ensemble d'itemsets fermés disjonctifs $\mathcal{IFD}$ menus de leurs supports disjonctifs, il est possible de dériver le support disjonctif exact de chaque sous-ensemble d'un itemset quelconque à partir de $\mathcal{IFD}$. En effet, nous avons $\forall\ I_1 \subseteq I$, $\mathrm{h}(I_1) \in \mathcal{IFD}$. Donc, il est possible de retrouver le support disjonctif exact de I_1, puisque $supp_{\vee}(\mathrm{I}) = supp_{\vee}(\mathrm{h(I)})$ et support disjonctif exact de I_1 peut être déduit de celui de I.

Théorème 4. *L'ensemble $\mathcal{IFD}$ d'itemsets fermés disjonctifs, associés à leurs supports disjonctifs respectifs, est une représentation concise exacte de l'ensemble de tous les itemsets [Hamrouni, 2009].*

Soit I un itemset de $\mathcal{I}$, alors le support disjonctif de I et ceux de ses sous-ensembles peuvent être dérivées exactement de $\mathcal{IFD}$. Par la suite, en appliquant une identité d'inclusion-exclusion (cf Théorème 1, page 28) utilisant les supports disjonctifs obtenus, on est en mesure de trouver le support conjonctif de I.

Conclusion : l'ensemble $\mathcal{IFD}$ constitue une représentation concise exacte non pas seulement d'itemsets *fréquents* mais plutôt de tous les itemsets, i.e., même les supports d'itemsets *non fréquents* peuvent être dérivés en utilisant $\mathcal{IFD}$.

Représentation basée sur les Itemsets essentiels fermés

Dans la représentation concise basée sur les essentiels fréquents, certains itemsets essentiels fréquents caractérisent le même ensemble de transactions. Ceci constitue une forme de redondance au sein de cette représentation. Pour pallier ce problème, Hamrouni et al. ont pensé à appliquer l'opérateur de fermeture disjonctive sur les essentiels fréquents caractérisant un même ensemble de transactions dans le but de les faire représenter par un même itemset [Hamrouni et al., 2007a].
Ainsi, la représentation concise basée sur les itemsets essentiels fermés, notée par $\mathcal{IEF}$ combine deux représentations condensées existantes, à savoir la représentation condensée d'itemsets fermés disjonctifs et celle des essentiels. Une caractéristique intéressante de cette représentation est la dérivation du support disjonctif de chaque motif fréquent et par suite ses supports négatif et conjonctif en utilisant la règle de De Morgan et les identités d'inclusion-exclusion.

La proposition suivante établit le lien entre les itemsets fermés disjonctifs et les itemsets essentiels.

Proposition 5. *Soit $\mathcal{IE}$ l'ensemble de tous les itemsets essentiels qui peuvent être extraits à partir d'un contexte d'extraction donné.*
$\forall$ ($I \subseteq \mathcal{I}$), $\exists$ ($I_1 \in \mathcal{IFD}$ et $I_2 \in \mathcal{IE}$) tel que $h(I_2) = h(I) = I_1$ et $I_2 \subseteq I$.

Dans ce qui suit, nous notons l'ensemble des fermés disjonctifs relatifs aux essentiels fréquents par $\mathcal{IFDE}$(acronyme de $\mathcal{I}$temsets $\mathcal{F}$ermés $\mathcal{D}$isjonctifs $\mathcal{E}$ssentiels), et par $\mathcal{IEF}$ l'ensemble des essentiels fréquents, (acronyme de $\mathcal{I}$temsets $\mathcal{E}$ssentiels $\mathcal{F}$réquents).

Théorème 5. *L'ensemble $\mathcal{IFDE} \cup \mathcal{IEF}$ d'itemsets, associés à leurs supports disjonctifs, est une représentation concise exacte de l'ensemble d'itemsets fréquents $\mathcal{IF}$.*

Ensuite, cette représentation a été améliorée par [Hamrouni et al., 2014] et appelée représentation basée sur l'espace de recherche disjonctif, puisqu'elle est basée seulement sur des éléments particuliers de l'espace de recherche disjonctif. En plus, cette dernière représentation se distingue par le fait qu'elle soit homogène dans le sens où elle est composée uniquement par des motifs disjonctifs et elle évite donc l'exploration de l'espace de recherche conjonctif. En effet, cette dernière ne nécessite pas d'ajouter des informations supplémentaires de l'espace de recherche conjonctif pour vérifier si un motif est fréquent ou non. Cette représentation est déterminée grâce à un algorithme valide et correct qui s'est montré plus rapide que l'algorithme d'extraction d'itemsets essentiels.

Conclusion : les expérimentations ont montré que le nombre d'itemsets disjonctifs essentiels fermés est beaucoup plus réduit que celui des ensembles des fermés fréquents et des essentiels fréquents, $\mathcal{IFDE} \subset \mathcal{IEF}$ et $\mathcal{IFDE} \subset \mathcal{IFD}$. Cette représentation ne permet la dérivation du support exact que pour les itemsets fréquents.

Autres représentations condensées

En plus, Hamrouni et al. ont essayé de développer deux nouvelles représentations en se basant sur l'ensemble $\mathcal{IFDE}$ mais en réduisant à chaque fois l'ensemble d'itemsets à ajouter à cet ensemble [Hamrouni et al., 2007b]. Ainsi, l'ensemble d'itemsets ajouté à $\mathcal{IFDE}$ est chaque fois plus petit que celui de $\mathcal{IEF}$.

Définition 23. ***Ensemble de tous les itemsets fermés disjonctifs ajoutés***
Soit $\mathcal{IE}$ l'ensemble d'itemsets essentiels qui peuvent être extraits à partir d'un contexte d'extraction donné. L'ensemble d'itemsets fermés disjonctifs de taille impaire de la bordure négative $\mathcal{IFDB}$ est défini comme suit :
$\mathcal{IFDB} = \{\ h(I) \in \mathcal{IFD} \mid (I \in Bd^{-}(\mathcal{IFD}) \cap \mathcal{IE}) \wedge (|I| \text{ est impair})\}$.

Théorème 6. *L'ensemble $\mathcal{IFD} \cup \mathcal{IFDB}$ d'itemsets fermés disjonctifs associés à leurs supports disjonctifs respectifs, est une représentation concise exacte de l'ensemble d'itemsets fréquents $\mathcal{IF}$.*

Définition 24. ***Ensemble d'itemsets fermés disjonctifs ajoutés***
L'ensemble $\mathcal{IFDA}$ acronyme de (Itemsets Fermés Disjonctifs Ajoutés) est défini comme suit : $\mathcal{IFDA} = \{\ I \in \mathcal{IFDB} \mid (I \notin \mathcal{IFDE})$ et $(\exists\ I' \in \mathcal{IFDE}$ tel que $I \subset I')\ \}$.

Théorème 7. *L'ensemble $\mathcal{IFD} \cup \mathcal{IFDA}$ d'itemsets fermés disjonctifs associés à leurs supports disjonctifs respectifs, est une représentation concise exacte de l'ensemble d'itemsets fréquents $\mathcal{IF}$.*

Conclusion : ces deux dernières représentations concises exactes sont à la base de l'ensemble $\mathcal{IFDE}$ et elles ne permettent la déduction du support disjonctif exact que pour les itemsets fréquents.
Les démonstrations ont pu conclure la relation d'inclusion suivante en nombre d'itemsets : $\mathcal{IFD} \cup \mathcal{IFDA} \subset \mathcal{IFD} \cup \mathcal{IFDB} \subset \mathcal{IFDE} \cup \mathcal{IEF}$.

Il importe de mettre en exergue que pour le cas des représentations approximatives dans le cadre de l'exploration disjonctive, nous signalons que ça sera l'une des nos première contributions dans cette thèse. C'est ainsi que nous proposerons une représentation condensée d'itemsets disjonctifs fréquents qui est *approximative* et qui sera détaillée ultérieurement.

2.2.2 Extraction de règles d'association généralisées

Dans ce qui suit, nous nous intéressons aux principaux travaux de la littérature concernant la fouille de règles d'association généralisées. Ainsi, avant d'entamer la recherche dans ces travaux, nous présentons dans un premier paragraphe les règles d'association classiques (i.e., des conjonctions d'items en prémisse et en conclusion).

Règles d'association classiques

Les contributions reliées à la fouille de règles d'association se sont limitées à leur début aux règles classiques (c'est à dire les règles avec des conjonctions des items dans la prémisse et dans la conclusion), e.g., R : $X \Rightarrow Y$ où $X = x_1 \wedge x_2 \wedge \ldots \wedge x_n$ et $Y = y_1 \wedge y_2 \wedge \ldots \wedge y_m$ [Agrawal et al., 1993]. Pour plus de détail, un survol sur les principes fondamentaux de la fouille de règles d'association est présenté chez [Ceglar et Roddick, 2006].

Remarque

La phase de génération de règles d'association est beaucoup moins coûteuse que la génération d'itemsets fréquents, car il n'est plus nécessaire de faire des parcours coûteux de la base de transactions.

Ainsi, [Agrawal et Srikant, 1994] ont proposé une optimisation qui vise à réduire le nombre de règles pouvant être générées à partir d'un $k-itemset$, et qui est égal à $2^k - 1$. Cette optimisation utilise la propriété suivante :

Propriété 3. *Soit X un itemset fréquent, nous avons : $\forall\ Y \subset X,\ Y \neq \emptyset$ $(X\text{-}Y) \rightarrow Y$ satisfait le seuil de confiance $\Rightarrow \forall\ \tilde{Y} \subset Y,\ \tilde{Y} \neq \emptyset$, $(X\text{-}\tilde{Y}) \rightarrow \tilde{Y}$ satisfait aussi le seuil de confiance.*

Ceci signifie que si une règle a une conclusion Y confiante, alors toutes les règles ayant pour conclusions des sous-ensembles de Y sont aussi confiantes.

Selon [Agrawal et Srikant, 1994], la génération de règles d'association est réalisée en deux étapes. Durant la première étape (algorithme 4), l'algorithme génère uniquement

les règles valides ayant un seul item en conclusion.
Durant la deuxième étape, (algorithme 5), les conclusions des règles obtenues dans l'algorithme 4 sont combinées pour générer toutes les conclusions possibles à deux items pouvant exister dans une règle générée à partir de F_k et ainsi de suite. L'algorithme 5 fait appel à la fonction de Apriori donnée par l'algorithme 2

Algorithme 4 : Génération des règles d'association

Données : Ensemble d'itemsets fréquents *Freq*, le seuil de la confiance γ.
Résultat : L'ensemble des règles d'association R.

```
1  R= ∅ ;
2  pour chaque k-itemset F_k ∈ Freq, k ≥ 2 faire
3      H_1 = { 1-itemsets fréquents sous-ensembles de F_k } ;
4      pour chaque h_1 ∈ H_1 faire
5          conf=supp(F_k)/supp(F_k−h_1) ;
6          si conf ≥ γ alors
7              r :(F_k-h_1)→ h_1 ;
8              R = R ∪ r
9          sinon
10             supprimer h_1 de H_1
11         fin
12     fin
13 fin
14 Procédure Gen-Règles (F_k, H_1);
15 retourner R ;
```

Algorithme 5 : Procédure Gen-Règles (F_k, H_m)

```
1  si (k > m+1) alors
2      H_{m+1}= Apriori-Gen(H_m);
3      pour chaque h_{m+1} ∈ H_{m+1} faire
4          conf=supp(F_k)/supp(F_k−h_{m+1}) ;
5          si conf ≥ γ alors
6              r :(F_k-h_{m+1})→ h_{m+1} ;
7              R = R ∪ r
8          sinon
9              supprimer h_{m+1} de H_{m+1}
10         fin
11     fin
12 fin
13 Gen-Règles (F_k, H_{m+1});
```

Les règles d'association classiques communiquent des informations sur les relations de co-occurrence entre les items. Cependant, d'autres relations entre les items (occurrence complémentaire, absence d'occurrence entre les items, etc) peuvent survenir et offrir

des connaissances intéressantes aux utilisateurs finaux [Hamrouni et al., 2010]. Ces différentes relations entre les items cachent des nouvelles informations pouvant être exploitées dans des règles plus riches sémantiquement, i.e., les règles généralisées. Ces dernières généralisent les règles classiques pour étudier encore les connecteurs de disjonction et de négation entre les items.

Définition 25. ***Règle généralisée*** *[Hamrouni et al., 2010]*
Soit $\mathcal{I}$ l'ensemble des items, x_i et $y_j \in \mathcal{I}$ tels que $i \in$ [1, n] et $j \in$ [1, m]. Une règle d'association est de la forme :

$$\varrho(x_1, x_2, \ldots, x_n) \Rightarrow \upsilon(y_1, y_2, \ldots, y_n)$$

avec $\varrho(x_1, x_2, \ldots, x_n)$ et $\upsilon(y_1, y_2, \ldots, y_n)$ sont deux itemsets qui n'ont aucun item en commun, et qui sont liés par des différents connecteurs qui seront repris pour les associer au bon calcul du support.
Ainsi, une règle généralisée peut être considérée comme un n-uplet de la forme ($X \Rightarrow Y$, connecteur 1, connecteur 2), où les connecteurs seraient repris pour les associer au bon calcul de support.

Nous précisions ici que la notion de règle généralisée ainsi que les notations introduites ci-dessus se justifient de la manière suivante.
Si l'on considère un itemset I, chacun de ses éléments i peut être associé à la formule atomique $i \in X$ où X est une variable représentant un itemset quelconque. Il est alors possible de considérer des formules logiques combinant les formules atomiques avec les connecteurs logiques habituels $\vee$, $\wedge$ et $\neg$.

Exemple 9. *Pour $I = \{a, b\}$, $\varphi_1 = (a \in X) \wedge (b \in X)$ et $\varphi_2 = (a \in X) \vee \neg(b \in X)$ sont deux formules que l'on peut considérer à partir de I.*

Si maintenant on considère un itemset J et une formule φ construite à partir de l'itemset I, on dit que J satisfait φ, si en substituant X par J dans φ, la formule ainsi obtenue est aussi satisfaite. Par exemple pour $J = \{a, b, c\}$, φ_1 est satisfaite, alors que φ_2 ne l'est pas.

Dans le cas d'une base de transactions, on peut définir le support d'une formule φ, que nous notons $supp(\varphi)$, comme étant le ratio entre le nombre de transactions dont l'itemset satisfait φ par le nombre total de transactions.

Par exemple, en retenant les deux formules φ_1 et φ_2 ci-dessus, pour un ensemble de transactions fixé, le support de φ_1 est égal au support conjonctif de I noté $supp_\wedge(I)$ ou $supp(I)$ dans la définition 3 (page 12) et le support de φ_2 ne correspond à aucun support défini jusque là. On remarque également que le support disjonctif de I noté $supp_\vee(I)$ précédemment est égal au support de la formule $a \in X \vee b \in X$.

Dans la mesure où les formules définies à partir d'itemsets seront utilisées par la suite, nous les noterons plus simplement en ne faisant figurer que les items et les connecteurs les constituant. Ainsi, φ_1 sera notée $a \vee b$.

De manière générale, les travaux reliés à la fouille de règles généralisées peuvent être classés selon les possibilités d'occurrence d'items dans la prémisse et dans la conclusion, à savoir des règles avec conjonction, avec négation et avec disjonction des items.

Règles d'association avec négation

De nombreux travaux se sont intéressés aux règles d'association avec négation des items. Ces règles sont dites *négatives* et elles sont utiles dans l'analyse des paniers de la ménagère pour identifier des produits, qui peuvent être en conflit avec d'autres produits ou des produits qui complètent certains autres produits [Mani, 2012].

Brin et al. en 1997 ont été les premiers à voir travailler sur cet axe de recherche sans utiliser le terme "règles négatives". Ils ont développé, dans [Brin et al., 1997], la fouille de règles identifiant des *corrélations* (règles généralisantes) en tenant compte de *l'absence* et de la *présence* des items comme une base pour générer ces règles. Pour mesurer la signification de ces corrélations, ils utilisent le test statistique de corrélation *Chi-carré* pour différencier les itemsets corrélés de ceux non-corrélés. Cette mesure de corrélation a permis de réduire le problème de la fouille à la simple recherche d'une frontière entre les motifs corrélés et les non-corrélés dans l'espace de recherche.
La même équipe a proposé en 1998 dans [Silverstein et al., 1998] une nouvelle version de ces règles dites *règles d'indépendance* identifiant des dépendances statistiques dans la *présence* et l'*absence* des items dans un itemset quelconque. De même, ils ont défini une mesure de dépendance permettant de distinguer des itemsets dépendants de ceux non indépendants dans un treillis. L'avantage de ces deux versions de règles par rapport aux règles standards est qu'elles permettent d'analyser un large éventail de données tenant compte de la présence et de l'absence des items dans un itemset quelconque.

Ainsi, ces deux approches ([Brin et al., 1997] et [Silverstein et al., 1998]) utilisent des méthodes et des mesures statistiques, en plus d'autres opérations supplémentaires pour déterminer la forme exacte des règles négatives.

Par la suite, [Savasere et al., 1998] ont présenté une nouvelle idée pour fouiller des règles d'association *négatives fortes*. Ils combinent les itemsets fréquents positifs avec les connaissances du domaine issues d'une taxonomie pour fouiller des association négatives. Cependant, cet algorithme est difficile à généraliser puisqu'il est trop dépendant du domaine et il nécessite une taxonomie définie. Une approche similaire à [Savasere et al., 1998] a été décrite par [Yuan et al., 2002] pour la fouille de règles négatives en utilisant une taxonomie. Les auteurs affirment que la règle $X \rightarrow \neg\, Y$ a un support $s\%$ dans l'ensemble de données, si $s\%$ des transactions contiennent l'itemset X et ne contiennent pas l'itemset Y.

Ces deux dernières approches utilisent des méthodes heuristiques incorporant les connaissances du domaine ; qui même si elles réussissent à extraire des règles négatives intéressantes, les connaissances de domaine peuvent souvent ne pas être facilement disponibles.

Dans l'objectif de généraliser le processus de génération de règles d'association, les auteurs de [Wu et al., 2002, Wu et al., 2004] dérivent un nouvel algorithme basé sur APRIORI pour générer des règles d'association positives et négatives. Ils ajoutent outre les mesures traditionnelles (support et confiance) une autre mesure appelée *minimum-intérêt* pour mieux élaguer les itemsets fréquents générés. Toutefois, les auteurs ne discutent ni comment définir ce paramètre ni quel sera l'impact sur les résultats quand on change ce paramètre.

Dans l'intuition de permettre aux clients de remplacer un ensemble d'items achetés par d'autres, [Teng et al., 2002] proposent l'algorithme SRM[1] pour extraire des règles dites des règles de substitution. Ces règles sont des règles négatives et sont de type $X \rightarrow \neg Y$.
Cette approche se déroule en deux étapes. Dans la première, les auteurs définissent l'ensemble des items *concrets*,i.e., les items dont leurs itemsets ont une importante valeur de la mesure *chi carré* et une importante valeur du support. Dans la deuxième étape, les auteurs calculent le coefficient de corrélation pour chaque paire de ces itemsets pré-extraits. Puis, à partir de ces paires qui sont négativement corrélés, ils extraient les règles de substitution souhaitées (de la forme $X \rightarrow \neg Y$). Une règle de substitution est construite à la base de deux itemsets *concrets* et elle est notée $X \triangle Y$. Ceci signifie que X remplace Y si et seulement si X et Y sont corrélés négativement et la règle négative $X \rightarrow \neg Y$ existe.

Exemple 10. *Cet exemple est une extension de l'exemple présenté dans [Teng et al., 2002]. Pour illustrer la fouille des associations négatives, une deuxième table de la figure 2.3 est créée, et où pour chaque transaction, le complément de chaque item absent est marqué par 'zéro', celui présent par 'un'.*

En utilisant un support minimal égal à $0,2$ *et un coefficient de corrélation égal à* $0,5$*, [Teng et al., 2002] extraient à partir de la table transactionnelle donnée par la figure 2.3 les itemsets concrets fréquents suivants. Les itemsets positifs sont séparés d'itemsets négatifs par une double ligne horizontale.*
Á partir de ces itemsets extraits (c.f figure 2.4), un ensemble de règles d'association peut être généré. À partir de l'itemset ad, la règle d'association $\neg d \rightarrow a$ *existe avec tel support et telle confiance.*

De manière plus explicite, [Boulicaut et al., 2000 (b)] abordent le problème des règles d'association qui impliquent des négations a $\wedge$ b $\Rightarrow \neg$ c ou bien $\neg$ a $\wedge$ b $\Rightarrow$ c. Cette

1. *S*ubstitution *R*ule *M*ining

TID	Liste d'items
T_1	a c d
T_2	b c
T_3	c
T_4	a b f
T_5	a c d
T_6	e
T_7	b f
T_8	b c f
T_9	a b e
T_{10}	a d

TID	Vecteur de bits
T_1	101100
T_2	011000
T_3	001000
T_4	110001
T_5	101100
T_6	000010
T_7	010001
T_8	011001
T_9	110010
T_{10}	100100

FIGURE 2.3 – Base transactionnelle et vecteur des bits.

2-itemsets concrets
a d
b f
b d

3-itemsets concrets
a c d
a b d

FIGURE 2.4 – Itemsets extraits.

approche est classée dans le cadre de la fouille des règles intéressantes parmi les règles fréquentes. Ainsi, ils proposent une approche d'extraction des motifs généralisés (contenant des occurrences et des négations des occurrences d'itemsets) pour la découverte des règles avec négations, en tenant compte des contraintes de monotonie et d'anti-monotonie.

Toujours dans le cadre de la fouille des règles d'association avec négation, Antonie et Zaïan proposent dans [Antonie et Zaïan, 2004] un algorithme pour la fouille de règles positives (corrélation positive) et de règles négatives (corrélation négative). Cet algorithme étend le cadre standard du support et confiance par un seuil de *coefficient de corrélation*. L'algorithme découvre des règles d'association négatives avec forte corrélation entre les items antécédents et les items conséquents, mais cette approche n'est pas complète.

Le travail de [Cornelis et al., 2006] est avéré efficace pour la fouille de règles positives et négatives sans ajouter aucune autre mesure d'intérêt supplémentaire à part celle du support et de la confiance.

Dans la même intuition, Ramasubbareddy et al. extraient des règles d'association négatives *indirectes* dans [Ramasubbareddy et al., 2010] sans ajouter aucune autre mesure d'intérêt supplémentaire. Ces règles indirectes sont considérés comme un nouveau type de motifs non fréquents qui fournit un outil pour interpréter les motifs non fréquents et réduire le nombre de motifs non fréquents inutiles.
En fait, ces règles permettent de connecter indirectement deux items qui co-apparaissent

rarement via l'utilisation d'un itemset fréquent appelé *médiateur*. Ce dernier réussit à trouver les paires d'items non fréquents réellement intéressants dans une table de données. Cette approche est à la fois simple et efficace : sans ajouter aucune autre mesure supplémentaire ni des balayages additionnels à la base de données.

Dans le même contexte, sans ajouter aucun balayage supplémentaire de la base de données, mais en utilisant cette fois ci une mesure d'intérêt supplémentaire à savoir la *conviction* à coté du cadre support-confiance, Mani a proposé un algorithme permettant de fouiller des règles d'association négatives [Mani, 2012]. Ces règles présentent un filtre par rapport au nombre total de règles négatives satisfaisant le mesure *conviction*.

Conclusion : les contributions dans cet axe sont nombreuses et variées grâce à la diversité des méthodes de fouille et d'optimisation (e.g., réduction du nombre de balayages de la base de données, prise en compte des items intéressants, utilisation d'autres mesures d'intérêt, etc).

Règles d'association avec disjonction

L'intérêt à l'étude des motifs disjonctifs a touché à certaines applications du monde réel telles que l'analyse des tickets des caisses [Nanavati et al., 2001], l'analyse des données médicales [Ralbovsky et Kuchar, 2007], l'analyse des réseaux sociaux et de la bioinformatique [Zhao et al., 2006], etc.

Par conséquent, certaines approches de fouille de règles d'association se sont intéressées à l'utilisation de l'opérateur de disjonction dans le processus d'extraction des règles d'association. Ainsi, l'intérêt aux règles disjonctives a vu le jour avec Rastogi et Shim [Rajeev et Kyuseok, 1998]. Cette tentative fut la première à introduire des disjonctions dans les règles d'association mais sans utiliser la terminologie *règles disjonctives*.

En liaison étroite avec ce travail, s'inscrivent deux travaux visant à optimiser les règles disjonctives extraites dans [Rajeev et Kyuseok, 1998]. Le premier est celui de [Zelenko, 1999], où il étudie l'optimisation des règles d'association disjonctives en optimisant en plus des problèmes évoqués dans [Rajeev et Kyuseok, 1998], le problème de la règle la plus courte. À cela s'ajoute le deuxième travail qui est celui de [Elble et al., 2003], dans le cadre de trouver le support optimal dans les règles d'association pour un seul attribut numérique quelconque.

Un autre travail important est celui de [Kim, 2003], où l'auteur s'intéresse principalement à l'extraction des règles d'association avec des conclusions contenant des items mutuellement exclusifs, c'est à dire la présence de l'un des items mène à l'absence des autres et ceci en utilisant la disjonction inclusive associée à l'opérateur $\vee$. Ceci est encore plus spécifié avec Nanavati et al. 2001 [Nanavati et al., 2001] mais en utilisant la disjonction exclusive dont l'opérateur associé est noté $\bigoplus$.
Dans ce dernier travail, les auteurs évoquent certaines limites de règles d'association

conjonctives, en justifiant que les informations véhiculées par les règles d'association généralisées et en particulier les règles disjonctives ne peuvent pas être obtenues même par une collection de règles d'association conjonctives. Ainsi, ils proposent deux types de règles d'association : les règles disjonctives *simples* et les règles disjonctives *généralisées* notées *d-règles*. Les règles disjonctives simples sont celles avec une prémisse ou une conclusion (mais pas les deux) composées d'une disjonction d'items. Par contre les règles disjonctives généralisées sont des règles disjonctives telles que leurs prémisses ou leurs conclusions contiennent une conjonction des disjonctions des items. Dans les deux types de règles disjonctives, les disjonctions peuvent être inclusives ou exclusives.

Le rapport technique de [Sampaio et al., 2008] constitue une bonne étude de cas pour la fouille de règles d'association disjonctives. Les auteurs présentent le cas des logiciels orientés-objets et montrent que ces derniers sont sujets à beaucoup des changements. Ainsi, avant de mener n'importe quel changement, il est important d'estimer le coût et d'identifier quels sont les autres éléments qui doivent être changés aussi. Deux questions peuvent être posées dans ce cas : (i) étant donné que la classe A est "utilisée", quelle est la probabilité que les classes B ou C ou D soient aussi automatiquement "utilisées" avec A ? et (ii) si les classes B ou C ou D sont "utilisées", quelle est la chance que la classe A soit aussi "utilisée" ?
Les réponses à ces deux questions sont formulées par les règles disjonctives : $A \Rightarrow B \vee C \vee D$ et $B \vee C \vee D \Rightarrow A$. De manière générale et pour résoudre ce problème, les auteurs proposent un modèle de règles d'association disjonctives accompagné d'un algorithme DAR [2] qui induit des règles conformes avec ce modèle.

Finalement, le travail de [Hamrouni et al., 2010] présente un processus complet pour la fouille de règles d'association généralisées. Cependant ce travail se limite à quatre formes bien définies à savoir : disjonction $\Rightarrow$ disjonction, négation de disjonction $\Rightarrow$ négation de disjonction, disjonction $\Rightarrow$ négation de disjonction et négation de disjonction $\Rightarrow$ disjonction. En outre, ce travail ne traite que les négations sur les disjonctions, alors que la négation sur des conjonctions est aussi sémantiquement riche.

Conclusion : les règles d'association disjonctives sont de plus en plus étudiées dans cette dernière décennie grâce à leur richesse sémantique et à l'occasion qu'elles donnent aux items non fréquents d'être fouillés et d'apparaître fréquents à côté des autres items fréquents.

2.3 Fouille de motifs rares

Les motifs rares sont des motifs dont la fréquence est faible mais pouvant donner lieu à des règles exprimant une corrélation élevée. La fouille des symptômes anormaux dans les applications médicales est un exemple standard de la fouille de motifs rares. Nous

2. L'acronyme DAR : *D*isjunctive *A*ssociation *R*ules

détaillons, dans ce qui suit, les principales approches relatives à la fouille et d'itemsets et des règles d'association rares.

2.3.1 Extraction des itemsets rares

Quel que soit l'ensemble de données dans lequel les motifs rares sont fouillés, les règles d'association avec support minimal et confiance importante sont difficiles à fouiller. Ceci est encore plus vrai quand on utilise les approches standards de fouille de règles d'association. Le problème réside dans la fixation de la valeur seuil de *minsup*. En effet, si ce dernier est fixé trop faible pour la raison de trouver des itemsets fréquents impliquant des items rares, alors on risque une explosion au niveau du nombre d'itemsets fréquents. Dans le cas contraire, fixé trop élevé, alors on risque l'absence de motifs fréquents et par suite risque de perte d'informations utiles.

Afin de remédier à ce problème, certaines approches ont considéré plus qu'une valeur seuil de *minsup* au long du processus d'extraction d'itemsets respectivement des règles fréquentes.
Parmi ces approches, il y a celles qui font varier la valeur de *minsup* d'un item à un autre [Liu et al., 1999] sous prétexte que pas tous les items sont de même fréquence d'apparition et celles qui font varier cette mesure d'un niveau à un autre dans le cas de la fouille de règles multi-niveaux [Han et Fu, 1995].

L'intuition derrière le travail de [Liu et al., 1999] se résume comme suit. Dans une base de données quelconque, il y a des items qui apparaissent très fréquents, d'autres très rares. Si le *minsup* est fixé élevé alors les règles impliquant les items rares ne seront pas trouvées. Pour trouver alors des règles impliquant à la fois des items fréquents et des items rares, le *minsup* doit être fixé très faible. Cependant, ceci peut causer une explosion au niveau des règles trouvées.
[Liu et al., 1999] ont proposé l'algorithme MSApriori[3] pour résoudre ce problème. Cette technique vise à fouiller des items non fréquents en assignant différents seuils de supports minimaux aux différents items de la base de données selon leurs fréquences respectives.
Par conséquent, chaque item dans la base de données peut avoir un seuil du support minimal nommé MIS (Minimal Item Support) défini par l'utilisateur et ceci afin de mieux refléter sa nature et/ou sa fréquence. Ainsi, en fournissant différents seuils des supports minimaux, l'utilisateur peut exprimer différents besoins des supports pour différentes règles. Ces dernières doivent satisfaire différents seuils de *minsup* en fonction des items qu'elles contiennent. En opposition à l'algorithme MSApriori [Liu et al., 1999], où un seul seuil de *minsup* suffit pour fouiller des différentes règles.
Un itemset est dit fréquent s'il satisfait la valeur de *minsup* la plus faible des items qu'il contient et le support minimal d'une règle d'association est alors défini en termes des supports minimaux d'items qui y apparaissent. Ainsi, le seuil du support minimal pour

3. Multiple Support Apriori

une règle est le plus petit (Minimal Item Support) parmi les items qu'elle contient.

Exemple 11. *Soit la règle R :* $a_1\ a_2\ \ldots\ a_n \rightarrow a_{n+1}\ \ldots\ a_k$*, cette règle satisfait son support minimal, si son support est* $\geq$ *min (MIS(*a_1*), MIS(*a_2*), ..., MIS(*a_k*)).*

Les auteurs ont défini le seuil du support pour chaque item comme suit :
— MIS(i)= M(i), si (M(i)$\geq$ LS)
— sinon MIS(i)= LS.

Où, M(i) = $\beta \times$ supp(i), avec β est un paramètre ($0 \leq \beta \leq 1$) et supp(i) le support de i. Le seuil *LS*, acronyme de ("Least Support"), est le plus petit support d'item minimal autorisé et spécifié par l'utilisateur.

Exemple 12. *Cet exemple est une extension de celui de [Wan et Zeitouni, 2011].*
Par exemple soit l'itemset L = {1, 2, 3}, MIS(1)= 10%, MIS(2)= 20%, MIS(3)= 30% et supp(L)= 15% qui est supérieur à MIS(1) (i.e., le plus petit). L est considéré alors fréquent même si supp(2)< 20%

L'inconvénient de cette approche est qu'en utilisant ces multiples support minimaux, nous risquons de perdre quelques itemsets dans la phase de génération de motifs fréquents et par la suite éliminer de nombreuses règles dans la phase de génération de règles.

Pour faire face à cet inconvénient, Wan et Zeitouni empruntent dans [Wan et Zeitouni, 2011] le modèle proposé dans [Liu et al., 1999] en proposant une nouvelle définition d'itemset fréquents permettant de réduire les itemsets inutiles sans trop perdre d'informations intéressantes. Ainsi, ils proposent une nouvelle définition d'itemset fréquent. Cette définition considère qu'un itemset I est fréquent si tout sous-ensemble de I satisfait le MIS le plus petit de ses items. Par conséquent, cette définition nous permet de réduire les itemsets les plus inutiles sans perdre trop d'informations intéressantes.

Exemple 13. *Soit l'itemset L = {1, 2, 3}, MIS(1)= 10%, MIS(2)=20% , MIS(3)= 30%. L est fréquent si et seulement si supp(1)$\geq$ 10%, supp(2)$\geq$ 20%, supp(3)$\geq$ 30%, supp(1, 2)$\geq$ 10%, supp(1, 3)$\geq$ 10%, supp(2, 3)$\geq$ 20% et supp(1, 2, 3) $\geq$ 10%.*

Passons, maintenant, aux approches qui font varier la valeur de *minsup* d'un niveau à un autre. Dans [Han et Fu, 1995], les auteurs présentent une méthode efficace pour extraire des règles d'association à multiples niveaux et où les items (i.e., les concepts) sont représentés dans une *taxonomie*. Ainsi, ils considèrent des support minimaux potentiellement différents entre les niveaux mais tout en gardant la même valeur pour chaque niveau. Cette variation du support d'un niveau à un autre est justifiée par le fait que les concepts (items) situés en haut de la taxonomie possèdent des supports plus élevés que ces situés en bas de la taxonomie. En fait, chercher des règles d'associations entre des items placés en bas de la taxonomie, nécessite la réduction de la valeur seuil de *minsup*.

Toutefois, comme certaines combinaisons d'objets dans un niveau donné peuvent être soit très fréquentes ou très rares, cette méthode ne parvient pas à trouver les règles intéressantes.

Pour faire face à ce problème d'utilisation de plusieurs seuils de support et dans le but d'extraire que des règles intéressantes, certaines autres approches sont orientées vers trouver d'autres mesures d'intérêt pour fouiller que ces règles intéressantes. Parmi ces mesures, nous citons les mesures du *support pondéré* chez respectivement [Tao et al., 2003] et [Wang et al. 2000] et du *support relatif* chez [Yun et al., 2003].

En effet, grâce au *support pondéré*, les utilisateurs peuvent assigner des poids selon leurs besoins et trouver des motifs non fréquents mais présentant une valeur ajoutée. Un point critique concernant ces deux contributions ([Tao et al., 2003] et [Wang et al. 2000]) est ce qu'il est difficile d'assigner les seuils de support minimal adéquats et/ou les poids aux différents items. En plus, cette tâche devient non faisable quand on considère un grand nombre d'items.

Pour la mesure du *support relatif* [Yun et al., 2003], les auteurs proposent d'utiliser une mesure relative plutôt qu'une mesure absolue, puisque les items diffèrent l'un de l'autre par nature. Par exemple, acheter un bien de luxe dans une transaction donnée est beaucoup moins fréquent que l'achat du lait ou d'autre produit alimentaire. Par suite, les fréquences correspondantes ne peuvent pas être interprétées de la même manière.

C'est pour cela que, la mesure du *support relatif* a été introduite dans l'algorithme RSAA [4] [Yun et al., 2003] pour la fouille de règles d'association à partir des données rares mais signifiantes et en utilisant le support relatif. L'idée était de combiner les deux approches Apriori [Agrawal et al., 1993] et MSApriori [Liu et al., 1999].

Pour l'algorithme Apriori basé uniquement sur la fréquence, il ne peut pas extraire des règles avec faible support et confiance élevée. La fixation du seuil du support à des valeurs faibles engendre des règles inutiles. Pour MSApriori, le MIS (Minimal Item Support) est déterminé par la valeur de β (cf exemple 11 page 41). Ainsi, cette méthode ignore la fréquence de chaque item dans la base de données et exige de fixer une valeur appropriée de β. L'idée est alors de penser au *support relatif*.
Par conséquent :

- Les données rares signifiantes : ne satisfont pas un support minimal mais apparaissent pour être associés fortement avec d'autres données.
- Les supports : un *premier support* est utilisé dans le processus de la découverte d'itemsets fréquents et un *deuxième support* pour la découverte des items rares, sous-entendu *premier support* > *deuxième support*.
- Le Support Relatif (SuppR) d'un item i (supposé rare) exprime la relation de cet item avec un itemset candidat $(i_1, i_2, \ldots, a_k)$.

4. Relative Support Apriori Algorithm

— Le Support Relatif (SuppR) d'un itemset est égal à :
SuppR (i_1, i_2, ..., a_k)= max ($\frac{supp(i_1,i_2,...,a_k)}{supp(i_1)}$, $\frac{supp(i_1,i_2,...,a_k)}{supp(i_2)}$, ..., $\frac{supp(i_1,i_2,...,a_k)}{supp(i_k)}$).

L'algorithme RSAA considérant le *support relatif* est montré plus efficace que Apriori et MSApriori.

Et depuis, les recherches sur la fouille d'itemsets rares ont progressé, même si ce n'est pas avec la même fréquence que les travaux s'intéressant à la fouille d'itemsets fréquents.

Pillai et al. [Pillai et Vyas, 2011] ont étudié la fouille d'itemsets rares utiles. Ils affirment que, dans certaines applications du monde réel telles que le marketing en détail, le diagnostic médical, la segmentation de la clientèle, etc, l'utilité d'itemsets est basée non pas sur la fréquence mais plutôt sur d'autres facteurs tels que : le coût, le profit ou le revenu, ou sur d'autres préférences des utilisateurs.
En effet, les itemsets utiles peuvent être fréquents ou rares. De même, les itemsets rares peuvent être utiles comme ils peuvent ne pas l'être. Les itemsets rares fournissent des informations utiles dans différents domaines de prise de décision.
Ainsi, ils ont proposé dans [Pillai et Vyas, 2011], un algorithme appelé HURI [5] pour la fouille d' itemsets rares. Cet algorithme est construit en deux phases : une pour la génération d'itemsets rares et une pour la génération d'itemsets rares utiles selon les intérêts des utilisateurs. En outre, l'algorithme utilise le concept de Apriori-Inverse [Koh et Rountree, 2005], pour la génération d'itemsets rares utiles pour les utilisateurs. Les mêmes auteurs ont étudié dans [Pillai et al., 2012] l'évaluation de la performance et l'analyse de la complexité de l'algorithme HURI proposé dans [Pillai et Vyas, 2011]. Dans le même contexte, mais cette fois-ci en incorporant l'aspect temporel, la même équipe a proposé dans [Pillai et Vyas, 2014] une approche appelée THURI [6] pour extraire efficacement des itemsets rares de grand intérêt à partir des bases de données temporelles.

Une autre technique de fouille d'itemsets rares a été élaborée dans l'algorithme Apriori-Inverse [Koh et Rountree, 2005], où les auteurs ont défini le concept des règles sporadiques avec support minimal et confiance élevée. Ils proposent alors l'algorithme Apriori-Inverse, qui ignore tous les itemsets candidats dont le support dépasse un seuil du support maximal. Un peu plus tard, dans leur livre intitulé *Rare Association Rule Mining and Knowledge Discovery : Technologies for Infrequent and Critical Event Detection*, les auteurs couvrent tous les sujets en relation avec les techniques d'extraction, les mesures d'intérêt, les domaines d'application dans le monde réel, etc.

Toujours dans le même contexte de la fouille de motifs rares et dans le cas d'une base de données médicales, Maumus et al. et Szathmary et al. ont étudié respectivement, le

5. High Utility Rare Itemsets
6. Temporal High Utility Rare Itemsets

problème de l'identification de la cause de Maladies Cardio Vasculaires (MCV) comme un cadre applicatif pour l'extraction de motifs rares [Maumus et al., 2006], [Szathmary et al., 2006] et [Szathmary et al., 2007].

