MINING CONCISE REPRESENTATIONS OF FREQUENT PATTERNS THROUGH CONJUNCTIVE AND DISJUNCTIVE SEARCH SPACES

Tarek Hamrouni

UNIVERSITY OF TUNIS EL MANAR
FACULTY OF SCIENCES OF TUNIS

UNIVERSITY OF ARTOIS
CRIL-CNRS, LENS

Thesis

Defended to obtain the Diploma of
Doctor in Computer Science

Elaborated by:

Tarek HAMROUNI
(Master in Computer Science, Faculty of Sciences of Tunis)

MINING CONCISE REPRESENTATIONS OF FREQUENT PATTERNS THROUGH CONJUNCTIVE AND DISJUNCTIVE SEARCH SPACES

Thesis Committee

Reviewers:	Prof. Habib OUNELLI	University of Tunis El-Manar, Tunis, Tunisia (***Chair***)
	Prof. Marzena KRYSZKIEWICZ	University of Technology, Warsaw, Poland
Examiners:	Prof. Salem BENFERHAT	University of Artois, Lens, France
	Associate Prof. Sadok BEN YAHIA	University of Tunis El-Manar, Tunis, Tunisia
Supervisors:	Prof. Khaled BSAÏES	University of Tunis El-Manar, Tunis, Tunisia
	Prof. Engelbert MEPHU NGUIFO	University of Blaise Pascal, Clermont Ferrand 2, France

August 4th, 2009

Acknowledgements

I would like to thank the many people that contributed to the realization of this thesis.

I would like to thank my thesis committee. Foremost, I express my gratitude to Professor Habib Ounelli to have agreed to chair the evaluating committee of my thesis.

I would like to thank Professor Marzena Kryszkiewicz and Professor Habib Ounelli for reading through drafts of this thesis, and providing me with valuable insights and comments. I also thank Professor Salem Benferhat for agreeing to participate to the thesis committee.

I would like to thank Professor Khaled Bsaïes for accepting to co-supervise my thesis. I am very thankful for his generosity and for being always present for helping me. I also express my sincere gratitude to Associate Professor Sadok Ben Yahia and Professor Engelbert Mephu Nguifo for their guidance and assistance during my doctoral studies. Their advices and research experiences have helped to always go on in this field. Now, I am grateful that I have chosen this exciting research field for my work.

I am also grateful to Professor Petko Valtchev for insightful discussions and comments about my work during my five-month visit to the UQÀM (Montréal, Canada), financed by the Merit scholarship program for foreign students managed by the "Fonds québécois de la recherche sur la nature et les technologies (FQRNT)". Collaboration with him resulted in an important part of Chapter 4. I am very grateful for this opportunity.

One of the main contributions in this thesis – Chapter 6 – was partly carried out thanks to a joint work with Islem Denden. I thank him for all the efforts he did for improving the quality of our work. I also really appreciate Nassima Ben Younes's support throughout the implementation of the tool used in Chapter 7.

I would like to thank researchers who kindly provided us the source codes of their algorithms which helped us in the evaluation of our contributions: thanks then to Toon Calders and Bart Goethals for providing the source code of the NDI algorithm, Alain Casali and Stéphane Lopes for providing the source code of the MEP algorithm, Guozhu Dong and Chunyu Jiang for providing the source code for the SSMG_Miner algorithm, Juho Muhonen and Hannu Toivonen for providing the source code of the FIRM algorithm, and Takeaki Uno for providing the source code of the LCM algorithm.

I would also thank Jean-Jacques Givry for English proofreading of a previous version of the thesis.

During my thesis, I have been a teaching contract employee at the Faculty of Sciences of Tunis. In this respect, I would like to thank all the employees of the Faculty of Sciences of Tunis and especially those of the Computer Science Department who have given me unconditional supports during these years.

My sincere thanks go to Madam Chiraz Latiri Cherif and Mister Sami Zghal to have had the encouraging words which helped me during some difficult periods.

I am also grateful for my friends for funny times we have spent together, ..., and most of all for just being there for me when I have needed them.

Last, but not least, I would like to express my gratitude to my family. I really don't quite know how to thank my parents. They have consistently and faithfully supported me through all my study periods both good and bad. They have been always there when I have needed them and they have been great encouragers. I would like to thank my brothers and my sister for always supporting me, for their love and strength they are giving to me all the time in my life. Without the devotion and support of my family, this project would not have been possible. I will never forget their support as long as I live.

To my Parents for their endless love and support throughout my life...

Abstract

The last years witnessed an explosive progress in networking, storage, and processing technologies resulting in an unprecedented amount of digitalization of data. There is hence a considerable need for tools or techniques to delve and efficiently discover valuable, non-obvious information from large databases. In this situation, Knowledge Discovery in Databases offers a complete process for the non-trivial extraction of implicit, previously unknown, and potentially useful knowledge from data. Amongst its steps, data mining offers tools and techniques for such an extraction. Much research in data mining from large databases has focused on the discovery of association rules which are used to identify relationships between sets of items in a database. The discovered association rules can be used in various tasks, such as depicting purchase dependencies, classification, medical data analysis, etc. In practice however, the number of frequently occurring itemsets, used as a basis for rule derivation, is very large, hampering their effective exploitation by the end-users. In this situation, a determined effort focused on defining manageably-sized sets of patterns, called concise representations, from which redundant patterns can be regenerated. The purpose of such representations is to reduce the number of mined patterns to make them manageable by the end-users while preserving as much as possible the hidden and interesting information about data.

Many concise representations for frequent patterns were so far proposed in the literature, mainly exploring the conjunctive search space. In this space, itemsets are characterized by the frequency of their co-occurrence. A detailed study proposed in this thesis shows that closed itemsets and minimal generators play a key role for concisely representing both frequent itemsets and association rules. These itemsets structure the search space into equivalence classes such that each class gathers the itemsets appearing in the same subset (*aka* objects or transactions) of the given data. A closed itemset includes the most specific expression describing the associated transactions, while a minimal generator includes one of the most general expressions. However, an intra-class combinatorial redundancy would logically results from the inherent absence of a unique minimal generator associated to a given closed itemset. This motivated us to carry out an in-depth study aiming at only retaining irreducible minimal generators in each equivalence class, and pruning the remaining ones. In this respect, we propose lossless reductions of the minimal generator set thanks to a new substitution-based process. We then carry out a thorough study of the associated properties of the obtained families. Our theoretical results will then be extended to the association rule framework in order to reduce as much as possible the number of retained rules without information loss. We then give a thorough formal study of the related inference mechanism allowing to derive all redundant association rules, starting from the retained ones. In order to validate our approach, computing means for the new pattern families are presented together with empirical evidences about their relative sizes *w.r.t.* the entire sets of patterns.

We also lead a thorough exploration of the disjunctive search space, where itemsets are characterized by their respective disjunctive supports, instead of the conjunctive ones. Thus, an itemset verifies a portion of data if at least one of its items belongs to it. Disjunctive itemsets thus convey knowledge about complementary occurrences of items in a dataset. This exploration is motivated by the fact that, in some applications, such information – conveyed through disjunctive support – brings richer knowledge to the end-users. In order to obtain a redundancy-free representation of the disjunctive search space, an interesting solution consists in selecting a unique element to represent itemsets covering the same set of data. Two itemsets are equivalent if their respective items cover the same set of data. In this regard, we introduce a new operator dedicated to this task. In each induced equivalence class, minimal elements are called essential itemsets, while the largest one is called disjunctive closed itemset. The introduced operator is then at the roots of new concise representations of frequent itemsets. We also exploit the disjunctive search space to derive generalized association rules. These latter rules generalize classic ones to also offer disjunction and negation connectors between items, in addition to the conjunctive one. Dedicated tools were then designed and implemented for extracting disjunctive itemsets and generalized association rules. Our experiments showed the usefulness of our exploration and highlighted interesting compactness rates.
Keywords: Association rule, Closed itemset, Closure operator, Concise representation, Data mining, Disjunctive closed itemset, Disjunctive support, Equivalence class, Essential itemset, Generalized association rule, Itemset, Minimal generator.