Si la règle d'association : {un niveau élevé de cholestérol} → { MCV} est fréquente (i.e., extraite à partir d'un motif fréquent) alors les individus ayant un fort taux de cholestérol ont un risque élevé de MCV.
Cependant, s'il existe un nombre important de "végétariens" dans la base de données, alors une règle d'association rare {végétarien} → {MCV} implique qu'un végétarien a un risque faible de contracter une MCV. Ainsi dans ce cas, le motif {végétarien} est fréquent, le motif {MCV} est fréquent alors que le motif {végétarien, MCV} est un motif rare.

L'ensemble des motifs rares minimaux forme un ensemble générateur minimal à partir duquel tous les motifs rares peuvent être retrouvés. La même équipe s'intéresse dans [Szathmary et al., 2010] à la génération de règles d'association rares à partir de l'ensemble d'itemsets rares extraits dans [Szathmary et al., 2007].

Dans [Tsang et al., 2013], les auteurs affirment que les règles d'association rares sont parfois plus intéressantes que celles fréquentes puisqu'elles représentent des associations inattendues ou inconnues. Ainsi, la méthode présentée dans [Tsang et al., 2013] diffère des autres méthodes classiques utilisant une approche par niveau comme Apriori, tant qu'elle utilise la structure d'arbre. Cette approche consiste en la proposition d'un algorithme nommé RP-Tree[7] pour la fouille d'un sous-ensemble de règles d'association rares. Cet algorithme est moins coûteux en termes de coût d'exécution dans les étapes de génération des candidats et d'élagage, puisqu'il n'a pas besoin de générer et tester toutes les combinaisons possibles des items rares. Les expérimentations ont montré que le temps d'exécution de cet algorithme varie en fonction de la longueur de la transaction et de la taille de l'itemset rare.

En termes de conclusion, et selon [Weiss, 2004], il est important d'étudier la rareté dans le contexte de fouille de données parce que les objets rares sont typiquement plus difficiles à identifier que les objets communs ou fréquents et que la plupart des algorithmes de fouille de données ont une grande difficulté à traiter la rareté.

2.3.2 Extraction des règles d'association rares

L'étude de la rareté mérite une attention particulière parce qu'elle présente d'importantes difficultés pour les algorithmes de fouille de données. Nous présentons dans ce qui suit les différentes approches qui traitent la fouille de règles d'association dans le cadre de rareté. Nous classons ces approches entre celles qui négligent la mesure support et se suffisent de la mesure confiance, et celles qui ajoutent au delà de la mesure confiance d'autres mesures.

7. Rules Pattern-Tree

Négligence de la mesure support

Les propositions de [Wang et al. 2000, Wang et al., 2001] étaient de trouver des règles d'association valides directement à partir de leur confiance. Bien que la confiance ne possèdait pas une propriété de fermeture vers le bas, les auteurs utilisent un élagage basé sur la confiance dans leur processus de génération des règles.
En effet, dans [Wang et al. 2000], les auteurs abandonnent le support minimal et utilisent les règles d'association qui satisfont seulement la confiance minimale appelées *règles-confiantes* dans le but de construire un classificateur. Cette approche permet de retrouver des règles confiantes sans examiner toutes les règles, en fournissant un élagage basé sur la confiance exploitant une certaine propriété de monotonie de la confiance. Cette propriété s'appelle "existential upward closure".
En effet, la confiance ne bénéficie pas de la fermeture vers le haut comme la cas du support. Par exemple, si abc est fréquent alors ab, ac , bc , a, b et c sont tous fréquents. Cependant, si être jeune et masculin ensemble influent positivement sur l'achat des services Internet alors Age.jeune $\rightarrow$ acheter.oui et Genre.M $\rightarrow$ acheter.oui pourraient avoir des confiances plus faibles que Age.jeune et Genre.M $\rightarrow$ acheter.oui.
De même, Age.agé, Genre.F $\rightarrow$ acheter.oui peut avoir une confiance faible à celle de Age.agé $\rightarrow$ acheter.oui et Genre.F $\rightarrow$ acheter.oui. Donc, l'élagage simple basé sur la fermeture vers le haut ou vers le bas n'est pas opérationnel.
C'est ainsi que, les auteurs ont proposé la propriété suivante : "existential upward closure".

Théorème 8. *Soit $X \rightarrow Y$ une règle d'association et A_i n'importe quel attribut, qui n'est pas dans $X \rightarrow Y$.*

- *(i) certaines A_i-spécialisations de $X \rightarrow Y$ ont au moins la confiance de $X \rightarrow Y$.*
- *(ii) si $X \rightarrow Y$ est confiante, alors certaines A_i-spécialisations de $X \rightarrow Y$ le sont.*

Exemple 14. *Soit les règles suivantes :*
r_1 *: Age.jeune $\rightarrow$ acheter.oui*
r_2 *: Age.jeune, Genre.F $\rightarrow$ acheter.oui, et*
r_3 *: Age.jeune Genre.M $\rightarrow$ acheter.oui.*

Au moins l'une de deux règles r_2 et r_3 possède autant de confiance que r_1.

Dans [Wang et al., 2001], le problème était également de trouver toutes les règles qui satisfont une confiance minimale mais pas nécessairement un support minimal. Étant donné que la stratégie d'élagage classique basée sur le support est inapplicable parce qu'on néglige la mesure du support. La monotonie de la confiance est expliquée comme suit : si une règle de taille k est confiante (pour un seuil de confiance donné), alors pour tout autre attribut A_i qui n'appartient pas à cette règle, les spécialisations de taille $k+1$ utilisant l'attribut A_i doivent être confiantes. Pour générer les règles confiantes candidates de taille k, nous avons besoin uniquement d'examiner les règles confiantes de taille $k+1$.

Considération d'autres mesures de corrélation

D'autres approches se sont orientées vers la prise en compte d'autres mesures de corrélation appropriées (outre que le support et la confiance) pour fouiller les motifs fortement corrélés.
Une mesure *h-confiance* a été étudiée dans [Xiong et al., 2003, Xiong et al., 2006].
Dans [Xiong et al., 2003], les auteurs affirment que la stratégie d'élagage basée sur le support n'est pas tout à fait efficace surtout quand les ensembles de données sont avec des distributions de supports asymétriques. Dans ce cas, cette stratégie tend ou bien à générer plusieurs motifs faux impliquant des items à partir des différents niveaux ou bien à oublier des motifs avec faibles supports mais qui sont très intéressants. Pour surmonter ce problème, les auteurs proposent le concept de motifs *hyper-cliques* qui utilisent la mesure *h-confiance* pour identifier les motifs d'affinité.

Définition 26. ***Mesure h-confiance***
Le $h-confiance$ *d'un itemset* $I = \{i_1, i_2, \ldots, i_n\}$ *est définie comme h-conf(I) = min* $[conf(i_1 \rightarrow i_2, \ldots, i_n), conf(i_2 \rightarrow i_1, i_3 \ldots, i_n), \ldots, conf(i_n \rightarrow i_1, i_2 \ldots, i_{n-1})]$*, où* $conf$ *est la mesure de confiance standard.*

Dans [Xiong et al., 2006], un nouvel algorithme appelé HCM [8] a été développé et qui utilise les propriétés de cross-support et d'anti-monotonie de la mesure $h-confiance$ pour la découverte efficace des motifs hyper-cliques. En effet, si I et $I^{'}$ deux itemsets et si $I \subseteq I^{'}$, alors $h-confiance(I) \geq h-confiance(I^{'})$. La mesure h-confiance satisfait la propriété de cross-support qui peut aider efficacement à éliminer les motifs faux impliquant des items avec des niveaux de support considérablement différents. En plus, cette propriété de support-cross n'est pas limitée à la mesure h-confiance et peut être généralisée à d'autres mesures d'association (e.g., la mesure d'intérêt).

A part ces deux travaux [Xiong et al., 2003, Xiong et al., 2006] qui s'intéressaient à la mesure *h-confiance*, d'autres se sont intéressés à la mesure *bond* [Bouasker et al., 2008, Ben Younes et al., 2010].

Dans le but de chercher plus d'informations concernant la corrélation entre les items d'un tel itemset, les chercheurs se sont intéressés à l'étude d'autres mesures de corrélation. Étant donné que les travaux de recherche de motifs fréquents s'intéressent principalement à la fréquence d'apparence et non pas aux dépendances au sein des ensembles des items. C'est ainsi que, dans [Ben Younes et al., 2010], les auteurs présentent une nouvelle représentation concise exacte des motifs fréquents corrélés en adaptant la mesure de corrélation *bond*. Cette mesure introduite par [Omiecinski, 2003], calcule le rapport entre le support conjonctif et le support disjonctif d'un itemset. Pour un itemset quelconque I de $\mathcal{I}$, $bond(I) = \frac{supp(I)}{supp_{\vee}(I)}$.

Le nombre de motifs corrélés étant très important permet de définir une représentation condensée de la manière suivante. Cette représentation condensée est définie à

8. Hyper-Clique-Miner

travers la définition d'un nouvel opérateur de fermeture lié à la mesure *bond*.

Les auteurs introduisent un nouvel opérateur de fermeture associé à la mesure *bond*, noté f_{bond}. L'application de f_{bond} sur n'importe quel motif est exactement égale à l'intersection de ses deux fermetures conjonctive f_c et disjonctive f_d.

Définition 27. ***Opérateur de fermeture*** f_{bond}
$f_{bond}(I) = \{\ i \in \mathcal{I} \mid i \in f_c(I) \wedge i \in f_d(I)\ \}$

De même, les auteurs ont défini les motifs corrélés fréquents par apport à un seuil de corrélation $min - bond$ et un seuil de support $minsup$ comme suit :

Définition 28. ***Motifs Corrélés Fréquents***
$MCF = \{\ I \in \mathcal{I} \mid bond(I) \geq min\text{-}bond\ et\ supp(I) \geq minsup\}$.

Ayant défini les MCF et l'opérateur de fermeture f_{bond}, les auteurs définissent par la suite la représentation condensée des motifs fréquents corrélés fermés associés à f_{bond}. Cette représentation est une représentation exacte de l'ensemble des MCF.

Définition 29. ***Représentation de Motifs Corrélés Fréquents Fermés***
$RMCFF = \{\ (I, supp_{\wedge}(I), supp_{\vee}(I)) \mid I \in MCFF\}$.

Dans la même intuition, [Bouasker et al., 2008] s'intéressent à la fouille de motifs corrélés qui sont peu fréquents /rares et qui représentent une problématique intéressante dans la fouille de données. Il s'agit en fait de la fouille de motifs corrélés rares selon la mesure de corrélation *bond*. Une nouvelle piste de recherche prometteuse dans le domaine de la fouille de données, et qui consiste à l'intégration des mesures de corrélation lors de l'extraction de motifs rares pour : (i) améliorer la qualité des connaissances extraites formant un ensemble plus réduit contenant que des motifs intéressants qui sont rares mais fortement corrélés : et (ii) renforcer la qualité de règles d'association dérivées à partir de ces motifs.
Par exemple, des motifs fortement corrélés mais peu fréquents dans les transactions d'une supermarché, peuvent être omis dans un processus classique de fouille de motifs fréquents.
De même, il a été montré dans [Bouasker et al., 2008, Ben Younes et al., 2010], que la forme disjonctive des motifs peut être dérivée à partir des motifs corrélés, en utilisant les techniques de la règle de De Morgan et des identités d'inclusion-exclusion [Hamrouni et al., 2010], (voir page 28).

Dans [Omiecinski, 2003], l'idée était de combiner les deux mesures précédentes *h-confiance* et *bond*. Ainsi, l'auteur discute trois mesures d'intérêt alternatives aux règles d'association à savoir *any-confiance*, *all-confiance* et la mesure *bond*.

Soit $I = \{i_1, i_2, \ldots, i_m\}$ un itemset de $\mathcal{I}$ de support $supp(I)$ et soit aussi $max{-}item{-}sup$ d'un itemset I le support maximal des items de I. La $all{-}confiance$ de I est la confiance minimale parmi l'ensemble des règles d'association $i_j \rightarrow I - i_j$, tel que $i_j \in I$.

Définition 30. ***Mesure all-confiance, Mesure any-confiance***
All-confiance d'un itemet I est définie par :

$$\frac{supp(X)}{\max_{i \in I}(supp(i))} \tag{2.1}$$

Any-confiance d'un itemet I est définie par :

$$\frac{supp(I)}{\min_{i \in I}(supp(i))} \tag{2.2}$$

Il est facile à démontrer que pour un itemset I : any-confiance(I) $\geq$ all-confiance(I) $\geq$ bond(I). D'autre part, la propriété importante de fermeture vers le bas ne s'applique qu'aux mesures *all-confiance* et *bond* et elle ne s'applique pas à la mesure *any-confiance*.

Les auteurs de [Omiecinski, 2003] confirment aussi que si les règles d'association ont une valeur minimale pour *all-confiance* ou pour *bond*, alors ces règles vont avoir une limite inférieure pour leur support minimal. En plus, les règles produites à partir de ces règles vont avoir aussi une certaine limite inférieure sur leur confiance minimale.

Il est important de noter que la plupart des approches de fouille de règles d'association rares se concentrent sur les motifs *conjonctifs* construits en utilisant des items fréquents ou non fréquents. Un risque majeur lorsqu'on considère des conjonctions d'items non fréquents est que ces itemsets ont un support très faible et ainsi, il est difficile de caractériser ces motifs lorsque les données sont bruitées.
Cette problématique a été étudiée avec [Hussain et al., 2000] à travers la fouille de règles d'exception en considérant une nouvelle mesure objective d'intérêt relatif, qui lie le sens commun aux règles de référence dans les données lors de l'estimation de l'intérêt.
De manière générale, les auteurs définissent une nouvelle mesure efficace fournissant un moyen de plus dans la fouille des règles d'exception avec d'autres mesures d'intérêt telles que le support et la confiance. Cette mesure malheureusement non nommée est en fonction de trois paramètres : le support (S), la confiance (C) et les connaissances sur les règles du sens commun (K), ainsi l'intérêt $I = f(S, C, K)$.

Il est vrai que l'intérêt est une question relative qui dépend d'autres connaissances préalables. Cependant, cette estimation, peut être biaisée (déformée) à cause de la connaissance du domaine qui est incomplète ou non exacte. Même s'il est possible d'estimer l'intérêt, il n'est pas si trivial de juger l'intérêt à partir d'un énorme ensemble de règles fouillées.

Pour notre cas (i.e., notre contribution dans le cadre de cette thèse), nous comptons fouiller des règles d'association disjonctives à partir d'itemsets disjonctifs-fréquents minimaux. Ces derniers sont extraits à partir des items non fréquents. Pour garantir que nous extrayons uniquement des motifs utiles et intéressants, nous supposons et calculons

un critère additionnel, basé sur une mesure de similarité entre les items, pour la fouille de nos motifs non fréquents.

Par opposition à [Bouasker et al., 2008, Ben Younes et al., 2010], où les motifs corrélés sont générés à partir des items fréquents, notre travail considère à la base des items non fréquents pour fouiller des disjonctions fréquentes minimales. Par conséquent, nous ne pouvons pas utiliser leurs résultats dans notre approche.

De même, nous notons que dans [Han et Fu, 1995], les auteurs considèrent une taxonomie pour fouiller les itemsets fréquents construits à partir des items du même niveau de la taxonomie. Cependant, notre approche diffère fondamentalement de ce travail. En effet, dans [Han et Fu, 1995], seuls les items fréquents sont considérés (tandis que nous considérons les items non fréquents) et le seuil du support change d'un niveau à un autre dans la taxonomie (tandis que nous gardons le même seuil du support durant tout le processus de la fouille).

2.4 Conclusion

Dans ce chapitre, nous avons passé en revue l'ensemble des approches de la littérature en liaison avec la fouille de motifs fréquents et de motifs rares. Dans le cadre de l'extraction de motifs fréquents (itemsets, règles d'association et représentations concises), nous avons évoqué le problème de la fouille de règles d'association généralisées et en particulier les règles négatives et les règles disjonctives. La fouille de règles disjonctives reste un problème difficile puisque ces dernières n'ont pas la propriété de fermeture du support conjonctif, qui est présente dans le cas des règles d'association conjonctives.
Dans le chapitre suivant, nous mettrons l'accent sur une (parmi d'autres) technique permettant de fouiller que des règles d'association utiles, à savoir la supposition d'une structure hiérarchique e.g., une taxonomie sur l'ensemble des items de la base de données.

Chapitre 3

Taxonomies et règles d'association

3.1 Introduction

Dans ce chapitre, nous étudions l'utilisation des taxonomies comme un cas particulier des structures hiérarchiques générales, dans le domaine de l'extraction d'itemsets peu fréquents/rares et de la génération de règles d'association. Nous présentons en première section l'architecture de la taxonomie et les mesures de similarité associées pour étudier l'homogénéité entre ses différents concepts. Dans la deuxième section, nous étudions les approches en liaison avec l'utilisation des taxonomies dans le processus d'extraction de règles d'association intéressantes.

3.2 Taxonomies et mesures de similarité

Dans cette section, nous étudions les structures hiérarchiques et les mesures de similarité associées.

3.2.1 Structures hiérarchiques, ontologies et taxonomies

Les structures hiérarchique, les ontologies et les taxonomies, si elles partagent l'organisation hiérarchique des concepts entre eux, n'ont pas les mêmes usages, ni les mêmes objectifs.
Selon [Fankam et al., 2009], une ontologie peut être définie comme étant une représentation formelle, référentielle et consensuelle de l'ensemble des concepts partagés d'un domaine.
De façon plus opérationnelle, l'ontologie cherche à décrire de façon formelle un domaine de connaissances, en identifiant les types d'objets de ce domaine, leurs propriétés et leurs relations. Les relations dans les ontologies permettent de mieux comprendre les choses, e.g, la relation d'inclusion (classe, sous-classe), relations d'opérations ensemblistes (union, intersection, etc), relations de caractéristiques des propriétés (transitivité, de cause/effet), etc.
Sur le plan sémantique, l'ontologie exprime donc des connaissances qui sont validées

par une communauté donnée. Les concepts y sont souvent désignés par des nœuds et les relations entre eux par des arcs. Les concepts peuvent être caractérisés par des attributs dont les valeurs sont de différents types : nombres, booléens, intervalles, mots, etc.

Selon [Candolle, 1813], une taxonomie est une classification systématique et hiérarchisée des taxons dans diverses catégories selon les caractères qu'ils ont en commun, des plus généraux aux plus particuliers. Il s'agit d'une structure arborescente utilisée pour désigner tout système de classification/ catégorisation. Elle organise les concepts selon des relations hiérarchiques de type "est-un", "père-fils" ou "générique-spécifique".
Selon [Saint-Dizier, 2006], les taxonomies se caractérisent par deux propriétés fondamentales :

— *La transitivité :* si C est un sous-type de B et B un sous-type de A alors C est un sous-type de A.
— *L'héritage descendant des propriétés :* si B est un sous-type de A, alors B hérite de A toutes ses propriétés.

Les ontologies impliquent généralement une portée d'informations plus large. Les gens font souvent référence à une taxonomie comme un "arbre", et en étendant cette analogie, une ontologie comme une "forêt". Une ontologie pourrait englober un certain nombre de taxonomies, chaque taxonomie organise un sujet d'une manière particulière.

Différents types des schémas de classification

De nombreux schémas de classification ont pour but d'aider les utilisateurs à naviguer dans des informations et leur fournir un accès clair et sans ambiguïté à l'information [Cruse, D. A., 1986]. Dans ce qui suit, nous proposons les particularités de certains schémas de classification selon lesquels les concepts sont organisés.

— *Les vocabulaires contrôlés :* listes restrictives des termes préférés. Ces termes sont utilisés dans un but spécifié, généralement pour l'indexation, l'étiquetage et la catégorisation. Un terme est une représentation de concept (sous forme de mot ou groupes de mots), un concept pouvant être représenté par plusieurs termes (c'est le cas des synonymes). Les vocabulaires sont dit *contrôlés* dans le sens où seuls les termes de la liste peuvent être utilisés pour le domaine couvert. [Cassel, J., 2011]
— *Les hiérarchies :* quand nous pensons aux taxonomies, les systèmes de classification hiérarchiques sont ceux qui viennent généralement à l'esprit. Une taxonomie hiérarchique est une sorte de vocabulaire contrôlé où chaque terme est relié à un terme désigné plus large (sauf si c'est le terme de niveau supérieur) et à un ou plusieurs termes étroits (sauf si c'est un terme de niveau inférieur). Ainsi, tous les termes sont organisés en une seule grande structure hiérarchique. [Cassel, J., 2011]
— *Les thésaurus :* contrairement aux taxonomies, les thésaurus (ou thesauri) ne sont pas (seulement) hiérarchiques, mais constituent cependant un élargissement des taxonomies en intégrant, au-delà des relations hiérarchiques, d'autres propriétés pour décrire les sujets. Un thésaurus est une sorte de dictionnaire qui contient des synonymes ou des expressions alternatives (peut-être même des antonymes)

pour chaque terme en entrée. Un thésaurus est donc un type plus structuré que le vocabulaire contrôlé, il fournit des informations sur chaque terme et sur ses relations avec d'autres termes dans le même thésaurus. [Cassel, J., 2011]

— *Navigation par facettes :* ou "recherche à facettes" désigne une classification selon plusieurs facettes. La classification à facettes définit une façon de décrire une ressource selon plusieurs axes (les facettes), chaque facette contenant des termes qui peuvent être décrits dans un thésaurus, un terme appartenant à une seule facette.

— *Les ontologies :* une ontologie offre un degré de sophistication supérieur aux schémas précédents. Elle peut être considérée comme un type de taxonomie avec encore plus de relations complexes entre les termes que dans un thésaurus. En fait, une ontologie vise à décrire un domaine par ses termes (appelé *individus* ou instances) et leurs relations. Les relations entre les termes dans une ontologie ne sont pas limitées à la relation large/étroit et ses connexes. Au contraire, il peut y avoir un certain nombre de types de relations spécifiques à un domaine, comme propriétaire/appartient à, produit/est produit par, a des membres/est un membre de, etc [Cassel, J., 2011].

Applications des taxonomies

Une façon plus pratique pour classer les taxonomies est de le faire selon leurs application et leurs utilisations. Cependant, une taxonomie peut certainement servir de multiples fonctions. Les taxonomies servent principalement une des trois fonctions suivantes, mais il peut certainement être des combinaisons de ces différentes fonctions [Sharman et al., 1999] :

— *support d'indexation :* une taxonomie est une liste des termes convenus pour l'indexation humaine ou le catalogue des documents multiples et/ou pour l'indexation réalisée par plusieurs indexeurs pour garantir de la cohérence [Zarri et al., 1999].

— *support de récupération/retrieval :* une taxonomie qui sert pour l'indexation sert également pour la récupération chez l'utilisateur final. Il y a aussi des taxonomies conçues pour aider la récupération des résultats de recherche sans soutenir l'indexation humaine. Ces taxonomies sont typiquement des tables de mapping des termes et de leurs synonymes conçues pour faciliter la recherche en ligne.

— *support d'organisation et de navigation :* une taxonomie, comme une hiérarchie, peut fournir un système de catégorisation ou de classification des objets ou des informations.

3.2.2 Mesures de similarité

Afin de pouvoir établir des relations entre les concepts d'une telle ontologie, il est nécessaire de trouver une mesure qui permet d'en évaluer leur similitude et/ou leur dissimilitude. Cependant, il est essentiel de faire la différence entre la notion de la relation sémantique et celle de la similarité. En effet, deux concepts sont dits *similaires* s'ils satisfont une certaine relation sémantique de ressemblance. Par ailleurs, deux concepts

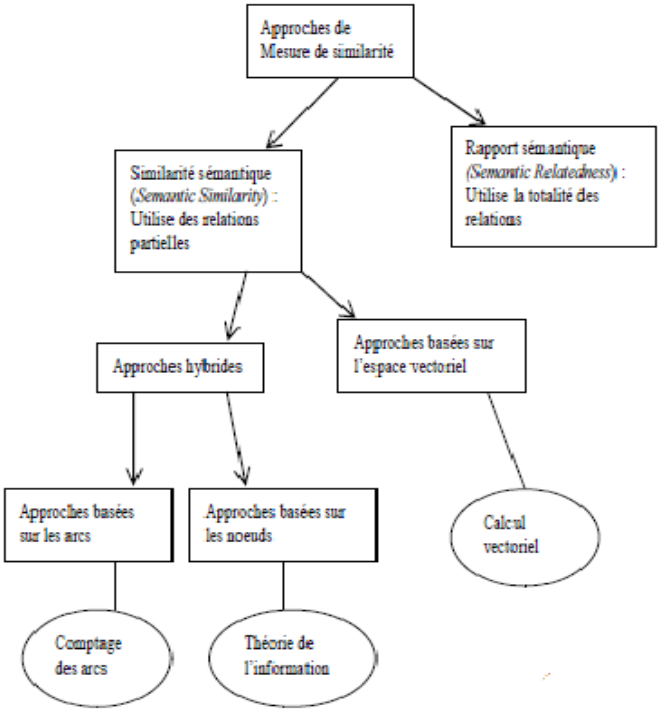


FIGURE 3.1 – Approches de mesure de similarité.
[Slimani et al., 2007]

notés comme dis-similaires peuvent également être liés sémantiquement mais par d'autres types de relations telles que : antonymie, métonymie, etc.

Selon [Rezgui et al., 2013], nous pouvons distinguer trois classes de mesures sémantiques entre les concepts et qui sont couramment utilisées :

Similarité sémantique quand la mesure calcule/évalue si les deux concepts sont sémantiquement similaires, i.e., ils partagent des propriétés et des attributs communs.

Parenté sémantique quand la mesure calcule/évalue si les deux concepts sont apparentés sémantiquement, i.e., ils sont connectés dans leur fonction. C'est un cas général de la similarité sémantique, étudiée dans les travaux de [Resnik et al., 1995] et [Budanitsky et Hirst, 2006].

Distance Sémantique quand la mesure calcule/évalue deux concepts qui sont sémantiquement distants/loin. En effet, selon [Budanitsky et Hirst, 2006], la distance sémantique est l'inverse de la parenté sémantique, i.e., plus deux termes sont sémantiquement liés, plus ils sont sémantiquement proches.

Nous nous intéressons dans le cadre de cette thèse seulement aux mesures de la similarité sémantique. Pour plus des détails sur les autres classes de mesures, nous invitons le lecteur intéressé à lire l'article [Rezgui et al., 2013].

Dans la littérature, nombreuses sont les approches de mesure de similarité, qui peuvent être classées en : approche reposant uniquement sur la structure hiérarchique, approche utilisant le contenu informationnel et incluant d'autres informations à part celles de la structure hiérarchique, approche hybride et approche basée sur l'espace vectoriel (cf figure 3.1). Les mesures de similarité varient donc du simple calcul du nombre d'arcs à l'intégration d'autres mesures statistiques.

Approche basée sur les arcs : basée sur la longueur des chemins dans un arbre pour déterminer la distance entre deux concepts. Cette approche suppose que les arcs d'une taxonomie représentent des distances uniformes. Dans [Rada et al., 1989], les auteurs calculent la distance conceptuelle entre les concepts par le chemin le plus court et la considèrent comme un moyen efficace pour évaluer la similarité sémantique, sans tenir compte des positions des arcs dans la hiérarchie des concepts. Cependant, dans [Zhong et al., 2002], les auteurs tiennent compte de la position des concepts dans la hiérarchie. De même, une autre mesure proposée par [Wu et Palmer, 1994] et très liée à celle de [Zhong et al., 2002], tient compte de la profondeur du plus petit généralisateur commun dans le calcul de la mesure de similarité.

Le problème avec cette classe de mesures est que chaque mesure de similarité est liée à une application particulière. De plus, elles ont toutes l'avantage d'être faciles à implémenter.

Approche basée sur les nœuds : utilise des mesures tenant compte du contenu informationnel pour déterminer la similarité conceptuelle. Dans [Resnik et al., 1995], l'auteur propose une nouvelle mesure de similarité sémantique basée sur le contenu informationnel.

Dans [Lin, 1998], l'auteur a essayé de proposer une définition de la mesure de similarité universelle, i.e., la mesure est applicable dans différents domaines.

Approche hybride : elle combine les propriétés de deux premières approches à savoir l'approche basée sur les arcs et l'approche basée sur les nœuds/le contenu informationnel. Ainsi, certaines autres mesures ont été dérivées de ces deux dernières mesures et plusieurs manières sont possibles pour déterminer la similarité sémantique. Parmi ces mesures, nous citons celle de [Jiang et Conrath, 1997], et où les auteurs proposent une nouvelle approche pour mesurer la similarité sémantique entre les concepts. Cette approche hérite de la façon de calcul des arcs à partir de l'approche basée sur les graphes/les arcs, ainsi que de l'approche basée sur les nœuds la manière de calculer le contenu informationnel.

De plus, la mesure de *Leacock and Chodorow* [Leacock et Chodorow, 1998], tient compte de la longueur du chemin entre les concepts dans une ontologie restreinte aux liens taxonomiques et à la profondeur de la taxonomie. De même, elle permet

d'éviter le calcul de la teneur en information, mais elle maintient le concept de la théorie de l'information.

Approche basée sur l'espace vectoriel : elle utilise un vecteur caractéristique k-dimensions représentant chaque objet/concept et puis calcule la similarité en se basant sur la mesure de cosine ou la distance euclidienne [Salton et McGill, 1983] et [Baeza-Yates et Ribeiro-Neto, 1999]. La définition de la similarité entre deux vecteurs d'objets est obtenue par leurs contenus internes. Parmi ces similarités, nous citons la similarité de *Jaccard*, la similarité de *Cosine*, la similarité de *Dice* [Lin, 1998], etc.

Dans cette thèse, nous utilisons le modèle basé sur les graphes (structure arborescente) et une mesure de similarité basée sur les arcs. Ce modèle suppose que la hiérarchie des concepts est structurée en fonction de la similarité sémantique.

Par conséquent, des concepts de l'ontologie sont similaires si la distance qui les sépare est faible. (Respectivement, ils sont dis-similaires si la distance qui les sépare est importante).

C'est dans ce cadre, se situe la mesure de similarité de Shekar et Natarajan [Shekar et Natarajan, 2004] pour le calcul de degrés de parenté entre les items d'une base de données transactionnelle. Cette mesure, appelée *Item-Relatedness* ou en français *Parenté-globale* et est classée comme étant une mesure sémantique, est celle que nous utilisons par la suite.

Pour qualifier si une règle d'association est intéressante ou non, la similarité est étudiée dans une règle d'association en deux niveaux : entre les items de la prémisse et entre les items de la conclusion.

L'objectif principal de la mesure introduite dans [Shekar et Natarajan, 2004] est de mesurer la parenté entre les items des règles d'association déjà découvertes. En fait, les auteurs ont proposé d'utiliser une taxonomie floue dans le but de décrire les relations entre les items des règles d'association. La différence entre les taxonomies simples et les taxonomies floues, est ce que ces dernières permettent à un nœud d'avoir des parents multiples (ce qui n'est pas le cas pour les taxonomies simples). Par ailleurs, elles permettent les relations pondérées de type *est-un*. Pour bien illustrer la mesure introduite par Shekar et al., nous présentons ses différentes composantes et les calculs nécessaires en se référant à la figure 3.2. Ainsi, cette dernière illustre un exemple d'une taxonomie des produits alimentaires où les rectangles représentent les concepts à classifier effectivement et les cercles des concepts plus généraux.

Nœud de plus-haut-niveau du chemin $[H_{A,B}(P)]$: le nœud qui apparaît au plus haut niveau dans le chemin p qui connecte A et B. Par exemple, $H_{Pomme,Salade}$(P) = Produits alimentaires.

Nœud de plus-haut-niveau d'appartenance $[HA_{A,B}(P)]$: est donné par

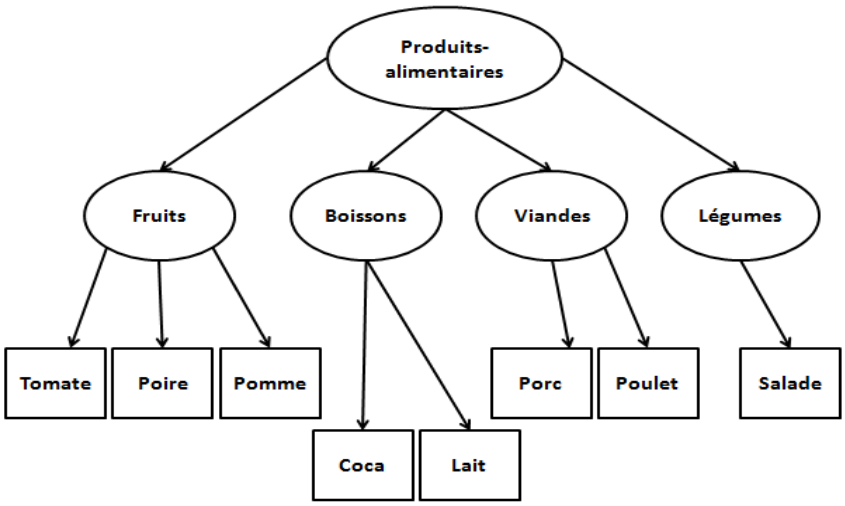


FIGURE 3.2 – Exemple d'une taxonomie.

le minimum des valeurs d'appartenance de deux items A et B dans le nœud de plus-haut-niveau du chemin $H_{A,B}$(P), (sachant que dans le cas d'une taxonomie floue, chaque item possède une valeur d'appartenance vers un père donné.) Supposons que la valeur d'appartenance de $Pomme$ est de 0.5, celle de $Salade$ est de 0.6, alors $HA_{Pomme,Salade}$(P) = 0.5.

Plus-haut niveau de parenté [$HP_{A,B}$**(P)**] : la parenté entre deux items est déterminée par le niveau du nœud situé au niveau le plus élevé qui les connecte dans le chemin p. $HP_{A,B}$(P) = le niveau ($H_{A,B}$(P)). Par exemple, $HP_{Pomme,Salade}$(P)=0.

Nœud de séparation de parenté [$NSP_{A,B}$**(P)**] : la longueur du chemin le plus simple p connectant A et B en terme de nœuds (concepts). Par exemple $NSP_{Pomme,Salade}$(P)= 3.

Ainsi, la mesure de Shekar et Natarajan [Shekar et Natarajan, 2004] est définie comme suit :

$$OR_{A,B}(P) = \frac{(1+HP_{A,B}(P))\times(HA_{A,B}(P))}{NSP_{A,B}(P)}$$

Shekar et al. comparent cette mesure de parenté introduite à d'autres mesures de similarité sémantique étant donné que les notions de similarité et de parenté sont liées : *(i)* mesure de [Resnik, 1999] ; et *(ii)* mesure de [Wu et Palmer, 1994]. La mesure de

Shekar et al. considère les différentes relations possibles entre deux items et aussi tient compte de la capacité des items de pouvoir être substitués l'un par l'autre.

3.3 Taxonomies dans le processus d'extraction de règles d'association

Les ontologies (respectivement les taxonomies) sont couramment utilisées dans différentes étapes du processus ECD (Extraction de Connaissances à partir des Données). En effet, dans l'étape de pré-traitement, il s'est avéré que les ontologies peuvent proposer les techniques les plus appropriées de pré-traitement. Dans l'étape de traitement, les ontologies ont servi à choisir les algorithmes adéquats pour le traitement. Enfin, dans l'étape de post-traitement, les ontologies ont contribué dans la génération des motifs pour choisir le modèle le plus intéressant concernant les connaissances découvertes [Cannataro et Comito, 2003].

L'utilisation des ontologies dans le domaine de la fouille de règles d'association a commencé avec l'utilisation d'un cas particulier des ontologies qui sont les taxonomies. Les premières idées ont été introduites par [Srikant et Agrawal, 1995] avec le concept de règles généralisées dans le but de généraliser/spécifier les règles.
Une étude des travaux de la littérature nous a permis de classer les travaux reliant les ontologies à la fouille des différents types de règles d'association en trois catégories, à savoir une catégorie où les ontologies sont utilisées dans une phase de pré-traitement, une catégorie dans la phase du traitement et une catégorie dans la phase du post-traitement.

3.3.1 Ontologies lors de la phase de pré-traitement

Dans [Xiangdan et al., 2005], les auteurs affirment que l'utilisation des ontologies permet d'inclure les connaissances spécifiques du domaine et ceci pour faciliter le processus d'extraction de règles d'association, et trouver des règles à plusieurs niveaux. Dans le même contexte, [Brisson, 2006] et [Brisson et Collard, 2009] ont développé la méthodologie KEOPS, qui vise à guider le processus d'intégration des connaissances de l'utilisateur dans le processus de la fouille de données. A cette première étape, KEOPS propose de construire une base de données orientée fouille et ceci en mappant les données originales avec ceux de l'ontologie.
Une nouvelle approche proposée par [Bellandi et al., 2007], et puis développée par la suite dans [Bellandi et al., 2008] utilisait les ontologies lors de l'étape de pré-traitement. Cette approche consiste à fournir certaines contraintes telles que les contraintes d'élagage, qui filtrent les items non intéressants et les contraintes d'abstraction permettent la généralisation des items à l'égard de l'ontologie. Par conséquent, l'ensemble de données est d'abord pré-traité selon les contraintes extraites à partir de l'ontologie, par la suite l'étape de traitement aura lieu.
En outre, [Zeman et al., 2009] ont proposé dans une approche liée aux ontologies consistant à fournir un appui pour la phase de préparation de données. Cette approche propose

d'établir un mapping entre les entités des ontologies et les colonnes de la matrice de données.

3.3.2 Ontologies lors de la phase de traitement

Dans [Escovar et al., 2005], les auteurs proposent l'algorithme SSDM [9], qui considère la sémantique de données pour révéler des règles d'association plus compréhensibles. Pour générer ces règles, l'algorithme SSDM utilise les concepts de la logique floue pour définir la similarité sémantique entre les items. Cependant, le formalisme conceptuel soutenu par des ontologies ordinaires peut ne pas être suffisant pour représenter des informations incertaines qui se trouvent dans de nombreuses applications. Pour répondre à ce problème, l'idée de [Escovar et al., 2006] était d'étendre l'algorithme SSDM pour utiliser des ontologies floues incluant des valeurs concernant les degrés de similarité entre les concepts qui sont traités par SSDM. Ainsi, les règles obtenues sont plus riches sémantiquement et reflètent plus la similarité sémantique chez les données.
Le travail de [Miani et al., 2009] propose l'algorithme NAFO [10] pour la fouille de règles d'association généralisées non redondantes. La contribution de ce travail consiste en la généralisation des itemsets non fréquents en plus de la généralisation des règles extraites et aussi le traitement des redondances, tout en considérant des itemsets flous.

3.3.3 Ontologies lors de la phase de post-traitement

Dans [Marinica et Guillet, 2010], les auteurs intègrent de manière explicite les connaissances du décideur afin de filtrer et cibler les règles utiles. Les connaissances de l'utilisateur sont modélisées dans une ontologie liée aux données. Dans la même orientation, [Mansingh et al., 2010] proposent une méthode hybride pour la fouille de règles d'association intéressantes. Cette méthode combine une analyse objective et une analyse subjective pour la fouille de règles d'association intéressantes en utilisant les ontologies.

Dans [Domingues et Rezende, 2011], les auteurs ont proposé l'algorithme GART [11] utilisant la taxonomie pour généraliser les règles d'association et l'analyse des règles généralisées.
Les travaux de [Ferraz et Garcia, 2008, Ferraz et Garcia, 2013] peuvent être classés et dans la phase de pré-traitement et dans la phase de post-traitement. Dans leur premier travail [Ferraz et Garcia, 2008], les auteurs utilisent une ontologie du domaine pour filtrer les règles d'association en augmentant leur sémantique et diminuant leur cardinalité. La comparaison des résultats obtenus dans la phase de pré-traitement et de post-traitement indique clairement la supériorité de la seconde option. Dans [Ferraz et Garcia, 2013], les mêmes auteurs étudient les effets de l'utilisation d'une technique de pré-traitement appelée *Sem Prune*, qui est construite sur une ontologie de domaine. Cette technique est

9. *S*emantically *S*imilar *D*ata *M*iner
10. *N*on-redundant *A*ssociation Rules based on *F*uzzy *O*ntology
11. *G*eneralization of *A*ssociation *R*ules using *T*axonomies

destinée aux deux phases de pré et de post-traitement dans l'extraction de motifs.

Une approche récente est celle de [Radhika et Vidya, 2012], qui proposent un algorithme appelé WARM [12] pour élaguer et filtrer les règles découvertes en se basant sur des poids relationnels ontologiques.
Un dernier travail concernant la post-fouille de règles d'association est celui de [Subashchadrabose et Sivakumar, 2013], dans lequel les auteurs proposent une nouvelle approche pour intégrer les connaissances de l'utilisateur en utilisant les ontologies et les schémas des règles dans l'étape de post-traitement des règles d'association.

3.4 Conclusion

Dans ce chapitre, nous avons mis l'accent en premier lieu sur les taxonomies et les mesures de similarité associées. En second lieu, nous avons évoqué la thématique de l'utilisation des structures hiérarchiques et des connaissances du domaine dans la fouille de règles d'association. En effet, nous avons étudié l'utilisation des taxonomies dans le processus d'extraction de règles d'association, sur laquelle porte notre dernière contribution dans cette thèse. Les études ont montré l'efficacité de ces structures dans les différentes phases du processus de fouille de motifs (à savoir pré-traitement, traitement ou post-traitement). Cette efficacité a porté principalement sur : *(i)* la qualité des règles générées (richesse sémantique) ; et *(ii)* sur la cardinalité des règles extraites (i.e., des règles intéressantes et non redondantes ont été générées).

En termes de conclusion pour la partie état de l'art, nous pouvons résumer que notre étude a porté principalement sur les trois axes suivants, pour lesquels nous décrirons une contribution dans la partie qui suit.

— En premier lieu, nous avons énoncé certaines représentations condensées dans le cadre de la fouille d'itemsets fréquents. Pour cela, nous proposons dans le chapitre 4 de notre contribution une représentation condensée d'itemsets disjonctifs-fréquents minimaux extraits à partir d'itemsets non fréquents.
— En second lieu, nous avons évoqué l'utilité des taxonomies dans le processus de la fouille d'itemsets fréquents et la génération de règles d'association. En fait, les études ont montré que ces structures permettent un pré-traitement sur les données qui garantit la fouille uniquement de règles intéressantes. Dans ce cadre, nous avons considéré une mesure de similarité permettant de trouver les itemsets homogènes (i.e., plus proches sémantiquement) et de construire des règles d'association entre ces itemsets disjonctifs fréquents homogènes.
— En troisième lieu, nous montrons que outre les règles disjonctives extraites dans le chapitre 5, il est possible de générer *seize* formes possibles de règles d'association et ceci en se basant sur différentes formes de calcul introduites dans le chapitre 1. Ces formes considèrent différents types de supports d'itemsets à savoir *conjonctifs*

12. *W*eighted *A*ssociation *R*ules *M*iner

, *disjonctifs* et *négatifs*. Pour mieux illustrer cette diversité liée aux connecteurs logiques, nous allons considérer alors dans le chapitre 6 une approche complète de fouille de règles d'association généralisées à partir d'itemsets extraits dans le chapitre 4 de nos contributions.

Deuxième partie

CONTRIBUTIONS

Chapitre 4

Fouille d'itemsets disjonctifs-fréquents minimaux

4.1 Introduction

Dans la première partie de ce manuscrit, nous avons mis le lecteur dans le contexte de nos travaux de thèse. C'est ainsi que dans cette deuxième partie, nous développons nos contributions dans le cadre de la thèse.

Dans une première contribution, nous proposons un algorithme de type APRIORI, que nous appelons DISAPRIORI, dédié à la fouille des tous les itemsets disjonctifs-fréquents minimaux satisfaisant un seuil de support *minsup*.

En effet, nous extrayons tous les Itemsets Disjonctifs-Fréquents Minimaux, *(IDFM)*, à partir d'une table transactionnelle et nous montrons que cet ensemble constitue une représentation concise approximative dans le sens où étant donné un itemset disjonctif quelconque nous pouvons connaître son état de fréquence, i.e.,dire s'il est fréquent minimal, fréquent ou non fréquent. Cependant, nous ne pouvons déterminer son support disjonctif seulement dans le cas où il est fréquent minimal ou non fréquent.

4.2 Préliminaires

Dans cette section, nous présentons les préliminaires de base pour la fouille d'itemsets disjonctifs-fréquents à partir d'une table de données.
Dans ce qui suit, nous notons par Δ l'ensemble de transactions, $\mathcal{I}$ l'ensemble des items de Δ et nous supposons qu'une transaction est un ensemble d'items (ou itemset).

Dans le cadre du calcul du support disjonctif de X, une transaction t de la base de données Δ supporte X, s'il existe un item i appartenant à X et à la transaction t. Le support disjonctif de X est défini via l'ensemble de transactions qui supportent X, noté par $T_{\vee}(X)$, et défini par :

$$T_{\vee}(X) = \{t \in \Delta \mid (\exists i \in \mathcal{I})(i \in X \wedge i \in t)\}$$

Définition 31. ***Support disjonctif d'un itemset***

Le support disjonctif, noté $supp_{\vee}(X)$, est le ratio du nombre des transactions qui contiennent au moins un item de l'itemset X divisé par le nombre total des transactions.

$$supp_{\vee}(X) = \frac{|\{t \in \Delta | (\exists i \in \mathcal{I})(i \in X \wedge i \in t)\}|}{|\Delta|}$$

Ainsi, étant donné un seuil de support σ, X est dit disjonctif-fréquent, si $supp_{\vee}(X) \geq \sigma$.

Propriété 4. ***Propriété de monotonie du support disjonctif***
Soit $\mathcal{I}$ un ensemble d'items, le $supp_{\vee}$ est une mesure monotone, ce qui est exprimé de manière formelle comme suit :

$$\forall X, Y \subseteq \mathcal{I} : X \subseteq Y \Rightarrow supp_{\vee}(X) \leq supp_{\vee}(Y).$$

Preuve
$X \subseteq Y \Rightarrow \exists\ X' \subseteq Y$ *tel que* $Y = X \cup X'$
$supp_{\vee}(Y) = \frac{T_{\vee}(Y)}{|\Delta|} = \frac{T_{\vee}(X \cup X')}{|\Delta|} = \frac{T_{\vee}(X) \cup T_{\vee}(X')}{|\Delta|} \geq \frac{T_{\vee}(X)}{|\Delta|}$, *par suite* $supp_{\vee}(Y) \geq supp_{\vee}(X)$. □

Exemple 15. *Nous considérons le contexte d'extraction de la figure 4.1 :*

T_1	$a\ b$
T_2	$a\ c\ d\ e$
T_3	$c\ d\ e$
T_4	$d\ e\ f$
T_5	$a\ b\ c\ d\ e$
T_6	$a\ b\ c$

FIGURE 4.1 – Contexte d'extraction.

Ainsi, l'itemset abc a le support disjonctif $supp_{\vee}(abc) = \frac{\{T_1, T_2, T_3, T_5, T_6\}}{6} = \frac{5}{6}$, *et il est dit disjonctif-fréquent pour un minsup* $\leq \frac{5}{6}$. *Par suite, il est dit non disjonctif-fréquent pour un minsup* $> \frac{5}{6}$.

Par ailleurs, l'itemset abc est un sur-itemset de ab et ab est un sous-itemset de abc.

Définition 32. ***Itemset disjonctif-fréquent minimal***
Soit X un itemset et σ un entier positif compris entre 0 *et* 1. *X est dit* disjonctif-fréquent *si* $supp_{\vee}(X) \geq \sigma$.
De plus, I est dit disjonctif-fréquent minimal *si :*

— *(i) X est disjonctif-fréquent, et*
— *(ii) aucun sous-itemset de X n'est disjonctif-fréquent.*

Par conséquent, un itemset disjonctif-fréquent minimal est tel que tous ses sur-ensembles sont fréquents et tous ses sous-ensembles sont non fréquents.

Exemple 16. *Selon la figure 4.1, et pour un minsup égal à $\frac{5}{6}$, les itemsets af, cf sont fréquents minimaux.*

Définition 33. ***Itemset non disjonctif-fréquent maximal***
Un itemset I est dit non disjonctif-fréquent maximal si et seulement si :

- *(i) il est non disjonctif-fréquent, et*
- *(ii) il est maximal i.e., tous ses sur-itemsets sont disjonctifs-fréquents.*

Exemple 17. *Selon la figure 4.1, et pour un minsup égal à $\frac{5}{6}$, l'itemset (ab) est non disjonctif-fréquent maximal. En effet, (AB) est non disjonctif-fréquent car son support $= \frac{4}{6}$, en plus tous ses sur-itemsets sont fréquents (abc, abd, abe, abf), des supports respectivement $\frac{5}{6}$, $\frac{6}{6}$, et $\frac{5}{6}$.*

4.3 Itemsets disjonctifs-fréquents minimaux

En vue d'extraire tout les itemsets disjonctifs-fréquents minimaux à partir d'une table transactionnelle, nous avons conçu un algorithme par niveau appelé DisApriori. Cet algorithme adopte une recherche en largeur d'abord et utilise la structure de données arbre pour calculer efficacement les supports d'itemsets candidats.

Notre algorithme, décrit par par l'algorithme 6, considère en entrée une base transactionnelle et extrait tous les itemsets disjonctifs-fréquents minimaux comme suit.
Dans les lignes 1 et 2, il calcule le support de chaque item et le compare au seuil *minsup* et décide si l'item est fréquent ou non. Par la suite, l'algorithme continue à partir du niveau 2 et il se termine au niveau k (i.e., il extrait et affiche tous les non disjonctifs-fréquents de ce niveau). Deux conditions d'arrêt sont possibles : *(i)* s'il existe des non disjonctifs-fréquents au niveau k mais il est impossible d'établir une auto-jointure entre eux pour passer au niveau suivant $k+1$; *(ii)* il n'y a pas des candidats non disjonctifs-fréquents au niveau suivant k à afficher (i.e., tous les candidats de niveau k sont disjonctifs-fréquents).

Ainsi à chaque niveau (à partir du niveau 2), l'algorithme procède par *(i)* génération des candidats C_k où les sous-ensembles non disjonctifs-fréquents sont étendus par un item à chaque fois (ligne 4) ; *(ii)* l'élagage des candidats, qui consiste à vérifier si un sous-ensemble d'un candidat quelconque est disjonctif-fréquent. Cette fonction est décrite dans les algorithmes 7 et 8. Dans ce cas, nous savons que le candidat est fréquent et il est supprimé de la liste des candidats. Ensuite, le support disjonctif des candidats de C_k est calculé par un balayage de la table Δ (lignes 5-11).
Enfin, les itemsets candidats qui valident le seuil de support disjonctif sont regroupés dans l'ensemble d'itemsets disjonctifs-fréquents minimaux de niveau k, noté par F_k, et les itemsets non disjonctifs-fréquents de niveau k dans $\neg\ F_k$.

Il est à noter qu'afin d'optimiser la génération d'itemsets candidats et par suite le calcul de leurs supports disjonctifs, nous supposons que, dans l'algorithme DISAPRIORI, les itemsets sont ordonnés par ordre lexicographique. Ceci évitera toute génération redondante d'itemsets.

Cet ordre lexicographique est respecté dans la (ligne 3 de l'algorithme 7). Deux itemsets p et q de F_{k-1} vont former un nouveau candidat c de cardinalité k si et seulement s'ils ont en commun $(k-2)$ itemsets, ce qui est exprimé grâce à l'ordre lexicographique dont le pseudo-code est donné par l'algorithme 6.

L'algorithme DISAPRIORI suit la même stratégie qu'APRIORI en tenant compte de quelques points de différence dans chaque étape :

1. L'ensemble de tous les itemsets disjonctifs candidats au niveau k, noté par C_k, est généré à partir de l'ensemble de tous les itemsets disjonctifs non-fréquentes calculés dans le niveau précédent, noté par $\neg F_{k-1}$.
2. L'ensemble C_k est élagué en utilisant les propriétés d'élagage de DISAPRIORI comme suit : si un candidat a un sous-ensemble qui est fréquent alors nous sommes sûrs que ce candidat est fréquent, et par la suite nous devons le soustraire à partir de l'ensemble C_k. En effet, DISAPRIORI est basé sur la propriété de monotonie suivante.
3. Enfin, l'algorithme scanne la table Δ pour déterminer les itemsets fréquents parmi les candidats, en calculant leurs supports disjonctifs.

Algorithme 6 : L'algorithme DisApriori

Données : La table Δ et le seuil du support σ.
Résultat : L'ensemble Freq de touts les itemsets disjonctifs-fréquents minimaux.

```
1  F1= {1-item fréquent};
2  ¬ F1= {1-item non fréquent};
3  pour k=2;¬ F_{k-1} ≠ ∅ ; k++ faire
4  |  C_k= DisApriori-Gen(¬ F_{k-1});
5  |  pour chaque tuple t ∈ Δ faire
6  |  |  pour chaque c ∈ C_k faire
   |  |  |  // c = a_1 ... a_p
7  |  |  |  si ∃ j ∈ {1 ... p} s.t. t.A_j=a_j alors
8  |  |  |  |  supp_∨(c) + +;
9  |  |  |  fin
10 |  |  fin
11 |  fin
12 |  F_k={c ∈ C_k/supp_∨(c)≥ σ };
13 |  ¬ F_k={c ∈ C_k/supp_∨(c)< σ };
14 fin
15 retourner Freq=∪_k F_k et ¬Freq=∪_k ¬F_k ;
```

Algorithme 7 : La fonction DisApriori-Gen

```
1  C_k= ∅;
2  pour chaque itemset p ∈ ¬ F_{k-1} faire
3  |  pour chaque itemset q ∈ ¬ F_{k-1} faire
4  |  |  si (p[1]=q[1] ∧ ... ∧p[k-2]=q[k-2]∧p[k-1]<q[k-1]) alors
5  |  |  |  c=p⋈q;
6  |  |  fin
7  |  |  si a-sous-ensemble-fréquent(c, F_{k-1})=faux alors
8  |  |  |  ajouter c to C_k;
9  |  |  fin
10 |  fin
11 fin
12 retourner C_k ;
```

Algorithme 8 : La fonction a-sous-ensemble-fréquent

```
1  pour chaque sous-ensemble s de niveau (k-1) de c faire
2  |  si s ∈ F_{k-1} alors
3  |  |  retourner vrai;
4  |  fin
5  fin
```

Correction : la preuve de correction de l'algorithme proposé est basée sur celle de l'algorithme Apriori [Agrawal et Srikant, 1994]. Donc, nous devons montrer que $\neg F_k \subseteq C_k$. Dans l'exploration disjonctive, chaque sur-ensemble d'un itemset fréquent doit être fréquent c'est à dire, il vérifie un certain seuil de support minimal.
En fait, l'auto-jointure est équivalente à joindre les itemsets de $\neg F_{k-1}$ pour passer au niveau suivant k et par la suite supprimer les éléments de jointure pour lesquels les $k-1$ éléments sont dans F_{k-1}, parce que nous sommes sûrs qu'ils sont fréquents. La condition $p.item_{k-1} < q.item_{k-1}$ assure tout simplement qu'il n'y aura pas de duplications générées. De plus, l'étape d'élagage a le même principe que celui de l'algorithme Apriori.

Complétude : une fois que, notre algorithme a été démontré correct, nous devons montrer qu'il est complet. Afin qu'il soit complet, notre algorithme doit réussir à extraire tous les itemsets disjonctifs-fréquents.

Pour le prouver, nous procédons par contra-position. Nous supposons un itemset disjonctif-fréquent minimal I qui n'a pas été calculé par notre algorithme DisApriori, et nous montrons que ceci est une contradiction.

Selon la définition 32, tous les sur-itemsets de I sont disjonctifs-fréquents et tous ses sous-itemsets sont non disjonctifs-fréquents. Ainsi, puisque l'algorithme DisApriori arrive à son terme dans le dernier niveau d'itemsets non disjonctifs-fréquents, (i.e., l'algorithme s'arrête quand il n'y aura plus de non disjonctifs-fréquents) alors tous les sur-itemsets d'itemsets non disjonctifs-fréquents de ce dernier niveau sont fréquents. Ainsi,

il n'y aura aucun itemset disjonctif-fréquent minimal qui n'est pas extrait à ce dernier niveau. Donc, l'itemset disjonctif-fréquent minimal I ne peut être que généré par l'algorithme DisApriori.

Par conséquent, et conformément aux définitions 32 et 33, les itemsets non disjonctifs-fréquents du dernier niveau sont des itemsets non disjonctifs-fréquents maximaux et leurs sur-itemsets sont disjonctifs-fréquents plus précisément des disjonctifs-fréquents minimaux dans le cas où on ajoute des itemsets non disjonctifs-fréquents à ces maximaux non disjonctifs-fréquents. En conclusion, tous les itemsets disjonctifs-fréquents minimaux sont générés et vérifiés par l'algorithme DisApriori.

L'ensemble de tous les itemsets disjonctifs-fréquents minimaux générés par l'algorithme DisApriori constitue une représentation concise qui sera notée dans le reste de manuscrit $IDFM$. Étant donné un itemset $I = i_1 \ldots i_n$, avec $n \in [1 \ldots |\mathcal{I}|]$, l'algorithme DisApriori permet de dire si I est disjonctif-fréquent minimal, fréquent ou non disjonctif-fréquent. En effet, si I apparaît dans l'ensemble de F_k, alors il est fréquent minimal, sinon s'il apparaît dans l'ensemble des non fréquents, c'est à dire $\neg F_k$, alors il est non disjonctif-fréquent sinon il est disjonctif-fréquent.

Exemple 18. *En se référant à la base transactionnelle de la figure 4.1, nous considérons ce qui suit :*

- *$I = f$, $I \in \neg F_1$ alors I est non disjonctif-fréquent.*
- *$I = af$, $I \in F_2$ alors I est disjonctif-fréquent minimal.*
- *$I = abf$, I n'apparaît ni dans F_3 ni dans $\neg F_3$ alors il est disjonctif-fréquent.*

Après avoir décidé l'état de fréquence d'un itemset disjonctif quelconque, nous allons essayer de retrouver son support disjonctif exact. Si l'itemset est disjonctif-fréquent minimal ou non disjonctif-fréquent alors l'algorithme DisApriori donne son support disjonctif exact. Dans le cas contraire, il faudrait calculer son support en parcourant la table Δ, sauf des cas spécifiques qui seront présentés dans la section suivante.

4.4 Application de l'extraction des itemsets disjonctifs-fréquents aux requêtes fréquentes

Dans la section précédente, nous nous sommes intéressés à la fouille d'itemsets disjonctifs-fréquents à partir d'une table transactionnelle.

Nous nous intéressons dans cette section aux requêtes de *sélection* disjonctives à partir d'une table relationnelle Δ définie sur un ensemble d'attributs $\mathbf{U}$.

Définition 34. ***Requête de sélection disjonctive***
Une requête de sélection disjonctive est de la forme $(A_1 = a_1 \vee A_2 = a_2 \ldots \vee A_n = a_n)$, avec $n \in [1 \ldots |\mathrm{U}|]$, et $\forall\ i \in [1 \ldots n]$, A_i est un attribut de U et $a_i \in dom(A_i)$. Afin

de simplifier la notation, une requête de sélection disjonctive de la forme $(A_1 = a_1 \vee \ldots \vee A_n = a_n(\Delta))$ *sera notée par* $< a_1 \vee \ldots \vee a_n >$.
Nous notons par $\mathcal{Q}(\Delta)$ *l'ensemble de toutes les requêtes de sélection disjonctives tel que leurs réponses soient non vides.*

Les requêtes disjonctives surviennent fréquemment dans les scénarios de l'interrogation des ensembles de données hétérogènes comme dans le cas des entrepôts de données biologiques, *Uniprot* (notre cadre applicatif dans le chapitre 6), Bio2RDF, etc [Kim et Anyanwu]. Parmi les applications pour lesquelles les requêtes disjonctives fréquentes ont un intérêt fort, nous citons à titre d'exemple les différentes explications des symptômes observés dans un diagnostique erroné, les ambiguïtés dans la compréhension du langage naturel, l'héritage biologique qui consiste à savoir si une cellule hérite de telle ou telle autre caractéristique de ses parents, etc. En effet, dans le contexte du dernier exemple, nous supposons qu'une fécondation a eu lieu entre deux cellules parents. Ainsi, et pour chaque attribut, la cellule résultante va hériter ou bien de la valeur de cet attribut chez la mère ou bien de celle chez le père. Par conséquent, le recours à une requête disjonctive ayant la condition suivante serait d'une grande utilité : couleur-yeux.enfant = (couleur-yeux.père OR couleur-yeux.mère).

De plus, les requêtes disjonctives (par exemple, Caddr="Paris" OU Caddr="Ny") et les requêtes sélectionnant des attributs vérifiant des intervalles de valeurs précis (e.g., Qty ENTRE "1" ET "10") dénoncent certaines imprécisions au niveau des requêtes [Nambiar, 2009], et elles sont d'une grande utilité dans certains domaines.

Exemple 19. *Nous considérons la table relationnelle* Δ *de la figure 4.2 définie sur l'ensemble des attributs* **U** *:* $\{$*Cid, Cname, Caddr, Pid, Ptype, Qty*$\}$*, ayant la signification suivante :*

TID	Cid	Pid	Cname	Caddr	Ptype	Qty
T_1	c_1	p_1	John	Paris	milk	10
T_2	c_1	p_2	John	Paris	beer	10
T_3	c_2	p_1	Mary	Paris	milk	1
T_4	c_2	p_2	Mary	Paris	beer	5
T_5	c_3	p_3	Paul	NY	beer	10
T_6	c_3	p_2	John	Paris	milk	10

FIGURE 4.2 – La table Δ.

- Cid, Cname et Caddr représentent respectivement l'identification du Client, le nom du Client, et l'adresse du Client ; Pid et Ptype représentent respectivement l'identification du Produit et le type de Produit ; Qty représente la quantité i.e., le nombre des produits vendus.

Chaque attribut A_i de **U** possède une liste des valeurs possibles qui constitue son domaine noté par dom(A_i).
Dans la suite, nous introduisons brièvement les notions clés utilisées dans le reste du chapitre.

Exemple 20. *Considérons les requêtes de sélection suivantes :*
$q_1 = \sigma_{Caddr=Paris}(\Delta)$
$q_2 = \sigma_{Ptype=beer}(\Delta)$
$q_3 = \sigma_{Caddr=Paris \vee Ptype=beer}(\Delta)$
$q_4 = \sigma_{Caddr=NY \wedge Qty=5}(\Delta)$.

Selon la table Δ donnée par la figure 4.2, les requêtes q_1, q_2 et q_3 sont dans $\mathcal{Q}(\Delta)$, alors que la requête q_4 n'est pas dans $\mathcal{Q}(\Delta)$.

Définition 35. ***Support d'une requête disjonctive***
La réponse à une requête disjonctive $q =< a_1 \vee \ldots \vee a_n >$ de $\mathcal{Q}(\Delta)$, notée $ans_\Delta(q)$, est égale au nombre des tuples t_i de Δ, tel que $\forall\ i \in [1 \ldots |TID|]\ \exists\ j \in [1 \ldots n]\ t_i.A_j = a_j$. Pour toute requête q dans $\mathcal{Q}(\Delta)$, le support de q dans Δ, noté par $supp_\Delta(q)$, est défini par le ratio de la cardinalité de sa réponse par le nombre total des tuples, i.e., $supp_\Delta(q) = \frac{|ans_\Delta(q)|}{|TID|}$.

Étant donné un seuil de support *minsup*, une requête est dite fréquente dans Δ, si $supp_\Delta(q)$ est supérieur ou égal au seuil de support *minsup*.

Remarque

Le support d'une requête disjonctive se calcule de la même façon que le support disjonctif d'un itemset vu dans la section précédente. Il s'agit de calculer les transactions ou les tuples qui satisfont au moins un item de l'itemset ou bien une sélection de la requête.

Nous notons pour chaque table Δ et pour chaque requête $q =< a_1 \vee \ldots \vee a_n >$ dans $\mathcal{Q}(\Delta)$, $0 \leq supp(q) \leq 1$.
Dans le cas conjonctif, nous pouvons extraire pour un attribut quelconque deux valeurs différentes dans une même requête, mais nous savons bien que la réponse est vide. Par exemple, dans la figure 4.2, la requête ($\sigma_{Caddr=Paris \wedge Caddr=Ny}(\Delta)$) aurait une réponse vide dans la table Δ.
Cependant, dans le cas disjonctif, il est possible d'extraire la requête ($\sigma_{Caddr=Paris \vee Caddr=Ny}(\Delta)$), et son support est égal à la cardinalité de l'ensemble des tuples satisfaisant (Caddr=Paris) ou (Caddr=Ny).

Exemple 21. *Dans la table de l'exemple 20 nous vérifions :*
- $q_1 = (\sigma_{Caddr=Paris}(\Delta)) = \{T_1, T_2, T_3, T_4 \text{ et } T_6\} = 5$, *et* $supp_\Delta(q_1) = \frac{5}{6}$
- $q_2 = (\sigma_{Caddr=Paris \vee Ptype=beer}(\Delta)) = \{T_1, T_2, T_3, T_4, T_5 \text{ et } T_6\} = 6$, *et* $supp_\Delta(q_2) = \frac{6}{6}$
- $q_3 = (\sigma_{Ptype=beer}(\Delta)) = \{T_2, T_4, \text{ et } T_5\} = 3$, *et* $supp_\Delta(q_3) = \frac{3}{6}$

Fixons minsup égal à $\frac{5}{6}$, nous vérifions facilement que les requêtes q_1 et q_2 sont disjonctives-fréquentes alors que la requête q_3 est non disjonctive-fréquente.

Définition 36. ***Sous-requête disjonctive, sur-requête disjonctive***
Soit $q =< a_1 \vee ... \vee a_n >$ et $q_1 =< a_{11} \vee ... \vee a_{1m} >$ deux requêtes disjonctives dans $\mathcal{Q}(\Delta)$ avec $n \leq m$, sachant que les séléctions de q appartiennent à q_1. Dans ce cas, q est dite une sous-requête de q_1, et q_1 est dite une sur-requête de q.

Exemple 22. *Considérons les requêtes suivantes :*
- $q = (\sigma_{Caddr=Paris \vee Cname=John}(\Delta))$
- $q_1 = (\sigma_{Caddr=Paris \vee Cname=John \vee Ptype=Milk}(\Delta))$

Les sélections de q sont incluses dans q_1 ; et donc q est une sous-requête de q_1, et q_1 est une sur-requête de q.

Propriété 5. *Considérons q_1 et q_2 deux requêtes disjonctives, si q_1 est une sous-requête de q_2, alors $supp_\Delta(q_1) \leq supp_\Delta(q_2)$.*

Preuve
Cette propriété se démontre via la propriété de monotonie de la définition 32. □
Remarque

La propriété de monotonie est vérifiée dans les deux cas du calcul du support disjonctif *: le cas d'un itemset (section 4.3) et le cas d'une requête de sélection disjonctive.*

Définition 37. ***Requête disjonctive-fréquente minimale***
Une requête disjonctive $(\sigma a_1 \vee \ldots \vee a_n(\Delta))$ est dite fréquente minimale si et seulement si :
-(i) elle est fréquente, et
-(ii) soit $n = 1$ ou
-(iii) soit toutes ses sous-requêtes sont non disjonctives-fréquentes.

Comme conséquence de la Propriété 5, si q est non disjonctive-fréquente alors toute sous-requête $q^{'}$ de q est aussi non disjonctive-fréquente. En d'autres termes, si $q^{'}$ est une sous-requête de q et elle est fréquente, alors q est fréquente. Ceci explique pourquoi l'ensemble de toutes les requêtes de sélection disjonctives peut être obtenu à partir des requêtes de sélection disjonctives-fréquentes minimales.

Il est important de noter que, par analogie aux itemsets disjonctifs-fréquents minimaux extraits dans la section précédente, et en se basant sur l'ensemble des requêtes de sélection fréquentes minimales et leurs supports, il n'est pas possible de calculer le support d'une requête de sélection disjonctive-fréquente non minimale. Néanmoins, soient $q_1 = \sigma_{D_1}(\Delta)$ et $q_2 = \sigma_{D_2}(\Delta)$ deux requêtes de sélection disjonctives-fréquentes minimales et considérons la requête de sélection disjonctive $q = \sigma_{D_1 \vee D_2}(\Delta)$. Nous allons

chercher la possibilité de déterminer le support de q à partir de celui de q_1 et q_2.

Puisque, la réponse à q est l'union de celles de q_1 et q_2, nous avons $supp(q) = supp(q_1)+supp(q_2)-supp(\sigma_{D_1 \wedge D_2}(\Delta))$. Ainsi, calculer le support de q, nécessite connaître le support de la conjonction $\sigma_{D_1 \wedge D_2}(\Delta)$, qui n'est pas une requête de sélection disjonctive.

Dans la suite, nous montrons que le support de q peut être inféré de ceux de q_1 et q_2, si q_1 et q_2 traitent le même attribut. Soit $q = \sigma_D(\Delta)$, où $D = (A = a_1) \vee \ldots \vee (A = a_n)$, une requête fréquente qu'on cherche à déterminer son support.

Nous supposons que pour tout $k \leq n$, les supports de toutes les requêtes de la forme $\sigma_{D_k}(\Delta)$, où $D_k = (A = a_1) \vee \ldots \vee (A = a_k)$, sont connus et soit $q = \sigma_D(\Delta)$, où $D = (A = a_1) \vee \ldots \vee (A = a_n) \vee (A = a_{n+1})$. Dans ce cas, notons par D_n la condition de sélection $(A = a_1) \vee \ldots \vee (A = a_n)$. Nous savons que si $(A = a_{n+1})$ apparaît dans D_n, alors D est équivalent à D_n et alors $supp(q) = supp(\sigma_{D_n}(\Delta))$. Sinon, nous avons $supp(q) = supp(\sigma_{D_n}(\Delta)) + supp(\sigma_{(A=a_{n+1})}(\Delta)) - supp(\sigma_{D_n \wedge (A=a_{n+1})}(\Delta))$, et nous connaissons le dernier support est égal à 0 puisque, pour n'importe quel $i = 1, \ldots, n$, Δ ne contient pas un tuple t tel que $t.A = a_i$ et $t.A = a_{n+1}$. Par conséquent, nous obtenons que $supp(q) = supp(\sigma_{D_n}(\Delta)) + supp(\sigma_{(A=a_{n+1})}(\Delta))$.

Exemple 23. *Soit $q =< Cname = Mary \vee Cname = Paul >$ une requête disjonctive-fréquente minimale, dont on connait le support qui est égal à 3.Également, à travers l'algorithme* Disapriori *on connait les supports des requêtes $q_1 =< Cname = Mary >$ et $q_2 =< Cname = Paul >$, puisqu'elles sont toutes les deux non disjonctives-fréquentes. Maintenant, nous supposons les deux requêtes $q^{'} =< Cname = Mary \vee Cname = Paul \vee Cname = Paul >$ et $q^{''} =< Cname = Mary \vee Cname = Paul \vee Cname = John >$. Nous avons bien supp(q) = supp($q^{'}$), car la sélection $Cname = Paul$ apparaît déjà dans q. Pour $q^{''}$, la sélection $Cname = John$ porte sur le même attribut, par la suite supp($q^{''}$) = supp(q) + supp($\sigma_{(Cname=John)}(\Delta)$)= $\frac{3}{6} + \frac{3}{6} = \frac{6}{6} = 1$.*

4.5 Étude expérimentale

Notre implémentation a été effectuée en C++ et les tests ont été réalisés sur une machine Intel (R) Core (TM) i3 CPU avec 3 Go de mémoire principale et sur la version Ubuntu *10.10* de Linux.
Dans la suite, nous traitons l'expérimentation de notre approche à travers une série des tests. Notre première expérimentation est basée sur des bases de données benchmark à partir de `http://archive.ics.uci.edu/ml/`.

Ces données sont fréquemment utilisées pour évaluer les algorithmes de fouille de données. Les caractéristiques de ces bases de données sont présentées dans la figure 4.3.

De plus, nous avons réalisé une étape de pré-traitement sur ces bases de données avant d'en extraire les itemsets fréquents minimaux. Ce pré traitement consiste à ne

pas permettre à une valeur du domaine d'un attribut quelconque d'apparaître dans le domaine de valeurs d'un autre attribut. Autrement dit, la paire *(attribut, valeur)* doit être unique dans la base de données.

Base	♯ Instances	♯ Attributs
Balloons	16	4
Lenses	24	4
MONK's	432	7
Car Evaluation	1728	6
SPECT Heart	276	22
Nursery	12960	8
Solar Flare	1066	10
Balance Scale	625	4

FIGURE 4.3 – Caractéristiques des bases de données de test.

Notre première expérimentation sur les bases de données benchmark porte sur le temps d'exécution en fonction de la taille T de la base quand la valeur du seuil minimal de support est fixée à 0.5 × T.

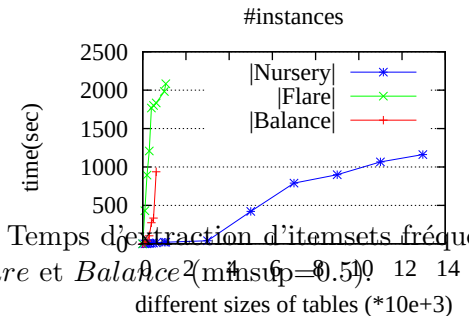


FIGURE 4.4 – Temps d'extraction d'itemsets fréquents en fonction des tailles des bases $Nursery$, $Flare$ et $Balance$ (minsup=0.5).

Pour les bases $Balloons$, $Lenses$ et $MONK's$, le temps d'exécution est très faible et proche de 0s. Pour les deux bases $CarEvaluation$ et $SPECTHeart$, nous avons obtenu un temps d'exécution réduit égal respectivement à 1.78s et 30.26s.

Pour la base $Nursery$, nous notons que le temps d'exécution atteint des valeurs maximales par exemple 1161s (par rapport à la base $Flare$ 2085) bien que cette base contient le nombre maximal des instances qui est égal à 12960 enregistrements, comparé aux deux autres bases ($Flare$ et $Balance$). Ceci est expliqué alors par la nature des données dans la base $Nursery$ qui n'est pas la même dans les deux autres bases.

Si nous comparons le temps d'exécution pour fouiller 1000 instances dans les bases *Nursery* et *Flare*, nous enregistrons respectivement 20.83s et 1983.09s, or ceci peut être expliqué par la différence dans le nombre d' attributs dans les deux bases respectivement 8 et 10, mais aussi le nombre total de couples (attribut = valeur) pour les deux bases. En effet, dans la base *Flare*, différentes valeurs au sein du domaine de valeurs d'un même attribut qui peuvent atteindre 7 valeurs différentes pour un même attribut.

Également, si nous comparons les bases *Flare* et *Balance* pour un même nombre d'instances égal à 500, nous enregistrons respectivement 1800.22s et 334.55s ceci peut être expliqué par la différence au niveau de nombre des attributs qui est égal respectivement à 10 et 8 et du nombre des différentes valeurs par domaine d'attribut.
Sachant que, le choix du nombre d'instances égal à 500 ou à 1000 est fait selon la répartition origine de la base, donc nous extrayons les 1000 (respectivement 500) premières instances pour les tester.

Nous concluons que le temps d'exécution nécessaire pour la fouille de toutes les requêtes disjonctives-fréquentes minimales augmente exponentiellement avec et le nombre d'attributs et le nombre de paires (attribut, valeur).

Notre deuxième expérimentation concerne le temps d'exécution nécessaire à l'égard de différentes valeurs du seuil du support *minsup*, dans le cas où la taille de la table a été fixée à $T = 100$, $T = 200$ et $T = 300$ transactions. Pour faire ceci, nous ne pouvons pas expérimenter les bases benchmark ci-dessus, parce que si nous limitons la taille des bases *Nursery*, *Flare* ou *CarEvaluation* aux 100 ou 200 premières transactions, nous risquons d'introduire un biais sur les valeurs des attributs, ce qui peut affecter nos résultats. Donc, nous menons notre expérimentation sur des ensembles des données qui ont été générés par notre générateur, adapté de générateur de données de IBM `http://www.almaden.ibm.com`.

Nos expérimentations sur ces données s'intéressent précisément à une table composée de 3 attributs et générée à partir d'un schéma en étoile. La taille de la table est un paramètre que nous fixons avant sa génération. La figure 4.5 montre la variation du temps d'exécution à l'égard des différentes valeurs du seuil minimal du support.

Pour des seuils de support minimal inférieurs à 0.2, donc l'ensemble des requêtes non disjonctives-fréquentes à partir desquelles sont générées les candidates du niveau suivant est très réduit, ce qui justifie des valeurs faibles du temps d'exécution. Pour des seuils de support minimal supérieurs à 0.2, 60 et 100, le nombre des requêtes disjonctives-fréquentes diminue et donc l'ensemble des requêtes non disjonctives-fréquentes à partir desquelles sont générées les candidates augmente. Ceci implique une augmentation importante du temps d'exécution global.

Nous notons également dans les résultats présentés dans la figure 4.5, qu'à partir d'un seuil de *misup* égal à $0.5 \times$ T, le temps d'exécution est constant. Ceci peut être ex-

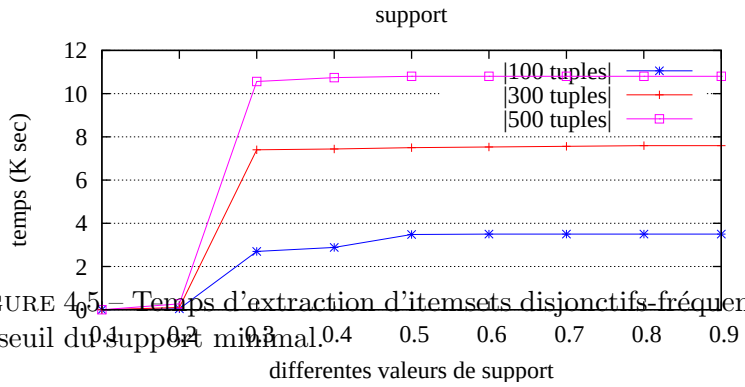


FIGURE 4.5 – Temps d'extraction d'itemsets disjonctifs-fréquents en fonction des valeurs du seuil du support minimal.

pliqué comme suit : le temps estimé pour retrouver les non disjonctifs-fréquents lorsque le *minsup* est égal à 0.5 est très proche de celui quand *minsup* est égal à 0.9. Autrement dit, le nombre des non disjonctifs-fréquents lorsque le *minsup* est égal à 0.5 est proche de celui lorsque *minsup* est égal à 0.9. Ainsi, la majorité des requêtes non disjonctives-fréquentes dont la fréquence est inférieure à 0.9 ×T, sont en réalité avec une fréquence inférieure à 0.5 ×T. Ceci est dû alors à la nature de données où les items sont rares et de faible fréquence.

En effet, nous allons comparer la figure 4.5 à la figure 5 dans l'article de [Jen et al., 2009] qui illustre le temps d'exécution nécessaire pour la fouille des requêtes *conjonctives* fréquentes à l'égard des différentes valeurs du seuil du support minimal, sachant que la taille de la table T a été fixée à 2000 instances.

En examinant les courbes, dans le cas disjonctif : pour des valeurs faibles du seuil du support, le temps d'exécution est faible, et pour des valeurs importantes du seuil du support, le temps d'exécution est important. Cependant, nous observons le contraire dans le cas conjonctif : en effet à des valeurs faibles du seuil du support, correspond un temps d'exécution très important, mais à des valeurs importantes du seuil du support correspond un temps d'exécution réduit.

Ceci peut être expliqué comme suit : Dans le cas conjonctif, la génération des nouveaux candidats et la phase d'élagage sont basées sur l'ensemble des requêtes fréquentes. Cet ensemble est grand quand le seuil du support est faible, et en conséquence le temps d'exécution nécessaire à l'extraction est important.