Résumé

Durant ces dernières années, les quantités de données collectées, dans divers domaines d'application de l'informatique, deviennent de plus en plus importantes. Cela suscite le besoin d'analyser et d'interpréter ces données afin d'en extraire des connaissances utiles. Dans cette situation, le processus d'Extraction de Connaissances à partir des Données est un processus complet visant à extraire des connaissances cachées, nouvelles et potentiellement utiles à partir de grands volumes de données. Parmi ces étapes, la fouille de données offre les outils et techniques permettant une telle extraction. Plusieurs travaux de recherche en fouille de données concernent la découverte des règles d'association, permettant d'identifier des liens entre ensembles de descripteurs (ou attributs ou items) décrivant un ensemble d'objets (ou individus ou transactions). Les règles d'association ont montré leur utilité dans plusieurs domaines d'application tels que la gestion de la relation client en grande distribution (analyse du panier de la ménagère pour déterminer les produits souvent achetés simultanément, et agencer les rayons et organiser les promotions en conséquence), la biologie moléculaire (analyse des associations entre gènes), etc.

De manière générale, la construction des règles d'association s'effectue en deux étapes : l'extraction des ensembles d'items (ou itemsets) fréquents, puis la génération des règles d'association à partir de des itemsets fréquents. Dans la pratique, le nombre de motifs (itemsets fréquents ou règles d'associations) extraits ou générés, peut être très élevé, ce qui rend difficile leur exploitation pertinente par les utilisateurs. Pour pallier ce problème, certains travaux de recherche proposent l'usage d'un noyau de motifs, appelés représentations concises, à partir desquels les motifs redondants peuvent être régénérés. Le but de telles représentations est de condenser les motifs extraits tout en préservant autant que possible les informations cachées et intéressantes sur des données.

Dans la littérature, beaucoup de représentations concises des motifs fréquents ont été proposées, explorant principalement l'espace de recherche conjonctif. Dans cet espace, les itemsets sont caractérisés par la fréquence de leur co-occurrence. Ceci fait l'objet de la première partie de ce travail. Une étude détaillée proposée dans cette thèse prouve que les itemsets fermés et les générateurs minimaux sont un moyen de représenter avec concision les itemsets fréquents et les règles d'association. Les itemsets fermés structurent l'espace de recherche dans des classes d'équivalence tels que chaque classe regroupe les itemsets apparaissant dans le même sous-ensemble (appelé aussi objets ou transactions) des données. Un itemset fermé inclut l'expression la plus spécifique décrivant les transactions associées, alors qu'un générateur minimal inclut une des expressions les plus générales. Cependant, une redondance combinatoire intra-classe résulte logiquement de l'absence inhérente d'un seul générateur minimal associé à un itemset fermé donné. Ceci nous a motivé à effectuer une étude approfondie visant à maintenir seulement les générateurs minimaux irréductibles dans chaque classe d'équivalence, et d'élaguer les autres. À cet égard, il est proposé une réduction sans perte d'information de l'ensemble des générateurs minimaux grâce à un nouveau processus basé sur la substitution. Une étude complète des propriétés associées aux familles obtenues est présentée. Les résultats théoriques sont ensuite étendus au cadre de règles d'association afin de réduire autant que possible le nombre de règles maintenues sans perte d'information. Puis, est présentée une étude formelle complète du mécanisme d'inférence permettant de dériver toutes les règles d'association redondantes, à partir de celles maintenues. Afin de valider l'approche proposée, les algorithmes de construction de ces représentations concises de motifs sont présentés ainsi que les résultats des expérimentations réalisées en terme de concision et de temps de calcul.

La seconde partie de ce travail est consacrée à une exploration complète de l'espace de recherche disjonctif des itemsets, où ceux-ci sont caractérisés par leurs supports disjonctifs. Ainsi dans l'espace disjonctif, un itemset vérifie une transaction si au moins un de ses items y est présent. Les itemsets disjonctifs véhiculent ainsi une connaissance au sujet des occurrences complémentaires d'items dans un ensemble de données. Cette exploration est motivée par le fait que, dans certaines applications, une telle information peut être utile aux utilisateurs. Lors de l'analyse d'une séquence génétique par exemple, le fait d'engendrer une information telle que "présence d'un gène X ou la présence d'un gène Y ou ..." présente un intérêt pour le biologiste.

Afin d'obtenir une représentation concise de l'espace de recherche disjonctif, une solution intéressante consiste à choisir un seul élément pour représenter les itemsets couvrant le même ensemble de données. Deux itemsets sont équivalents si leurs items respectifs couvrent le même ensemble de données. À cet

égard, un nouvel opérateur consacré à cette tâche, a été introduit. Dans chaque classe d'équivalence induite, les éléments minimaux sont appelés itemsets essentiels, alors que le plus grand élément est appelé itemset fermé disjonctif. L'opérateur présenté est alors à la base de nouvelles représentations concises des itemsets fréquents. L'espace de recherche disjonctif est ensuite exploité pour dériver des règles d'association généralisées. Ces dernières règles généralisent les règles classiques pour offrir également des connecteurs de disjonction et de négation d'items, en plus de celui conjonctif. Des outils (algorithme et programme) dédiés ont été alors conçus et mis en application pour extraire les itemsets disjonctifs et les règles d'association généralisées. Les résultats des expérimentations effectuées ont montré l'utilité de notre exploration et ont mis en valeur la concision des représentations concises proposées.
Mots clés : Fouille de données, Classe d'équivalence, Itemset, Itemset essentiel, Itemset fermé, Itemset fermé disjonctif, Générateur minimal, Opérateur de fermeture, Règle d'association, Règle d'association généralisée, Représentation concise, Support disjonctif.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

With the development of computer tools, we noted these last years a flood of information stored in large databases [Berry and Linoff., 2004]. The need to interpret and analyze these data raises much interest. Thus, the setting of new data analysis solutions became a real challenge for the scientific community. To overcome the lack of extracted knowledge from stored data, new methods were hence proposed, gathered under the generic term of Knowledge Discovery in Databases (KDD) [Fayyad *et al.*, 1996, Han and Kamber, 2006]. According to Frawley *et al.* [Frawley *et al.*, 1992]: "The Knowledge Discovery in Databases indicates the interactive and iterative process for extracting implicit knowledge, previously unknown and potentially useful starting from stored data in databases".

Within a KDD process, the data mining is the step focusing on the mining part of interesting patterns. For this purpose, this multidisciplinary field is at the confluence of various others, such as statistics, database management, mathematics, artificial intelligence, etc. In few years, the data mining became a research field in full progress aiming at exploiting the great quantities of data collected in various domains using computer sciences. The term "data mining" gathers different complementary tasks, such as prediction, grouping by similarity, classification, cluster analysis, etc. These tasks are divided into several techniques, such as association rules, decision trees, neural networks, etc. [Han and Kamber, 2006].

In this thesis, we are interested in two pattern classes, namely *frequent itemsets* and *association rules*. In data mining, frequent itemsets and association rules are among the most popular research topics [Han *et al.*, 2007]. These pattern classes are closely related since the extraction of the former is usually considered as a starting point for getting the latter. Association rule mining is a fundamental topic in data mining and has been extensively investigated since its inception in [Agrawal *et al.*, 1993]. Its key idea consists in looking for relationships between sets of items, commonly called *itemsets*, where the presence of some items suggests that others follow from them. A typical example of a successful application of association rules was the market basket analysis [Agrawal *et al.*, 1993], where the discovered rules can lead to important marketing and management strategic decisions. In this case, each transaction consists of a list of bought articles (or products). The purpose was to identify the groups of articles frequently bought together. The association analysis, applied to sales transactions, is then called *market basket analysis* [Agrawal *et al.*, 1993]. It starts from the finest data which compose a transaction: sales of the elementary articles. The mining of associations then aims at finding relations or correlations which could

exist between products (for example, **80**% of customers who buy tomato and salad also buy oil.), but also between product sales (for example, when the sales of milk increase then the sales of chocolate increase with a confidence of **60**%).

The use of association rules was then extended to various applications analyzing for example economic, financial, or medical data. In the general case, given a set of items (or attribute) and a set of objects (or transactions), the frequent pattern mining problem consists of getting out, from a dataset, patterns having a number of occurrences (*i.e.*, conjunctive support or support for short) greater than or equal to a user-defined threshold. An association rule is defined under the general form: *If Condition*(s), *then Result*(s) where the *Condition*(s) and *Result*(s) parts are composed by sets of items from the dataset.