Cependant, dans le cas disjonctif, la génération des nouveaux candidats et leurs tests sont basés sur l'ensemble des requêtes non disjonctives-fréquentes. Cet ensemble est très important quand le seuil du support est important, c'est pour cela que le temps d'exé-

cution nécessaire pour le fouiller est assez important. A l'opposé mais pour des valeurs faibles du seuil du support, nous avons peu de requêtes non disjonctives-fréquentes et par la suite le temps d'exécution est faible.

4.6 Conclusion

Dans ce chapitre, nous avons considéré le problème de la fouille d'itemsets disjonctifs-fréquents minimaux à partir d'une table transactionnelle. Cette contribution a fait le sujet de notre publication [Hilali-Jaghdam et al., 2011].

Nous avons proposé l'algorithme par niveaux DisApriori capable d'extraire tous ces itemsets et nous avons montré que l'ensemble de ces itemsets constitue une représentation concise approximative de l'ensemble d'itemsets disjonctifs-fréquents. En effet, cet algorithme est capable de déterminer l'état de fréquence de n'importe quel itemset disjonctif (plus précisément d'indiquer s'il est disjonctif-fréquent minimal, non disjonctif-fréquent ou disjonctif-fréquent), mais notre algorithme permet de déterminer son support disjonctif que dans les cas où il est disjonctif-fréquent minimal ou non disjonctif-fréquent. De plus, nous avons étendu cet algorithme au cas du modèle relationnel pour fouiller des requêtes de sélection disjonctives-fréquentes à partir d'une base de données relationnelle Δ.

Chapitre 5

Fouille de règles d'association disjonctives en utilisant les taxonomies

5.1 Introduction

Dans le domaine de la fouille de règles d'association intéressantes, l'intérêt porte généralement sur les items fréquents, à partir desquels les itemsets fréquents sont extraits et les règles d'association sont engendrées.
Cependant, dans certains cas, les données contient des items non fréquents qui peuvent révéler des connaissances utiles que la plupart des algorithmes standards ne parviennent pas à les exploiter.

Par exemple, si les items sont des produits (p_1, p_2, p_3 et p_4), il se peut que aucun des produits p_1 et p_2 ne se vende bien (i.e., aucun d'eux n'est fréquent dans les transactions) mais la vente des produits p_1 ou p_2 soit fréquente (i.e., les transactions qui contiennent p_1 ou p_2 sont fréquentes).
De plus, nous supposons qu'une mesure de similarité entre les produits est donnée et que les produits p_1 et p_2 sont similaires selon cette mesure pour être considérés comme étant de même catégorie.
Par conséquent, l'ensemble $\{ p_1, p_2 \}$ peut être considéré pour la fouille de règles de la forme $\{p_1, p_2\} \rightarrow \{ p_3, p_4 \}$ (supposons que p_3 et p_4 sont deux produits non fréquents similaires aussi et tels que $\{ p_3, p_4 \}$ est fréquent). Ces règles peuvent être pertinentes pour l'utilisateur et elles signifient que la plupart de clients qui achètent p_1 ou p_2 achètent aussi p_3 ou p_4.

L'objectif de ce travail est d'extraire de telles règles d'association dans lesquelles les items non fréquents sont utilisés pour construire des itemsets disjonctifs fréquents et dont les éléments sont suffisamment similaires.
Pour faire ceci, nous considérons les itemsets disjonctifs-fréquents minimaux extraits

dans le chapitre 4, ainsi qu'une mesure d'homogénéité, qui permet de caractériser le degré d'homogénéité des itemsets.
Un itemset I est dit *homogène* si toutes les paires d'items possibles dans I ont un degré de similarité supérieur ou égal à un seuil d'homogénéité donné. L'homogénéité peut alors être vue comme un critère d'intérêt sémantique permettant de sélectionner des itemsets pertinents, comme c'est le cas dans [Marinica et Guillet, 2010].
Pour définir cette notion d'homogénéité, nous supposons une taxonomie de type "est-un" sur les différents items de la base de données, puis nous considérons une mesure de similarité appelée *Overall-Relatedness* permettant de quantifier l'homogénéité d'itemsets disjonctifs-fréquents minimaux dans cette base.
Pour résumer, nous proposons d'extraire des implications intéressantes entre uniquement des itemsets disjonctifs-fréquents minimaux homogènes. Nous nous intéressons également aux règles valides.
En se basant sur les itemsets disjonctifs-fréquents minimaux homogènes, nous cherchons des règles d'association de la forme $\rho : D_1 \rightarrow D_2$ où D_1 et D_2 sont tels que *(i)* le support et la confiance de ρ (les notions de support et de confiance dans notre contexte sont définies dans ce chapitre) sont respectivement supérieurs ou égaux à des seuils donnés de support et de confiance ; et *(ii)* D_1 et D_2 sont des itemsets disjonctifs-fréquents "aussi petits que possible" homogènes et disjoints.

5.2 Homogénéité et itemsets homogènes

Comme nous l'avons déjà mentionné, l'intérêt d'itemsets est mesuré non pas seulement en se basant sur leur support disjonctif, mais aussi sur leur *homogénéité*. L'homogénéité d'itemsets est définie en utilisant une *mesure d'homogénéité*, que nous appelons *Overall-Relatedness*.

5.2.1 Mesure d'homogénéité : Overall-Relatedness

Dans ce qui suit, nous étudions la mesure d'homogénéité entre une paire d'items, puis nous la généralisons pour le cas d'un ensemble d'items (i.e., itemset).

La mesure *Overall-Relatedness* (*OR*) pour une paire d'items mesure le niveau d'homogénéité de deux items qui est équivalent à calculer la distance sémantique entre ces deux items.

Définition 38. *Une taxonomie est une structure hiérarchique qui représente des concepts liés à travers la relation* "est-un" *ou* "is-a".

Il est important à noter que nous avons considéré le cas des taxonomies *arbre* dans notre travail.

Dans ce qui suit, nous rappelons brièvement les définitions de Shekar et Natarajan [Shekar et Natarajan, 2004], qui sont précédemment citées (cf. page 55) et nous considérons la taxonomie des produits alimentaires de la figure 5.1 pour illustrer nos définitions.

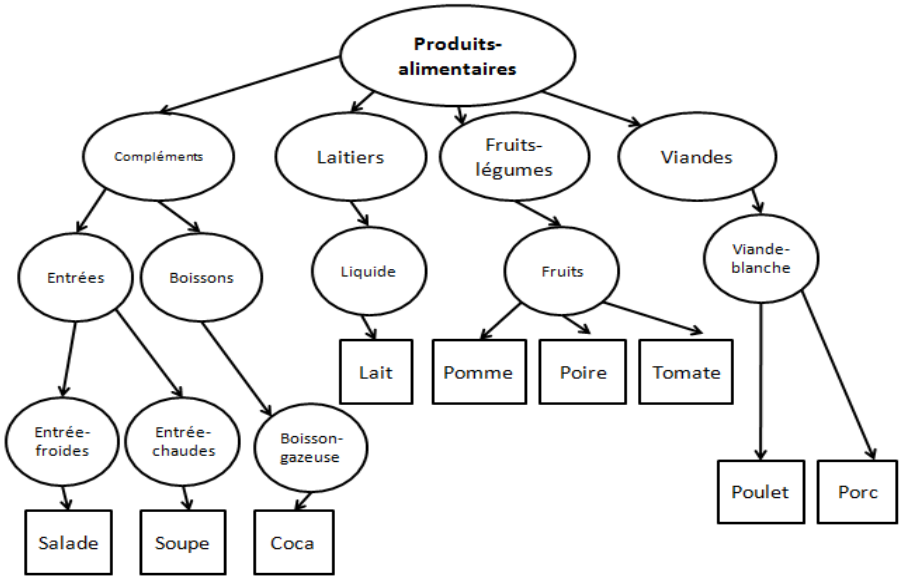


FIGURE 5.1 – Taxonomie des produits alimentaires.

Soit T une taxonomie et A et B deux nœuds distincts de T. Si on note p le chemin unique dans T qui connecte A et B, alors :

H(A,B) : dénote le nœud de plus-haut-niveau du chemin p.

HR(A,B) dénote le niveau maximal des nœuds de p.

NSR(A,B) dénote le nombre de nœuds de p, si l'on exclut A et B.

Exemple 24. *En considérant la taxonomie de la figure 5.1, nous avons :*
H(Pomme, Salade) = Produits-alimentaires,
HR(Pomme, Salade)= 0,
NSR(Pomme, Salade)= 6.

La mesure d'homogénéité *Overall-Relatedness* pour une paire de concepts, notée $OR(c_1, c_2)$, est définie comme suit :

Définition 39. ***Overall-Relatedness***
Soit T une taxonomie et A et B deux nœuds de T.
- *si A=B alors OR(A, B) = 1*
- *sinon,* $OR(A, B) = sim(A, B) = \frac{1+HR(A,B)}{k*NSR(A,B)}$

où HR et NSR ont été définis ci-dessus et où k désigne la profondeur de T.

Remarque

Cette définition de la mesure d'homogénéité diffère de celle de Shekar et Natarajan [Shekar et Natarajan, 2004], puisque les auteurs ont considéré une taxonomie floue (avec des degrés d'appartenance différents) ce qui n'est pas notre cas.

5.2.2 Calcul de la mesure Overall-Relatedness

Dans ce qui suit, nous expliquons en premier lieu la manière dont sont représentés les concepts, puis en deuxième lieu comment calculer la distance sémantique entre deux items c_1 et c_2.
Pour calculer la mesure Overall-Relatedness entre les concepts d'une taxonomie donnée, nous construisons un fichier texte incluant pour chaque concept son chemin. Ce fichier texte sera parmi les paramètres de l'algorithme du calcul de la mesure Overall-Relatedness entre deux concepts quelconques. Le fichier correspondant à la taxonomie de la figure 5.1 est donné par la figure 5.2.

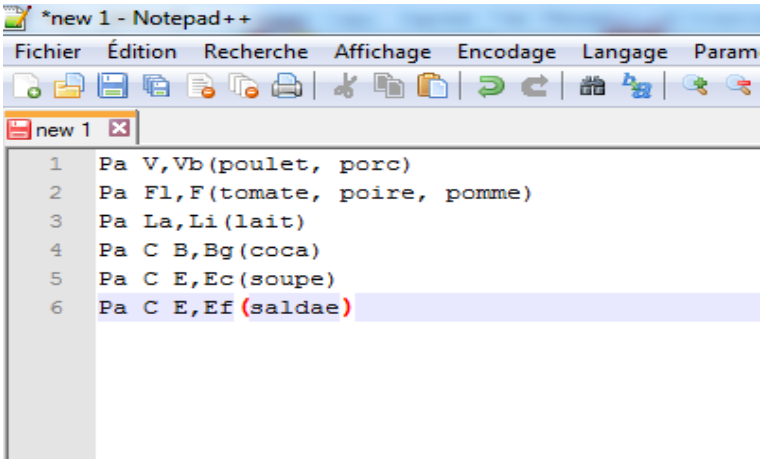


FIGURE 5.2 – Fichier de la taxonomie de la figure 5.1.

Chaque ligne de ce fichier correspond à un concept donné de la taxonomie. Si un ou plusieurs concepts sont frères, alors on les représente dans la même ligne.
Nous expliquons la première ligne tout en sachant qu'il en est de même pour les autres lignes. Les concepts sont toujours à la fin de la ligne et sont toujours entre parenthèses et séparés par une virgule. Le début de la ligne correspond au chemin de ce (ou de ces) concept(s) depuis le père racine. Ainsi, nous avons au niveau de la racine Pa(Produits Alimentaires), puis V(Viandes), puis Vb(Viande Blanche). Pour différentier le père de ce (ou de ces) concept(s) des autres ancêtres, on le précède par une virgule.
Chaque ligne est implémentée à l'aide d'une liste et le niveau de chaque concept est déterminé par sa position dans la liste. De même, il est facile de déterminer le HR de

deux concepts donnés qui correspond à la position de leur prédécesseur commun dans la liste.

Après avoir expliqué comment sont représentés les concepts, nous passons à expliquer comment calculer leur mesure Overall-Relatedness. D'abord, nous expliquons le calcul de la distance entre deux concepts c_1 et c_2, $NSR(c_1, c_2)$ qui est présenté par l'algorithme 9. Ce calcul se réalise en plusieurs étapes :

1. Déterminer le niveau de chaque concept dans la taxonomie ;
2. Initialiser la mesure $NSR(c_1, c_2) = 0$, alors deux cas peuvent se présenter :

   les deux concepts sont du même niveau : nous partons des deux concepts c_1 et c_2, et nous remontons à travers la relation *"is-a"* vers leur ancêtre commun en ajoutant à chaque remontée la valeur 2 à $NSR(c_1, c_2)$. Étant donné, que nous comptons les nœuds et pas les arcs, nous retranchons la valeur 1 de la valeur finale de NSR.

   les deux concepts ne sont pas du même niveau : nous partons du concept avec le niveau le plus élevé (i.e., celui situé plus bas de la taxonomie), nous remontons dans la taxonomie à travers la relation *"is-a"* et nous ajoutons la valeur 1 à de $NSR(c_1, c_2)$ à chaque remontée. Nous nous arrêtons quand nous avons atteint le niveau de l'autre concept, et nous revenons au cas précédent.

Exemple 25. *Nous considérons la taxonomie donnée par la figure 5.1 : les produits de cette taxonomie sont eux même les items de la base de données représentée par la figure 5.3 et qui servira comme un exemple illustratif jusqu'à la fin du chapitre.*
Les nœuds {Fruits-légumes, Laitiers, compléments, Viandes, Entrées, Boissons, Liquide, Fruits, Viande-blanche, Entrée-froides, Entrée-chaudes et Boisson-gazeuse} sont des nœuds fictifs (i.e., n'existaient pas réellement dans la base de données).
De même, considérons l'ensemble $\mathcal{I}$ = {salade, tomate, poulet, porc, pomme, soupe, poire, lait, coca}, et nous supposons l'ensemble de transactions Δ présenté dans la table de la figure 5.3. Pour simplifier, pour chaque $j = 1, \ldots, 8$, la transaction avec tId égal à j est dénotée par t_j. Par exemple t_1 fait référence à la première transaction dans Δ, qui est (1, { salade, tomate, poulet }).

tId	*I*
T_1	salade, tomate, poulet
T_2	salade, tomate, porc
T_3	tomate, poulet
T_4	salade, tomate, poulet
T_5	pomme, soupe
T_6	poire
T_7	lait, soupe
T_8	coca

FIGURE 5.3 – Un exemple d'une base transactionnelle.

Algorithme 9 : L'algorithme NSR

Données : Les deux concepts : C_1 et C_2.
Résultat : NSR(C_1, C_2).

```
1  NSR=0;
2  N1 = niveau du concept C1 dans la taxonomie;
3  N2 = niveau du concept C2 dans la taxonomie;
4  si N1=N2 alors
5  |  si même ancêtre commun alors
6  |  |  Return NSR=1;
7  |  sinon
8  |  |  tant que pas ancêtre commun faire
9  |  |  |  remonter vers ancêtre commun;
10 |  |  |  NSR=NSR+2;
11 |  |  fin
12 |  fin
13 |  Return NSR-1;
14 sinon
15 |  si N1 > N2 alors
16 |  |  tant que N1 ≠ N2 faire
17 |  |  |  remonter vers N2;
18 |  |  |  NSR=NSR+1;
19 |  |  fin
20 |  sinon
21 |  |  tant que N2 ≠ N1 faire
22 |  |  |  remonter vers N1;
23 |  |  |  NSR=NSR+1;
24 |  |  fin
25 |  fin
26 fin
27 retourner NSR(C1, C2)
```

Exemple 26. *En se référant à la table transactionnelle de la figure 5.3, nous allons calculer la mesure d'homogénéité pour un ensemble de paires d'items :*

- $OR(tomate, poire) = \frac{1+2}{4*1} = 0.75$,
- $OR(tomate, pomme) = \frac{1+2}{4*1} = 0.75$,
- $OR(tomate, lait) = \frac{1+0}{4*5} = 0.05$,
- $OR(poire, pomme) = \frac{1+2}{4*1} = 0.75$,
- $OR(poire, lait) = \frac{1+0}{4*5} = 0.05$ *et*
- $OR(pomme, lait) = \frac{1+0}{4*5} = 0.05$.

Alors nous expliquons comment sont obtenues les valeurs dans le premier cas, soit $OR(tomate, poire)$. *Nous avons* $HR(tomate, poire)$ *est égal au niveau de l'ancêtre commun de tomate et poire qui est* $Fruits$ *et de niveau égal à* 2, *si on considère que le niveau de la racine* $Produits - alimentaires$ *est égal à* 0. K *est le profondeur de la*

taxonomie et il est égal à 4. $NSR(tomate, poire)$ *est égal à* 1. *En fait, tomate et poire sont du même niveau ont le même ancêtre commun.*

Nous utilisons un seuil d'homogénéité dit *min-interest* prédéfini par l'utilisateur pour décider si un itemset est homogène ou hétérogène selon la valeur de sa mesure *Overall-Relatedness.*

Exemple 27. *Pour un* $min - interest = 0.5$, *seuls les paires d'items : (tomate, poire), (tomate, pomme) et (poire, pomme) sont homogènes.*

Nous définissons maintenant le degré d'homogénéité d'un itemset quelconque.

Définition 40. ***Itemset homogène***
Étant donné une taxonomie T *définie sur l'ensemble des items* $\mathcal{I}$ *et un seuil d'homogénéité* h, *un itemset* I *est dit homogène par rapport à* h, *si* $min_{i,i' \in I}(sim(i, i^{'})) \geq h$.

Exemple 28. *Nous allons considérer dans la suite l'itemset ("tomate, poire, pomme, lait"). La mesure d'homogénéité pur cet itemset est calculée de la manière suivante :* OR*("tomate, poire, pomme, lait")* $= min$ $\{sim(tomate, poire)$, $sim(tomate, pomme)$, $sim(tomate, lait)$, $sim(poire, pomme)$, $sim(poire, lait)$, $sim(pomme, lait)\} = min\{$ *0.75, 0.75, 0.05, 0.75, 0.05 , 0.05*$\} = 0.05$.
Soit h = *0.05 le seuil d'homogénéité, alors l'itemset (tomate, poire, pomme, lait) est classé homogène.*

La propriété suivante démontre la monotonie de la mesure d'homogénéité.

Propriété 6. *Pour tous itemsets* X *et* Y, *tels que* $X \subseteq Y$, *nous avons* $OR(X) \geq OR(Y)$.

Preuve
Nous avons OR(X)= $min_{x,x' \in X}(sim(x, x^{'}))$ et OR(Y) = $min_{y,y' \in Y}(sim(y, y^{'}))$. Nous avons $X \subseteq Y$, par conséquent, $min_{x,x' \in X}(sim(x, x^{'})) \geq min_{y,y' \in Y}(sim(y, y^{'}))$. $\square$.

Si on appelle *hétérogène* tout ensemble non homogène, alors l'itemset X est hétérogène, s'il existe x et $x^{'}$ dans X tels que $sim(x, x^{'}) < h$.

Proposition 6. *Si* X *est hétérogène alors, pour tout* Y, $X \cup Y$ *est hétérogène.*

Preuve
X *est hétérogène s'il existe* x *et* $x^{'}$ *dans* X *tels que* $sim(x, x^{'}) < h$. *Nous supposons* $Z = X \cup Y$, *il existe* z *et* $z^{'}$ *dans* Z *tels que* $sim(z, z^{'}) < h$, *ce sont en particulier le* x $x^{'}$ *de* X, *alors* Z *est hétérogène. Par conséquent, si* X *est hétérogène alors, pour tout* Y, $X \cup Y$ *est hétérogène.* $\square$

5.3 Fouille d'Itemsets Disjonctifs-Fréquents Minimaux Homogènes

Dans section précédente, nous avons montré comment classifier les itemsets en deux classes : les itemsets homogènes et les itemsets non homogènes grâce à la mesure *Overall-Relatedness*. Dans cette section, nous nous intéressons en premier lieu à la fouille d'itemsets disjonctifs-fréquents minimaux homogènes *(IDFMH)*. En second lieu, nous proposerons un algorithme en profondeur pour leur fouille et nous prouverons qu'il est correct et complet.

5.3.1 Calcul des Itemsets Disjonctifs-Fréquents Minimaux Homogènes

Les Itemsets Disjonctifs-Fréquents Minimaux Homogènes *(IDFMH)*[13] sont une extension d'itemsets disjonctifs-fréquents minimaux extraits dans le chapitre 4 à l'aide de l'algorithme Disapriori.

Dans cette section, nous discutons un algorithme en profondeur correct et complet pour l'extraction de tous les itemsets disjonctifs-fréquents minimaux homogènes.
L'avantage des algorithmes en profondeur, selon [Pennerath, 2009], est leur capacité de calculer rapidement le support d'un itemset courant I en mémorisant la liste L(I) des transactions qui le contiennent, dans une structure appelée $Tid - set$.

En effet, grâce à un codage vertical de données associant à chaque item "i" la liste L(i) des transactions présentant cet item, il est alors possible de calculer très rapidement le support des itemsets I $\cup$ i résultant de l'ajout d'un item à l'itemset courant, comme étant égal à la cardinalité de L(I) $\cap$ L(i) [Pennerath, 2009].

Pour les algorithmes en profondeur, nous citons [Zaki, 2000], [Han et al., 2007].

L'algorithme Eclat [Zaki, 2000], est fondateur de la seconde grande famille d'algorithmes de recherche de règles d'association en profondeur (étant donnée que la première famille était celle de l'algorithme Apriori et ses dérivés). L'algorithme compile pour chaque item l'ensemble des identifiants des transactions où il est présent (TIDset) : le support des itemsets candidats est alors égal à la cardinalité de l'intersection des TIDsets des items présents.

L'algorithme FP-Growth [Han et al., 2007] est différent des autres algorithmes de recherche des motifs fréquents . La différence est qu'il procède sans génération de candidats et qu'il se sert d'une structure de données compacte appelée FP-tree.

De manière générale, les algorithmes de parcours en profondeur sont connus à être plus efficaces que les algorithmes de parcours en largeur, à condition que la base de

13. Itemset Disjonctifs-Fréquents Minimaux

transactions (ou sa présentation) puisse être stockée en mémoire centrale.

Revenons alors à notre algorithme IDFMH pour l'extraction des *(IDFMH)* ; ce dernier se base sur le même principe que l'algorithme ECLAT. Nous considérons le format vertical de la base de données, où pour chaque itemset nous disposons de son tidset, i.e. de l'ensemble de toutes les transactions contenant cet itemset. Le format vertical a l'avantage de rendre le calcul du support plus simple puisqu'il s'agit de calculer le *OU* disjonctif entre deux items appartenant à deux tidsets différents, étant donné que nous calculons le support disjonctif.
L'algorithme IDFMH dont le pseudo-code est donné par l'algorithme 10, permet l'extraction des *IDFMH*. L'entrée de cet algorithme est la table Δ, le seuil de support σ, ainsi que le seuil d'homogénéité noté h.
L'algorithme commence par balayer la base de données pour calculer les supports de tous les items. Ceux dont le support est inférieur au seuil σ sont stockés dans $nonfreq - hom_1$ (ligne 1) dans l'ordre décroissant de leurs supports.
Ensuite, l'algorithme procède en profondeur pour chaque item de l'ensemble $nonfreq - hom_1$. L'algorithme termine lorsqu'il a fini de traiter tous les items de $nonfreq - hom_1$.

Nous distinguons une différence au niveau du traitement des nouveaux candidats entre ceux de taille $k = 2$ et ceux de taille $k \geq 3$.
En effet, pour les candidats de taille $k = 2$, nous calculons la mesure sim entre les deux items formant le candidat. Cependant, pour les candidats de taille $k \geq 3$, nous utilisons de même sim entre les deux items de chaque paire d'items du candidat pour qualifier son homogénéité.
Ainsi, la première catégorie est traitée par l'algorithme 10 itératif et celle de la deuxième catégorie est traitée par l'algorithme 12 récursif.
Pour chacune des deux catégories, deux étapes sont traitées : *(i)* la génération de candidats (ligne 5) dans l'algorithme 10 et (ligne 3) dans l'algorithme 12 ; et *(ii)* le calcul des supports disjonctifs de candidats (ligne 9) dans l'algorithme 10 et (ligne 9) dans l'algorithme 12.

De même, nous mentionnons que pour chacun des algorithmes 10 et 12, le parcours en profondeur est construit en ordonnant les items formant les itemsets selon un ordre lexicographique, e.g., l'itemset {d, b, a, f} est présenté par la séquence ordonnée d'items {a, b, d, f}. Ce classement d'items par ordre lexicographique permet d'éviter des générations redondantes d'itemsets. Ceci est donné par la (ligne 7) de l'algorithme 10 et dans la (ligne 5) de l'algorithme 12.

Dans les deux algorithmes 10 et 12, la génération de candidats se fait par auto-jointure. Seuls les candidats homogènes sont alors retenus pour en calculer leurs supports. Pour chaque nouveau candidat, le calcul de l'homogénéité précède le calcul du support disjonctif, parce que l'accès à la taxonomie pour calculer l'homogénéité nécessite moins d'espace mémoire que celui à la base de données pour le calcul du support.

Algorithme 10 : L'algorithme IDFMH

Données : La table Δ, le seuil du support σ et le seuil de similarité h.
Résultat : L'ensemble IDFMH(Δ) de tous les IDFMHs.

```
// scanner Δ pour calculer l'ensemble des tous les items non
   fréquents
nonfreq − hom1 = { i ∈ I | supp∨(i) < σ};
IDFMH(Δ) = ∅;
si nonfreq − hom1 ≠ ∅ alors
    pour chaque i de nonfreq − hom1 faire
        // génération des candidats de taille 2
        cand − homi = ∅;
        non − freq − homi = ∅;
        pour chaque i' ≠ i et i' > i dans nonfreq − hom1 faire
            si sim(i, i') ≥ h alors
                cand − homi= cand − homi ∪ {(i∪ i')};
                // calcul du support disjonctif
                Procedure Calcul support(i, i');
                si supp∨(i, i') >= σ alors
                    IDFMH(Δ)= IDFMH(Δ) ∪ {(i, i')}
                sinon
                    si test de non-fréquence = vrai alors
                        non − freq − homi = non − freq − homi ∪ {(i, i')};
                    fin
                fin
            fin
        fin
        Procédure traitement (Δ, σ, h, non − freq − homi);
    fin
fin
retourner IDFMH(Δ) ;
```

Algorithme 11 : Procédure calcul support (D).

Données : La table Δ, D
Résultat : $supp_{\vee}$(D)
// D désigne l'itemset disjonctif dont on cherche à calculer son support
1 $supp_{\vee}$(D)=0;
// scanner Δ pour calculer les $supp_{\vee}$ de tous les D
2 **pour chaque** *(tId, I)* $\in \Delta$ **faire**
3 **si** $D \cap I \neq \emptyset$ **alors**
4 $supp_{\vee}$(D) = $supp_{\vee}$(D) +1
5 **fin**
6 **fin**

Algorithme 12 : Procédure traitement.

Données : La table Δ, le seuil du support σ, le seuil de similarité h et $non-freq-hom_i$.
Résultat : L'ensemble IDFMH(Δ) de tous les IDFMHs.
1 **pour chaque** p *dans* $non-freq-hom_i$ **faire**
// génération des candidats de taille 3 et plus
2 $cand-hom_p = \emptyset$;
3 $non-freq-hom_p = \emptyset$;
4 **pour chaque** p *et* q *dans* $non-freq-hom_i$ *tel que* $q > p$ **faire**
5 **si** *sim(p[k-1], q[k-1])* $\geq h$ **alors**
6 $cand-hom_p = cand-hom_p \cup \{$(p[1]... p[k-2] p[k-1] q[k-1])$\}$;
7 cand=p[1]... p[k-2] p[k-1] q[k-1];
8 Calcul support(*cand*);
9 **si** $supp_{\vee}$*(cand)* >= σ *Et test de minimalité (cand) = vrai* **alors**
10 IDFMH(Δ)= IDFMH(Δ) $\cup \{cand\}$
11 **sinon**
12 **si** *test de non-fréquence (cand) = vrai* **alors**
13 $non-freq-hom_p = non-freq-hom_p \cup \{$cand$\}$;
14 **fin**
15 **fin**
16 **fin**
17 **fin**
18 Procédure traitement(Δ, σ, h , $non-freq-hom_p$);
19 **fin**

— Au niveau $k = 3$, alors pour que cet itemset soit homogène, il faut que toutes ses paires le soient. Ainsi, soit I un itemset de taille $k = 3$, qui est obtenu par auto-jointure de deux itemsets p et q de taille $k-1$ et tels que $p[1] = q[1]$ et $p[2] < p[2]$. Nous avons bien p et q sont homogènes car ce sont les parents de I, reste alors à vérifier l'homogénéité de sous-itemset direct w tel que $w[1] = p[2]$ et $w[2]$

$< q[2]$.

Exemple 29. *Soit abc, on a bien ab et ac sont homogènes (après avoir terminé la génération des candidats homogènes de a), donc il suffit de tester l'homogénéité de bc à l'aide de* $OR(b, c)$.

— Au niveau $k > 3$, alors pour que cet itemset soit homogène, il faut que toutes ses paires le soient. Ainsi, soit I un itemset de taille $k > 3$, ce dernier est obtenu par auto-jointure de deux itemsets homogènes p et q de taille $k-1$ telsque $p[1] = q[1] \wedge \ldots \wedge p[k-2] = q[k-2] \wedge p[k-1] < q[k-1]$. Donc toutes les paires d'items qui composent respectivement p et q sont homogènes. Pour que I soit homogène, il suffit de tester $\text{sim}(p[k-1], q[k-1])$.

Exemple 30. *Soit abcd un itemset qui est obtenu par auto-jointure de deux itemsets homogènes abc et abd. On a bien abc et abd qui sont homogènes, alors ab, ac, bc, ad et cd sont tous homogènes. Pour que abcd soit homogène, alors il suffit de tester l'homogénéité de cd uniquement.*

Une fois le calcul d'homogénéité effectué, les candidats homogènes pour chaque item sont stockés dans $cand - hom_i$ (ligne 9) de l'algorithme 10 ou par itemset dans $cand - hom_p$ (ligne 7) de l'algorithme 12. Ensuite, l'algorithme calcule le support disjonctif de chaque candidat homogène en balayant la base de données.

Si le support est supérieur ou égal au seuil σ, alors deux cas se présentent :

— l'itemset est de taille *deux* : il est alors retourné par l'algorithme 10 comme étant un disjonctif-fréquent minimal homogène (ligne 12).

— l'itemset est de taille *trois ou plus* : l'itemset est qualifié disjonctif-fréquent mais pas minimal. Pour vérifier la minimalité de cet itemset disjonctif-fréquent, nous avons recours au *test de minimalité* de l'algorithme 13. Si ce test retourne vrai, alors l'algorithme retourne ce candidat parmi la liste de *IDFMH* (ligne 10) de l'algorithme 12.

Si le support du candidat est strictement inférieur à σ, l'algorithme vérifie s'il doit continuer le traitement en profondeur de cet itemset ou seulement l'afficher comme étant non disjonctif-fréquent. Pour faire cette tâche, les deux algorithmes 10 et 12 font appel au *test de non-fréquence* de l'algorithme 14. Si ce dernier renvoie vrai, alors l'algorithme stocke le candidat dans $non - freq - hom_i$ (ligne 15) de l'algorithme 10, respectivement dans $non - freq - hom_p$ (ligne 13) de l'algorithme 12 pour le traiter de nouveau en profondeur.

L'algorithme 10 retourne la liste de tous les disjonctifs-fréquents minimaux homogènes de toutes les tailles (ligne 23).

Dans ce qui suit, nous expliquons en détail les deux tests à savoir le *test de minimalité* et le *test de non-fréquence* :

Le test de minimalité : un itemset de taille k est dit disjonctif-fréquent minimal si aucun de ses sous-itemsets directs de taille $k-1$ n'est disjonctif-fréquent (ligne 2) de l'algorithme 13. Pour tout itemset de taille k, nous devons pas tester ses deux

sous-itemsets directs qui lui ont donné naissance par la relation d'auto-jointure, car pour ces deux derniers, on est certain qu'ils ne sont pas disjonctifs-fréquents, mais plutôt les autres sous-itemsets.
Cependant, pour les itemsets de cardinalité $k = 2$ ils sont tous fréquents minimaux, étant donné que tous les items singletons (sous-itemsets directs) sont non fréquents.

Le test de non-fréquence : est appelé pour chaque itemset de taille $k \geq 2$, dont il a été prouvé qu'il est non disjonctif-fréquent. Ce test a pour but de vérifier que le support de cet itemset est strictement supérieur aux supports de tous ses deux sous-itemsets directs. En effet, si le support d'un itemset est égal à celui de l'un de ses sous-itemsets, il est inutile de procéder à son augmentation.

Exemple 31. *Nous allons expliquer l'utilité du test de la non-fréquence ci-dessus. Soient, a, b et c trois items non fréquents de supports respectifs* 3, 2 *et* 2*, ordonnés dans l'ordre décroissant de leurs supports. Quand nous traitons en profondeur le a donc on va générer ab, que l'on suppose lui aussi non fréquent de support égal à* 3 *et que les transactions contenant le b sont incluses dans celles contenant le a. Par conséquent,* $supp_{\vee}ab = supp(a) = 3$*. Il est inutile de traiter le ab en profondeur pour la raison suivante : car si on va augmenter encore le ab par c, nous obtenons alors abc dont le support est équivalent à celui de ac. Cependant, le ac est à traiter par la suite, comme c'était pour le ab.*

Exemple 32. *Soit l'itemset* salade coca poire *de l'exemple de la table 5.3 et* $minsup = \frac{5}{8}$*, nous avons salade, coca et poire sont tous non fréquents de supports respectifs* $\frac{3}{8}$*,* $\frac{1}{8}$ *et* $\frac{1}{8}$*. Pour l'item salade, nous construisons l'itemset* salade coca *qui est non disjonctif-fréquent de support disjonctif* $\frac{4}{8}$*, le test de non-fréquence de l'algorithme 14 renvoie vrai (car les supports respectifs de salade et coca sont égaux à* $\frac{3}{8}$ *et*$\frac{1}{8}$*). Ainsi, il faut traiter* "salade coca" *en profondeur. Nous obtenons* "salade coca poire" *qui est dit disjonctif-fréquent de support disjonctif égal à* $\frac{5}{8}$*. Pour décider que* "salade coca poire" *est fréquent minimal, nous vérifions si* "coca poire" *est disjonctif-fréquent étant donné que* "salade coca" *et* "salade poire" *sont déjà non disjonctifs-fréquents. Tout calcul fait,* "coca poire" *est non disjonctif-fréquent et par suite* "salade coca poire" *est fréquent minimal.*

Algorithme 13 : Test de minimalité.

Données : $cand_k$, $cand_{k-1}$, σ.
Résultat : Booléen.
// $cand_k$ désigne l'itemset disjonctif-fréquent dont on cherche à vérifier sa minimalité
// $cand_{k-1}$ désigne l'ensemble de tous les sous-itemsets directs de $cand_k$

```
1 pour chaque w de cand_{k-1} faire
2     si (w ≠ p) ET (w ≠ q) alors
          // p et q sont tels que p[1]=q[1] ∧ ... ∧
             p[k-2]=q[k-2]∧p[k-1]<q[k-1]
3         si (supp_∨(w) ≥ σ) alors
4             retourner faux;
5         fin
6     fin
7 fin
```

Algorithme 14 : Test de non-fréquence.

Données : cand
Résultat : Booléen.

```
1  pour chaque itemset p ∈ cand_{k-1} faire
2      pour chaque itemset q ∈ cand_{k-1} faire
3          si (p[1]=q[1]∧ ... ∧p[k-2]=q[k-2]∧p[k-1]<q[k-1]) alors
               // p et q sont les parents directs selon la relation de
                  jointure de cand
4              si (supp_∨(p) = supp_∨(cand)) alors
5                  retourner faux;
6              sinon
7                  si (supp_∨(q) = supp_∨(cand)) alors
8                      retourner faux;
9                  fin
10             fin
11         fin
12     fin
13 fin
14 retourner vrai.
```

5.3.2 Correction et complétude de l'algorithme IDFMH

Dans cette sous-section, nous montrons que l'algorithme IDFMH est correct, i.e., il extrait des itemsets qui sont disjonctifs-fréquents minimaux et homogènes et complet i.e., calcule *tous* les itemsets disjonctifs-fréquents minimaux homogènes.

Correction de l'algorithme IDFMH

L'algorithme IDFMH est un algorithme en profondeur qui génère *tous les itemsets* possibles (i.e., de différentes cardinalités) à partir de l'ensemble des items non fréquents. Pour chaque itemset engendré, l'algorithme vérifie s'il est homogène ou non.

Ainsi, seuls les itemsets homogènes sont pris en considération pour leur traitement en profondeur. Par la suite, l'algorithme IDFMH calcule le support disjonctif de chaque itemset homogène. Dans le cas où l'itemset est prouvé fréquent, l'algorithme passe à tester sa minimalité. En conclusion, l'algorithme IDFMH n'extrait que des itemsets disjonctifs-fréquents minimaux homogènes.

Complétude de l'algorithme IDFMH

Nous montrons que tous les itemsets disjonctifs-fréquents minimaux ont été extraits. Nous supposons qu'il existe un itemset disjonctif-fréquent minimal qui n'a pas été extrait par notre algorithme.
Nous savons que l'algorithme IDFMH arrive à son terme quand il finit de traiter en profondeur tous les items non fréquents un par un, et à ce moment là tous les itemsets disjonctifs-fréquents minimaux ont été extraits.
Nous nous intéressons alors au traitement en profondeur d'un item non fréquent quelconque. Ainsi, l'algorithme peut s'arrêter dans l'un de deux cas suivants.

— En premier cas, si un itemset est prouvé non disjonctif-fréquent et qu'il faut procéder à son traitement en profondeur, mais qu'on ne trouve pas de frères pour faire la jointure pour générer ses fils et poursuivre son traitement.
— En deuxième cas, l'algorithme s'arrête au niveau d'un itemset si tous ses fils sont prouvés disjonctifs-fréquents.

Maintenant, nous montrons que dans aucun de ces deux cas, l'algorithme risque d'omettre la génération d'un itemset disjonctif-fréquent minimal. Dans le premier cas, si un itemset ne trouve plus d'itemsets frères pour faire de jointure et générer des fils. Alors, ceci garantit qu'il n'y aura plus possibilité de création de nouveaux itemsets qu'ils soient disjonctifs-fréquents ou non. Par conséquent, tous les itemsets disjonctifs-fréquents minimaux sont extraits.
Dans le deuxième cas, si un itemset a tous ses fils (sur-ensembles) qui sont fréquents, alors l'algorithme s'arrête et tous les itemsets disjonctifs-fréquents minimaux qui ont pour préfixe cet itemset sont déjà extraits.

Discussion

Notre algorithme procède par trois tests à savoir : le test d'homogénéité, le test de fréquence et le test de minimalité. Cependant, il contient des redondances au niveau du deux derniers tests, i.e., test de fréquence et test de minimalité. Nous détaillons comment se déroulent ces tests dans ce qui suit :

— test d'homogénéité : pour ce test , il n'y a pas de redondance puisque nous construisons une matrice de similarité au fur et à mesure qu'on calcule des OR des

nouvelles paires. Quand il s'agit d'une nouvelle paire à calculer son homogénéité, on vérifie d'abord si elle existe déjà au niveau de la matrice ou non.

— test de fréquence : ce test n'est pas redondant quand il est utilisé pour juger si un itemset est fréquent ou non. En effet, pour un itemset donné, nous calculons une seule fois son support pour décider s'il est fréquent ou non. Cependant, on peut refaire le calcul du support d'un itemset quand il est un sous-itemset d'un itemset prouvé disjonctif-fréquent et qu'on cherche à vérifier sa minimalité.

— test de minimalité : ici nous attestons à une redondance. En effet, le support d'un itemset peut être calculé plus qu'une fois.

Exemple 33. *Soit abc un itemset qui est prouvé disjonctif-fréquent. Pour que abc soit minimal, il faut que aucun de ses sous-itemsets soit fréquent, en particulier bc. Donc, nous devons calculer le support disjonctif de l'itemset bc pour décider la minimalité de abc. De l'autre côté, se selon l'algorithme 11, une fois on a finit de traiter l'item a et on va explorer l'item b, on doit passer certainement par re-calculer le support de bc.*

5.4 Fouille de règles d'association disjonctives

Les itemsets disjonctifs-fréquents minimaux homogènes extraits dans la section précédente sont à la base de la construction de règles d'association disjonctives intéressantes qu'on développe dans ce qui suit :

5.4.1 Formalisme et propriétés de base

Dans ce que suit, nous introduisons les définitions et propriétés utiles pour la caractérisation et la fouille de règles d'association disjonctives.

Itemsets indépendants

Dans ce qui suit, nous présentons la définition d'itemsets indépendants qui est utilisée pour la construction de règles d'association entre deux itemsets disjonctifs-fréquents minimaux homogènes.

Définition 41. ***Itemsets indépendants***
Soient X et Y deux itemsets, $T_\vee(X)$ et $T_\vee(Y)$ les deux ensembles de transactions qui supportent disjonctivement et respectivement X et Y. X et Y sont dits indépendants si $T_\vee(X) \cap T_\vee(Y) = \emptyset$.

Si $X \cap Y \neq \emptyset$, alors $T_\vee(X \cap Y) \neq \emptyset$. Or, $T_\vee(X \cap Y) \subseteq T_\vee(X) \cap T_\vee(Y)$ et donc, si X et Y ont une intersection non vide alors X et Y ne sont pas indépendants.

En d'autres termes, si X et Y sont indépendants alors $X \cap Y = \emptyset$. La réciproque de l'implication précédente est fausse comme le montre l'exemple suivant.

Exemple 34. *Considérons l'exemple de la table 5.3, $X = \{pomme, soupe\}$ et $Y = \{lait\}$. On a $T_\vee(X) = \{t_5, t_7\}$ et $T_\vee(Y) = \{t_7\}$, ce qui montre que $X \cap Y = \emptyset$ n'implique pas que $T_\vee(X) \cap T_\vee(Y) = \emptyset$, puisque nous avons $T_\vee(X) \cap T_\vee(Y) = \{t_7\}$.*

Pour le calcul du support d'une règle d'association R : X → Y ayant comme prémisse et conclusion deux itemsets disjonctifs-fréquents, nous définissons tout d'abord l'ensemble $T(R)$ de transactions supportant la règle R et qui peut être traduit par le fait qu'une transaction t supporte une règle d'association R : X→Y, si t supporte X *et* Y. Il est à mentionner que le support d'une règle disjonctive suit la même idée que le cas classique de règles d'association.
Si R est la règle $X \to Y$, nous définissons $T(R)$ par :

$$T(R) = T_\vee(X) \cap T_\vee(Y).$$

Support et confiance d'une règle disjonctive

Dans ce paragraphe, nous introduisons les définitions du support et de la confiance d'une règle disjonctive.

Définition 42. ***Support d'une règle disjonctive***
Le support d'une règle d'association disjonctive $R : X \to Y$, noté supp-r($X \to Y$) est défini par le ratio de la cardinalité de l'ensemble T($X \to Y$) par le nombre total de transactions dans Δ :

$$supp - r(X \to Y) = \frac{|T(X \to Y)|}{|\Delta|} = \frac{|T_\vee(X) \cap T_\vee(Y)|}{|\Delta|}.$$

Étant donné un seuil de support σ, la règle disjonctive $X \to Y$ est dite disjonctive-fréquente, si supp-r($X \to Y$) $\geq \sigma$.

Remarque

Si X et Y sont indépendants, alors supp-r(X→Y) = supp-r(Y→X) = 0, et donc on ne peut pas avoir une règle intéressante entre X et Y.

Définition 43. ***Confiance d'une règle disjonctive***
La confiance d'une règle d'association disjonctive $R : X \to Y$, notée $d - conf(X \to Y)$ est le ratio du support de la règle par le support de sa prémisse, soit

$$d\text{-}conf(X \to Y) = \frac{supp - r(X \to Y)}{supp_\vee(X)} = \frac{|T_\vee(X) \cap T_\vee(Y)|}{|T_\vee(X)|}.$$

Exemple 35. *Nous supposons la table de transactions de la table 5.3, alors pour $X = \{salade, porc\}$ et $Y = \{tomate\}$, nous avons $supp - r(X \to Y) = \frac{|t_1, t_2, t_4|}{8} = 0{,}375$ et d-conf($X \to Y$) $= \frac{0{,}375}{0{,}5} = 0{,}75$.*

La propriété 7 est une conséquence importante des définitions 41 (page 94) et 42 (page 95).

Propriété 7. *Pour tous itemsets X, X_1 et X_2, si X et X_1 sont indépendants alors*

- *supp-r($X_1 \rightarrow X_2$) = supp-r($X_1 \rightarrow (X_2 \cup X)$).*
- *d-conf($X_1 \rightarrow X_2$) = d-conf($X_1 \rightarrow (X_2 \cup X)$).*

Preuve
X et X_1sont indépendants alors $T_\vee(X_1) \cap T_\vee(X) = \emptyset$.
Nous avons supp-r($X_1 \rightarrow X_2 \cup X$) $= \frac{|T_\vee(X_1) \cap T_\vee(X_2 \cup X)|}{|\Delta|} = \frac{|(T_\vee(X_1) \cap T_\vee(X_2)) \cup (T_\vee(X_1) \cap T_\vee(X))|}{|\Delta|}$, or $T_\vee(X_1) \cap T_\vee(X) = \emptyset$. Donc, $\frac{|T_\vee(X_1) \cap T_\vee(X_2 \cup X)|}{|\Delta|} = \frac{|(T_\vee(X_1) \cap T_\vee(X_2))|}{|\Delta|}$ et supp-r($X_1 \rightarrow X_2 \cup X$) = supp-r($X_1 \rightarrow X_2$) et d-conf($X_1 \rightarrow X_2 \cup X$) = d-conf($X_1 \rightarrow X_2$). □

5.4.2 Propriétés et critères de sélection de règles d'association disjonctives

Dans ce qui suit, nous nous intéressons en premier lieu à étudier les propriétés de règles d'association disjonctives. En second lieu, nous introduisons les critères de leur sélection.

Propriétés de règles d'association disjonctives

Dans ce paragraphe, nous étudions certaines propriétés de règles d'association disjonctives valides et leurs influences dans l'exécution de leur algorithme d'extraction.

Propriété 8. *Pour tous itemsets X_1, X_2 et X_i, nous avons : supp-r($X_1 \rightarrow X_2$) $\leq$ $supp_\vee(X_i)$, pour i = 1, 2.*

Preuve
Selon la définition 42, si la transaction t supporte la règle disjonctive $X_1 \rightarrow X_2$, alors t supporte disjonctivement X_1 et X_2. Ainsi, l'ensemble de transactions qui supporte la règle $X_1 \rightarrow X_2$ est inclus dans celui qui supporte disjonctivement X_1 et X_2. Par la suite, supp-r($X_1 \rightarrow X_2$) $\leq$ $supp_\vee(X_i)$, pour i = 1, 2. □

La propriété 8 implique que le nombre de règles disjonctives fréquentes est susceptible d'être limité, car nous avons en particulier supp-r($X_1 \rightarrow X_2$) $\leq$ $supp_\vee(X_1)$. Or, X_1 est un disjonctif-fréquent minimal, son support dépasse le seuil le moins possible. Il est donc probable que la règle ne soit pas fréquente car son support est plus faible que celui de X_1.
Ainsi, pour avoir plus de règles nous devons étendre la partie gauche et/ou la partie droite de ces règles. Nous verrons par la suite que toutes les extensions ne sont pas pertinentes, et que le nombre des extensions doit être limité tant que possible pour des raisons d'efficacité.

La propriété suivante compare le support d'une règle d'association disjonctive avant et après l'agrandissement de sa partie gauche.

Propriété 9. *supp-r($X_1 \rightarrow X_2$) $\leq$ supp-r(($X_1 \cup X$) $\rightarrow X_2$), pour chaque itemset X.*

Preuve
Nous avons supp-r($X_1 \rightarrow X_2$) $= \frac{|T_\vee(X_1) \cap T_\vee(X_2)|}{|\Delta|}$ et supp-r(($X_1 \cup X$) $\rightarrow X_2 = \frac{|T_\vee(X_1 \cup X) \cap T_\vee(X_2)|}{|\Delta|}$
$= \frac{|T_\vee(X_1) \cap T_\vee(X_2) \cup T_\vee(X) \cap T_\vee(X_2)|}{|\Delta|}$ *= supp-r($X_1 \rightarrow X_2$)+ $\frac{|T_\vee(X) \cap T_\vee(X_2)|}{|\Delta|}$.*
Par suite, supp-r(($X_1 \cup X$) $\rightarrow X_2$) $\geq$ supp-r($X_1 \rightarrow X_2$). $\square$

Remarque

Il importe de mentionner que, nous ne pouvons pas comparer d-Conf($X_1 \rightarrow X_2$) à d-Conf(($X_1 \cup X$) $\rightarrow X_2$), pour chaque itemset X.
En effet, nous avons d-Conf($X_1 \rightarrow X_2$) $= \frac{supp-r(X_1 \rightarrow X_2)}{supp_\vee(X_1)}$ et d-conf(($X_1 \cup X$) $\rightarrow X_2$) $= \frac{supp-r(X_1 \cup X \rightarrow X_2)}{supp_\vee(X_1 \cup X)}$.

En effet, les deux fractions n'ont ni le même dénominateur, ni le même numérateur donc nous ne pouvons pas les comparer.

La propriété 9 et la remarque précédente montrent que l'agrandissement de la partie gauche de la règle n'est pas une solution, parce dans ce cas le support augmente, mais nous ne pouvons rien confirmer pour la confiance.

D'autre part, la sémantique des implications logiques considèrent que l'implication $X \rightarrow Y$ est vraie lorsque l'interprétation de X est un sous-ensemble de l'interprétation de Y.
De plus, une implication $X \rightarrow Y$ peut être assimilée à une règle d'association de confiance égale à 1. Par suite, augmenter la partie gauche d'une implication (i.e., règle d'association) ne conserve pas la sémantique d'une inclusion, et c'est plutôt l'agrandissement de la partie droite qui le garde.

La propriété suivante compare le support d'une règle d'association disjonctive avant et après l'agrandissement de sa partie droite.

Propriété 10. *supp-r($X_1 \rightarrow X_2$) $\leq$ supp-r($X_1 \rightarrow$ ($X_2 \cup X$)), pour chaque itemset X.*

Preuve
Même preuve que la propriété 9. $\square$

La propriété suivante compare la confiance d'une règle d'association disjonctive avant et après l'agrandissement de sa partie droite.

Propriété 11. *d-Conf($X_1 \rightarrow X_2$) $\leq$ d-Conf($X_1 \rightarrow$ ($X_2 \cup X$)), pour chaque itemset X.*

Preuve
Nous avons d-Conf($X_1 \rightarrow X_2$) = $\frac{supp-r(X_1 \rightarrow X_2)}{supp_\vee(X_1)}$ *et d-conf($X_1 \rightarrow (X_2 \cup X)$) =* $\frac{supp-r(X_1 \rightarrow X_2 \cup X)}{supp_\vee(X_1)}$. *Or, les deux fractions ont le même dénominateur, donc la comparaison de deux numérateurs suffit pour comparer les deux fractions. Or, selon la propriété 10, supp-r($X_1 \rightarrow X_2$) $\leq$ supp-r($X_1 \rightarrow (X_2 \cup X)$) et par suite d-Conf($X_1 \rightarrow X_2$) $\leq$ d-Conf ($X_1 \rightarrow (X_2 \cup X)$).* $\square$

Les propriétés 10 et 11 montrent que l'expansion de la partie droite de la règle augmente et le support et la confiance.

Le résultat de l'algorithme 10 est utilisé pour construire et évaluer les règles disjonctives. Cependant, dans notre approche, et contrairement au cas standard, l'évaluation de ces règles nécessite de balayer de nouveau la base de données. Il en est ainsi parce que, le fait de connaître les supports disjonctifs de D_1 et D_2 ne signifie pas que supp-r($D_1 \rightarrow D_2$) peut être calculé sans balayer la base de données.

De plus, selon la propriété 8, le support disjonctif de la règle $D_1 \rightarrow D_2$ est inférieur aux supports disjonctifs de D_1 et D_2. Par conséquent, considérer seulement les règles $D_1 \rightarrow D_2$ où D_1 et D_2 sont des *IDFMH* est susceptible de produire un nombre limité de règles.

D'autre part, les propriétés 8 et 9 affirment que le support disjonctif de la règle $D_1 \rightarrow D_2$ augmente lorsque les deux itemsets D_1 ou D_2 est agrandi. C'est pourquoi nous cherchons des règles d'association de la forme $D_1 \rightarrow D_2$ où D_1 et D_2 sont des itemsets disjonctif-fréquents homogènes qui pourraient ne pas être des $IDFMH$.

Par ailleurs, nous restreignons naturellement ces ensembles pour qu'ils soient les plus petits possible pour que les règles produites soient le plus concis possible.

Nous rappelons que par les propriétés 9 et 10, élargir la partie droite d'une règle d'association augmente le support et la confiance disjonctifs d'une règle d'association. Cependant, élargir la partie gauche d'une règle d'association augmente le support disjonctif de la règle mais pas toujours sa confiance disjonctive.

En fait, élargir la partie gauche des règles n'est pas pertinent dans notre approche. En effet, une règle $D_1 \rightarrow D_2$, dont la confiance est 1 et pour laquelle supp-r($D_1 \rightarrow D_2$) = $supp_\vee(D_1)$, satisfait $T_\vee(D_1) \subseteq T_\vee(D_2)$.

Ainsi, augmenter la confiance d'une règle $D_1 \rightarrow D_2$ telle que sa confiance disjonctive n'est pas égale à 1 tend à faire en sorte que $T_\vee(D_1)$ soit un sous-ensemble de $T_\vee(D_2)$, ce qui ne peut pas être réalisé en élargissant D_1 (puisque élargir D_1 implique que $T_\vee(D_1)$ est aussi élargi).

Nous caractérisons maintenant l'ensemble de règles que nous cherchons à extraire.

Étant donnés les seuils du support, de la confiance et d'homogénéité : σ, γ et h, ces règles sont de la forme $D_1 \rightarrow D_2$ où :

1. D_1 est un $IDFMH$ et D_2 est un itemset disjonctif-fréquent homogène ;
2. D_1 et D_2 sont disjoints ;
3. $supp - r(D_1 \rightarrow D_2) \geq \sigma$ et $d - conf(D_1 \rightarrow D_2) \geq \gamma$;
4. pour chaque règle $D_1 \rightarrow D_2$ satisfaisant les trois items au dessus, et pour chaque $D \subset D_2$, la règle $D_1 \rightarrow D$ ne satisfait pas les trois items ci-dessus.

Nous appelons ces règles *règles d'association disjonctives intéressantes*, et nous fournissons par la suite un algorithme pour leur fouille.

Pour construire les membres droits des règles d'association, D_2, nous allons leur ajouter des items de façon à ce que l'itemset résultant reste toujours homogène.

Par conséquent, nous limitons les items à ajouter à D_2 aux items i de $\mathcal{I}$, tels que pour tout i_2 de D_2, $sim(i, i_2) \geq h$. Pour formaliser cette étape, nous donnons la définition suivante.

Définition 44. ***Ens-h***
Pour tout itemset X, on définit l'ensemble $Ens-h(X)$ par : Ens-h(X) = $\{i \in \mathcal{I} \mid X \cup \{i\}$ est homogène$\}$.

Le calcul de l'ensemble *Ens-h(X)* d'un itemset X est donné par l'algorithme 15.

Propriété 12. *Ens-h(X)* $= \bigcap_{x \in X} \{i \in \mathcal{I} \mid OR(x, i) \geq h\}$

Preuve
Selon la définition 44, Ens-h(X) = $\{i \in \mathcal{I} \mid X \cup \{i\}$ est homogène$\}$ = $\{i \in \mathcal{I} \mid \forall\, x \in X, OR(x, i) \geq h\}$ = $\bigcap_{x \in X} \{i \in \mathcal{I} \mid OR(x, i) \geq h\}$. $\square$

Nous notons que la propriété 6 (page 85) implique que si X n'est pas homogène, alors *Ens-h(X)* est vide. En effet, selon la définition 44, *Ens-h(X)* est l'ensemble des items i tel que $X \cup \{i\}$ soit homogène. Or si X n'est pas homogène alors $\forall\, i$, $X \cup \{i\}$ n'est pas homogène.
Mais la réciproque n'est pas toujours vraie. En effet, on peut trouver *Ens-h(X)* vide, alors que X est homogène.

Exemple 36. *Soit la taxonomie de la figure 5.1, la base de données de la table 5.3, et $h = 0.05$. Nous avons sim(tomate, poire) = 0.75, donc cette paire est homogène.*
Pour Ens-h(tomate), nous ne gardons que les paires dont l'homogénéité est ≥ 0.05 parmi les paires suivantes : sim(tomate, lait)=0.05, sim(tomate, porc)=0.05, sim(tomate, poire)=0.75, sim(tomate, pomme)=0.75, sim(tomate, poulet)=0.05, sim(tomate, salade)=0.04, sim(tomate, coca)=0.04 et sim(tomate, soupe)=0.04.

Par la suite Ens-h(tomate) $= \{tomate, lait, porc, pomme, poire, poulet\}$.

De même pour Ens-h(poire), sim(poire, salade)=0.04, sim(poire, poulet)=0.05, sim(poire, porc)=0.05, sim(poire, pomme)=0.75, sim(poire, tomate)=0.75, sim(poire, lait)=0.05, sim(poire, coca)=0.04 et sim(poire, soupe)=0.04, par la suite Ens-h(poire) $= \{poire, poulet, porc, pomme, tomate, lait\}$.
Par conséquent, Ens-h(tomate, poire) $= \{poulet, porc, pomme, tomate, poire, lait\}$.

Propriété 13. *Ens* $- h(X)$ *contient* X *si* X *est homogène. Cependant,* Ens-h(X) *n'est pas forcément homogène, même si* X *est homogène.*

Preuve
X *est homogène alors* $min_{x,x' \in X}(sim(x, x')) \geq h$ *et Ens-h(X)* $= \{i \in \mathcal{I} \mid X \cup \{i\} \geq h\}$.
Donc Ens-h(X) va contenir en particulier X.
Nous avons Ens-h(X) $= \{i \in \mathcal{I} \mid X \cup \{i\} \geq h\}$ *donc* $\forall i \in$ *Ens-h(X)* $\mid i \notin X$ *et* $\forall x \in X,\ sim(x, i) \geq h$.
Cependant, il $\exists\ \{i, i'\} \mid \{i, i'\} \in$ *Ens-h(X),* $\{i, i'\} \notin X$ *et sim(i, i') est à inconnue.*
Par la suite, Ens-h(X) n'est pas forcément homogène. □

Exemple 37. *Soient a, b, c et d des items de* $\mathcal{I}$ *et soit* $X = ab$ *un itemset homogène, (i.e., sim(a, b)* $\geq h$*). Nous allons chercher Ens-h(ab) parmi la liste des items* $\mathcal{I} = \{a, b, c, d\}$. *Ainsi, selon la propriété 13, Ens-h(ab)=* $\{a, b\}$. *De même, nous supposons que* $\forall\ i \in \{c, d\}$, *nous avons* $AB \cup \{i\} \geq h$, *alors, Ens-h(ab)=* $\{a, b, c, d\}$. *Cependant, pour que abcd soit homogène, il faut que ab, ac, ad, bc, bd, cd soient tous homogènes. Or, nous savons pas si cd est homogène ou non.*

Algorithme 15 : L'algorithme Ens-h

Données : La table Δ, h, X
Résultat : Ens-h(X)

```
1  Ens-h(X)= ∅
2  pour chaque x_i ∈ X faire
3      pour chaque i ∈ Δ faire
4          si OR (i, x_i) ≥ h alors
5              Ens-h(x_i) = Ens-h(x_i) ⋃ {i}
6          fin
7      fin
8  fin
9  Ens-h(X)= Ens-h(X) ⋃ {i | i ∈ Ens-h(x_i) ∀ x_i ∈ X}
10 retourner Ens-h(X)
```

Dans l'algorithme 16, lorsqu'un item i est ajouté à l'ensemble $X_2 \cup E$, où E est un sous-ensemble de *Ens-h*(X_2), il faut que i soit parmi *Ens-h*(X_2), car sinon $X_2 \cup E \cup i$ ne peut pas être homogène. Cependant, selon la propriété 13, il faut tout de même vérifier si $X_2 \cup E \cup \{i\}$ est homogène ou non (ligne 25). De plus, puisque $X_2 \cup E$ est supposé

homogène, ce test ne nécessite que de tester si, pour tout e de E, $sim(i, e) \geq h$. Malheureusement, lorsque i est dans *Ens-h(X_2)*, on ne sait pas si $X_2 \cup E \cup \{i\}$ est homogène. Pour tester cela, on invoque la fonction booléenne *Hm(E, i)*, qui vérifie si $E \cup \{i\}$ est un ensemble homogène, sachant que E est homogène. Ainsi, *Hm (E,i)* calcule les $sim(e, i)$ pour tout e dans E et les compare à h. *Hm (E,i)* renvoie vrai si tous les résultats sont supérieurs ou égaux à h.

Exemple 38. *Soit la taxonomie illustrée par la figure 5.1 et la base de données de la table 5.3, minsup σ = 0.5, minconf γ = 0.5 et $h = 0.05$.*

Les deux itemsets {coca, salade} et {lait, poulet} sont deux itemsets disjonctifs-fréquents minimaux (i.e., coca, lait, poulet et salade sont des items non fréquents).
*De même sim(*lait, poulet*) = 0.05 et sim(*coca, salade*)= 0.1, donc les deux itemsets* coca, salade *et* lait, poulet *sont disjonctifs-fréquents minimaux homogènes.*

Considérons la règle d'association R : coca, salade $\Rightarrow$ lait, poulet*, alors le supp-r(R) = $\frac{2}{8}$ et d-conf(R)= 0.5. Puisque cette règle ne satisfait pas le seuil de minsup, il faut donc augmenter son membre droit pour obtenir une règle valide.*

*Ens-h(*lait, poulet*) = {lait, poulet, tomate, porc, pomme, poire}. Pour augmenter la règle R, on doit ajouter parmi Ens-h(*lait, poulet*) les éléments qui n'existent ni dans la prémisse ni dans la conclusion de la règle R, i.e., {tomate, porc, pomme, poire}. Nous passons à vérifier Hm({*lait, poulet*}, i) pour tout i de Ens-h(*lait, poulet*), or Hm({*lait-poulet*}, i) est vrai pour tout i de Ens-h(*lait-poulet*).*

Ainsi, quatre règles sont possibles : R_1 = coca, salade $\Rightarrow$ lait, poulet, tomate*,* R_2 = coca, salade $\Rightarrow$ lait, poulet, porc*,* R_3 = coca, salade $\Rightarrow$ lait, poulet, pomme *et* R_4 = coca, salade $\Rightarrow$ lait, poulet, poire*.*

Après le calcul des supports et des confiances, nous constatons que R_1 et R_2 ont le même support $\frac{3}{8}$ et la même confiance 0.75*. Cependant, ces règles sont toujours non valides. Pour R_3 et R_4 elles ont la même confiance* 0.5 *et le même support $\frac{2}{8}$ qui est celui de la règle R avant de procéder à l'augmentation. Nous allons donc augmenter uniquement les deux règles R_1 et R_2 à la prochaine étape.*

*Nous avons Ens-h(*lait, poulet, tomate*)= {lait, poulet, tomate, porc, pomme, poire} pour R_1 et Ens-h(*lait, poulet, porc*)= {lait, poulet, porc, tomate, pomme, poire} pour R_2.*
*Pour augmenter R_1 et R_2, on doit ajouter parmi respectivement Ens-h(*lait, poulet, tomate*) et Ens-h(*lait, poulet, porc*) les éléments qui n'existent ni dans la prémisse ni dans la conclusion des règles, i.e., {porc, pomme, poire} pour R_1 et {tomate, pomme, poire} pour R_2.*
*Nous passons à vérifier Hm({*lait, poulet, tomate*}, {i}) pour tout i de Ens-h(*lait, poulet,

tomate*). Tout calcul fait montre que Hm(*{lait, poulet, tomate}*, {i}) est vrai pour tout i de Ens-h(*lait, poulet, tomate*).*
De même pour R_2, nous vérifions que Hm({lait, poulet, porc}*, {i}) est vrai pour tout i de Ens-h(*lait, poulet, porc*).*

*Nous obtenons que tous les éléments de deux ensembles Ens-h(*lait, poulet, tomate*) et Ens-h(*lait, poulet, tomate*) peuvent former des nouvelles règles.*
Ainsi, les règles suivantes à tester (i.e., les trois premières règles sont relatives à la règle R_1, respectivement et les trois dernière à R_2) :

— R_{11} = coca, salade ⇒ lait, poulet, tomate, porc,
— R_{12} = coca, salade ⇒ lait, poulet, tomate, pomme,
— R_{13} = coca, salade ⇒ lait, poulet, tomate, poire
— R_{21} = coca, salade ⇒ lait, poulet, porc, tomate,
— R_{22} = coca, salade ⇒ lait, poulet, porc, pomme,
— R_{23} = coca, salade ⇒ lait, poulet, porc, poire

Nous remarquons que les règles se répètent (R_{11}, R_{12} et R_{13}) et (R_{21}, R_{22} et R_{23}). Alors, nous nous limitons à calculer les supports et les confiances des trois premières règles i.e., R_{11}, R_{12} et R_{13}. Après le calcul des supports et des confiances, toutes ces règles ont le même support $\frac{3}{4}$ et la même confiance 0.75*. De même nous remarquons que le support de toutes ces règles est le même que celui de la règle dont elles dérivent, alors nous arrêtons le processus d'augmentation de ces règles non valides.*

Critères de sélection des règles intéressantes

En termes de formalisation de notre approche présentée dans le paragraphe précédent, nous fournissons les critères de sélection de règles disjonctives intéressantes.

Les règles d'association que nous extrayons seront alors de la forme R : $X_1 \rightarrow X_2 \cup Y_2$ telles que :

— Γ^1 : X_1 et X_2 sont deux itemsets disjonctifs-fréquents minimaux homogènes.
— Γ^2 : Y_2 est un itemset disjonctif tel que $X_2 \cup Y_2$ soit un itemset disjonctif fréquent homogène.
— Γ^3 : $X_1 \cap (X_2 \cup Y_2) = \emptyset$.
— Γ^4 : supp-r(R) $\geq \sigma$ (seuil de support prédéfini par l'utilisateur)
— Γ^5 : d-conf(R) $\geq \gamma$ (seuil de confiance prédéfini par l'utilisateur)
— Γ^6 : X_1 et X_2 ne sont pas indépendants
— Γ^7 : pour tout item i de $Y_2 \setminus X_1$, X_2 et $\{i\}$ ne sont pas indépendants.
— Γ^8 : $X_2 \cup Y_2$ est un ensemble minimal (selon l'inclusion ensembliste) tel que R satisfait les items précédents.

La propriété 7 montre que le critère Γ^7 est une conséquence des autres critères Γ^j, pour $1 \leq j \leq 8$ et $j \neq 7$. En effet, soit $X_1 \rightarrow X_2 \cup Y_2$ une règle qui satisfait critères Γ^j, pour $1 \leq j \leq 8$ et $j \neq 7$ et pas Γ^7. Dans ce cas, soit $\mathcal{Z}$ l'ensemble (non vide) de tous les items de Y_2 tel que X_1 et $\mathcal{Z}$ sont indépendants.

Donc, si on note $Y_2^{'}$ l'ensemble $Y_2 \setminus \mathcal{Z}$, on a d'après la propriété 7 (page 96), supp-r$(X_1 \rightarrow X_2 \cup Y_2) =$ supp-r$(X_1 \rightarrow X_2 \cup Y_2{}')$ et conf$(X_1 \rightarrow X_2 \cup Y_2) =$ conf$(X_1 \rightarrow X_2 \cup Y_2{}')$.

Par suite, la règle $X_1 \rightarrow X_2 \cup Y_2{}'$ satisfait Γ^4 et Γ^5. Comme on peut voir que $X_1 \rightarrow X_2 \cup Y_2{}'$ satisfait aussi Γ^1, Γ^2, Γ^3, Γ^6 et Γ^7, et comme $Y_2{}' \subset Y_2$, on obtient une contradiction car dans ce cas $X_1 \rightarrow X_2 \cup Y_2{}'$ ne satisfait pas Γ^8.

Ce résultat montre que si l'on satisfait les critères Γ^j, pour $1 \leq j \leq 8$ et $j \neq 7$, on n'a pas à tester la satisfaction du critère Γ^7.

5.4.3 Algorithme de fouille de règles disjonctives

Nous supposons que l'algorithme retourne l'ensemble *Result* et reçoit en entrée les paramètres suivants :

— Δ : la base de données.
— F^h : l'ensemble d'itemsets disjonctifs-fréquents minimaux homogènes générés par l'algorithme Disapriori.
— σ : le seuil de support.
— γ : le seuil de confiance.

Pour chaque couple (X_1, X_2), on considère un candidat de la forme (X_1, X_2, E $\cup$ $\{i\}$, s, *S-ancien*, *S-nouveau*) où :

— X_1 et X_2 sont deux itemsets disjonctifs-fréquents minimaux homogènes ;
— i est un item de Ens-h$(X_2) \setminus (X_1 \cup X_2 \cup E)$;
— $s = supp_{\vee}(X_1)$;
— $S - ancien = supp - r(X_1 \rightarrow X_2 \cup E)$;
— $S - nouveau = supp - r(X_1 \rightarrow X_2 \cup E \cup \{i\})$;

Algorithme 16 : L'algorithme de génération de règles d'association

Données : Δ, F^h, σ et γ.
Résultat : Result

```
1  Result= ∅
2  Etape 1
3  C= ∅
4  pour chaque (X1, X2) de F^h × F^h faire
5  |  si (X1 ∩ X2 = ∅) alors
6  |  |  C = C ∪ {(X1, X2, ∅, s, 0, 0)}
7  |  fin
8  fin
9  Calculer tous les supports S-nouveau=supp-r (X1 → X2) des éléments de C.
10 pour chaque c= (X1, X2, ∅, s, 0, S-nouveau) de C faire
11 |  si S-nouveau ≥ σ ET conf(X1 → X2 ≥ γ ) alors
12 |  |  Result = Result ∪ {X1 → X2}
13 |  |  C=C \{c}
14 |  sinon
15 |  |  si S-nouveau = S-ancien alors
16 |  |  |  C= C\{c}
17 |  |  fin
18 |  fin
19 fin
20 C-ancien=C
21 Etape 2
22 si C-ancien ≠ ∅ alors
23 |  pour chaque c = (X1, X2, ∅, s, S-nouveau, 0) dans C-ancien faire
24 |  |  pour chaque i de (Ens-h(X2) \ (X2 ∪ X1)), i >_I max_I(E) faire
25 |  |  |  c^i= (X1, X2, {i} , s , S-nouveau, 0) ;
26 |  |  |  Calculer le support S − nouveau = supp − r(X1 → X2 ∪ {i}) de c^i;
27 |  |  |  si S-nouveau ≥ σ ET conf(X1 → X2 ∪ {i} ≥ γ ) alors
28 |  |  |  |  Result = Result ∪ {X1 → X2 ∪ {i}}
29 |  |  |  |  C-ancien = C-ancien \{c^i}
30 |  |  |  sinon
31 |  |  |  |  si S-ancien = S-nouveau alors
32 |  |  |  |  |  C-ancien= C-ancien \{c^i}
33 |  |  |  |  sinon
34 |  |  |  |  |  Procédure traitement-RDisj (Δ, σ , γ, c^i )
35 |  |  |  |  fin
36 |  |  |  fin
37 |  |  fin
38 |  fin
39 fin
40 Retirer de Result toutes les règles X1 → X2 ∪ Y2 telles que Result contient une
   règle X1 → X2' ∪ Y2' avec X2' ∪ Y2' ⊆ X2 ∪ Y2.
41 retourner Result
```

Algorithme 17 : Procédure traitement-RDisj

Données : Δ, σ , γ et c^i.
Résultat : Result

1 **pour chaque** *j de (Ens-h(X_2) \ ($X_2 \cup E \cup X_1$)), $j > i$, $j >_{\mathcal{I}} max_{\mathcal{I}}(E)$ ET Hm ($E \cup \{j\}$)* **faire**
2 $\quad c^{ij} = (X_1, X_2, \mathrm{E} \cup \{\mathrm{j}\}$, s, S-nouveau, 0);
3 $\quad$ Calculer le support $S - nouveau = supp - r(X_1 \rightarrow X_2 \cup E \cup \{j\})$;
4 $\quad$ **si** *S-nouveau* $\geq \sigma$ *ET conf(*$X_1 \rightarrow X_2 \cup E \cup \{j\} \geq \gamma$ *)* **alors**
5 $\quad\quad$ Result = Result $\cup \{X_1 \rightarrow X_2 \cup \mathrm{E} \cup \{\mathrm{j}\}\}$
6 $\quad\quad$ C-ancien = C-ancien $\backslash\{c^{ij}\}$
7 $\quad$ **sinon**
8 $\quad\quad$ **si** *S-ancien = S-nouveau* **alors**
9 $\quad\quad\quad$ C-ancien= C-ancien $\backslash\{c^{ij}\}$
10 $\quad\quad$ **sinon**
11 $\quad\quad\quad$ Procédure traitement-RDisj (Δ, σ , γ, c^{ij})
12 $\quad\quad$ **fin**
13 $\quad$ **fin**
14 **fin**

5.4.4 Explication de l'algorithme

Étape 1

Nous initialisons l'ensemble des candidats C= $\emptyset$ (ligne 3) de l'algorithme 16, et pour chaque couple d'itemsets disjonctifs-fréquents minimaux homogènes (X_1, X_2) de $F^h \times F^h$ (ligne 4), nous testons si $X_1 \cap X_2 = \emptyset$. Si c'est le cas, alors nous générons la règle-candidate $X_1 \rightarrow X_2$, représentée par (X_1, X_2, $\emptyset$, s, 0, 0). Ce candidat est ajouté à l'ensemble C (ligne 6) : C= C $\cup \{(X_1, X_2, \emptyset, s, 0, 0)\}$.
Donc à la fin de cette étape, nous aurons dans C l'ensemble des candidats créés à partir de tous les couples d'itemsets disjonctifs-fréquents minimaux homogènes.

Afin de calculer le support de tous ces candidats à la fois, nous balayons une seule fois la base de données Δ et nous calculons $S - nouveau = supp - r(X_1 \rightarrow X_2)$ pour tout candidat c qui s'écrit alors c= (X_1, X_2, $\emptyset$, s, 0, S-nouveau) (ligne 10).
Dans la suite, pour chaque candidat c= (X_1, X_2, $\emptyset$, s, 0, S-nouveau) de C (ligne 10), nous testons si le support(*S-nouveau*) et la confiance ($\frac{S-nouveau}{s}$) de la règle-candidate $X_1 \rightarrow X_2$ dépassent les seuils définis (ligne 11) (noter que $s = supp - d(X_1)$ a été déjà calculé dans l'algorithme Disapriori pour les itemsets disjonctifs-fréquents minimaux homogènes et que $S - ancien = 0$).
Lors du test :

— Si la règle est valide, alors nous l'ajoutons à l'ensemble $Result = Result \cup \{X_1 \cup X_2\}$ (ligne 12) et nous retirons le candidat c correspondant de l'ensemble C :

C=C \{c} (ligne 13).

— Si la règle n'est pas valide, alors, afin de s'assurer que la conclusion de la règle est un ensemble minimal, nous testons si X_1 et X_2 sont indépendants. Ce test qui est en théorie, cherche à vérifier que $T_\vee(X_1) \cap T_\vee(X_2) = \emptyset$, est réalisé en testant que $S - nouveau = S - ancien$ (ligne 15), puisque $S - ancien = 0$. Si le test réussit (i.e., les deux itemsets X_1 et X_2 sont indépendants), alors le candidat c est retiré de l'ensemble de candidats C (ligne 16). Nous attribuons ensuite à l'ensemble $C - ancien$ le contenu de l'ensemble C (ligne 20).

À titre d'optimisation, nous comparons le support de $X_1 \rightarrow X_2$ à zéro avant de passer à calculer la confiance de $X_1 \rightarrow X_2$. Ainsi, si le support est nul, alors ce candidat est directement retiré de l'ensemble C et pas la peine de calculer sa confiance.
De même, une fois le support est non nul, nous vérifions en réalité les confiances de deux règles $X_1 \rightarrow X_2$ et $X_2 \rightarrow X_1$ et nous procédons par la suite de la même manière pour chacune d'elles.

Étape 2

Ainsi, nous utilisons comme point de départ l'ensemble de règles-candidates invalides de *l'étape 1*, contenues dans l'ensemble *C-ancien*. Si *C-ancien* est non vide (ligne 22), pour chaque candidat $c = (X_1, X_2, \emptyset, s, S - nouveau, 0)$ de l'ensemble *C-ancien* (ligne 23), nous générons un ou plusieurs nouveaux candidats en grandissant la conclusion de la règle-candidate $X_1 \rightarrow X_2$.

Pour grossir la conclusion $X_2 \cup$ E de la règle candidate invalide (notons que E=$\emptyset$ pour la première fois), nous lui ajoutons un item i appartenant à l'ensemble Ens-h(X_2) \ ($X_1 \cup X_2$) ; et ceci pour maintenir la condition que les règles obtenues sont telles que les parties gauche et droite sont des itemsets disjoints.
De plus, afin de ne pas générer plusieurs fois le même ensemble, nous considérons l'ordre total $>_\mathcal{I}$ défini sur $\mathcal{I}$, et nous ajoutons les items i tels que $i > max_\mathcal{I}$(E) (ligne 24).
Si ces conditions sont satisfaites, le candidat c^i = (X_1, X_2, $\{i\}$, s, S-nouveau, 0) est généré (ligne 25). Il faut noter que lors de la construction du candidat c^i à partir du candidat c, la valeur du paramètre *S-ancien* dans c^i correspond au support de la règle correspondante à c qui a généré c^i. Cette règle est alors $X_1 \rightarrow X_2$ et son support est donc stocké dans le champ *S-nouveau* de c.
Ensuite, nous calculons le support de c^i (ligne 26) : $S - nouveau = Supp - r(X_1 \rightarrow X_2 \cup \{i\})$. Une fois le support de c^i est calculé, nous testons la validité de la règle. Si S-nouveau $\geq \sigma$ et conf ($X_1 \rightarrow X_2 \cup \{i\}$) $\geq \gamma$, alors on ajoute ce candidat à l'ensemble *Result* et on le retire de l'ensemble de candidats (ligne 28). Si la règle n'est pas valide (support et/ou confiance inférieurs aux seuils), nous vérifions si l'itemset X_1 et i sont indépendants (ligne 31). Si c'est le cas alors toute règle de la forme $X_1 \rightarrow X_2 \cup E'$ où E' contient i ne peut convenir car $X_2 \cup E'$ n'est pas minimal.
En effet, X_1 et i sont indépendants lorsqu'il n'existe aucune transaction qui supporte disjonctivement X_1 et i, ce qui signifie qu'il ne peut pas exister de règle intéressante

entre X_1 et i. Par la suite, le support de la règle $X_1 \rightarrow X_2 \cup E'$, où E' contient i, est le même que celui de la même règle avant de lui ajouter l'item i. Cette indépendance se traduit par *S-ancien* = *S-nouveau* (ligne 31).

Ainsi, la règle candidate $X_1 \rightarrow X_2 \cup \{i\}$ n'a plus à être considérée par la suite, car l'ajout de i n'augmente pas son support et le candidat c est enlevé de l'ensemble *C-nouveau* : *C-nouveau* = *C-nouveau* \{c} (ligne 32) de l'algorithme 16.

Après cette étape, la règle-candidate qui n'ont pas été validée à cause du support et/ou de la confiance sera traitée par l'algorithme 17 afin d'augmenter sa conclusion en profondeur. Pour augmenter cette fois, la conclusion $X_2 \cup \{E\}$ de c^i (notons que E=$\{i\}$), nous lui ajoutons un item j appartenant à l'ensemble Ens-h(X_2) \ ($X_1 \cup X_2 \cup$ E) ; et ceci pour maintenir la condition que les règles obtenues sont telles que les parties gauche et droite sont des itemsets disjoints.
De plus, afin de ne pas générer plusieurs fois le même ensemble, nous considérons l'ordre total $>_{\mathcal{I}}$ défini sur $\mathcal{I}$, et nous ajoutons les items j tels que $j > max_{\mathcal{I}}$(E) (ligne 1).