In practice, the number of frequent itemsets, and hence association rules, can be overwhelmingly large hampering their effective exploitation by the end-users. In order to reduce the number of mined rules, statistical measures were introduced, amongst the most known are the *support* and *confidence* [Geng and Hamilton, 2006]. Nevertheless, if the minimal support threshold is set too low or the data is highly correlated, no matter how efficient the frequent pattern mining algorithm is, generating all frequent patterns is impossible. Moreover, the set of patterns presents redundancy in the sense that many patterns convey the same information [Ashrafi *et al.*, 2007]. To overcome this problem, several proposals have been made to construct only a manageably-sized set of patterns from which we can regenerate all frequent patterns along with to their exact frequencies. Such a reduced set is better known as *exact concise* (or *condensed*) *representation*. A concise representation only stores a non-redundant cover of all frequent patterns. In many practical situations, this cover is considerably smaller than the complete collection of all frequent patterns. Therefore, a concise representation can be used in those situations where it is impossible or inefficient to get out all frequent patterns.

1.1 Motivations and Contributions

Within the traditional association analysis, the conjunction connector – linking items – got the monopoly [Ceglar and Roddick, 2006]. This was motivated by the original application pertaining to market basket analysis. In this respect, a growing number of approaches explored the conjunctive search space where items are characterized by the frequency of their *simultaneous occurrence* (or *co-occurrence*). The aim of such an exploration is to get out a lossless nucleus of itemsets, from which the remaining ones can be derived. Beyond high compactness rates, an exact concise representation makes it possible to guess the frequency status of an itemset and to exactly retrieve its exact support in the case that itemset is (potentially) interesting *w.r.t.* statistical measures. Many exact concise representations of frequent patterns were thus proposed in the literature [Bastide *et al.*, 2000b, Boulicaut *et al.*, 2003, Bykowski and Rigotti, 2003, Calders and Goethals, 2007, Casali *et al.*, 2005a, Kryszkiewicz, 2002, Liu *et al.*, 2007, Muhonen and Toivonen, 2006, Pasquier *et al.*, 1999b].

Among the numerous concise representations, the ones based on closed itemsets [Pasquier *et al.*, 1999b] and minimal generators [Bastide *et al.*, 2000b] (*aka* free itemsets [Boulicaut *et al.*, 2003] or key itemsets [Stumme *et al.*, 2002] or intent reducts [Xie and Liu, 2005]) got a large interest since their respective proposals. The representation based on closed itemsets heavily relies on an operator [Ganter and Wille, 1999] which makes it possible mapping an important number of elements – from the frequent itemset search space – into a single element within that of frequent closed itemsets. On its side, the minimal

generator-based representation takes advantage from its efficient computation thanks to the interesting structural properties offered by the minimal generator set [Stumme *et al.*, 2002].

In fact, these itemsets are closely linked. Indeed, once applied, the aforementioned operator partitions the set of frequent itemsets into equivalence classes. Each class contains itemsets characterizing the same set of objects. These itemsets hence share the same closure obtained by intersecting the associated objects. The closed itemset is then the *unique maximal* set of items characterizing a set of objects. While, *often several* minimal generators constitute the *minimal* elements of each class. Semantically speaking, a closed itemset thus includes the most specific expression, while a minimal generator includes one of the most general expressions describing the associated set of objects.

The aforementioned link between closed patterns and minimal generators explains why they are often simultaneously used for concisely representing pattern classes, like frequent itemsets (*e.g.* [Li *et al.*, 2005, Li *et al.*, 2007, Phan Luong, 2002, Soulet and Crémilleux, 2008, Xie *et al.*, 2006]), association rules (*e.g.* [Bastide *et al.*, 2000a, Ben Yahia *et al.*, 2009b, Kryszkiewicz, 2002]), sequential patterns (*e.g.* [Balcázar and Casas-Garriga, 2007, Lo *et al.*, 2008]), etc. In this respect, the interesting structural properties of the minimal generator set made it a key step for mining important pattern classes as well as for knowledge interpretation. For example,

(*i*) they allow the efficient mining (*resp.* construction) of the set of frequent itemsets [Bastide *et al.*, 2000b] and of frequent closed itemsets [Pasquier *et al.*, 1999b, Stumme *et al.*, 2002] (*resp.* partially ordered structure [Hamrouni *et al.*, 2005b]). Indeed, minimal generators are the first elements of their respective equivalence classes to be reached. Furthermore, "to be (frequent) minimal generator" induces an anti-monotone constraint, exploited in achieving better performances. Indeed, each superset of an itemset not fulfilling the constraint is ensured not to be a (frequent) minimal generator. Hence, this constraint dramatically facilitates the localization of the elements to be retained, and thus reduces the cost of the processing to be carried out [Mannila and Toivonen, 1997].

(*ii*) they are at the roots of various concise representations of frequent itemsets [Calders *et al.*, 2005, Kryszkiewicz, 2002, Liu *et al.*, 2007]. They are also used in other fields, like graph theory (as minimal transversals [Berge, 1989]), database design (as minimal keys [Maier, 1983]), etc. Interestingly, minimal generators are used for mining complex pattern classes, like sequential patterns [Balcázar and Casas-Garriga, 2007, Lo *et al.*, 2008], etc.

(*iv*) they play a key role in the rule set construction since they are at the origin of a variety of compact subsets of the implication/association rule sets of a context [Ceglar and Roddick, 2006, Kryszkiewicz, 2002, Pasquier, 2009], which are hence called as *generic bases*. Traditionally, a generic basis is considered as an irreducible nucleus of the underlying rule set from which *redundant* ones can be derived without any loss of information [Pasquier, 2009]. In this context, many proposals have shown that generic bases, containing association rules whose implications are between minimal generators and closed itemsets, convey the maximum of information since they are of minimal premises and of maximal conclusions [Bastide *et al.*, 2000a, Kryszkiewicz, 1998, Pasquier, 2009]. For these reasons, such association rules are considered as the most informative ones [Bastide *et al.*, 2000a], since they preserve the minimum description length principle (MDLP) [Grunwald, 2007, Rissanen, 1978].

(*iii*) they are, according to the MDLP, the preferred representation of an equivalence class in applications like model selection, classification, etc. [Li *et al.*, 2006]. Indeed, being usually strictly smaller in size than their closed itemsets (unless themselves closed), they offer minimal combinations of conditions necessary to identify a class of situations. This reduces the economic cost of a decision process.

Unfortunately, the number of minimal generators is usually larger than that of closed itemsets. This is explained by the fact that several minimal generators cohabit in the same equivalence class and, hence, convey the same information. Thus, a same piece of knowledge is redundantly conveyed by distinct minimal generators. For example, this occurs in the case of association rules where minimal generators are used in the premise part. The same problem applies for the equivalence classes having their supersets for minimal seeds, which involves a highly combinatorial redundancy.

This situation motivated us to explore the issue of how to reduce the number of minimal generators per equivalence class without information loss. Our aim is thus to approach as much as possible the "ideal" case which consists in only retaining a unique irreducible minimal generator per equivalence class. This issue thus concerns the study of the internal interchangeability of the minimal generators. This consists in localizing minimal generators mutually reachable by permutation of their respective subsets. This will allow to group them in *finer* equivalence classes, induced by a *dedicated substitution operator*. Then, only a representative in each class will be retained, while the remaining ones will be omitted since redundant.

The obtained results can be useful for the different extensions and applications of minimal generators or their similar constructs. Here, they will be applied to generic bases of association rules to reduce without information loss the number of rules to be retained. A new approach will then be proposed, dedicated to the extraction of a lossless subset of generic association rules based on redundancy-free minimal generators as a starting point.