Si ces conditions sont satisfaites, le candidat c^{ij} = (X_1, X_2, E $\cup \{i\}$, s, S-nouveau, 0) est généré (ligne 2). Il faut noter que lors de la construction du candidat c^{ij} à partir du candidat c^i, la valeur du paramètre *S-ancien* dans c^{ij} correspond au support de la règle correspondante à c^i qui a généré c^{ij}. Cette règle est alors $X_1 \rightarrow X_2 \cup \{i\}$ et son support est donc stocké dans le champ *S-nouveau* de c^i.
Ensuite, nous calculons le support de c^{ij} (ligne 3) : $S - nouveau = Supp - r(X_1 \rightarrow X_2 \cup E \cup \{j\})$. Une fois le support de c^{ij} est calculé, nous testons la validité de la règle. Si S-nouveau $\geq \sigma$ et conf ($X_1 \rightarrow X_2 \cup$ E $\cup$ {j}) $\geq \gamma$, alors on ajoute ce candidat à l'ensemble *Result* et on le retire de l'ensemble de candidats (ligne 5 et 6). Si la règle n'est pas valide (support et/ou confiance inférieurs aux seuils), nous vérifions si l'itemset X_1 et j sont indépendants (ligne 8). Si c'est le cas alors toute règle de la forme $X_1 \rightarrow X_2 \cup E'$ où E' contient j ne peut convenir car $X_2 \cup E'$ n'est pas minimal.
En effet, X_1 et j sont indépendants lorsqu'il n'existe aucune transaction qui supporte disjonctivement X_1 et j, ce qui signifie qu'il ne peut pas exister de règle intéressante entre X_1 et j. Par la suite, le support de la règle $X_1 \rightarrow X_2 \cup E'$, où E' contient j, est le même que celui de la même règle avant de lui ajouter l'item j. Cette indépendance se traduit par *S-ancien* = *S-nouveau* (ligne 8).

Ainsi, la règle candidate $X_1 \rightarrow X_2 \cup E \cup \{j\}$ n'a plus à être considérée par la suite, car l'ajout de j n'augmente pas son support et le candidat c^{ij} est enlevé de l'ensemble *C-ancien* : *C-ancien* = *C-ancien* \ $\{c^{ij}\}$ (ligne 9) de l'algorithme 17.

Si c'est pas le cas, alors nous décidons d'augmenter de nouveau la règle correspondante au candidat c^{ij}, par un appel récursif de l'algorithme 17. A la fin, l'algorithme retourne l'ensemble *Result*.

5.4.5 Correction et complétude de l'algorithme

Le but de cette section est de montrer que notre algorithme 16 est correct (i.e., calcule les règles qui satisfont *tous* les critères énoncés précédemment) et complet (i.e., calcule *toutes* les règles satisfaisant ces critères). Ces critères sont notés Γ^1, ..., Γ^8 et leur ensemble est noté Γ. De plus, on note Γ^0 l'ensemble des critères de Γ à l'exception du dernier critère (la minimalité de $X_2 \cup Y_2$)).

Correction de l'algorithme

Nous commençons par montrer que les règles de *Result* retournées par l'algorithme satisfont les critères de Γ.

Nous avons supp-d$(X_1 \cup X_2) \geq$ supp-d(X_i), pour i = 1, 2, et donc, si X_1 et X_2 sont disjonctifs-fréquents minimaux, alors pour tout Y_2, $X_2 \cup Y_2$ est aussi disjonctif fréquent. Comme, en outre, X_1 et X_2 sont supposés homogènes, toutes les règles retournées dans *Result* satisfont Γ. Pour satisfaire Γ^2, il reste à voir si $X_2 \cup Y_2$ est homogène, ce qui est vérifié (ligne 25). Donc, Γ^1 et Γ^2 sont satisfaits.

De plus, $X_1 \cap (X_2 \cup Y_2) = \emptyset$ est testé dans (ligne 5) lorsque $Y_2 = \emptyset$ et dans (ligne 25) lorsque ($Y_2 \neq \emptyset$. Donc Γ^3 est satisfait. De même, Γ^4 et Γ^5 sont satisfaits puisqu'ils ont été testés (ligne 11) (lorsque $Y_2 = \emptyset$) et (ligne 32) lorsque ($Y_2 \neq \emptyset$).

Il est important de remarquer comme conséquence de la proposition 6, que toutes les règles de *Result* sont telles que l'union de leurs parties gauche et droite est hétérogène. Si on part du fait que X et Y sont *indépendants* lorsque $T_\vee$(X) $\cap$ $T_\vee$(Y) $= \emptyset$, en supposant $\sigma > 0$, alors le test de (ligne 11) indique que *S-nouveau* $= supp_\vee(X_1 \cap X_2)$ est strictement positif et donc que X_1 et X_2 ne sont pas indépendants. Donc, Γ^6 est satisfait pour toute règle de *Result*.
La satisfaction de Γ^7 est une conséquence de la propriété 7 car on montre que *Result* satisfait tous les critères autres que Γ^7.
Par conséquent, pour montrer que toutes les règles de *Result* satisfont Γ, il reste à montrer que Γ^8 est satisfait, ceci est vérifié par la (ligne 43) de l'algorithme 16.

Complétude de l'algorithme

Nous montrons que l'algorithme 16 est *complet*. Pour cela, nous montrons que *Result* contient toutes les règles qui satisfont Γ.

Soit $X_1 \rightarrow X_2 \cup Y_2$ une règle qui satisfait Γ et qui n'appartient pas à *Result*.
Il est facile de voir que le cas $Y_2 = \emptyset$ est impossible car le calcul des disjonctifs-fréquents minimaux homogènes est complet et car la (ligne 4) montre que toutes les règles possibles de ce type sont considérés par l'algorithme et la (ligne 12) montre que toutes les règles possibles de ce type qui satisfont Γ sont mises dans *Result*.

Considérons maintenant le cas où $Y_2 \neq \emptyset$. Dans ce cas, on écrit $Y_2 = E \cup \{i\}$ où i est tel que $i >_{\mathcal{I}} max_{\mathcal{I}}(E)$, et on suppose que $X_1 \rightarrow X_2 \cup E \cup \{i\}$ satisfait Γ mais n'est pas dans *Result*. D'après Γ^8, $X_1 \rightarrow X_2 \cup E$ ne satisfait pas Γ^0. De plus, si à une étape donnée de l'algorithme la règle $X_1 \rightarrow X_2 \cup E$ est dans *C-ancien* (ligne 22), alors $X_1 \rightarrow X_2 \cup Y_2$ est dans *Result*. Par conséquent, si on montre que $X_1 \rightarrow X_2 \cup E$ est dans *C-ancien* (ligne 19), alors $X_1 \rightarrow X_2 \cup E$ est dans *Result*.
Par conséquent, si on montre que $X_1 \rightarrow X_2 \cup E$ est dans *C-ancien*, alors on obtient une contradiction qui prouve la complétude de notre algorithme.

Or, comme $X_1 \rightarrow X_2 \cup E \cup \{i\}$ satisfait Γ, il est facile de voir que $X_1 \rightarrow X_2 \cup E$ satisfait Γ^j pour $j = 1, 2, 3, 6, 7$. Par conséquent, $X_1 \rightarrow X_2 \cup E \cup \{i\}$ ne satisfait pas Γ^4 ou Γ^5.
Par la suite, dans le cas où E est vide, les tests des (ligne 11) et (ligne 14) échouent et donc, $X_1 \rightarrow X_2 \cup E$ est dans *C-ancien* par la (ligne 31). Lorsque E n'est pas vide, $X_1 \rightarrow X_2 \cup E$ s'écrit alors $X_1 \rightarrow X_2 \cup E^{'} \cup \{i^{'}\}$ et, pour les mêmes que ci-dessus, $X_1 \rightarrow X_2 \cup E^{'}$ satisfait Γ^j pour $j = 1, 2, 3, 6, 7$ mais ne satisfait pas Γ^4 ou Γ^5. Dans ce cas, les tests des (lignes 32 et 36) échouent, ce qui implique que $X_1 \rightarrow X_2 \cup E$ est dans *C-ancien* par la (ligne 41).

5.5 Étude expérimentale

Notre implémentation a été effectuée en C++ et les tests ont été réalisés sur une machine Intel (R) Core (TM) i3 CPU avec 3 Go de mémoire principale et sur la version Ubuntu *10.10* de Linux.
Dans ces expérimentations, nous utilisons les données *Suisses* à partir de *http ://www.swisspanel.ch*. La base de données Panel Suisse de Ménages (PSM), s'intéresse au changement social dans la population Suisse, notamment la dynamique de l'évolution des ses conditions de vie.
En particulier, nous traitons les données relatives à un questionnaire individuel appelé $Vague12$ et réalisé entre Septembre 2010 et Février 2011. Dans cette base de données, nous distinguons deux catégories : une référencée "H" et contenant une liste de 217 questions qui ont été posées aux "Habitants" et l'autre référencée "P" et contenant une liste de 616 questions qui ont été posées aux "Personnes".
Pour nos expérimentations, nous avons utilisé la deuxième catégorie contenant les questions d'ordre général et qui ont été posées à la totalité des "Personnes", qu'ils soient des habitants ou non.
Pour explorer ces données, nous avons utilisé SAS Universal Viewer `http://support.sas.com/software/products/univiewer/`, qui est une application libre de visualisation et d'impression des jeux de données. La catégorie "P" est composée de 430 variables sur 11330 observations. Ces variables sont de type numérique d'où il fallait procéder à leur discrétisation avant de les traiter par nos algorithmes. Ainsi, chaque variable est discrétisée en un certain nombre d'items, formant notre base de données, qui est l'entrée de nos algorithmes 10 et 16.

Nous avons obtenu, ainsi après la discrétisation, 1666 items répartis sur 450 variables. Nous avons également construit notre taxonomie en nous basant sur les données de la base (PSM) et en utilisant un éditeur d'ontologies semi-automatique **OntoGen2.0** [Fortuna et al., 2006] à partir de `http://ontogen.ijs.si/?page_id=10`. La taxonomie obtenue est construite sur 7 niveaux.
Nos expérimentations se divisent en deux parties à savoir celles pour l'extraction d'itemsets disjonctifs-fréquents minimaux homogènes et celles relatives aux règles d'association disjonctives.

5.5.1 Fouille d'itemsets disjonctifs-fréquents minimaux homogènes

Dans ce qui suit, nous étudions le nombre de *IDFMH* générés et le temps de calcul nécessaire pour leur extraction, ceci pour les deux cas suivants : variation de la valeur de $min - interest$ et fixation de $minsup$ et variation de la valeur de $minsup$ et fixation de $min - interest$.

Variation de la valeur de $min - interest$ et fixation de $minsup$

En premier lieu, nous étudions le nombre de *IDFMH* générés en faisant varier la valeur de $min - interest$ et gardant celle de $minsup$ fixe. Les résultats obtenus sont illustrés par la figure 5.4.

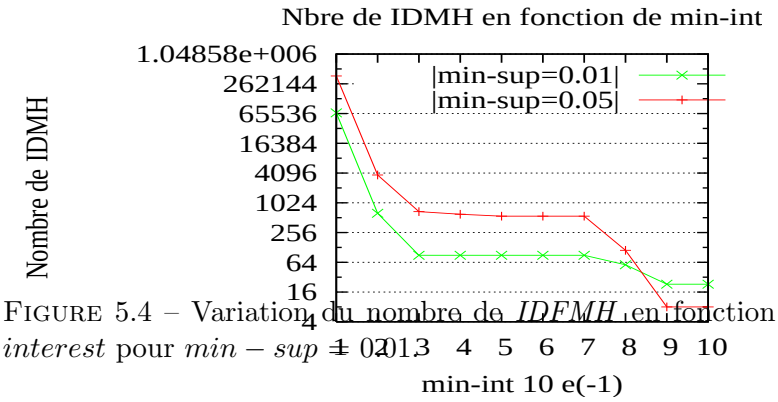


FIGURE 5.4 – Variation du nombre de *IDFMH* en fonction de la variation de $min - interest$ pour $min - sup = 0.01$

Selon cette dernière figure, nous remarquons que plus la valeur de $min - interest$ augmente, plus le nombre de *IDFMH* générés diminue. Ceci est justifié par le fait que plus on augmente la valeur seuil d'homogénéité, plus le nombre d'itemsets homogènes qui la satisfont diminue.

De même, il est à signaler que le nombre de *IDFMH* réalise une chute entre des valeurs faibles de $min-interest$ (i.e., 0.1 et 0.3). Ensuite, il se maintient presque constant pour le reste de valeurs de $min-interest$, pour lesquelles, les itemsets sont peu sensibles. Ceci est interprété par le fait que la majorité d'*IDFMH* extraits satisfaient une valeur faible de $min-interest$ qui est inférieure à 0.3. Par conséquent, les itemsets disjonctifs fréquents minimaux qu'on a pu extraire sont faiblement homogènes.

En deuxième lieu, nous étudions le temps de calcul nécessaire pour générer les *IDFMH* en variant la valeur de $min-interest$ et gardant celle de $minsup$ fixe. Pour ceci, nous avons fait varier la valeur de $min-interest$ de 0.1 à 1, et garder celle de $minsup$ toujours fixe à 0.01. Les résultats obtenus sont illustrés par la figure 5.5.1.

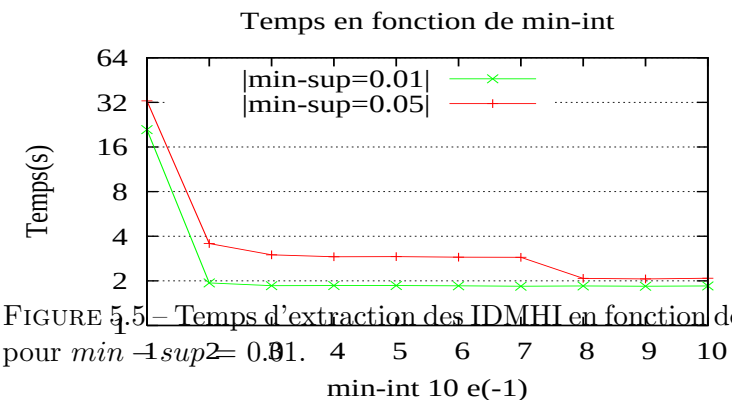


FIGURE 5.5 – Temps d'extraction des IDMHI en fonction de la variation de $min-interest$ pour $minsup = 0.01$.

Selon cette dernière figure, nous remarquons que plus la valeur de $min-interest$ augmente, plus le temps d'extraction nécessaire à générer les *IDFMH* diminue. De même, nous remarquons une chute dans la courbe entre les valeurs 0.1 et 0.2 de $min-interest$, i.e., où il y a un nombre important de *IDFMH*. Ces résultats sont à tendance égale à ceux illustrés par la figure 5.4. En fait, le temps d'extraction des *IDFMH* est toujours proportionnel à leur nombre.

Variation de la valeur de $minsup$ et fixation de $min-interest$

En premier lieu, nous étudions le nombre de *IDFMH* générés en faisant varier la valeur de $minsup$ et gardant celle de $min-interest$ fixe.

En fait, nous avons fait varier la valeur de $minsup$ de 0.01 à 0.1, en gardant celle de $min-interest$ fixe à des valeurs faibles 0.2 (cf. tableau de la figure 5.8) et 0.5 (cf. tableau de la figure 5.9) (car nous avons réalisé une saturation pour le cas de $min-interest = 0.1$)

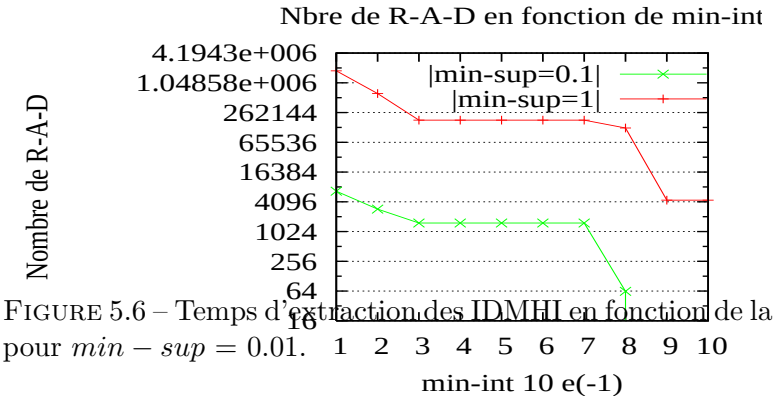


FIGURE 5.6 – Temps d'extraction des IDMHI en fonction de la variation de $min-interest$ pour $min-sup = 0.01$.

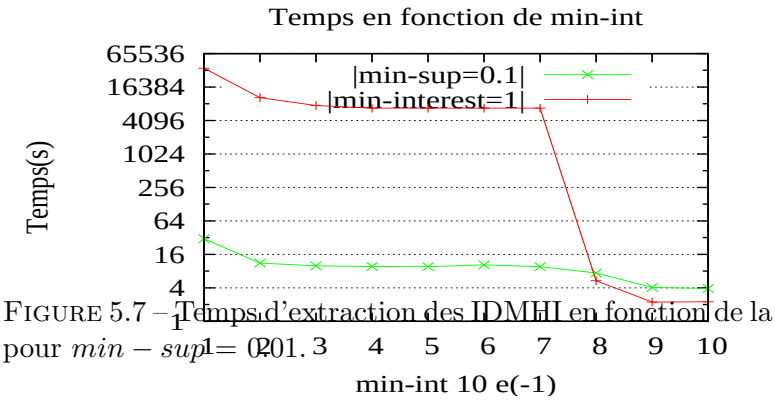
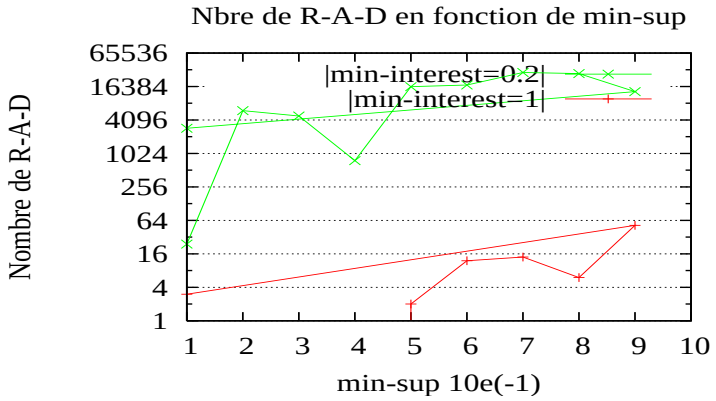


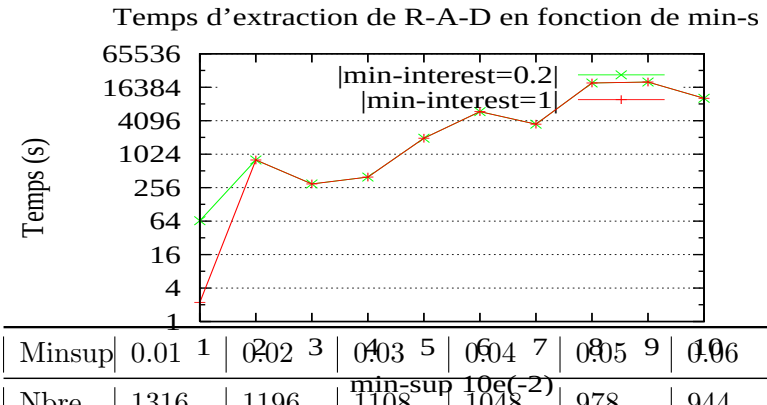
FIGURE 5.7 – Temps d'extraction des IDMHI en fonction de la variation de $min-interest$ pour $min-sup = 0.01$.

et des valeurs importantes 0.9 et 1.

En effet, pour ces deux dernières valeurs, nous avons obtenu exactement les mêmes valeurs à quelques milli-secondes de différence au niveau du temps d'exécution, donc nous nous limitons à donner les résultats seulement pour $min-interest = 1$ (cf. tableau de la figure 5.10).

D'abord, nous mentionnons que nous fournissons le nombre des fréquents i.e., singletons ($Nbrefreq$) dans les tableaux précédents parce que cette dernière change en fonction de la valeur de $minsup$ ce qui n'est pas le cas dans la sous-section précédente





Minsup	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.1
Nbre freq	1316	1196	1108	1048	978	944	921	889	867	849
Nbre IDFMH	628	3136	642	788	883	1080	1506	1872	2438	2317
Temps-exécution(s)	1.916	2.652	3.584	3.729	3.821	3.875	4.007	4.103	4.294	4.410

FIGURE 5.8 – Variation de *minsup* pour Min-interest=0.2.

où elle a été maintenue constante.

Par la suite, nous nous intéressons à la variation du nombre de fréquents ($Nbrefreq$) et du nombre de $IDFMH$ ($NbreIDFMH$) en fonction de la valeur seuil de *minsup*.

— Pour le $Nbrefreq$, nous remarquons qui est le même (i.e., pour les trois valeurs de

Minsup	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.1
Nbre freq	1316	1196	1108	1048	978	944	921	889	867	849
Nbre IDFMH	89	182	263	347	549	700	923	1181	1474	1852
Temps-exécution(s)	1.932	1.940	1.952	1.990	2.059	2.150	2.161	2.221	2.272	2.358

FIGURE 5.9 – Variation de *minsup* pour Min-interest=0.5.

Minsup	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.1
Nbre freq	1316	1196	1108	1048	978	944	921	889	867	849
Nbre IDFMH	23	36	15	10	8	2	0	3	2	7
Temps-exécution(s)	1.845	1.906	1.939	1.975	2.033	2.066	2.094	2.129	2.158	2.170

FIGURE 5.10 – Variation de *minsup* pour Min-interest=1.

$min-interest$) pour une même valeur de *minsup*. Ceci est justifié par le fait que tous les singletons sont supposés homogènes et que la valeur de $min-interest$ n'a aucun effet sur leurs nombres respectifs.
Sinon, plus la valeur de *minsup* augmente, plus le nombre de ce dernier diminue. En effet, ce phénomène est tout à fait attendu, puisque pour des valeurs importantes de *minsup*, peu d'itemsets qui vont les satisfaire.

— Pour le cas du nombre de $IDFMH$, il est tout à fait dépendant de la valeur de $min-interest$. C'est ainsi que, pour une même valeur de *minsup*, plus la valeur $min-interest$ augmente plus le nombre de $IDFMH$ diminue.
Sinon, le nombre de $IDFMH$ réalise une valeur importante pour des valeurs de *minsup* faibles pour laquelle les itemsets sont trop sensibles.

Dans la figure 5.11, nous illustrons le comportement de la variation du nombre de *IDFMH* en fonction des valeurs de *minsup* pour deux valeurs différentes de $min-interest$.

En deuxième lieu, nous étudions le temps de calcul nécessaire pour générer les *IDFMH* en faisant varier la valeur de *minsup* et en gardant celle de $min-interest$ fixe. En effet, nous avons fait varier la valeur de *minsup* de 0.01 à 0.1, en gardant celle de $min-interest$ fixe à une valeur faible (0.2), moyenne (0.5) et importante (1).

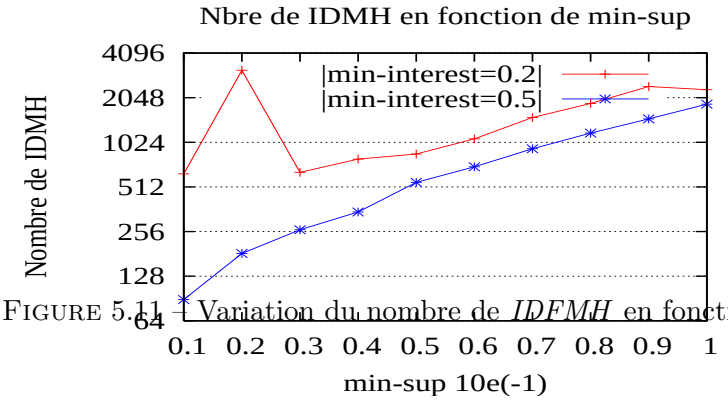


FIGURE 5.11 – Variation du nombre de *IDFMH* en fonction de la variation de *minsup*.

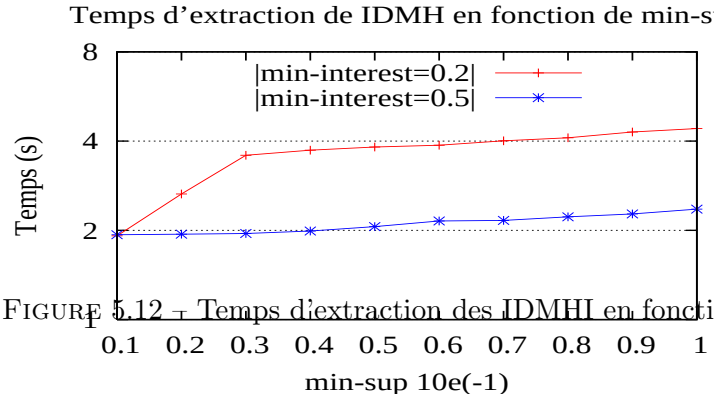


FIGURE 5.12 – Temps d'extraction des IDMHI en fonction de la variation de *minsup*.

Nous remarquons que plus la valeur de *minsup* augmente, plus le temps d'extraction nécessaire à extraire les *IDFMH* augmente. De même, nous remarquons que le temps requis pour $min - interest = 0.5$ et $min - interest = 1$ est presque le même. Ceci est justifié par le fait que $min - interest = 0.5$ est considérée comme une valeur importante de seuil d'homogénéité, pour la quelle peux d'itemsets qui la satisfont.

5.5.2 Fouille des règles d'association disjonctives intéressantes

Dans ce cadre, nous nous intéressons aux expérimentations relatives à l'algorithme 16. Nous extrayons alors des règles d'association intéressantes et valides en fonction de la variation de *minsup* et de $min - interest$.

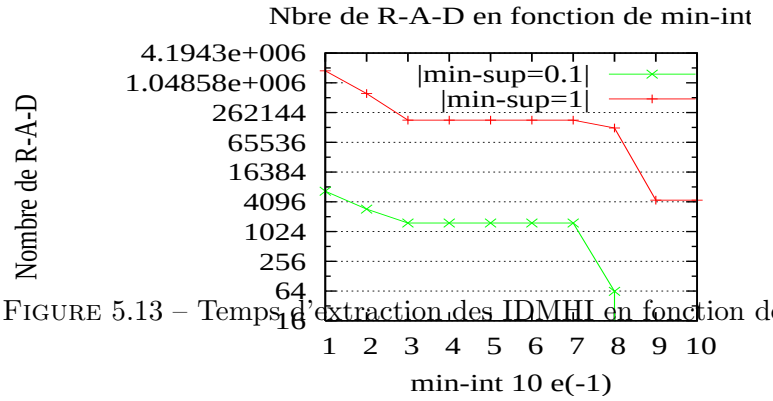


FIGURE 5.13 – Temps d'extraction des IDMHI en fonction de la variation de *minsup*.

Ainsi, nous nous focalisons sur deux types d'analyses, à savoir une analyse quantitative des résultats obtenus et puis une analyse qualitative qui met en œuvre l'intérêt de ces règles extraites.

Analyse quantitative : À ce niveau là, nous étudions le nombre de règles disjonctives valides RA (i.e., qui satisfont $minsup$ et $minconf$) et le temps nécessaire pour leur extraction en fonction des paramètres suivants : $minsup$ et $min-interest$. De même, nous désignons par $nonRA$ les règles disjonctives non valides correspondant aux candidats $\{c^i\}$ dans l'algorithme 16, qu'on va encore augmenter leurs conclusions dans une nouvelle itération.

En premier lieu, nous gardons fixe $minsup$ à 1 et 0.1 et nous varions la valeur de $min-interest$ de 1 à 0.1. La valeur de $minconf$ est maintenue aussi fixe à 0.5. Les résultats obtenus sont illustrés respectivement dans les tableaux de figures 5.14 et 5.15.
Selon ces deux tableaux (i.e., 5.14 et 5.15), le nombre de règles valides augmente en diminuant la valeur de $min-interest$. Ceci est tout à fait logique, car en diminuant la valeur de $min-interest$ et pour une valeur constante de $minsup$, le nombre de $MHDI$ augmente et c'est de même pour les règles d'association. En plus, en diminuant la valeur de $min-interest$, nous obtenons plus de conclusions qui satisfont le critère d'homogénéité. Le temps d'exécution est toujours proportionnel au nombre de règles disjonctives valides et non valids extraites.
En plus, nous remarquons que le nombre de règles valides est maintenu constant pour les valeurs de min-interest entre 0.7 et 0.3 et a triplé à la valeur de min-interest égale à 0.2 jusqu'à arriver à une saturation pour la valeur de 0.1. Ceci est justifié par le fait que nos données sont plus sensibles à des valeurs faibles de $min-interest$, comme nous l'avons déjà mentionné dans les expérimentations de la sous-section précédente relatives aux $IDFMH$.

Nous remarquons aussi que le nombre de règles non valides pour $minsup = 1$ est nul. En fait, toutes les règles qui ont été formées à partir de $IDFMH$ satisfont les seuils de $minsup$, $min - interest$ et $minconf$. De même, nous avons vérifié que toutes ces règles ont une confiance égale à 1.

Min-interest	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
IDFMH	222262	9340	430	430	430	430	430	358	67	67
RA	sat	633 770	184 470	184 470	184 470	184 470	184 470	127 806	4422	4422
nonRA	sat	0	0	0	0	0	0	0	0	0
Temps Exé(s)	sat	.	9. 942	9. 667	9. 655	10. 361	9. 648	7. 417	4. 068	3. 913

FIGURE 5.14 – Variation de $min - interest$ pour Min-sup=1

Min-interest	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
IDFMH	222354	2299	1852	1852	1852	1852	1852	97	7	7
RA	sat	2910	1528	1528	1528	1528	1528	63	0	0
nonRA	sat	716 280263	466 854594	466 797016	466 854602	466 797016	466 807892	152 796	84	84
Temps Exé(s)	sat	10546. 432	7612 . 812	6848. 459	6842 . 264	6843. 340	6840. 226	5. 383	2. 226	2. 242

FIGURE 5.15 – Variation de $min - interest$ pour Min-sup=0.1

En deuxième lieu, nous gardons fixes $min - interest$ à 1 (valeur importante) et à 0.2 (valeur faible) et nous faisons varier la valeur de $minsup$ de 0.01 à 1. Les résultats obtenus sont illustrés dans les tableaux de deux figures 5.16 et 5.17.
Selon ces deux tableaux, le nombre de règles valides générées et le nombre de règles non valides sont étroitement liés au nombre des $IDFMH$ extraits précédemment. Ceci est tout à fait logique, étant donnée que les règles sont extraites à base de ces $IDFMH$. Ainsi, pour le cas de $minsup = 0.07$ et $min - interest = 1$, le nombre des $IDFMH$ était nul et par conséquent ceux de règles valides et de règles non valides sont ausi nuls.

De même, nous remarquons que le temps d'exécution est aussi proportionnel au nombre de $IDFMH$ extrait à chaque valeur de $minsup$.

Min-sup	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.1
IDFMH	23	36	15	10	8	2	0	3	2	7
RA	3	52	6	14	12	2	0	0	0	0
nonRA	1827	5492	450	130	33	0	0	8	4	84
Temps Exé(s)	2.064	2.262	2.068	2.070	2.110	2.094	0	2.143	2.185	2.214

FIGURE 5.16 – Variation de *minsup* pour min-interest=1.

Min-sup	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.1
IDFMH	621	2890	629	721	870	1037	1453	1674	2065	2299
RA	24	6021	4824	759	16248	17432	29376	28025	13260	2910
nonRA	3789824	29430273	1947509	25466797	134229838	408159336	27381783	1208811398	1280368745	716268336
Temps Exé(s)	65.007	807.431	298.874	396.639	1982.233	5956.517	3558.315	19533.321	20382.451	10380427

FIGURE 5.17 – Variation de *minsup* pour min-interest=0.2.

Analyse qualitative : À ce niveau là, nous nous intéressons à trouver et interpréter des règles valides intéressantes, i.e., des règles qui contiennent des itemsets pour la prémisse et pour la conclusion qui ne sont pas proches sémantiquement. Ces règles vont nous permettre de mieux exploiter la base de données *Suisses* et d'interpréter des informations utiles sur les habitudes et le mode de vie des "Personnes" *Suisses*.

À la base de l'analyse quantitative et des tests réalisés, nous décidons de traiter des règles d'association valides et intéressantes pour des valeurs bien déterminées de *minsup* égales à 1, 0.5, 0.1, 0.05 et 0.01 et des valeurs bien déterminées aussi de $min - interest$ correspondantes respectivement 1, 0.5 et 0.2.

En fait, pour $minsup = 0.01$, nous testons les deux valeurs de min-interest égales à 1 et 0.2. De même, pour la valeur de $min - interest = 1$, nous testons les trois valeurs de *minsup* égales à 1, 0.5 et 0.05. La valeur de $min - conf$ était constante et égale à 0.5.

Le nombre de règles valides résultantes et à interpréter est résumé dans le tableau de la figure 5.18.

Min-interest \ Minsup	1	0.5	0.1	0.05	0.01
1	4422	41		12	
0.5			1528	974	
0.2			2910		24

FIGURE 5.18 – Nombre de règles valides en fonction de $minsup$ et de $min - interest$.

Ainsi, notre travail consiste à analyser ces *sept* fichiers correspondant aux règles extraites :

— fichier A.txt contenant 4422 règles valides et correspondant à minsup = 1 et min-interest=1.
— fichier B.txt contenant 41 règles valides et correspondant à minsup = 0.5 et min-interest=1.
— fichier C.txt contenant 12 règles valides et correspondant à minsup = 0.05 et min-interest=1.
— fichier D.txt contenant 1528 règles valides et correspondant à minsup = 0.1 et min-interest=0.5.
— fichier E.txt contenant 974 règles valides et correspondant à minsup = 0.05 et min-interest=0.5.
— fichier F.txt contenant 24 règles valides et correspondant à minsup = 0.01 et min-interest=0.2.
— fichier G.txt contenant 2910 règles valides et correspondant à minsup = 0.1 et min-interest=0.2.

Par la suite, nous codons chacun de ces fichiers, de façon qu'il contient des implications entre nos données réelles. Aini, chaque item de la base de données correspondra à une réponse à une question bien définie par laquelle a été interrogé la population *Suisse*.

Exemple 39. *Nous considérons la règle suivante du fichier A.txt :*
R *: 691 692 693 => 473 474 Supp = 11330 ,conf= 1.000000*
Cette règle R *sera codée à la forme* R' *et ceci pour faciliter son interprétation.*
R' *: Changement d'emploi/d'employeur-2ème raison :1,2,3(Occuper ou rechercher un emploi plus intéressant, Fin de contrat temporaire, Obligé de changer du fait de l'employeur) Changement d'emploi/d'employeur-2ème raison :5 (Garde des enfants ou d'autres personnes à charge) Changement d'emploi/d'employeur-2ème raison :7(autre raison) => Fumeurs-pipe :1oui Fumeurs-pipe :2non Supp = 11330 ,conf= 1.000000*

Lors de l'analyse de ces règles, nous nous limitons à examiner que les règles exactes (i.e., confiance est égale à 1). Dans cette liste de règles, nous cherchons celles qui nous paraissent intéressantes i.e., elles traitent dans la prémisse et dans

la conclusion des items pour lesquels l'implication n'était pas implicite.
Voici une liste de règles que nous jugeons intéressantes.

- implication entre : Autre événement grave (Année) => Souffre encore de cet événement grave(degrés)
- implication entre : Conflit dans l'entourage => Changement d'emploi/d'employeur-2ème raison
- implication entre : Poids individuel longitudinal, taille de l'échantillon => Poids transversal individuel, taille de l'échantillon inchangée
- implication entre : Poids individuel longitudinal, taille de l'échantillon => Poids individuel longitudinal, extrapolant à la taille de la population en 2004
- implication entre : Date de reception de la nationalité Suisse => Changement de métier
- implication entre : Permis de résidence(séjour annuel B/ d'établissement C) => Poids individuel taille de l'échantillon
- implication entre : Fumeurs-cigares / Fumeurs-pipe => Fumeurs-nombre de cigares par jour / Fumeurs-nombre de pipes par jour
- implication entre : Autres enfants nés-7ème enfant vit en Suisse => Satisfaction de vivre seul ou en commun
- implication entre : Conflits dans l'entourage-Mois => Souffre encore de ces conflits dans l'entourage(degrés).

5.6 Conclusion

Dans ce chapitre, nous avons proposé une nouvelle approche de fouille de règles d'association impliquant des items non fréquents. En effet, les items non fréquents sont groupés dans des itemsets pour produire des itemsets fréquents selon la mesure du support disjonctif. Dans le but de produire des règles aussi "compréhensibles" que possible, les itemsets disjonctifs fréquents ont été limités à être minimaux par rapport à l'inclusion ensembliste, et un critère d'homogénéité a été considéré pour filtrer les itemsets.

Nous avons implémenté un premier algorithme pour la fouille de tous les itemsets disjonctifs-fréquents minimaux homogènes et un deuxième pour le calcul de toutes les règles intéressantes.
Nos algorithmes ont été testés sur des données réelles issues de SHP.
Nous avons obtenu des résultats intéressantes que se soitent au niveau du nombre de motifs extraits (les $IDFMH$ et les règles d'association disjonctives intéressantes) ou au niveau du temps d'exécution et de la réalisabilité des ces expérimentations. De même, nous avons procédé à l'analyse et l'interprétation des résultats obtenus.
Dans le chapitre prochain, nous généralisons ces règles disjonctives vers d'autres règles tenant compte de différents types de supports que peut avoir un itemset.

Chapitre 6

Extraction de règles d'association généralisées

6.1 Introduction

Selon [Weiss, 2004], l'utilisation de l'opérateur de disjonction dans les règles d'association permet, entre autres, de fouiller des règles reliant des motifs fréquents et des motifs non fréquents (i.e., rares). Supposons que les deux itemsets X et Y sont non disjonctifs-fréquents et que l'itemset XY est disjonctif-fréquent c'est à dire $supp_{\vee}(XY)$ est supérieur ou égal à un seuil de support, nous étudions des règles de la forme R : X →Y.

Le support et la confiance de cette règle étudient respectivement la fréquence d'avoir l'ensemble X et Y et la fréquence d'avoir la conclusion Y étant donné la prémisse X, sachant que X et Y sont non disjonctifs-fréquents.

Notre démarche s'appuie à la base sur la représentation concise basée sur les itemsets disjonctifs-fréquents minimaux extraits dans le chapitre 4 (page 68).

Comme nous avons mentionné dans le chapitre 4, une telle représentation permet de déterminer les supports d'itemsets non disjonctifs-fréquents et disjonctifs-fréquents minimaux. Dans ce cadre, il importerait de mentionner qu'à l'aide des propriétés logiques du théorème 1 (page 13), il est possible de retrouver les supports conjonctifs et négatifs d'itemsets à partir de leurs supports disjonctifs.

C'est dans ce cadre, que s'inscrivent nos contributions de ce chapitre. En effet, nous extrayons des implications généralisées entre des motifs non disjonctifs-fréquents. Ces derniers sont difficiles à fouiller en utilisant des règles d'association conjonctives. Les règles à générer sont construites avec des différentes combinaisons des *expressions logiques* dans la prémisse et dans la conclusion, en considérant comme point de départ les règles d'association disjonctives.

Ainsi, nous disposons de tous les supports relatifs aux différentes expressions logiques qui peuvent être formées à partir de ces itemsets. Par conséquent, nous pouvons calculer les valeurs des mesures de qualité des règles d'association puisqu'elles sont exprimées généralement en fonction des supports de la prémisse et de la conclusion.
Il est important de mentionner que notre contribution dans ce chapitre est liée à celle de [Hamrouni et al., 2010] pour l'extraction de règles généralisées. Cependant, nos règles sont jugées plus générales.