In the literature, other item links such as the complementary/mutually occurrences – rather than item co-occurrences – were neglected [Steinbach and Kumar, 2007], and only some recent works highlight the added-value of this type of knowledge. Indeed, the focus has been mainly on mining items linked through the conjunctive connector, *i.e.*, conjunctive itemsets. Association rule forms that have been mainly of interest also convey relation between conjunctions of itemsets [Ceglar and Roddick, 2006]. However, in practice the following situations can arise. Suppose that a market basket data is under treatment, and the manager is searching for items c_1, c_2, ..., and c_n whose selling implies that of at least one of two competitive products a and b (or probably both), *i.e.*, the items fulfilling the condition: $c_1 \vee c_2 \vee \ldots \vee c_n \Rightarrow a \vee b$ is always true. Such a rule conveys knowledge about the items sold simultaneously with a or b. Since the disjunction connector $\vee$ is inclusive, the simultaneous selling of c_i and c_j $(i \neq j)$ is possible. On the other hand, in a textmining application related to text translation from a language l_1 to a language l_2, an analyst may be interested in the possible translations in the language l_2 of a given term t belonging to the language l_1. In this respect, t may have several translations tr_1, tr_2, ..., and tr_n in the language l_2 according to its usage context. Thus, a rule like $tr_1 \vee tr_2 \vee \ldots \vee tr_n \Rightarrow t$ is interesting since it summarizes the possible translation of t. In both cases, more computations may be performed to get more precise information about the effect of a given product (*resp.* terms) among c_i (*resp.* t_j) on the appearance of a and b (*resp.* t). Various other applications of disjunctive itemsets are possible in the contexts of social network analysis and bioinformatics, as previously mentioned in [Zhao

et al., 2006]. In such situations, the disjunction connector linking items can bring key information as well as a summarizing method of the conveyed knowledge. Such knowledge may not be obtained even by a collection of conjunctive patterns [Nanavati *et al.*, 2001].

Due to the close link between frequent itemsets and association rules, it is more advantageous to mine a concise representation of frequent itemsets that offers direct access to the disjunctive support of frequent itemsets. Such a representation can be used as a starting point for mining generalized rules (*i.e.*, also involving disjunction and negation of items) based on frequently occurring itemsets. In this respect, we initially focus on the *unique* representation in the literature exploring the disjunctive search space, namely that based on *essential itemsets* [Casali *et al.*, 2005a].

Being the equivalent of minimal generators within the disjunctive search space, essential itemsets bring interesting knowledge about the *complementary occurrence* of items in a dataset. However, several essential itemsets can characterize the same set of objects. This motivated us to propose a new *disjunctive closure operator* dedicated to the disjunctive search space to avoid such a redundancy.

The disjunctive closure operator will offer a reduced exact representation of frequent itemsets. This representation also allows the efficient derivation of the different types of itemset supports, *i.e.*, conjunctive, disjunctive and negative. In addition, the proposed closure operator constitutes an interesting tool for the efficient exploration of the disjunctive search space. Such an exploration can be used for example towards the derivation of *generalized association rules*. These latter rules generalize classic association rules – positive rules – to also offer disjunction and negation connectors between items, in addition to the conjunctive one. The associated quality measures can thus be derived using a representation based on disjunctive itemsets. In the literature, generalized association rules are useful in different applications. For example, they are used as an intermediate step for defining concise representations for frequent itemsets [Ceglar and Roddick, 2006]. They are also exploited to provide the end-users with some new forms of association rules [Grün, 1998, Kim, 2003, Nanavati *et al.*, 2001].

The main contributions of this thesis are thus twofold:

1. A study of the redundancy within the set of minimal generators that can be extracted from a dataset aiming at only retaining without information loss those that are actually irreducible. The obtained results will be applied to reduce the number of generic association rules.

2. An exploration of the disjunctive search space through the introduction of a new disjunctive closure operator. In particular, we propose a new exact concise representation of frequent itemsets only based on disjunctive closed itemsets. In addition, new association rule forms will be efficiently derived, aiming to offer richer knowledge to the end-users.

The evaluation protocol of these contributions consists of experimental studies carried out on dense and sparse benchmark databases commonly used for evaluating data mining contributions as well as a comparison with other methods reported in the literature.

1.2 Thesis Organization

The rest of this thesis is organized as follows:

Chapter 2 briefly describes the mathematical background of frequent itemset and association rule mining. Moreover, a characterization of the approaches dedicated to the extraction of subsets of association rules is described.

Chapter 3 is dedicated to a thorough analysis of the main concise representations of frequent itemsets. Their respective link with the central concept of minimal generator will also be highlighted. This chapter also presents a critical comparative study of the surveyed concise representations. This chapter extends the work we proposed in [Ben Yahia *et al.*, 2006].

Chapter 4 focuses on the lossless reduction of the minimal generator set. A pioneer attempt, proposed in the literature, is thoroughly studied. The associated system is shown to be extracted with a loss of information. It is thus followed by our first solution which is lossless. Interestingly, the associated system constitutes a perfect cover of the minimal generator set, since being a subset of this latter set. We also introduce a second system as a lossless reduction of the minimal generator set having for advantage its interesting structural properties. New mining tools are designed for getting the proposed systems. Experimental results prove that our approach allows pruning a large number of minimal generators and, thus, to reach as much as possible the ideal case, *i.e.*, one irreducible minimal generator per closed itemset. The main content of this chapter was published in [Hamrouni *et al.*, 2007b, Hamrouni *et al.*, 2008a].

Chapter 5 is motivated by the fact that the results obtained in the previous chapter are extensible to generic association rules. Our aim is thus to show that these latter rules are no longer irredundant and can be further reduced. The proposed approach in this chapter is based on our perfect cover of the minimal generator set. An axiomatic system is also proposed for losslessly deriving redundant association rules starting from retained ones. To efficiently extract the lossless subsets of generic association rules, a new algorithm is proposed. Its main feature is that it constructs the precedence links between closed itemsets and simultaneously derives these latter itemsets only using minimal generators as a starting point. Carried out experiments show interesting compactness rates of the proposed cover set, as well as its efficient extraction. The main content of this chapter was published in [Hamrouni *et al.*, 2006, Hamrouni *et al.*, 2008a].

Chapter 6 offers a new disjunctive closure operator. The purpose of this operator is to structure the disjunctive search space into associated disjunctive equivalence classes. The structural properties of this operator are thoroughly studied. Its introduction makes it possible to concisely represent the disjunctive search space through only maintaining disjunctive closed itemsets. A new concise representation of frequent itemsets is then proposed based on the disjunctive closure of frequent essential itemsets. A dedicated algorithm to its extraction is also proposed. The obtained experimental results show that the disjunctive closed pattern-based representation highlights interesting compactness rates whenever compared to the main concise representations of the literature. Our main publications related to this chapter are [Denden *et al.*, 2008, Hamrouni *et al.*, 2007a, Hamrouni *et al.*, 2009b].

Chapter 7 introduces a novel approach for extracting generalized association rules. It thus starts by extending the framework of classic association rules through taking into account various possible connectors as well as negative items. An overview of the possible mined forms of generalized

association rules is then presented, in addition to how are calculated the associated supports in the general case. Aiming at reducing the number of extracted rules, a selection process of generalized rules is then described. We also propose new algorithms covering the whole process of generalized association rules extraction. The experimental results show that our approach allows the efficient extraction of various association rule forms. A part of this chapter was published in [Hamrouni *et al.*, 2008b].

Chapter 8 concludes this thesis and points out our perspectives for future work.

Part I

Overview of the Extraction of Concise Representations for Frequent Patterns

Chapter 2

Preliminary Notions

2.1 Introduction

The main interest of association rule mining is the identification of significant and hidden relations or correlations between data contained in a database. Such relations can be useful for the end-users, like domain experts, decision makers, etc. These latter can exploit them for various objectives aiming at improving their decision quality.

In this chapter, we present the problem of association rules mining based on frequent itemsets. We also recall the mathematical background of Formal Concept Analysis (FCA). This latter is used as a starting point for the derivation of lossless subsets of association rules, called *generic bases*.

The organization of the chapter is as follows: Section 2.2 presents the basic definitions related to frequent itemsets search. Section 2.3 details the FCA mathematical settings. The association rule framework is described in Section 2.4 where the link between the FCA and the extraction of lossless association rule subsets is stated. Section 2.5 concludes this chapter.

2.2 Itemset Search Space

This section presents some basic definitions that will be used in the remainder.

2.2.1 Extraction Context and Itemsets

In this thesis, we will consider datasets represented using binary contexts defined as follows.

Definition 1 (Extraction context)
An extraction context (or context for short) is a triplet $\mathcal{K} = (\mathcal{O}, \mathcal{I}, \mathcal{M})$, where $\mathcal{O}$ is a finite set of objects (or transactions), $\mathcal{I}$ is a finite set of items (or attributes) and $\mathcal{M}$ is a binary (incidence) relation (i.e., $\mathcal{M} \subseteq \mathcal{O} \times \mathcal{I}$). A couple $(o, i) \in \mathcal{M}$ if the object $o \in \mathcal{O}$ has the item $i \in \mathcal{I}$.

Example 1 *Consider the context given in Table 2.1, used as a running example through this chapter. Here, $\mathcal{O} = \{$**1**, **2**, **3**, **4**, **5**$\}$ and $\mathcal{I} = \{$A, B, C, D, E, F$\}$. The couple $(3,$ E$) \in \mathcal{M}$ since it is crossed in the matrix, on the contrary of the couple $(5,$ B$)$ whose associated cell is not crossed in the matrix.*

	A	B	C	D	E	F
1	×	×	×	×		
2			×	×	×	
3	×	×			×	×
4	×	×	×	×	×	×
5			×	×		×

Table 2.1: An extraction context.

An itemset is a set of items. For example, {C, D, E} is an itemset composed by the items C, D and E. In the remainder, we use a separator-free form for the sets, *e.g.*, CDE stands for the itemset {C, D, E}. The terms *dataset* and (*extraction*) *context* are also used interchangeably throughout the remainder of the thesis. It is the same for *transaction* and *object*.

2.2.2 Itemset Supports: Links and Associated Constraints

An itemset can be characterized by different kinds of support. The latter are detailed in the following definition.

Definition 2 (Support of an itemset)
Let $\mathcal{K} = (\mathcal{O}, \mathcal{I}, \mathcal{M})$ be an extraction context. We distinguish three kinds of support associated to a non-empty itemset I:

- ***Conjunctive support:*** $Supp(\wedge I) = |\{o \in \mathcal{O} | (\forall\ i \in I, (o,i) \in \mathcal{M})\}|$,

- ***Disjunctive support:*** $Supp(\vee I) = |\{o \in \mathcal{O} | (\exists\ i \in I, (o,i) \in \mathcal{M})\}|$, *and,*

- ***Negative support:*** $Supp(\overline{I}) = |\{o \in \mathcal{O} | (\forall\ i \in I, (o,i) \notin \mathcal{M})\}|$.

Roughly speaking, the different supports are defined as follows:

- $\boldsymbol{Supp(\wedge I)}$ is the number of transactions containing all items of I. In this case, I can be seen as a conjunction of items (*i.e.*, $i_1 \wedge i_2 \wedge \ldots \wedge i_n$) such that the appearance of one of its items is conditioned by the appearance of all remaining ones to say that I satisfies a given transaction.

- $\boldsymbol{Supp(\vee I)}$ is the number of transactions containing at least one item of I. In this case, I can be seen as a disjunction of items (*i.e.*, $i_1 \vee i_2 \vee \ldots \vee i_n$) such that the presence of one item of I in a given transaction is sufficient to satisfy it independently from the remaining items.

- $\boldsymbol{Supp(\overline{I})}$ is the number of transactions that do not contain any item of I. In other words, they contain the respective negations of all items of I (*i.e.*, $\overline{i_1} \wedge \overline{i_2} \wedge \ldots \wedge \overline{i_n}$).

Example 2 *Consider the context depicted by Table 2.1. The different supports that can be associated to the itemset* BC *are:* $Supp(\wedge BC) = \mathbf{2}$, $Supp(\vee BC) = \mathbf{5}$, $Supp(\overline{BC}) = \mathbf{0}$.

Note that the conjunctive support of the empty set is equal to $|\mathcal{O}|$ since included in all objects. While the disjunctive support is not defined on this pattern since it does not contain any item.

The next proposition summarizes important properties related to the itemsets supports.

Proposition 1 *Let $i \in \mathcal{I}$, and $I, I_1 \subseteq \mathcal{I}$. The following properties hold:*

- $Supp(\wedge i) = Supp(\vee i)$.
- $Supp(\wedge I) \leq Supp(\vee I)$ *for* $I \neq \emptyset$.
- *If* $I \subseteq I_1$, *then* $Supp(\wedge I) \geq Supp(\wedge I_1)$.
- *If* $I \neq \emptyset$ *and* $I \subseteq I_1$, *then* $Supp(\vee I) \leq Supp(\vee I_1)$.

Given the respective disjunctive supports of the subsets of an arbitrary itemset, we are able to derive its conjunctive support using the *inclusion-exclusion identities* [Galambos and Simonelli, 2000, Narushima, 1982]. Furthermore, thanks to the *De Morgan's law*, we are able to straightforwardly derive its negative support. Lemma 1 shows these important equations.

Lemma 1 *Let $I \subseteq \mathcal{I}$ be an arbitrary itemset. Its conjunctive and negative supports are respectively derived as follows [Galambos and Simonelli, 2000]:*

$$\bullet\ Supp(\wedge I) = \sum_{\emptyset \subset I_1 \subseteq I} (\text{-}1)^{|I_1| \text{ - } 1}\ Supp(\vee I_1) \qquad (1)$$

$$\bullet\ Supp(\overline{I}) = |\mathcal{O}| - Supp(\vee I) \qquad (2)$$

Example 3 *Consider the context of Table 2.1. Given the respective disjunctive supports of BC' subsets, its conjunctive and negative supports are inferred as follows:*

- $Supp(\wedge BC) = (\text{-}1)^{|BC| - 1}\ Supp(\vee BC) + (\text{-}1)^{|B| - 1}\ Supp(\vee B) + (\text{-}1)^{|C| - 1}\ Supp(\vee C) = \text{-}\ Supp(\vee BC) + Supp(\vee B) + Supp(\vee C) = \text{-}\ \mathbf{5} + \mathbf{3} + \mathbf{4} = \mathbf{2}$.
- $Supp(\overline{BC}) = |\mathcal{O}| \text{ - } Supp(\vee BC) = \mathbf{5} \text{ - } Supp(\vee BC) = \mathbf{5} \text{ - } \mathbf{5} = \mathbf{0}$.

To prune the search space of itemsets, different types of constraints were investigated. Anti-monotone and monotone constraints, defined in the following, are the most used ones [Bonchi and Lucchese, 2006].

Definition 3 (Anti-monotone constraint)
Let $I \subseteq \mathcal{I}$. A constraint Q is said to be anti-monotone if $\forall\ I_1 \subseteq I$: I satisfies $Q \Rightarrow I_1$ satisfies Q.

Definition 4 (Monotone constraint)
Let $I \subseteq \mathcal{I}$. A constraint Q is said to be monotone if $\forall\ I_1 \supseteq I$: I satisfies $Q \Rightarrow I_1$ satisfies Q.

Example 4 *By setting a minimum conjunctive support threshold, we define an anti-monotone constraint, commonly called the frequency constraint. Dually, the disjunctive frequency constraint, relying on a minimum disjunctive support threshold, is a monotone one.*

The next proposition states an important result about the conjunction of constraints of the *same* type.

Proposition 2 *The conjunction of anti-monotone (resp. monotone) constraints results in an anti-monotone (resp. monotone) constraint.*

A proof of this proposition can be found in [Lee *et al.*, 2006].

Hereafter, $Supp(\wedge I)$ will simply be denoted $Supp(I)$. In addition, if there is no risk of confusion, the *conjunctive support* will be called *support*. The next section focuses on frequent itemsets, induced by the frequency constraint.

2.2.3 Frequent Itemsets

Since in practice we are mainly interested in itemsets that occur at least in a given number of transactions, we introduce the notion of *frequency*.

Definition 5 (Frequency of an itemset)
The frequency of an itemset $I \subseteq \mathcal{I}$ in a context $\mathcal{K}$, denoted by $Freq(I)$, is equal to $Freq(I) = \frac{Supp(I)}{|\mathcal{O}|}$.

In the remainder, we will mainly use the support of itemsets instead of their frequency.

Definition 6 (Frequent or Infrequent itemset)
An itemset I is said to be frequent in $\mathcal{K}$ if $Supp(I)$ is greater than or equal to a user-specified threshold, denoted minsupp. Otherwise, I is said to be infrequent or rare.