6.2 Règles d'association généralisées

Dans cette section, nous étudions les règles d'association généralisées : leurs notations et les méthodes de calcul de leurs mesures d'intérêt associées.

6.2.1 Formulation logique

Dans ce qui suit, nous rappelons ce que nous avons mentionné dans la partie état de l'art, exactement dans le chapitre 2 (page 34). En effet, dans la section où nous avons évoqué l'extraction de règles d'association généralisées, nous avons introduit les notations nécessaires pour leur définition.

Si l'on considère un itemset I, chacun de ses éléments i peut être associé à la formule atomique $i \in X$ où X est une variable représentant un itemset quelconque. Il est alors possible de considérer des formules logiques combinant les formules atomiques avec les connecteurs logiques habituels $\vee$, $\wedge$ et $\neg$.

Définition 45. *Soit I un itemset quelconque, alors plusieurs formules logiques peuvent lui correspondre, chaque formule doit impliquer tout i de I et des connecteurs logiques.*

Exemple 40. *Pour $I = \{a, b\}$, $\varphi_1 = (b \in X) \wedge (b \in X)$ et $\varphi_2 = (a \in X) \vee \neg(b \in X)$ sont deux formules que l'on peut considérer à partir de I.*

Si maintenant on considère un itemset J et une formule φ construite à partir de l'itemset I, on dit que J satisfait φ, si en substituant X par J dans φ, la formule ainsi obtenue serait aussi satisfaite. Par exemple pour $J = \{a, b, c\}$, φ_1 est satisfaite, alors que φ_2 ne l'est pas.

Dans le cas d'une base de transactions, on peut définir le support d'une formule φ, que nous notons $supp(\varphi)$, comme étant le ratio entre le nombre de transactions dont l'itemset satisfait φ par le nombre total de transactions.

Par exemple, en retenant les deux formules φ_1 et φ_2 ci-dessus, pour un ensemble de transactions fixé, le support de φ_1 est égal au support conjonctif de I noté $supp_\wedge(I)$ ou $supp(I)$ dans la définition 3 (page 12) et le support de φ_2 ne correspond à aucun support défini jusque là. On remarque également que le support disjonctif de I noté $supp_\vee(I)$

précédemment est égal au support de la formule (a$\in$ X) $\vee$ (b$\in$ X).

Dans la mesure où les formules définies à partir d'itemsets seront utilisées par la suite, nous les noterons plus simplement en ne faisant figurer que les items et les connecteurs les constituant. Ainsi, la formule φ_1 sera notée $a \vee b$.

Définition 46. ***Support et Confiance d'une règle d'association généralisée***
Soit $\mathcal{R}$ une règle d'association généralisée $\varrho(x_1, x_2, \ldots, x_n) \Rightarrow \upsilon(y_1, y_2, \ldots, y_m)$.

Le support de $\mathcal{R}$, noté par Supp($\mathcal{R}$), est égal au nombre de transactions qui satisfont les deux expressions logiques $\varrho(x_1, x_2, \ldots, x_n)$ et $\upsilon(y_1, y_2, \ldots, y_m)$. D'où,

$$Supp(\mathcal{R}) = Supp(\varrho(x_1, x_2, \ldots, x_n) \wedge \upsilon(y_1, y_2, \ldots, y_m)).$$

La confiance de $\mathcal{R}$, dénotée par Conf($\mathcal{R}$), est le ratio entre son support et le support de l'expression logique représentant la partie prémisse. D'où,

$$Conf(\mathcal{R}) = \frac{Supp(\varrho(x_1,x_2,\ldots,x_n) \wedge \upsilon(y_1,y_2,\ldots,y_m))}{Supp(\varrho(x_1,x_2,\ldots,x_n))}.$$

Ainsi, une règle d'association généralisée est jugée valide si et seulement si Supp($\mathcal{R}$)$\geq$ $minsup$ et Conf($\mathcal{R}$)$\geq$ $minconf$, tels que $minsup$ et $minconf$ sont deux seuils respectivement du support et de confiance définis par l'utilisateur.

6.2.2 Calcul des supports des règles d'association généralisées

Dans ce qui suit, nous étudions les propriétés nécessaires pour le calcul des mesures d'intérêt des règles généralisées.

Rappelons que dans le chapitre 4, nous avons pu extraire deux types de motifs : les motifs non disjonctifs-fréquents et les motifs disjonctifs-fréquents minimaux munis chacun de son support disjonctif exact.

Nous considérons les règles d'association généralisées qui sont à la base des règles disjonctives de la forme $\mathcal{R} : X \Rightarrow Y$, telles que :

— $X = x_1, \ldots, x_n$, $Y = y_1 \ldots y_m$, $n \neq m$,
— X et Y sont deux motifs non disjonctifs-fréquents, i.e., $supp_\vee$(X) $<$ $minsup$ et $supp_\vee$(Y) $<$ $minsup$,
— la disjonction $Z = X \vee Y$ est un motif *fréquent minimal*, i.e., $supp_\vee$(Z) $\geq$ $minsup$

Par la suite, nous tentons de généraliser la fouille de ce type de règles disjonctives vers celle d'autres types de règles plus générales.

Notons que dans le chapitre 4, nous avons retrouvé les supports disjonctifs exacts de tous les motifs *non disjonctif-fréquents* et *disjonctifs-fréquents minimaux*. En plus, grâce aux formules données par le théorème 1 présenté dans le chapitre 1 (page 13), nous pouvons calculer les supports (conjonctif, conjonctif avec négation et disjonctif avec négation) de n'importe quelle expression booléenne $X = x_1, \ldots, x_n$ connaissant son support disjonctif et les supports disjonctifs de tous ses sous-ensembles. Par conséquent, les supports de toute forme d'une expression booléenne servant de prémisse ou de conclusion pourront être calculés.

Exemple 41. *Dans cet exemple, nous rappelons le contexte d'extraction du chapitre 4 de la page (66) dans la figure suivante.*

T_1	$a\ b$
T_2	$a\ c\ d\ e$
T_3	$c\ d\ e$
T_4	$d\ e\ f$
T_5	$a\ b\ c\ d\ e$
T_6	$a\ b\ c$

FIGURE 6.1 – Contexte d'extraction(1).

Nous montrons comment il est possible de retrouver le support conjonctif respectivement négatif d'un itemset quelconque à partir de son support disjonctif, en appliquant le théorème 1 de la page (13).
Soit X= abc, $supp_{\vee}(X)$= $\frac{5}{6}$, alors $supp_{\bar{\vee}}(X)$ = $\frac{6}{6}$ - $\frac{5}{6}$ =$\frac{1}{6}$. Par contre, nous avons $supp_{\wedge}$ (X)= supp(a) + supp(b) + supp(c) - $supp_{\vee}$(ab) - $supp_{\vee}$(ac) - $supp_{\vee}$(bc) + $supp_{\vee}$(abc) = $\frac{4}{6}$ + $\frac{3}{6}$ + $\frac{4}{6}$ - $\frac{4}{6}$ - $\frac{5}{6}$ - $\frac{5}{6}$ + $\frac{5}{6}$ = $\frac{2}{6}$ et $supp_{\bar{\wedge}}(X)$ = $\frac{6}{6}$ - $\frac{2}{6}$ = $\frac{4}{6}$.

Par ailleurs, nous montrons qu'il est possible de retrouver le support et la confiance de toute règle d'association $\mathcal{R}$: X⇒Y avec X et Y deux expressions logiques différentes et ceci grâce aux formules suivantes de calcul des supports des règles d'association. Ces formules seront appelées ultérieurement, selon le besoin, pour le calcul des supports et des confiances des règles généralisées.

Formule 1 R : $x_1 \wedge \ldots \wedge x_n \Rightarrow y_1 \vee \ldots \vee y_m$.
$supp((x_1 \wedge \ldots \wedge x_n) \wedge (y_1 \vee \ldots \vee y_m)) = supp(x_1 \wedge \ldots \wedge x_n) - supp(x_1 \wedge \ldots \wedge x_n \wedge \bar{y_1} \wedge \ldots \wedge \bar{y_m})$ [Galambos et Simonelli, 2000] .
Cette formule calcule le support d'une règle d'association dans le cas où la prémisse est une conjonction de littéraux positifs et la conclusion est une disjonction de littéraux positifs.

Formule 2 R : $x_1 \wedge \ldots \wedge x_n \Rightarrow \bar{y_1} \wedge \ldots \wedge \bar{y_m}$.
$supp(x_1 \wedge \ldots \wedge x_n \wedge \bar{y_1} \wedge \ldots \wedge \bar{y_m}) = supp(x_1 \wedge \ldots \wedge x_n) + \sum_{\emptyset \subset S \subseteq \{y_1 \ldots y_m\}} (-1)^{|S|} supp(x_1 \wedge \ldots \wedge x_n \wedge S)$ [Toivonen, 1996].
Cette formule calcule le support d'une règle d'association dans le cas où la prémisse est une conjonction de littéraux positifs et la conclusion est une conjonction de littéraux négatifs.

Formule 3 $supp(X \wedge Y) = supp(X) + supp(Y) - supp(X \vee Y)$.
Cette formule calcule le support d'une règle d'association dans le cas général, pour n'importe quelle formule de X et de Y.

Formule 4 $supp(X \wedge \bar{Y})$= $supp(X \vee Y)$ - $supp$(Y).
Cette formule calcule le support d'une règle d'association dans le cas où la prémisse est positive et la conclusion est négative, en se faisant aider par le support disjonctif de l'union (prémisse, conclusion).

Selon la définition 3 (page 12), un itemset quelconque I peut lui être associé *quatre types de supports différents*. Ainsi, quatre types d'expressions logiques sont à former.

Exemple 42. *Soit I=abc, alors* $\varphi_1 = a \wedge b \wedge c$, $\varphi_2 = a \vee b \vee c$, $\varphi_3 = \neg a \vee \neg b \vee \neg c$ *et* $\varphi_4 = \neg a \wedge \neg\ b \wedge \neg\ c$.

Par la suite, quand nous passons à constituer les règles d'association, il est possible de considérer *seize formes différentes* de règles d'association avec des différentes combinaisons dans la prémisse et dans la conclusion (cf tableau de la figure 6.2).

6.3 Discussion sur l'état de fréquence des règles générales

Dans cette section, et comme présenté dans le tableau du tableau 6.3, nous discutons l'état de fréquence possible des différentes formes des règles d'association générées.

Nous partons de l'itemset (Prémisse-Conclusion) qui est un itemset disjonctif-fréquent minimal, sous entendu les deux itemsets ou disons les parties de la règle *Prémisse* et *Conclusion* sont deux itemsets non disjonctifs-fréquents (i.e., $supp_{\vee}$(prémisse)$\leq minsup$ et $supp_{\vee}(conclusion) \leq minsup$).

Nous nous intéressons maintenant à étudier le support de la règle $\mathcal{R}$: Prémisse $\Rightarrow$ Conclusion, pour chacune des formes du tableau 6.2.

Considérons la forme 1, nous avons $supp_{\vee}$(prémisse)$\leq minsup$. Par la suite, $supp_{\wedge}$(prémisse)$\leq$ $minsup$, et c'est de même pour la *conclusion*. Par conséquent, $supp_{\wedge}$(prémisse conclusion)$\leq$ $minsup$ (propriété dérivée de l'algorithme Apriori, i.e., l'auto-jointure de deux itemsets non fréquents dans le cas du calcul du support conjonctif) est un itemset non fréquent.

Par conséquent, le support de la règle $\mathcal{R}$: Prémisse $\Rightarrow$ Conclusion est inférieur à $minsup$ et la règle $\mathcal{R}$ est qualifiée non fréquente pour la forme 1.

Maintenant, étant donné l'état de fréquence d'un itemset à la forme disjonctive, nous allons déduire son état de fréquence à la forme conjonctive à l'aide de la propriété suivante :

Forme	Support & Confiance	Formules de réécriture
$R_1 : \vee X \Rightarrow \vee Y$	$supp_{\vee}(X) + supp_{\vee}(Y) - supp_{\vee}(Z)$. $\frac{supp_{\vee}(X)+supp_{\vee}(Y)-supp_{\vee}(Z)}{supp_{\vee}(X)}$.	1.
$R_2 : \vee X \Rightarrow \wedge Y$	- $\sum_{\emptyset \subset S \subseteq \{X_1 \ldots X_n\}} (-1)^{\lvert S \rvert} supp(Y_1 \wedge \ldots \wedge Y_m \wedge S)$. $\frac{-\sum_{\emptyset \subset S \subseteq \{X_1 \ldots X_n\}} (-1)^{\lvert S \rvert} supp(Y_1 \wedge \ldots \wedge Y_m \wedge S)}{supp_{\vee}(X)}$.	1.
$R_3 : \vee X \Rightarrow \wedge \bar{Y}$	$supp_{\vee}(X) + \sum_{\emptyset \subset S \subseteq \{X_1 \ldots X_n\}} (-1)^{\lvert S \rvert} supp(Y_1 \wedge \ldots \wedge Y_m \wedge S)$. $\frac{supp_{\vee}(X)+\sum_{\emptyset \subset S \subseteq \{X_1 \ldots X_n\}} (-1)^{\lvert S \rvert} supp(Y_1 \wedge \ldots \wedge Y_m \wedge S)}{supp_{\vee}(X)}$.	1, 2, et 4.
$R_4 : \vee X \Rightarrow \vee \bar{Y}$	$supp_{\vee}(Z)$ - $supp_{\vee}(Y)$. $\frac{supp_{\vee}(Z)-supp_{\vee}(Y)}{supp_{\vee}(X)}$.	4.
$R_5 : \wedge X \Rightarrow \vee Y$	- $\sum_{\emptyset \subset S \subseteq \{Y_1 \ldots Y_m\}} (-1)^{\lvert S \rvert} supp(X_1 \wedge \ldots \wedge X_n \wedge S)$. $\frac{-\sum_{\emptyset \subset S \subseteq \{Y_1 \ldots Y_m\}} (-1)^{\lvert S \rvert} supp(X_1 \wedge \ldots \wedge X_n \wedge S)}{supp_{\wedge}(X)}$.	1 et 2.
$R_6 : \wedge X \Rightarrow \wedge Y$	$supp_{\wedge}(Z)$. $\frac{supp_{\wedge}(Z)}{supp_{\wedge}(X)}$.	Aucune.
$R_7 : \wedge X \Rightarrow \wedge \bar{Y}$	$supp_{\wedge}(X)$ - $supp_{\wedge}(Z)$. $\frac{supp_{\wedge}(X)-supp_{\wedge}(Z)}{supp_{\wedge}(X)}$.	1.
$R_8 : \wedge X \Rightarrow \vee \bar{Y}$	$supp_{\wedge}(X) + \sum_{\emptyset \subset S \subseteq \{Y_1 \ldots Y_m\}} (-1)^{\lvert S \rvert} supp(X_1 \wedge \ldots \wedge X_n \wedge S)$. $\frac{supp_{\wedge}(X)+\sum_{\emptyset \subset S \subseteq \{Y_1 \ldots Y_m\}} (-1)^{\lvert S \rvert} supp(X_1 \wedge \ldots \wedge X_n \wedge S)}{supp_{\wedge}(X)}$.	2.
$R_9 : \vee \bar{X} \Rightarrow \vee Y$	$supp_{\vee}(Z)$- $supp_{\vee}(X)$. $\frac{supp_{\vee}(Z)-supp_{\vee}(X)}{supp_{\bar{\vee}}(X)}$.	4.
$R_{10} : \vee \bar{X} \Rightarrow \wedge Y$	$supp_{\wedge}(Y) + \sum_{\emptyset \subset S \subseteq \{X_1 \ldots X_n\}} (-1)^{\lvert S \rvert} supp(Y_1 \wedge \ldots \wedge Y_m \wedge S)$. $\frac{supp_{\wedge}(Y)+\sum_{\emptyset \subseteq S \subseteq \{X_1 \ldots X_n\}} (-1)^{\lvert S \rvert} supp(Y_1 \wedge \ldots \wedge Y_m \wedge S)}{supp_{\bar{\vee}}(X)}$.	2.
$R_{11} : \vee \bar{X} \Rightarrow \wedge \bar{Y}$	$supp_{\bar{\vee}}(X)$ - $supp_{\wedge}(Y)$ - $\sum_{\emptyset \subseteq S \subseteq \{X_1 \ldots X_n\}} (-1)^{\lvert S \rvert} supp(Y_1 \wedge \ldots \wedge Y_m \wedge S)$. $\frac{supp_{\bar{\vee}}(X)-supp_{\wedge}(Y)-\sum_{\emptyset \subseteq S \subseteq \{X_1 \ldots X_n\}} (-1)^{\lvert S \rvert} supp(Y_1 \wedge \ldots \wedge Y_m \wedge S)}{supp_{\bar{\vee}}(X)}$.	1 et 2.
$R_{12} : \vee \bar{X} \Rightarrow \vee \bar{Y}$	$supp_{\bar{\vee}}(Z)$. $\frac{supp_{\bar{\vee}}(Z)}{supp_{\bar{\vee}}(X)}$.	Aucune.
$R_{13} : \wedge \bar{X} \Rightarrow \vee Y$	$supp_{\vee}(Y) + \sum_{\emptyset \subseteq S \subseteq \{Y_1 \ldots Y_m\}} (-1)^{\lvert S \rvert} supp(X_1 \wedge \ldots \wedge X_n \wedge S)$. $\frac{supp_{\vee}(Y)+\sum_{\emptyset \subseteq S \subseteq \{Y_1 \ldots Y_m\}} (-1)^{\lvert S \rvert} Supp(X_1 \wedge \ldots \wedge X_n \wedge S)}{supp_{\bar{\wedge}}(X)}$.	1, 2 et 4.
$R_{14} : \wedge \bar{X} \Rightarrow \wedge Y$	$supp_{\wedge}(Y)$ - $supp_{\wedge}(Z)$. $\frac{supp_{\wedge}(Y)-supp_{\wedge}(Z)}{supp_{\bar{\wedge}}(X)}$.	1.
$R_{15} : \wedge \bar{X} \Rightarrow \wedge \bar{Y}$	$supp_{\bar{\wedge}}(X)$ + $supp_{\bar{\wedge}}(Y)$ - $supp_{\bar{\wedge}}(Z)$. $\frac{supp_{\bar{\wedge}}(X)+supp_{\bar{\wedge}}(Y)-supp_{\bar{\wedge}}(Y)}{supp_{\bar{\wedge}}(X)}$.	3.
$R_{16} : \wedge \bar{X} \Rightarrow \vee \bar{Y}$	$supp_{\bar{\vee}}(Y)$ - $supp_{\wedge}(X)$ - $\sum_{\emptyset \subseteq S \subseteq \{Y_1 \ldots Y_m\}} (-1)^{\lvert S \rvert} supp(X_1 \wedge \ldots \wedge X_n \wedge S)$. $\frac{supp_{\bar{\vee}}(Y)-supp_{\wedge}(X)-\sum_{\emptyset \subseteq S \subseteq \{Y_1 \ldots Y_m\}} (-1)^{\lvert S \rvert} supp(X_1 \wedge \ldots \wedge X_n \wedge S)}{supp_{\bar{\wedge}}(X)}$.	1 et 2.

FIGURE 6.2 – Formes, supports et confiances de règles d'association généralisées.

Propriété 14. *Pour un itemset I quelconque, $supp_{\wedge}(I) \leq supp_{\vee}(I)$.*

Preuve :
Le support conjonctif d'un itemset I est égal au nombre total des transactions satisfaisant tous *les i t.q $i \in I$ (divisé par le nombre total des transactions). Cependant, celui disjonctif est égal au nombre total des transactions satisfaisant l'existence d'*au moins *un i t.q $i \in I$.* $\square$

Étant donné que le support conjonctif d'un itemset est inférieur ou égal à son support disjonctif, alors si les Prémisses et les Conclusions étaient non disjonctives-fréquentes alors elles sont également non fréquentes.

Par conséquent, le support de la règle $\mathcal{R}$: Prémisse $\Rightarrow$ Conclusion (pour Prémisse et Conclusion calculées à leurs formes conjonctives) est inférieur à *minsup* et la règle $\mathcal{R}$ est qualifiée non fréquente pour les formes 2, 5 et 6.

Si nous considérons les règles avec des négations au niveau des Prémisse ou au niveau des Conclusion, i.e., les formes 3, 4, 7 et 8 et si la Prémisse et la Conclusion étaient non disjonctives-fréquentes, alors elles doivent être fréquentes à leurs formes négatives qu'elle soit une négation sur disjonction ou sur conjonction (c.f., preuve Règle de De Morgan). Par la suite, le support de la règle $\mathcal{R}$: Prémisse $\Rightarrow$ Conclusion va faire dans toutes ces formes la conjonction entre un terme fréquent (respectivement disjonctif-fréquent) et un terme non fréquent (respectivement non disjonctif-fréquent). Or, selon la propriété arithmétique, la conjonction de deux motifs, l'un fréquent l'autre non fréquent, donne toujours une conjonction non fréquente. Par la suite, la règle $\mathcal{R}$ est qualifiée non fréquente pour toutes ces formes.

Pour les règles avec des négations au niveau des Prémisse et Conclusion, i.e., les formes 11, 12, 15 et 16, si la Prémisse et la Conclusion étaient non disjonctives-fréquentes, alors elles doivent être fréquentes à leurs formes négatives qu'elle soit une négation sur disjonction ou sur conjonction (c.f., preuve Règle de De Morgan).
Par la suite, le support de la règle $\mathcal{R}$: Prémisse $\Rightarrow$ Conclusion est à étudier car la conjonction de deux motifs fréquents peut être fréquente ou non fréquente, et par la suite la règle $\mathcal{R}$ peut être qualifiée fréquente ou non.

Quant à la dernière catégorie, i.e., les formes 9, 10, 13 et 14, ces règles décrivent des implications entre des formes négatives (négation de disjonctions ou de conjonctions) et des formes positives (de disjonctions ou de conjonctions). Dans ce cas, les prémisses sont fréquentes (respectivement disjonctives-fréquentes) et les conclusions sont non fréquentes (respectivement non disjonctives-fréquentes). D'où la règle $\mathcal{R}$ est qualifiée non fréquente pour ces formes.

Conclusion : toutes les formes ont été démontrées qu'elles sont non fréquentes, sauf 11, 12, 15 et 16. Leur état dépend de la répartition des données de la base.

	Disjonction	Conjonction	Négation de conjonction	Négation de disjonction
Disjonction	1	2	3	4
Conjonction	5	6	7	8
Négation de disjonction	9	10	11	12
Négation de conjonction	13	14	15	16

FIGURE 6.3 – Estimation de l'état de fréquence des différentes formes de règles d'association.

6.4 Algorithme et expérimentations

Dans cette section, nous donnons l'algorithme pour la génération des règles généralisées et les expérimentations effectuées. Notons que, nous nous limitons dans notre implémentation et nos expérimentations aux formes 1, 4 et 9, i.e., qui ne nécessitent aucun balayage supplémentaire de la base de données et se suffisent par une simple application d'une formule mathématique.

6.4.1 Algorithme

Dans cette sous-section, nous discutons l'algorithme 18 pour la fouille de règles d'association généralisées, ainsi que l'algorithme 19. De même, nous fournissons un exemple illustratif pour explication. L'algorithme 18 est basé sur l'algorithme DISAPRIORI (page 68) pour la fouille d'itemsets disjonctifs-fréquents minimaux.

L'input de cet algorithme est un itemset disjonctif fréquent minimal de taille k ≥ 3. Ainsi, le but de cet algorithme est d'écrire toutes les possibilité de règles d'association Prémisse $\rightarrow$ Conclusion, tels que *Prmisse* et *Conclusion* sont non nuls, Prémisse $\cap$ Conclusion $= \emptyset$ et Prémisse $\cup$ Conclusion est l'itemset disjonctif fréquent minimal en question.
Une fois, les différentes combinaisons sont construites à partir de l'itemset $freq$, on les stocke dans la liste de règles formées *Règles* (ligne 16) de l'algorithme 18. Par la suite, nous procédons à calculer le support et la confiance de chaque règle de *Règles*, par l'appel de l'algorithme 19.
Ce dernier calcule alors les mesures (support et confiance) de chaque règle R de *Rgles* selon la forme définie.
Finalement, l'algorithme 18 retourne que les règles dont les supports et les confiances satisfont les seuils respectifs σ et γ.

Algorithme 18 : Règles d'association généralisées

Données : La table Δ, le seuil du support σ, le seuil de la confiance γ et Freq
Résultat : RA : ensemble des règles valides

```
   /* formulation des règles                                          */
1  pour chaque freq de Freq tel que k≥3 faire
2  |  pour i= 0 ; i ≤ k/2 ; i++ faire
3  |  |  pour j=0 ; j<k ; j++ faire
4  |  |  |  Premisse=0 ; Conclusion=0;
5  |  |  |  pour l=0 ; l<k ; l++ faire
6  |  |  |  |  r= (l+j)% k;
7  |  |  |  |  si l ≤ i alors
8  |  |  |  |  |  Premisse=Premisse + freq[r]
9  |  |  |  |  sinon
10 |  |  |  |  |  Conclusion=Conclusion + freq[r]
11 |  |  |  |  fin
12 |  |  |  fin
13 |  |  fin
14 |  fin
15 fin
16 Règles = Prémisse → Conclusion;
   /* calcul de leurs supports et de leurs confiances                 */
17 Procédure (Calcul supports et confiances) pour chaque R de Règles faire
18 |  si (supp(R) ≥ σ) ET (conf(R) ≥ γ) alors
19 |  |  RA = RA ∪ R
20 |  fin
21 fin
22 return RA
```

Exemple 43. *Nous considérons le contexte d'extraction de la figure 6.4 et le seuil du support minsup* $= \frac{6}{6}$.

T_1	$a\ b$
T_2	$a\ c\ d\ e$
T_3	$c\ d\ e$
T_4	$d\ e\ f$
T_5	$b\ c\ e$
T_6	$a\ c$

FIGURE 6.4 – Contexte d'extraction(2).

Ainsi, $supp_{\vee}(abcd) = \frac{\{T_1,T_2,T_3,T_3,T_5,T_6\}}{6} = \frac{6}{6}$, *et il est dit disjonctif-fréquent minimal, étant donnée qu'aucun de ses sous-ensemble n'est disjonctif-fréquent.*

Les différentes possibilités de règles qu'on peut dériver de l'itemset abcd et selon l'algorithme 18 sont les suivantes :

- R_1 *:* $a \rightarrow bcd$ *et* R_2 *:* $bcd \rightarrow a$
- R_3 *:* $ab \rightarrow cd$ *et* R_4 *:* $cd \rightarrow ab$
- R_5 *:* $abc \rightarrow d$ *et* R_6 *:* $d \rightarrow abc$

Au niveau de l'algorithme 19, nous allons donc considérer les trois formes suivantes : Forme 1, 4 *et* 9*. Nous fournissons, ainsi, les formules de calcul des supports et des confiances de la règle* R_1 *ci dessus, selon ces trois Formes.*

- $supp(R_1) = supp_\vee(a) + supp_\vee(bcd) - supp_\vee(abcd) = \frac{3}{6} + \frac{6}{6} - \frac{6}{6} = \frac{3}{6}$ *et* $conf(R_1) = \frac{supp_\vee(a)+supp_\vee(bcd)-supp_\vee(abcd)}{sup_\vee(a)} = 1$.
- $supp(R_1) = supp_\vee(abcd) - supp_\vee(bcd) = \frac{6}{6} - \frac{6}{6} = 0$ *et* $conf(R_1) = \frac{supp_\vee(abcd)-supp_\vee(bcd)}{sup_\vee(a)} = 0$.
- $supp(R_1) = supp_\vee(abcd) - supp_\vee(a) = \frac{6}{6} - \frac{3}{6} = \frac{3}{6}$ *et* $conf(R_1) = \frac{supp_\vee(abcd)-supp_\vee(a)}{1-sup_\vee(a)} = 1$.

Algorithme 19 : Calcul supports et confiances

```
Données : Règles
pour chaque R : X → Y de Règles faire
    suivant Forme faire
        cas où Forme1
            supp(R) = supp∨(X) + supp∨(Y) - supp∨(Z);
            conf(R) = supp(R) / Supp∨(X);
        cas où Forme4  supp(R) = supp∨(Z) - supp∨(Y);
        conf(R) = supp(R) / Supp∨(Y);
        cas où Forme9
            Supp(R) = supp∨(Z) - supp∨(X);
            conf(R) = supp(R) / supp∨(X);
            break;
        fin
        autres cas
            Rien faire;
        fin
    fin
fin
```

6.4.2 Étude expérimentale

Dans cette sous-section, nous décrivons l'interprétabilité et la performance de notre approche par le biais des résultats expérimentaux obtenus. Toutes nos expérimentations ont été réalisées sur un PC muni d'un processeur Intel(R) Core(TM) i3 ayant une fré-

quence d'horloge de $2,13$ GHz et 3.00 Go de RAM (2.43 Go de swap) tournant sur la plate-forme Linux Ubuntu. Le programme a été implémenté en langage C++.
Ces données proviennent de la banque de données Uniprot (`www.uniprot.org`), qui est une ressource complète des séquences de protéines et de leurs informations fonctionnelles [Terrapon, 2009]. En effet, les protéines peuvent être décomposées en une ou plusieurs unités structurales et/ou fonctionnelles élémentaires appelées domaines.
Pour tester les formes précédentes (i.e., 1, 4 et 9), nous filtrons en premier lieu la base *Uniprot* en gardant seulement les lignes contenant les domaines des protéines de la famille *PFAM* puis en second lieu nous filtrons selon les identifiants eucaryotes pour trouver exactement 124332 lignes de la base *Uniprot*.
De plus, puisque les données traitées sont réelles donc très variées, alors le taux de répétition d'un domaine d'une protéine quelconque est faible. Par conséquent et dans le but d'avoir des domaines des protéines qui sont relativement fréquents, nous faisons varier la valeur de *minsup* entre $0,001$ et $0,01$.

Nos expérimentations s'intéressent au temps d'exécution nécessaire pour la fouille de toutes les règles relatives aux formes sélectionnées, en fonction de la variation de *minsup*. Nous illustrons en réalité la totalité des règles indépendamment du fait qu'elles soient fréquentes (de support $\geq$ *minsup*) ou non, car comme nous avons pu montrer dans la section précédente que toutes les règles de formes 1, 4 ou 9 sont non fréquentes.
Ces règles non fréquentes, appelées aussi règles *rares*, reflètent des régularités qui sont non fréquentes mais qui peuvent fournir des connaissances intéressantes aux utilisateurs finaux. Ces règles correspondent à des corrélations apparaissant dans un nombre limité d'instances dans certains domaines tels que : la bio-informatique, la génétique, le diagnostique médical, etc.

Ainsi, la figure 6.5 montre la variation du temps d'exécution en fonction de la variation de *minsup*.

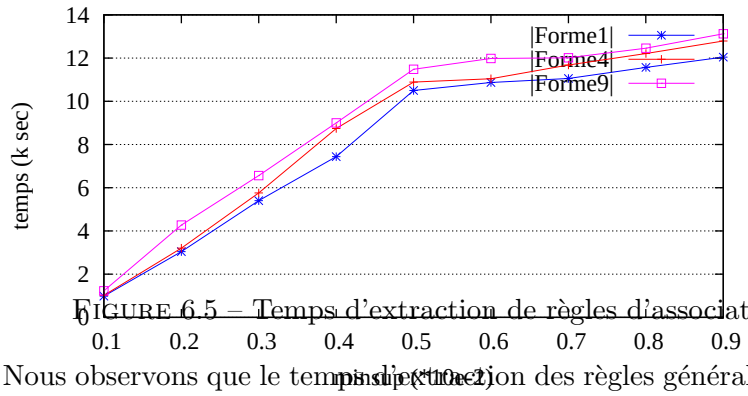


FIGURE 6.5 – Temps d'extraction de règles d'association généralisées.

Nous observons que le temps d'exécution des règles généralisées diminue en fonction

de la valeur de *minsup*. Ceci est expliqué par le fait que, dans le cas disjonctif, plus le support est élevé, plus le nombre des disjonctions non fréquentes est important et celui des disjonctions fréquentes est faible et que dans notre approche, l'extraction des règles généralisées est basée sur les disjonctions fréquentes minimales.

6.5 Conclusion

Dans ce chapitre, nous avons proposé une nouvelle approche plus générale d'extraction des règles d'association généralisées. Cette proposition a fait le sujet notre deuxième contribution dans le cadre de cette thèse [Hilali-Jaghdam et al., 2012].

En plus, notre contribution est considérée plus générique par rapport aux contributions précédentes dans le domaine de la fouille de règles d'association généralisées. Notre approche étend le système de règles d'association classiques en tenant compte des différents connecteurs logiques aussi bien la négation que se soit sur une disjonction ou une conjonction des items. Les expériences que nous avons réalisées sur des données réelles ont montré que notre approche est réaliste en terme de temps d'exécution.

CONCLUSION ET PERSPECTIVES

La fouille d'itemsets fréquents à partir de bases de données n'a pas cessé depuis les trois dernières décennies de réaliser des grands succès et de faire varier des problématiques selon les besoins du monde courant. En effet, la plupart des approches existantes sont dédiées au problème de la recherche de motifs fréquents malgré que les motifs non fréquents (rares) sont aussi prouvés intéressants dans certaines applications. En plus, quand nous fouillons des motifs intéressants, l'intérêt ne porte plus seulement sur la fréquence d'itemsets dans la base de données mais plutôt sur certains autres critères. De même, la fouille de motifs fréquents à partir de bases de données souffre du problème de l'abondance de motifs extraits qui nécessitent parfois de les fouiller de nouveau pour en choisir les plus utiles. Pour ceci, les recherches sont orientées vers considérer d'autres moyens pour permettre à fouiller que de motifs utiles.

Dans ce qui suit, nous allons résumer en première section nos contributions, puis en deuxième sections discuter quelques perspectives envisageables et relatives à nos contributions.

Résumé de nos contributions

Dans cette thèse, nous nous intéressons à la fouille de règles d'association disjonctives à partir d'items non fréquents qui est une nouvelle tentative dans cette problématique de fouille de données. Pour ceci, nous introduisons nos contributions une par une jusqu'à aboutir à ces règles.

En premier lieu de cette thèse, nous avons proposé un algorithme par niveau de type Apriori que nous l'appelons Disapriori pour la fouille de tous les itemsets disjonctifs-fréquents minimaux à part d'une base de données et selon un seuil de *minsup*. En effet, certaines applications du monde réel on montré que le calcul du support disjonctif (en comptant l'occurrence complémentaire d'items) est plus intéressant que celui du support conjonctif.
De même, nous avons montré que l'ensemble d'itemsets disjonctifs-fréquents minimaux constitue une représentation concise approximative de l'ensemble d'itemsets disjonctifs. Ainsi, l'algorithme Disapriori est capable de retrouver le support disjonctif seulement pour le cas des itemsets disjonctifs-fréquents minimaux et les itemsets non disjonctifs-fréquents mais pas le support d'itemsets disjonctifs-fréquents.
Par la suite, nous avons étendu l'algorithme Disapriori au cas du modèle relationnel pour fouiller des requêtes de sélection disjonctives-fréquentes minimales à partir d'une base de données relationnelle. Ces requêtes sont de grand intérêt dans certaines applications telles que : les explications des symptômes observés dans un diagnostique médical ou biologique, etc.

En second lieu, nous sommes partis de l'idée que, dans le domaine de la fouille de données, l'intérêt ne porte pas seulement sur la fréquence d'itemsets, mais plutôt sur d'autres critères. Pour ceci, nous supposons une taxonomie sur l'ensemble d'items de

la base de données et une mesure de similarité entre les items permettant de qualifier l'homogénéité d'itemsets et les classer en homogènes ou hétérogènes. Les contributions dans ce chapitre ont porté principalement sur :

— l'extraction de l'ensemble d'itemsets disjonctifs-fréquents minimaux homogènes (i.e., fréquents selon un seuil de support et homogènes selon un seuil d'homogénéité), à partir d'une base de données, à l'aide d'un algorithme de parcours en profondeur : IDFMH. Un itemset est jugé homogènes si sa mesure de similarité *Overall-Relatedness* satisfait un certain seuil d'homogénéité pré-défini.

— la construction de règles d'association disjonctives intéressantes à partir de l'ensemble d'itemsets disjonctifs-fréquents minimaux homogènes pré-extraits. Ces règles sont générées de manière itérative à l'aide d'un algorithme par niveaux DisRègles. Dans ce dernier, nous procédons par augmenter la conclusion d'une règle, qui n'a pas encore satisfait le seuil de *minsup*, à chaque itération.

En dernier lieu, nous procédons à généraliser les règles disjonctives extraites précédemment par des différentes autres formes de calcul des supports disjonctifs, conjonctifs et négatifs. Ainsi, ces règles généralisées véhiculent des informations intéressantes concernant la co-occurrence, l'occurrence complémentaire et l'absence de l'occurrence des items. Notre proposition considère au départ une forme particulière de règles d'association pour en dériver les différentes formes généralisées existantes.
Cette approche est considérée la plus générique dans le contexte de la fouille des règles d'association généralisées, puisqu'elle permet d'extraire *seize* formes généralisées de règles d'association.
De cette façon, nous considérons au départ des règles d'association disjonctives avec des prémisses et des conclusions qui sont des itemsets non disjonctifs-fréquentes et dont la disjonction φ = Prémisse $\vee$ Conclusion est disjonctive-fréquente minimale. Ensuite, nous calculons les mesures relatives aux différentes formes généralisées qui peuvent en être dérivées.

Il est à noter que, nous avons procédé à une autre contribution qui a porté sur la fouille d'itemsets conjonctifs-fréquents *homogènes* et *fermés*. Pour cela, nous avons proposé une nouvelle notion de fermeture qui prend en compte, non pas seulement le support des itemsets, mais aussi leur degrés d'homogénéité dans le contexte d'une taxonomie donnée.
De même, nous avons montré que connaître l'ensemble de tous les itemsets fermés fréquents et homogènes avec leurs supports et leurs degrés d'homogénéité, permet de connaître tous les itemsets fréquents et homogènes.

Perspectives

Les perspectives de recherche dans le cadre de cette thèse sont diverses et peuvent aller de l'amélioration de travaux déjà existants vers la proposition des nouveaux chemins

à aborder :

1. Comme travail futur, nous proposons faire l'extraction non pas seulement des requêtes disjonctives de sélection mais aussi des requêtes de projection-sélection et aller encore plus vers les requêtes de projection-sélection-jointure i.e., à partir de plusieurs tables. Ces requêtes sont sémantiquement plus riches que celles que nous avons extraites.
2. Pour l'algorithme de parcours en profondeur IDFMH que nous avons proposé , nous comptons optimiser cet algorithme. Cette optimisation doit porter en particulier sur la redondance qu'on rencontre au niveau de la vérification de la minimalité d'itemsets prouvé disjonctifs-fréquents.
3. Pour les règles d'association généralisées, nous allons terminer à implémenter le reste des formes et de faire l'interprétation des règles extraites sur des données bio-informatiques pour mettre en valeur leur utilité.
4. Pour notre contribution concernant la fouille d'itemsets conjonctifs fréquents fermés et homogènes, qui ne fait pas partie de nos travaux de thèse, nous avons réussi à proposer un algorithme par niveau pour fouiller ces itemsets. De même, nous avons rédigé un article dans ce contexte qui a été accepté (c'est le dernier article dans la liste de publications qui suit).
   Reste alors à valider nos résultats via une étude expérimentale, que nous comptons faire prochainement.

Publications dans le cadre de cette thèse :

- I. Hilali-Jaghdam, T.Y. Jen , D. Laurent et S. Ben Yahia.
Mining frequent disjunctive selection queries.
The 22 nd International conference on Database and Expert System Applications, DEXA 2011, Part II, Pages 90-96, Toulouse, France, Springer, Août 2011.

- I. Hilali-Jaghdam, T.Y. Jen , D. Laurent et S. Ben Yahia.
Fouille de règles d'association généralisées à travers les disjonctions : application aux données génomiques.
La 12 ème conférence internationale francophone sur l'Extraction et la Gestion de Connaissances, EGC 2012, RNTI-E-23, Éditions Hermann, pages 237-242, Bordeaux, France, Janvier 2012.