Example 5 *Consider the itemset `CDE` of the context given in Table 2.1. Both transactions* **2** *and* **4** *contain this itemset. Hence, $Supp(\texttt{CDE}) = \mathbf{2}$. The frequency of `CDE` is then equal to $\frac{2}{5}$. If $minsupp = \mathbf{1}$, then `CDE` is considered as frequent in $\mathcal{K}$ since $Supp(\texttt{CDE}) = \mathbf{2} \geq \mathbf{1}$.*

Notation 1 *For the sake of readability, in the tables and figures presenting experimental results, the value of minsupp can be sketched in percentage. Let us suppose this value equals to α. It indicates that `the minimum number of objects which must be satisfied` is equal to $\frac{\alpha \times |\mathcal{O}|}{100}$.*

By setting the *minsupp* threshold, we only consider frequent itemsets (and not the whole set of itemsets). Hereafter, we will denote by $\mathcal{FI}$ the set of frequent itemsets that can be extracted from a context $\mathcal{K}$ for a given *minsupp*. The next proposition sheds light on an important property of the set of frequent itemsets. It states that all subsets of a frequent itemset are also frequent. Conversely, the supersets of an infrequent itemset are also infrequent.

Proposition 3 *Let $I \subseteq \mathcal{I}$. We have [Agrawal* et al.*, 1996]:*

- *If $I \in \mathcal{FI}$, then $\forall\ I_1 \subseteq I$, $I_1 \in \mathcal{FI}$.*
- *If $I \notin \mathcal{FI}$, then $\forall\ I_1 \supseteq I$, $I_1 \notin \mathcal{FI}$.*

This result follows from the fact that the constraint induced by setting *minsupp* is anti-monotone (*cf.* Definition 3). Since the supersets of infrequent itemsets are expected to be infrequent, the set $\mathcal{I}$ (and consequently the context $\mathcal{K}$) will be reduced to frequent items. Infrequent ones will thus be pruned. The set of frequent itemsets induces an order ideal (or down-set) in $(\mathcal{P}(\mathcal{I}), \subseteq)$ when partially ordered *w.r.t.* set inclusion. An order ideal is defined as follows:

Definition 7 (Order ideal)
A subset S of $\mathcal{P}(\mathcal{I})$ is an order ideal in $(\mathcal{P}(\mathcal{I}), \subseteq)$ if it fulfills the following properties [Ganter and Wille, 1999]:
- If $x \in S$, then $\forall\ y \subseteq x$, $y \in S$.
- If $x \notin S$, then $\forall\ y \supseteq x$, $y \notin S$.

The set S is hence downward closed since for each $x \in S$, *all its subsets* are in S. An order ideal splits the power-set of items into two disjoint parts: the first contains itemsets fulfilling the associated constraint, while the second part contains those not fulfilling it. Both parts are delimited thanks to a positive and a negative border, respectively [Mannila and Toivonen, 1997]. The positive border contains the *maximal*, *w.r.t.* set inclusion, elements among those that fulfill the constraint associated to the order ideal. While the negative border gathers the *minimal*, *w.r.t.* set inclusion, elements among those that do not fulfill the constraint. These borders are formally defined as follows:

Definition 8 (Positive, Negative border)
Let $(\mathcal{P}(\mathcal{I}), \subseteq)$ be a partially ordered set of elements and S be a subset of $\mathcal{P}(\mathcal{I})$ s.t. S is an order ideal in $(\mathcal{P}(\mathcal{I}), \subseteq)$. S can be represented by its positive border $\mathcal{B}d^+(S)$ or its negative border $\mathcal{B}d^-(S)$ defined as follows:

$$\mathcal{B}d^+(S) = max_{\subseteq}\{I \in S\},$$
$$\mathcal{B}d^-(S) = min_{\subseteq}\{I \in \mathcal{P}(\mathcal{I}) \setminus S\}.$$

Dually, a monotone constraint induces an **order filter** [Ganter and Wille, 1999] in $(\mathcal{P}(\mathcal{I}), \subseteq)$. If an element belongs to this latter order, then it is the same for *all its supersets* (*cf.* Definition 4).

2.2.4 Concise Representations for Frequent Itemsets

Several reported works shed light on the huge number of frequent itemsets that can be extracted from a given context. In this situation, extracting a subset of itemsets constitutes an interesting solution for concisely representing frequent itemsets [Calders *et al.*, 2005, Ceglar and Roddick, 2006, Kryszkiewicz, 2002]. To be lossless, this subset should enable the derivation of the whole set of frequent itemsets, associated to their exact supports. In this case, it is called *exact concise representation of frequent itemsets*. Definition 9 summarizes this concept:

Definition 9 (Exact concise representation of frequent itemsets)
Let $\mathcal{E}$ be a set of itemsets. $\mathcal{E}$ is said to be an exact concise representation of the set of frequent itemsets if, starting from $\mathcal{E}$, we are able to guess whether an arbitrary itemset I is frequent or not. In addition, if I is frequent, then we can exactly determine its conjunctive support.

In fact, the concept of concise representation for frequent itemsets derives from a more general framework, called the *ϵ-adequate representation* introduced in [Mannila and Toinoven, 1996]. We begin by describing this framework. After that, we adapt it to our context. Intuitively, an ϵ-adequate representation is a representation which can substitute another one in order to answer the same request(s), more effectively, possibly at the cost of an error bounded by the parameter ϵ. Such a representation is defined as follows:

Definition 10 (ϵ-ADEQUATE REPRESENTATION)
Let $\mathcal{S}$ be a class of structures. Let $\mathcal{Q}$ be a class of queries for $\mathcal{S}$. The value of a query $Q \in \mathcal{Q}$ on a structure $s \in \mathcal{S}$ is assumed to be a real number in [0, 1] and is denoted by $Q(s)$. An ϵ-adequate representation for $\mathcal{S}$, w.r.t. a class of queries $\mathcal{Q}$, is a class of structures $\mathcal{C}$, a representation mapping rep: $\mathcal{S} \rightarrow \mathcal{C}$ and a query evaluation function m: $\mathcal{Q} \times \mathcal{C} \rightarrow$ [0, 1] s.t. $\forall\, Q \in \mathcal{Q}$, $\forall\, s \in \mathcal{S}$, $|Q(s) - m(Q, rep(s))| \leq \epsilon$.

In our case, the class of structures $\mathcal{S}$ is composed by the different set of frequent itemsets that can be drawn from all possible binary extraction contexts $\mathcal{EC}$, defined over a set of items $\mathcal{I}$, a set of objects $\mathcal{O}$, and for a minimum support threshold *minsupp*. Thus, $\mathcal{S} = \{\mathcal{FI}_{\mathcal{K}} \mid \mathcal{K} \in \mathcal{EC}\}$. The set of queries represents those searching for the frequency of itemsets of size no more than $|\mathcal{I}|$. This set is as follows: $\mathcal{Q} = \{Q_X \mid X \subseteq \mathcal{I}\}$ where the value of Q_X in a context $\mathcal{K} \in \mathcal{EC}$ is defined by $Q_X(\mathcal{K}) = \mathit{Freq}(X) = \frac{\mathit{Supp}(X)}{|\mathcal{O}|}$. While rep is a given concise representation of frequent itemset, $\mathcal{C}$ is the application of rep on the different contexts of $\mathcal{EC}$: $\mathcal{C} = \{rep(\mathcal{K}) \mid \mathcal{K} \in \mathcal{EC}\}$. Finally, m is the function by which the frequency of an arbitrary itemset is assessed starting from the representation rep.

To establish the link between an exact concise representation of frequent itemsets and the concept of ϵ-adequate representation, we note that exact representations form 0-adequate representations of the set of frequent itemsets. Indeed, for an arbitrary context and a given *minsupp* value, they allow the exact retrieval of the respective frequencies of frequent itemsets. The error ϵ is hence equal to 0. This is not the case of approximate concise representations, like the δ-free set-based one [Boulicaut *et al.*, 2003] and maximal frequent itemsets [Bayardo, 1998], from which only an approximation is possible when searching for the frequency of an arbitrary itemset.

An exact concise representation is also called *perfect cover* if it fulfills the conditions stated by the following definition:

Definition 11 (PERFECT COVER)
A cover of a set of patterns S is a set S_1 that allows recovering S without information loss. S_1 is said perfect if it is always a subset of S.

It is also important to note that exact concise representations are preferable to the whole set of frequent itemsets *w.r.t.* the minimal description length principle (MDLP) [Grunwald, 2007, Rissanen, 1978]. This principle states that the best theory describing a set of data is the one minimizing the description length of the theory plus the description length of the data described (or compressed) by the theory. It seeks to minimize the description length of the entire data. In the general case, this principle can be roughly described as follows:

Definition 12 (MINIMUM DESCRIPTION LENGTH PRINCIPLE (MDLP)) *[Grunwald, 2007]*
Given a set of hypothesis $\mathcal{H}$ learned from a set of data D, the best hypothesis $H \in \mathcal{H}$ is the one that minimizes:

$$\mathrm{L}(D, H) = \mathrm{L}(H) + \mathrm{L}(D|H)$$

in which

- $\mathrm{L}(H)$ *is the length in bits of the description of H, and,*
- $\mathrm{L}(D|H)$ *is the length, in bits, of the description of the data D when encoded with H.*

If we bring this principle into the context of concise representations of frequent itemsets, then the description length of the theory (here, a concise representation CR) given the input data (here, the set $\mathcal{FI}_\mathcal{K}$ of frequent itemsets associated to a context $\mathcal{K} \in \mathcal{EC}$) is computed as: $\mathrm{L}(\mathcal{FI}_\mathcal{K}, \mathrm{CR}) = \mathrm{L}(\mathrm{CR}) + \mathrm{L}(\mathcal{FI}_\mathcal{K}|\mathrm{CR})$. The MDLP thus seeks a concise representation that minimizes $\mathrm{L}(\mathcal{FI}_\mathcal{K}, \mathrm{CR})$.

To reduce the size of pattern sets, different proposals rely on Formal Concept Analysis [Ganter and Wille, 1999]. The next section is dedicated to its mathematical background.

2.3 Formal Concept Analysis

Formal Concept Analysis (FCA) mathematical foundations [Ganter and Wille, 1999] have been used as a theoretical basis for various tasks (*e.g.* [Sassi *et al.*, 2007, Stumme *et al.*, 1998, Valtchev *et al.*, 2004]). In our context, some concise representations of frequent patterns are based on FCA. Let us recall its basic constructs.

2.3.1 Galois Connection and Compound Operators

We begin by defining the Galois connection used to make the link between the power-sets $\mathcal{P}(\mathcal{I})$ and $\mathcal{P}(\mathcal{O})$ associated respectively to the set of items $\mathcal{I}$ and the set of objects $\mathcal{O}$.

Definition 13 (Galois connection)
Let $\mathcal{K} = (\mathcal{O}, \mathcal{I}, \mathcal{M})$ be an extraction context. The application ψ is defined from the power-set of objects (i.e., $\mathcal{P}(\mathcal{O})$) to the power-set of items (i.e., $\mathcal{P}(\mathcal{I})$). It associates to a set of objects O the set of items $i \in \mathcal{I}$ that are common to all objects $o \in O$:

$$\psi : \mathcal{P}(\mathcal{O}) \rightarrow \mathcal{P}(\mathcal{I})$$
$$O \mapsto \psi(O) = \{i \in \mathcal{I} \mid \forall\, o \in O,\, (o,\, i) \in \mathcal{M}\}$$

In a dual way, the application ϕ is defined from the power-set of items (i.e., $\mathcal{P}(\mathcal{I})$) to the power-set of objects (i.e., $\mathcal{P}(\mathcal{O})$). It associates to a set of items I the set of objects $o \in \mathcal{O}$ that contains all items $i \in I$:

$$\phi : \mathcal{P}(\mathcal{I}) \rightarrow \mathcal{P}(\mathcal{O})$$
$$I \mapsto \phi(I) = \{o \in \mathcal{O} \mid \forall\, i \in I,\, (o,\, i) \in \mathcal{M}\}$$

The couple of applications $(\psi,\, \phi)$ is a Galois connection between the power-set of $\mathcal{O}$ and that of $\mathcal{I}$ [Barbut and Monjardet, 1970, Ganter and Wille, 1999].

Definition 14 describes the properties that must satisfy an operator to be qualified as a closure or a kernel one [Ganter and Wille, 1999].

Definition 14 (Closure, Kernel operator)
Let $(S,\subseteq)$ be a partially ordered set and x, y be two elements of S. An operator h defined from $(S,\subseteq)$ to $(S,\subseteq)$ is called a closure operator if it is:
(i) extensive, i.e., $x \subseteq h(x)$,

(ii) isotone, i.e., $x \subseteq y \Rightarrow h(x) \subseteq h(y)$*, and,*
(iii) idempotent, i.e., $h(h(x)) = h(x)$.
Given the closure operator h *applied on the partially ordered set* $(S,\subseteq)$*, an element* $x \in S$ *is said to be closed if its image by* h *is equal to itself, i.e.,* $h(x) = x$.

If an operator h'*, defined from* $(S,\subseteq)$ *to* $(S,\subseteq)$*, is such that* $h'(x) \subseteq x$*, then* h' *has the property to be contractive. If it is also isotone and idempotent, then* h' *is said to be a kernel operator.*

The following definition introduces the closure operators associated to a Galois connection.

Definition 15 (Galois closure operators)
Let us consider the power-sets $\mathcal{P}(\mathcal{I})$ *and* $\mathcal{P}(\mathcal{O})$*, with the inclusion relation* $\subseteq$*, i.e., the partially ordered sets* $(\mathcal{P}(\mathcal{I}),\subseteq)$ *and* $(\mathcal{P}(\mathcal{O}),\subseteq)$*. The operators* $\gamma = \phi\ o\ \psi$ *from* $(\mathcal{P}(\mathcal{I}),\subseteq)$ *to* $(\mathcal{P}(\mathcal{I}),\subseteq)$*, and* $\omega = \psi\ o\ \phi$ *from* $(\mathcal{P}(\mathcal{O}),\subseteq)$ *to* $(\mathcal{P}(\mathcal{O}),\subseteq)$ *are closure operators of the Galois connection [Barbut and Monjardet, 1970, Ganter and Wille, 1999]. They define closure systems on* $(\mathcal{P}(\mathcal{I}),\subseteq)$ *and* $(\mathcal{P}(\mathcal{O}),\subseteq)$*, respectively. The operator* γ *generates closed subsets of items, while* ω *generates closed subsets of objects.*

This leads us to the definition of a formal concept.

Definition 16 (Formal concept)
A pair $c = (O,\ I) \in \mathcal{O} \times \mathcal{I}$*, of mutually corresponding subsets, i.e.,* $O = \psi(I)$ *and* $I = \phi(O)$*, is called a formal concept, where* O *is called extent of* c *and* I *is called its intent.*

Example 6 *The pair* (**14**, `ABCD`) *is a concept from the extraction context given by Table 2.1.*

The operators *join* and *meet* provide the least upper bound (LUB) and the greatest lower bound (GLB), respectively, of a couple of formal concepts.

Definition 17 (Join and Meet operators)
Let (O_1, I_1) *and* (O_2, I_2) *be two formal concepts. The operators join* $(\vee)$ *and meet* $(\wedge)$ *are respectively defined as follows [Ganter and Wille, 1999]:*

- $(O_1, I_1) \vee (O_2, I_2) = (\omega(O_1 \cup O_2), I_1 \cap I_2)$,
- $(O_1, I_1) \wedge (O_2, I_2) = (O_1 \cap O_2, \gamma(I_1 \cup I_2))$.

Proposition 4 presents the partial order on formal concepts *w.r.t.* set inclusion [Ganter and Wille, 1999].

Proposition 4 *A partial order on formal concepts is defined as:* $\forall\ c_1 = (O_1,\ I_1)$ *and* $c_2 = (O_2,\ I_2)$ *two formal concepts,* $c_1 \leq c_2$ *if* $O_2 \subseteq O_1$*, or equivalently* $I_1 \subseteq I_2$.