- I. Hilali-Jaghdam, T.Y. Jen , D. Laurent C. Marinica et S. Ben Yahia.
Mining Interesting Disjunctive Association Rules from Unfrequent Items.
International Workshop, ISIP 2013, Bangkok, Thailand, September 16–18, 2013. Information Search, Integration, and Personalization. Editeurs : Asanee Kawtrakul, Dominique Laurent, Nicolas Spyratos, Yuzuru Tanaka, pages 84-99.

- I. Hilali-Jaghdam, T.Y. Jen , D. Laurent C. Marinica et S. Ben Yahia. (à paraître)
Mining Frequent and Homogeneous Closed Itemsets.
International Workshop, ISIP 2014, HELP College of Arts and Technology, Kuala Lumpur, October 9–10, 2014. Information Search, Integration, and Personalization. Editeurs : Asanee Kawtrakul, Dominique Laurent, Nicolas Spyratos, Yuzuru Tanaka.

Bibliographie

[Han et Kamber, 1992] J. Han et M. Kamber. *Data mining concepts and technics.* Morgann Kaufmann, San Francisco, 1992.

[Hamrouni, 2009] T. Hamrouni. *Mining concise representations of frequent patterns through conjunctive and disjunctive search spaces.* Université d'Artois France, Août, 2009.

[Diop, 2003] C. Diop. *Étude et mise en œuvre des aspects itératifs de l'extraction de règles d'asoociation dans une base de données.* Université François Rabelais Blois, Tours, Chinon, Décembre, 2003.

[Pasquier, 2000] N. Pasquier. *Data mining : Algorithmes d'extraction et de réduction des règles d'association dans les bases de données.* Université Clermont Ferrand II, France, École Doctorale Sciences pour l'Ingénieur de Clermont Ferrand, Janvier, 2000.

[Pennerath, 2009] F. Pennerath. *Méthodes d'extraction de connaissances à partir de données modélisables par des graphes. Application à des problèmes de synthèse organique.* Computer Science. Université Henri Poincaré - Nancy I, Novembre, 2009.

[Salleb, 2003] A. Salleb. *Recherche de motifs fréquents pour l'extraction de règles d'association et de caractérisation.* Laboratoire d'Informatique Fondamentale d'Orléans LIFO, Université d'Orléans, France, Décembre, 2003.

[Dieng, 2011] C. Tidiane Dieng. *Étude et implantation de l'extraction de requêtes fréquentes dans les bases de données multidimensionnelles.* Université de Cergy-Pontoise, France, Juillet, 2011.

[Zighed et Rakotomalala, 2000] , D. A. Zighed and R. Rakotomalala. *Graphes d'Induction : Apprentissage et Data Mining.* Ed. Hermès - Lavoisier. Science Publications, 2000.

[Ganter et Wille, 1999] B. Ganter and R. Wille. *Formal Concept Analysis : Mathematical Foundations.* Springer Verlag Berlin Heidelberg, 1999.

[Nambiar, 2009] U. Nambiar. *Supporting Imprecision in Database Systems.* Encyclopedia of Data Warehousing and Mining. Second Edition, John Wang (Ed.). pp. 1884-1887, 2009.

[Galambos et Simonelli, 2000] J. Galambos and I. Simonelli. *Bonferroni-type Inequalities with Applications.* Springer, 2000.

[Ceglar et Roddick, 2006] A. Ceglar and J.F. Roddick. *Association mining.* ACM Computing Surveys, Vol. 38, no. 2, 2006.

[Codd, 1972] E. F. Codd. *Relational Completeness of Data Base Sublanguages.* Research reports, Volume 987 de Research report. San José, California Research Laboratory : Computer sciences. IBM Corporation, 1972.

[Sampaio et al., 2008] M. C. Sampaio, F. H. B. Cardoso, G. P. dos Santos Jr. and L. Hattori. *Mining Disjunctive Association Rules.* 15 August 2008.

[Srikant et Agrawal, 1995] R. Srikant and R. Agrawal. *Mining Generalized Association Rules.* In Proc. of the 21st Int. Conf. on Very Large Databases, pp. 407-419, September 1995, Zurich, Switzerland.

[Xiangdan et al., 2005] H. Xiangdan, G. Junhua, S. Xueqin and Y. Weili. *Application of Data Mining in Fault Diagnosis Based on Ontology.* In Proc. of the 3rd Int. Conf. on Inform. Technology and Applicat., pp. 260-263, 2005, Washington, USA.

[Hamrouni et al., 2007] T. Hamrouni, I. Denden, S. Ben Yahia et E. Mephu Nguifo. *Étude expérimentale des représentations concises des itemsets fréquents.* Octobre, 2007, Tunis, Tunisie - Lens, France.

[Frawley et al., 1992] W. J. Frawley and G. Piatetsky-Shapiro and C. J. Matheus. *Knowledge Discovery in Databases : An Overview.* AI Magazine, vol. 13, no. 3, pp 57-70, 1992.

[Shekar et Natarajan, 2004] B. Shekar et R. Natarajan. *A Framework for Evaluating Knowledge-Based Interestingness of Association Rules.* Int J. of Fuzzy Optimization and Decision Making, vol. 3, no. 2, pp. 157-185, 2004.

[Hamrouni et al., 2010] T. Hamrouni, S. Ben Yahia et E. Mephu Nguifo. *Genralization of Association Rules through disjunction.* Ann. of Math. and Artificial Intell., vol .59, no. 2, pp 201-222, 2010.

[Hamrouni et al., 2014] T. Hamrouni and S. Ben Yahia and E. Mephu Nguifo. Towards Faster *Mining of Disjunction-Based Concise Representations of Frequent Patterns.* Int. J. on Artificial Intell. Tools, vol. 23, no. 2, 33 pages, 2014.

[Weiss, 2004] G.M. Weiss. *Mining with rarity : A unifying framework.* J. of ACM-SIGKDD Explorations Newsletter, vol. 6, no. 1, pp. 7-19, 2004.

[Terrapon, 2009] N. Terrapon, O. Gascuel, E. Marechal et L. Brehelin. *Detection of new protein domains using co-occurrence : application to Plasmodium falciparum.* J. of Bioinformatics, vol. 25, pp. 3077-3083, 2009.

[Codd, 1970] E. F. Codd. *A Relational Model of Data for Large Shared Data Banks.* Mag. Commun. of the ACM, vol. 13, no. 6, pp. 377-387, 1970.

[Wang et al., 2001] K. Wang, Y. He et D. W. Cheung. *Mining confident rules without support requirement.* In ACM Int. Conf. on Inform. and Knowledge Manage., pp. 89-96, 2001.

[Yun et al., 2003] H. Yun, D. Ha, B. Hwang et K. Ho Ryu. *Mining association rules on significant rare data using relative support.* J. of Syst. and Software, vol. 67, no. 3, pp. 181-191, 2003.

[Xiong et al., 2006] H. Xiong, P. N. Tan et V. Koumar. *Hyperclique pattern discovery.* J. of Data Mining and Knowledge Discovery, vol .13, no. 2, pp. 219-242, 2006.

[Omiecinski, 2003] E. R. Omiecinski. *Alternative interest measures for mining associations in databases.* J. of IEEE Trans. on Knowledge and Data Engineering, vol. 15, no. 1, pp. 57-69, 2003.

[Hilali-Jaghdam et al., 2011] I. Hilali-Jaghdam, T.Y. Jen, D. Laurent et S. Ben Yahia. *Mining frequent disjunctive selection queries.* In Proc. of the 22nd Int. Conf. on Database and Expert Syst. Applicat., Part II (DEXA'11). Springer pp. 90-96, 2011, Toulouse, France.

[Hilali-Jaghdam et al., 2012] I. Hilali-Jaghdam, T.Y. Jen, D. Laurent et S. Ben Yahia. *Fouille de règles d'association généralisées à travers les disjonctions : application aux données génomiques.* La 12 ème conférence int. francophone sur l'Extraction et la Gestion de Connaissances, EGC 2012, RNTI-E-23, Éditions Hermann, pp. 237-242, Janvier 2012, Bordeaux, France.

[Hilali-Jaghdam et al., 2013] I. Hilali-Jaghdam, T.Y. Jen , D. Laurent C. Marinica et S. Ben Yahia. *Mining Interesting Disjunctive Association Rules from Unfrequent Items.* Int. Workshop, ISIP 2013, Bangkok, Thailand, September 16-18, 2013. Information Search, Integration, and Personalization. Editeurs : Asanee Kawtrakul, Dominique Laurent, Nicolas Spyratos, Yuzuru Tanaka.

[Hilali-Jaghdam et al., 2014] I. Hilali-Jaghdam, T.Y. Jen , D. Laurent C. Marinica et S. Ben Yahia. *Mining Frequent and Homogeneous Closed Itemsets.* Int. Workshop, ISIP 2014, HELP College of Arts and Technology, Kuala Lumpur, October 9-10, 2014. Editeurs : Asanee Kawtrakul, Dominique Laurent, Nicolas Spyratos, Yuzuru Tanaka

[Toivonen, 1996] H. Toivonen. *Sampling Large Databases for Association Rules.* In Proc. of the 22nd Int. Conf. on Very Large Data Bases. pp. 134-145, 1996, Bombay, India.

[Kim, 2003] H. D. Kim. *Complementary occurrence and disjunctive rules for market analysis in data mining.* In Proc. of the 2nd IASTED Int. Conf. on Inform. and Knowledge Sharing. pp. 155-157, Novembre, 2003, Scottsdale, USA.

[Kim et Anyanwu] H. Kim et K. Anyanwu. *Scalable Ontological Query Processing over Semantically Integrated Life Science Datasets using MapReduce.* In Proc. of the Conf. on Semantics in Healthcare and Life Sciences. February, 26-28, 2014 Boston, MA. North Carolina State University, United States.

[Zelenko, 1999] D. Zelenko. *Optimizing Disjunctive Association Rules.* In Proc. of the 3rd European Conf. on Principles of Data Mining and Knowledge Discovery. Volume 1704 of the series Lecture Notes in Computer Science, pp. 204-213, 1999, Springer-Verlag, London, UK.

[Elble et al., 2003] J. Elble, C. Heeren et L. Pitt. *Optimized Disjunctive Association Rules via Sampling.* In Proc. of the 3rd IEEE Int. Conf. on Data Mining. pp. 43-50. IEEE Computer Society, Washington, DC, USA.

[Rajeev et Kyuseok, 1998] R. Rajeev et S. Kyuseok. *Mining Optimized Association Rules with Categorical and Numeric Attributes.* In Proc. of the 14th Int. Conf. on Data Eng., pp. 503-512, 1998, Orlando, Florida, USA.

[Nanavati et al., 2001] A. A. Nanavati, K. P. Chitrapura, S. Joshi et R. Krishnapuram. *Mining generalized disjunctive association rules.* In Proc. of the 10th Int. Conf. on Inform. and Knowledge Manage., pp. 482-489, 2001, Atlanta, Georgia, USA.

[Agrawal et al., 1993] R. Agrawal, T. Imielienski et A. Swami. *Mining association rules between sets of items in large databases.* In Proc. of the 1993 ACM SIGMOD Int. conf. on Manage. of data, pp. 207-216, 1993, Washington, D.C.

[Agrawal et Srikant, 1994] R. Agrawal et R. Srikant. *Fast algorithms for mining association rules.* In Proc. of the 20th Int. Conf. on Very Large Data Bases, pp.487-499, September, 1994, Santiago, Chile.

[Kamber et al., 1997] M. Kamber, J. Han et J. Chiang. *Metarule-guided mining of multidimensional association rules using data cubes.* In Proc. of the 3rd Int. Conf. on Knowledge Discovery and Data Mining, pp. 207-210, August, 1997, Newport Beach, California.

[Bykowski et Rigotti, 2001] A. Bykowski et C. Rigotti. *A condensed representation to find frequent patterns.* In Proc. of the 20th ACM SIGMOD-SIGACT-SIGART Symp. on Principles of database syst. pp. 267-273, 2001. Santa Barbara, California, United States.

[Casali et al., 2006] A. Casali, R. Cicchetti, L. Lakhal et S. Lopes. *Motifs essentiels et inférence des fréquences.* In Proc. of 20èmes Journées Bases de Données Avancées, pp. 535-554, 2004, Montpellier

[Casali et al., 2005] A. Casali, R. Cicchetti et L. Lakhal. *Essential Patterns : A Perfect Cover of Frequent Patterns.* In Proc. of the 7th Int. Conf. on Data Warehousing and Knowledge Discovery, pp. 428-437, 2005, Copenhagen, Denmark.

[Hamrouni et al., 2007a] T. Hamrouni, I. Denden, S. Ben Yahia, E. Mephu Nguifo et Y. Slimani. *Les itemsets essentiels fermés, une nouvelle représentation condensées.* In Proc. de la 8 ème Conférence Française en Extraction et Gestion de Connaissances, pp. 241-252, 2007, Namur Belgique.

[Hamrouni et al., 2007b] T. Hamrouni, I. Denden, S. Ben Yahia et E. Mephu Nguifo. *A New Concise Representation of Frequent Patterns through Disjunctive Search Space.* In Proc. of the 5th conf. on Concept Lattices and their Applicat., October 24-26, pp. 50-61, 2007, Montpellier, France.

[Hamrouni, 2007] T. Hamrouni. *Itemsets fréquents et règles d'association : extraction et réduction.* Novembre, 2007, LARPAH (Faculté des Sciences de Tunis), CRIL (Université d'Artois de Lens).

[Boulicaut et al., 2000 (a)] J-F. Boulicaut, A. Bykowski et C. Rigotti. *Approximation of frequency queries by means of Free-Sets.* In Proc. of the 4th European Conf. on Principles of Data Mining and Knowledge Discovery in Databases, September 13-16, 2000, pp. 75-85, 2000, Lyon, France.

[Boulicaut et al., 2000 (b)] J.-F. Boulicaut, A. Bykowski et B. Jeudy. *Towards the tractable discovery of association rules with negation.* In Proc. of the 4th Int. Conf. on Flexible Query Answering Syst., Advances in Soft Computing, October, 2000. pp. 425-434, Warsaw, Poland.

[Kryszkiewicz et Gajek, 2002] M. Kryszkiewicz et M. Gajek. *Why to apply generalized disjunction-free generators representation of frequent patterns ?* In Proc. of the 13th Int. Symp. on Foundations of Intelligent Syst., pp. 382-392, June 2002, Lyon, France.

[Jen et al., 2009] T. Y. Jen, D. Laurent et N. Spyratos. *Mining frequent conjunctive queries in star schemas.* In Proc. of the Int. Database Eng. and Applicat. Symp., pp.97-108, 2009, Cetraro - Calabria, Italy. ACM, New York, NY, USA.

[Fayyad et al., 1996] Usama M. Fayyad, G. Piatetsky-Shapiro et P. Smyth. *From Data Mining to Knowledge Discovery : An Overview.* Advances in Knowledge Discovery and Data Mining. pp.1-34, 1996.

[Liu et al., 1999] B. Liu, W. Hsu, et Y. Ma. *Mining association rules with multiple minimum supports.* In Proc. of the 5th Int. Conf. on Knowledge Discovery and Data Mining, pp. 337-341, 1999, San Diego, CA, USA.

[Tao et al., 2003] F. Tao, F. Murtagh et M. Farid. *Weighted association rule mining using weighted support and significance framework.* In Proc. of the 9th Int. Conf. on Knowledge Discovery and Data Mining, pp. 661-666, 2003, Washington, DC, USA.

[Wang et al. 2000] W. Wang, J. Yang et P. S. Yu. *Efficient mining of weighted association rules (WAR).* In Proc. of the 6th Int. Conf. on Knowledge Discovery and Data Mining, pp. 270-274, 2000, Boston, MA, USA.

[Wang et al. 2000] K. Wang, S. Zhou et Y. He. *Growing decision trees on support-less association rules.* In Proc. of the 6th Int. Conf. on Knowledge Discovery and Data Mining, pp. 265-269, 2000, Boston, MA, USA

[Xiong et al., 2003] H. Xiong, P. N. Tan et V. Koumar. *Mining strong affinity association patterns in data sets with skewed support distribution.* In Proc. of the 3rd IEEE Int. Conf. on Data Mining, pp. 387-394, 2003.

[Bouasker et al., 2008] S. Bouasker, T. Hamrouni et S. Ben Yahia. *New Exact Concise Representation of Rare Correlated Patterns : Application to Intrusion Detection.* In Proc. of the 16th Pacific-Asia conf. on Advances in Knowledge Discovery and Data Mining, PAKDD (2), pp. 61-72, 2012.

[Ben Younes et al., 2010] N. Ben Younes, T. Hamrouni et S. Ben Yahia. *Bridging conjunctive and disjunctive search spaces for mining a new concise and exact*

representation of correlated patterns. In Proc. of the 13th Int. conf. on Discovery Science, October 6-8, 2010, vol. 6332 of Lecture Notes in Computer Science, Springer, pp. 189-204, Canberra, Australia.

[Hussain et al., 2000] F. Hussain, H. Liu, E. Suzuki et H. Lu. *Exception rule mining with a relative interestingness measure.* In Proc. of the Pacific-Asia Conf. on Knowledge Discovery and Data mining, vol. 6118 of Lecture Notes in Computer Science, Springer, pp. 86-97, 2000.

[Han et Fu, 1995] J. Han et Y. Fu. *Discovery of multiple-level association rules from large databases.* In Proc. of the 21th Int. Conf. on Very Large Data Bases, pp. 420- 431, 1995.

[Maumus et al., 2006] S. Maumus, A. Napoli, L. Szathmary et Y. Toussaint. *Réflexions sur l'extraction des motifs rares.* Communication dans un congrés, 13 ièmes rencontres de la Socité Francophone de Classification, Metz, France. M. Nadif, F.-X. Jollois (editors), Presses Universitaires de Montréal, pp. 157-162, 2006.

[Szathmary et al., 2007] L. Szathmary, A. Napoli et P. Valtchev. *Towards Rare Itemset Mining.* In Proc. of 19 th Int. Conf. on Tools with Artificial Intell., October 29-31, vol. 1, IEEE Computer Society, pp. 305-312, 2007, Patras,Greece.

[Wan et Zeitouni, 2011] T. Wan et K. Zeitouni. *Mining Association rules with Multiple Min-supports- Application to Symbolic Data.* Student J. (ISSN 1420-1011), vol. 5, no. 1, pp. 59-74, 2004.

[Antonie et Zaïan, 2004] M-L. Antonie et O. R. Zaïan. *Mining positive and negative association rules : an approach for confined rules.* In Proc. of the 8th European Conf. on Principles and Practice of Knowledge Discovery in Databases pp. 27-38, 2004.

[Brin et al., 1997] S. Brin, R. Motwani et C. Silverstein. *Beyond market basket : Generalizing association rules to correlations.* In Proc. of SIGMOD, Int. Conf. of Manage. of Data, pp. 265-276, 1997.

[Silverstein et al., 1998] C. Silverstein, S. Brin et R. Motwani. *Beyond market basket : Generalizing association rules to dependence rules.* Int. J. of Data Mining and Knowledge Discovery, vol. 2, no. 1, pp. 39-68, 1998.

[Wu et al., 2002] X. Wu, C. Zhang et S. Zhang. *Mining both positive and negative association rules.* In Proc. of the 9th Int. Conf. on Machine Learning, pp. 658–665, 2002. Sydney, NSW, Australia.

[Wu et al., 2004] X. Wu, C. Zhang et S. Zhang. *Efficient mining both positive and negative association rules.* ACM Trans. on Inform. Syst., vol. 22, no. 3, pp. 381-405, July 2004.

[Teng et al., 2002] W. Teng, M. Hsieh et M. Chen. *On the mining of substitution rules for statistically dependent items.* In Proc. of the IEEE Int. Conf. on Data Mining, pp. 442-449, 2002.

[Savasere et al., 1998] A. Savasere, E. Omiecinski et S. Navathe. *Mining for strong negative associations in a large database of customer transactions.* In Proc. of the 18th Int. Conf. on Data Engineering, pp. 494-502, 1998, San Jose, CA, USA.

[Yuan et al., 2002] X. Yuan, B. Buckles, Z. Yuan et J. Zhang. *Mining negative association rules.* In Proc. of the 7th Int. Symp. on Comput. and Commun., pp. 623-629, 2002.

[Cornelis et al., 2006] C. Cornelis, P. Yan, X. Zhang et G. Chen. *Mining Positive and Negative Association Rules from Large Databases.* In Proc. of the IEEE Conf. on Cybernetics and Intell. Syst., 2006.

[Ramasubbareddy et al., 2010] B. Ramasubbareddy, A. Govardhan et A. Ramamohanreddy. *Mining indirect Positive and Negative Association Rules.* In Proc. of the IEEE Int. Conf. on Software Eng., August 2011, Hefeai, China.

[Mani, 2012] T. Mani. *Mining Negative Association Rules.* IOSR Journal of Computer Engineering (IOSRJCE) ISSN : 2278-0661 Volume (3), Issue (6), Sep-Oct, pp. 43–47, 2012.

[Fankam et al., 2009] C. Fankam, L. Bellatreche, D. Hondjack, Y. A. Ameur et G. Pierra. *SISRO, Conception de Bases de Données à partir d'Ontologies de Domaine.* Technique et Science Informatiques, vol. 28, no. 10, pp. 1-29, 2009.

[Rada et al., 1989] R. Rada, H. Mili, E. Bicknell et M. Blettner. *Development and application of a metric on semantic nets.* IEEE trans. on syst., man, and cybern., vol. 19, no. 1, pp. 17-30, 1989.

[Wu et Palmer, 1994] Z. Wu et M. Palmer. *Verb Semantics and Lexical Selection.* In Proc. of the 32nd Ann. Meetings of the Assoc. for Computational Linguistics, pp. 133-138, 1994.

[Slimani et al., 2007] T. Slimani, B. B. Yaghlane et K. Mellouli. *Une extension de mesure de similarité entre les concepts d'une ontologie.* In Proc. of the 4th Int. Conf. : Sci. of Electron., Technologies of Inform. and Telecommun., March 25-29, 2007, TUNISIA.

[Resnik et al., 1995] P. Resnik. *Using information content to evaluate semantic similarity in a taxonomy.* In Proc. of the Int. Joint Conf. on Artificial Intell., pp. 448-453, 1995.

[Jiang et Conrath, 1997] J. Jiang et D.W. Conrath. *Semantic Similarity based on Corpus Statistics and Lexical Taxonomy.* In Proc. of the Int. Conf. on Research in Computational Linguistics, 1997, Taiwan.

[Lin, 1998] D. Lin. *An information-theoric definition of similarity.* In Proc. of the 15th Int. Conf. on Mach. Learning, pp. 296-304, 1998.

[Leacock et Chodorow, 1998] C. Leacock et M. Chodorow. *Combining Local Context and WordNet Similarity for Word Sense Identification.* In WordNet : An Electron. Lexical Database, C. Fellbaum, MIT Press, 1998.

[Resnik, 1999] P. Resnik. *Semantic similarity in a taxonomy : An information based measure and its application to problems of ambiguity in natural language.* J. of Artificial Intell. Research, 11, pp. 95-130, 1999.

[Salton et McGill, 1983] G. Salton et M. J. McGill. *Introduction to modern information retrieval.* McGraw-Hill, Inc. New York, NY, USA 1986.

[Baeza-Yates et Ribeiro-Neto, 1999] R. Baeza-Yates et B. Ribeiro-Neto. *Modern Information Retrieval.* ACM Press/ Addison-Wesley : New York ; Harlow, England ; Reading, Mass., 1999.

[Marinica et Guillet, 2010] C. Marinica et F. Guillet. *Knowledge-based interactive post-mining of association rules using ontologies.* IEEE Trans. on Knowledge and Data Engineering, vol. 22, no. 6), pp. 784-797, 2010.

[Radhika et Vidya, 2012] N. Radhika et K. Vidya. *Association Rule Mining based on Ontological Relational Weights.* Int. J. of Scientific and Research Publications, vol. 2, no.1, January 2012.

[Domingues et Rezende, 2011] M. A. Domingues et S. O. Rezende. *Using Taxonomies to Facilitate the Analysis of the Association Rules.* CoRR, December, 2011.

[Mansingh et al., 2010] G. Mansingh, K. M. O. Bryson et H. Reichgelt. *Using ontologies to facilitate post-processing of association rules by domain experts.* Int. J. of Inform. Sci., vol. 181, no. 3, pp. 419-434.

[Escovar et al., 2005] E. L. G. Escovar, M. Biajiz et M. T. P. Vieira. *SSDM : A Semantically Similar Data Mining Algorithm.* In Proc. of the 20th Simpósio Brasileiro de Bancos de Dados, de 3-7 Outubro, 2005, pp. 265-279, Uberlândia, MG, Brazil, Anais.

[Escovar et al., 2006] E. L. G. Escovar, C. A. Yaguinuma et M. Biajiz. *Using Fuzzy Ontologies to Extend Semantically Similar Data Mining.* In Proc. of the 21st Brazilian Symp. of Databases, Florianópolis, Brazil, pp. 16-30, 2006.

[Miani et al., 2009] R. G. Miani, C. A. Yaguinuma, M. T. P. Santos et M. Biajiz. *NARFO Algorithm : Mining Non-redundant and Generalized Association Rules Based on Fuzzy Ontologies.* In Proc. of the 11th Int. Conf. on Enterprise Inform. Syst., pp. 415-426, 2009. Milan, Italy.

[Bellandi et al., 2007] A. Bellandi, B. Furletti, V. Grossi et A. Romei. *Ontology-driven association rule extraction : A case study.* In Proc. of the Int. Workshop on Context and Ontologies : Representation and Reasoning, pp. 1-10, 2007.

[Bellandi et al., 2008] A. Bellandi, B. Furletti, V. Grossi et A. Romei. *Ontological support for association rule mining.* In Proc. of the 26th IASTED Int. Conf. on Artificial Intell. and Applicat., ACTA Press, pp. 110-115, 2008.

[Brisson, 2006] L. Brisson. *Knowledge extraction using a conceptual information system (excis).* In Proc. of Ontologies-Based Databases and Informat. Syst. Workshop in VLDB Conf., pp. 119-134, 2006.

[Brisson et Collard, 2009] L. Brisson et M. Collard. *How to Semantically Enhance a Data Mining Process ?* Chapter Enterprise Information Systems, vol. 19, pp. 103-116, 2009. Springer Berlin Heidelberg.

[Zeman et al., 2009] M. Zeman , V. Svátek et J. Rauch. *Ontology-Driven Data Preparation for Association Mining.* Online http ://keg.vse.cz/onto-kdd-draft.pdf.

[Ferraz et Garcia, 2008] I. N. Ferraz et A. C. B. Garcia. *Ontology in Association Rules Pre-Processing and post-processing.* In Proc. of the 3rd Int. Conf. on Informat. Technology and Applicat, pp. 87-91, 2008, Washington, USA.

[Ferraz et Garcia, 2013] I. N. Ferraz et A. C. B. Garcia. *Ontology in association rules.* SpringerPlus, vol. 2, 2013.

[Subashchadrabose et Sivakumar, 2013] R. Subashchadrabose et R. Sivakumar. *Post-mining of association rules using Ontologies and rule schemas.* Int. J. of Comput. and Communicat. Technology, ISSN (ONLINE) : 2231 - 0371, vol. 4, no. 1, pp. 33-37, 2013.

[Cannataro et Comito, 2003] M. Cannataro et C. Comito. *A data mining ontology for grid programming.* In Proc. of the 1st Int. Workshop on Semantics in Peer-to-Peer and Grid Computing, 2003.

[Ralbovsky et Kuchar, 2007] M. Ralbovsky et T. Kuchar. *Using disjunctions in association mining.* In Proc. of the 7th Industrial Conf. on Advances in Data Mining : theoretical aspects and applicat., pp. 339-351, 2007.

[Zhao et al., 2006] L. Zhao, M.J. Zaki et N. Ramakrishnan. *BLOSOM : A framework for mining arbitrary Boolean expressions.* In Proc. of the 12th ACM Int. Conf. on Knowledge Discovery and Data Mining, pp. 827-832, 2006, Philadelphia, PA, USA.

[Szathmary et al., 2006] L. Szathmary, S. Maumus, P. Pierre, Y. Toussaint et A. Napoli. *Vers l'extraction de motifs rares.* In Proc. of Extraction et gestion des connaissances, pp. 499-510, 2006, Lille, France. G. Ritschard, C. Djeraba (editors), RNTI-E-6, Cépaduès-Éditions Toulouse.

[Szathmary et al., 2010] L. Szathmary, P. Valtchev et A. Napoli. *Finding Minimal Rare Itemsets and Rare Association Rules.* In Proc. of 4th Int. Conf. on Knowledge Sci., Eng. and Manage., pp. 16-27, 2010. Belfast, Northern Ireland, UK, September 1-3, 2010.

[Mannila et Toivonen, 1996] H. Mannila et H. Toivonen. *Multiple uses of frequent sets and condensed representations (extended abstract).* In Proc. of the Second Int. Conf. on Knowledge Discovery and Data Mining, pp. 189-194, 1996, Portland, Oregon, USA.

[Mannila et Toivonen, 1997] H. Mannila et H. Toivonen. *Levelwise search and borders of theories in knowledge discovery.* In Proc. of the 3rd Int. Conf. on Data Mining Knowledge Discovery, pp. 241-258, 1997.

[Pillai et Vyas, 2011] J. Pillai et O.P. Vyas. *High Utility Rare Itemset Mining (HURI) : An approach for extracting high-utility rare itemsets.* J. on Future Eng. and Technology, vol. 7, no. 1, pp. 25, August-October 2011.

[Pillai et al., 2012] J. Pillai, O. P. Vyas et M K. Muyeba. *HURI - A Novel Algorithm for Mining High Utility Rare Itemsets.* In Proc. of the 2nd Int. Conf. on Advances in Computing and Informat. Technology, July 13-15, 2012, Chennai, India - vol. 2, pp. 531-540.

[Pillai et Vyas, 2014] J. Pillai et O.P. Vyas. *A Naïve Approach to High Utility Rare Itemset Mining Algorithm using Temporal Concept-THURI.* Int. J. of Advanced Research in Comput. and Communicat. Eng., vol. 3, no. 3, March 2014.

[Koh et Rountree, 2005] Y. S. Koh et N. Rountree. *Finding Sporadic Rules Using Apriori-Inverse.* In Proc. of the 9th Pacific-Asia conf. on Advances in Knowledge Discovery and Data Mining, Hanoi, Vietnam, May 18-20, pp. 97-106, 2005.

[Tsang et al., 2013] S. Tsang, Y. S. Koh et G. Dobbie. *Finding Interesting Rare Association Rules Using Rare Pattern Tree.* Trans. on Large Scale Data and Knowledge Centered Syst. VIII Lecture Notes in Computer Science, vol. 7790, 2013, pp. 157-173.

[Yun et al., 2003] H. Yun, D. Ha, B. Hwang et K. H. Ryu. *Mining association rules on significant rare data using relative support.* The J. of Syst. and Software, vol. 67, no. 3, pp. 181-191, 2003.

[Cruse, D. A., 1986] D. A. Cruse. *Lexical Semantics.* Cambridge University Press, Cambridge, UK, 1986.

[Cassel, J., 2011] J. Cassel. *What are Taxonomies ? Taxonomies ? That's classified information.* Avril 2011.

[Zarri et al., 1999] G. P. Zarri, Bertino E., Black B., Brasher A., Catania B., Deavin D., Di Pace L., Esposito F., Leo P., McNaught J., Persidis A., Rinaldi F. et Semeraro G.

J. Cassel. *CONCERTO, An Environment for the "Intelligent" Indexing, Querying and Retrieval of Digital Documents.* In Proc. of the 11th Int. Symp. on Found. of Intell. Syst., pp. 226-234, 1999, Warsaw, Poland.

[Sharman et al., 1999] R. Sharman, R. Kishore and R. Ramesh. *Ontologies : A Handbook of Principles, Concepts and Applications in Information Systems.* Springer, 2007.

[Vimieiro et Moscato, 2012] R. Vimieiro, and P. Moscato.

Mining disjunctive minimal generators with TitanicOR. Expert Syst. with Applicat., vol. 39, no. 9, pp. 8228-8238, July 2012.

[Mitchell, 1981] T. M. Mitchell. *Generalization as Search.* In Webber, B. L. et Nilsson, N. J., éditeurs : Readings in Artificial Intell., pp. 517-542, Morgan Kaufmann, Los Altos.

[Bastide et al., 1981] Y. Bastide, R. Taouil, N. Pasquier, G. Stumme, and L. Lakhal. *Levelwise search of frequent patterns with counting inference.* In 16èmes journées de bases de données avancées, 2000.

[Candolle, 1813] A. P. de Candolle. *Traité élémentaire de la botanique.* 1813.

[Budanitsky et Hirst, 2006] A. Budanitsky and G. Hirst. *Evaluating WordNet-based Measures of Lexical Semantic Relatednes.* Computational Linguistics, vol. 32, no. 1.

[Rezgui et al., 2013] K. Rezgui, H. Mhiri and K. Ghédira. *Theoretical Formulas of Semantic Measure : A Survey.* J. of Emerging Technologies in Web Intell., vol. 5, no. 4, November 2013.

[Zhong et al., 2002] J. Zhong, H. Zhu, J. Li, and Y. Yu. *Conceptual graph matching for search.* In Proc. of the 10th Int. Conf. on Conceptual Structures : Integration and Interfaces, London, UK : Springer-Verlag, 2002.

[De Morgan, 1847] De Morgan Augustus. *Formal Logic : or, The Calculus of Inference, Necessary and Probable.* London : Taylor and Walton, 1847.

[Han et al., 2007] J. Han, J. Pei, Y. Yin and R. Mao. *Mining frequent patterns without candidate generation : a frequent-pattern tree approach.* Data Mining and Knowledge Discovery, January 2004, vol. 8, no. 1, pp. 53-87.

[Zaki, 2000] Mohammed J. Zaki. *Scalable algorithms for association mining.* IEEE Trans. on Knowledge and Data Eng., vol. 12, no. 3, pp. 372-390, May 2000.

[Saint-Dizier, 2006] Saint-Dizier. Patrick. *Taxonomie.* In D. Godard, L. Roussarie et F. Corblin (éd.), Sémanticlopédie : dictionnaire de sémantique, GDR Sémantique et Modélisation, CNRS, `http://www.semantique-gdr.net/dico/`.

[Loubna, 2012] L. Kouki. *Extraction de connaissances.* École des Sciences de l'Information, CIS, Maroc.

[Fortuna et al., 2006] B. Fortuna, M. Grobelnik, and D. Mladenic. *Instructions for Ontogen 2.0.* Récupéré de `http://analytics.ijs.si/~blazf/ontogen/OntoGen2Help-2006-09-11.pdf`.

[Hamrouni et al., 2008] T. Hamrouni, S. Ben Yahia and E. Mephu Nguifo. *Succinct minimal generators : Theoretical foundations and applications.* International journal of foundations of computer science, vol 19, no 02, World Scientific, 2008, pp. 271–296.

[Hamrouni et al., 2008] T. Hamrouni, S. Ben Yahia and E. Mephu Nguifo. *Succinct system of minimal generators : A thorough study, limitations and new definitions.* Concept Lattices and Their Applications, Springer., 2008, pp. 80–95.

[Gasmi et al., 2007] G. Gasmi, S. Ben Yahia, E. Mephu Nguifo and S. Bouker. *Extraction of association rules based on literalsets.* International Conference on Data Warehousing and Knowledge Discovery, Springer, 2007, pp. 293–302.

[Gasmi et al., 2007] G. Gasmi, S. Ben Yahia, E. Mephu Nguifo and S. Bouker. *Extraction of association rules based on literalsets.* International Conference on Data Warehousing and Knowledge Discovery, Springer, 2007, pp. 293–302.

[Bouker et al., 2012] S. Bouker, R. Saidi, S. Ben Yahia, and E. Mephu Nguifo. *Ranking and selecting association rules based on dominance relationship.* Tools with Artificial Intelligence (ICTAI), 1 , IEEE, 2012, pp. 658–665.

[Ayouni et al., 2011] S. Ayouni, S. Ben Yahia and A. Laurent. *Extracting compact and information lossless sets of fuzzy association rules.* Fuzzy Sets and Systems, vol 183, no 1 , Elseiver, 2011, pp. 1-25.

[Ben yahia and Nguifo, 2004] S. Ben Yahia and E. Mephu Nguifo. *Emulating a Cooperative Behavior in a Generic Association Rule Visualization Tool.* 16th International Conference on Tools with Artificial Intelligence ICTAI, 110, CEUR-WS.org, 2004.

[Ben yahia and Nguifo, 2004] S. Ben Yahia and E. Mephu Nguifo. *Revisiting generic bases of association rules.* International Conference on Data Warehousing and Knowledge Discovery, Springer, 2004, pp. 58–67.

[Hamdi et al., 2013] S. Hamdi, A. Bouzeghoub, A. Lopes Gancarski and S. Ben Yahia. *Trust inference computation for online social networks.* Trust, Security and Privacy in Computing and Communications (TrustCom), IEEE, 2013, pp. 210–217.

[Brahmi et al., 2012] H. Brahmi, I. Brahmi and S. Ben Yahia. *OMC-IDS : at the crossroads of OLAP mining and intrusion detection.* Pacific-Asia Conference on Knowledge Discovery and Data Mining PAKDD, Springer, 2012, pp. 13–24.

[Jelassi et al., 2014] M. Nidhal Jelassi, C. Largeron, and S. Ben Yahia. *Efficient unveiling of multi-members in a social network.* Journal of Systems and Software vol 194, Elsevier, 2014, pp. 30–38.

[Ferjani et al., 2012] F. Ferjani, S. Elloumi, A. Jaoua, S. Ben Yahia, S. Ismail and S. Sheikha. *Formal context coverage based on isolated labels : An efficient solution for text feature extraction.* Information Sciences, vol 188, Elsevier, 2012, pp. 198–214.

[Bouzouita et al., 2006] I. Bouzouita, S. Elloumi and S. Ben Yahia. *GARC : A new associative classification approach.* International Conference on Data Warehousing and Knowledge Discovery, Springer, 2006, pp. 554–565.

Résumé

L'**E**xtraction de **C**onnaissances à partir de bases de **D**onnées vise à exploiter des grandes masses de données afin d'en extraire des connaissances nouvelles et utiles. La fouille de données, l'étape cœur de **ECD**, regroupe un ensemble de techniques telles que : le clustering, la classification, l'extraction de règles d'association, etc. Notre contribution se situe dans le cadre de l'extraction de règles d'association. Contrairement aux approches traditionnelles, nous nous intéressons aux items non fréquents à partir desquels des itemsets appelés disjonctifs-fréquents sont construits (un itemset est disjonctif-fréquent si le nombre de transactions contenant au moins de ses éléments est supérieur à un seuil fixé). En plus, afin de limiter le nombre de motifs extraits, nous supposons une ontologie définie sur l'ensemble de tous les items. Cette ontologie permet de définir une mesure d'homogénéité sur les itemsets et ainsi, de ne considérer que les itemsets disjonctifs-fréquents dont la mesure d'homogénéité est supérieure à un seuil donné. Enfin, les itemsets disjonctifs-fréquents homogènes sont utilisés pour la construction de règles d'association. Nos algorithmes ont été testés sur différents jeux de données, notamment sur des données réelles.

Mots-clés: itemset disjonctif-fréquent minimal, itemsets homogènes, règles d'association disjonctives intéressantes.

Abstract

Knowledge **D**iscovery in **D**atabases aims to exploit hudge volume of data to extract new and potentially useful knowledge. Data mining, the fundamental step of **KDD**, is built around a set of techniques such as clustering, classification. Our contribution concerns the extraction of association rules. More precisely, and contrary to standard approaches, we are interested in unfrequent items from which itemsets that we call disjunctive-frequent are built up (an itemset is said to be disjunctive-frequent if the number of transactions containing at least one of its elements is greater than a given threshold). Moreover, in order to restrict the number of mined patterns, we assume that an ontology is defined over the set of all items. Based on this ontology, we define a homogeneity measure over itemsets, so as to consider only those disjunctive-frequent itemsets whose homogeneity measure is above a given threshold. In this framework, we have designed and implemented algorithms for mining these patterns. These algorithms have been tested on various datasets, either synthetic or real.

Keywords: minimal disjunctive-frequent itemset, homogneous itemsts, disjunctive interesting association rules.