In the case where two formal concepts fulfill the condition of Proposition 4, they are said to be *comparable*. Otherwise, they are said to be *incomparable*. When partially sorted with set inclusion, formal concepts form a structure called *Galois (concept) lattice*, defined as follows.

Definition 18 (Galois (concept) lattice)
Given a context $\mathcal{K}$*, the set of formal concepts* $\mathcal{C}$ *is a complete lattice* $\mathcal{L}_{\mathcal{C}} = (\mathcal{C},\ \leq)$*, called Galois (concept) lattice, when* $\mathcal{C}$ *is considered with set inclusion between concepts intents (or extents) [Barbut and Monjardet, 1970, Ganter and Wille, 1999].*

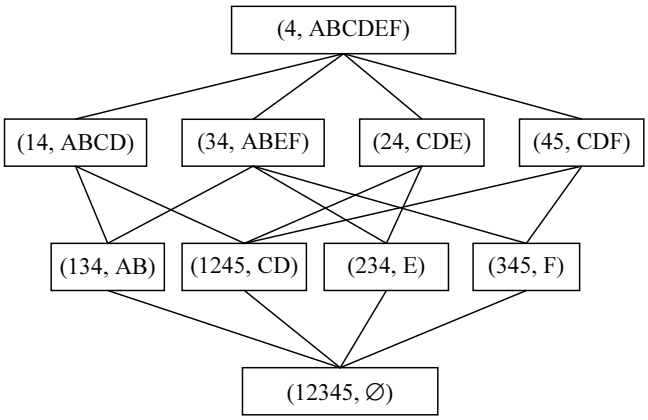


Figure 2.1: The Galois lattice associated to the extraction context of Table 2.1.

Example 7 *Figure 2.1 presents the Galois lattice associated to the context given by Table 2.1.*

The next definition presents two particular elements within a Galois lattice.

Definition 19 (Bottom, Top of a Galois lattice)
Within a Galois lattice, the element $\mathcal{O} \times \psi(\mathcal{O})$ is called `bottom` *of the lattice, and denoted $\bot$. While the element $\phi(\mathcal{I}) \times \mathcal{I}$ is called* `top` *of the lattice, and denoted $\top$. The extent (resp. intent) part of the bottom element is then the largest (resp. smallest) one, w.r.t. set inclusion, among the extents (resp. intents) of the lattice concepts. It is the reverse for the top element.*

Example 8 *Considering Figure 2.1, the bottom of the lattice is* (**12345**, $\emptyset$). *While the top element is* (**4**, `ABCDEF`).

2.3.2 Equivalence Classes, Closed Itemsets and Minimal Generators

Once applied, the closure operator γ induces an equivalence relation on the power-set of items $\mathcal{P}(\mathcal{I})$ splitting it into so-called *equivalence classes* [Bastide *et al.*, 2000b], which will further be denoted γ-equivalence classes. A γ-equivalence class is then defined as follows:

Definition 20 (γ-Equivalence class)
A γ-equivalence class contains a set of itemsets sharing the same set of objects and, hence, having the same closure computed using the operator γ.

Example 9 *Consider the context given by Table 2.1. Since the itemsets* `BD` *and* `ACD` *share the same set of objects, namely {***1**, **4***}, they belong to the same γ-equivalence class. They hence have a common closure, namely* `ABCD`.

In each γ-equivalence class, the largest itemset (*w.r.t.* set inclusion) is called a *closed itemset* while the minimal ones are called *minimal generators*. The respective definitions of these particular itemsets are given below.

Definition 21 (Closed itemset)
An itemset $I \subseteq \mathcal{I}$ is said to be closed if $\gamma(I) = I$ [Pasquier et al.*, 1999b].*

Example 10 *Given the context depicted by Table 2.1, the itemset ABCD is a closed one since it is the maximal set of items common to the set of objects {1, 4}. The itemset ACD is not closed since all objects containing the itemset ACD also contain the item B.*

The set of closed itemsets extracted from $\mathcal{K}$ will further be denoted $\mathcal{CI}$. Each closed itemset (CI) constitutes the intent part of a formal concept.

Definition 22 (Minimal generator)
• *An itemset $I_1 \subseteq \mathcal{I}$ is said to be a minimal generator of a closed itemset I if $\gamma(I_1) = I$ and, $\forall\ I_2 \subseteq \mathcal{I}$, if $I_2 \subseteq I_1$ and $\gamma(I_2) = I$, then $I_2 = I_1$ [Bastide* et al.*, 2000b].*
• *A minimal generator is also called a* **0***-free itemset [Boulicaut* et al.*, 2003] or a key itemset [Stumme* et al.*, 2002] or an intent reduct [Xie and Liu, 2005].*

Example 11 *Consider the* CI *ABCD described by the previous example. ABCD has AC as a minimal generator* (MG)*. Indeed, $\gamma(AC) = ABCD$ and the closure of each proper subset of AC is different from ABCD: $\gamma(\emptyset) = \emptyset$, $\gamma(A) = AB$ and $\gamma(C) = CD$. The* CI *ABCD has also other* MG*s which are AD, BC and BF. Hence,* $\text{MG}_{ABCD} = \{AC, AD, BC, BF\}$*. ABCD is then the largest element of its γ-equivalence class, whereas AC, AD, BC and BF are the minimal ones. All these itemsets share the objects {1, 4}.*

The set of MGs associated to a CI I (*resp.* an extraction context $\mathcal{K}$) will further be denoted MG_I (*resp.* $\mathcal{MG}$). The next proposition states the relation between elements of MG_I *w.r.t.* set inclusion.

Proposition 5 *The minimal generators of a closed itemset are incomparable w.r.t. set inclusion.*

Proof. The proof is based on the minimality status of a MG within its associated γ-equivalence class. Indeed, let g_1 and g_2 be two MGs of a CI I. Suppose that $g_1 \subset g_2$. This necessarily leads to the fact that g_2 is not a MG, which is in contradiction with the fact that $g_2 \in \text{MG}_I$. Thus, all elements of MG_I are incomparable *w.r.t.* set inclusion. $\diamondsuit$

The next proposition states an important property of the minimal generator set.

Proposition 6 *The set $\mathcal{MG}$ of minimal generators that can be extracted from a context $\mathcal{K}$ is an order ideal in $(\mathcal{P}(\mathcal{I}), \subseteq)$ [Stumme* et al.*, 2002].*

Since the conjunction of two anti-monotone constraints is also an anti-monotone constraint, the set $\mathcal{FMG}$ of frequent minimal generators is also an order ideal.

Example 12 *For minsupp = 1, Table 2.2 shows, for each frequent* CI*, its* MG*s and its support value.*

Using supports, CIs and MGs are characterized as follows.

Proposition 7 *Let $I, I' \subseteq \mathcal{I}$,*
• *I is a closed itemset iff $Supp(I) > max\{Supp(I') \mid I \subset I'\}$.*
• *I is a minimal generator iff $Supp(I) < min\{Supp(I') \mid I' \subset I\}$.*

Frequent CI	Frequent MGs	Support
∅	∅	5
E	E	3
F	F	3
AB	A, B	3
CD	C, D	4
CDE	CE, DE	2
CDF	CF, DF	2
ABCD	AC, AD, BC, BD	2
ABEF	AE, AF, BE, BF, EF	2
ABCDEF	ACE, ACF, ADE, ADF, BCE, BCF, BDE, BDF, CEF, DEF	1

Table 2.2: The list of frequent closed itemsets, and for each one, the corresponding minimal generators, and support.

Specific elements within γ-equivalence classes are called *pseudo-closed itemsets* and are defined as follows.

Definition 23 (Pseudo-closed itemset)
An itemset $X \subseteq \mathcal{I}$ is pseudo-closed iff $\gamma(X) \neq X$ and $\forall\, Y \subset X$, such that Y is a pseudo-closed itemset, we have $\gamma(Y) \subset X$ [Guigues and Duquenne, 1986].

Example 13 *Considering Table 2.2, the itemset EF is a pseudo-closed one since $\gamma(EF) = ABEF \neq EF$. In addition, all subsets of EF are not pseudo-closed itemsets since they are equal to their respective closures.*

2.3.3 Iceberg Lattice

In a Galois lattice, the concepts whose intents are frequent, *i.e.*, containing frequent CIs, constitute a join-semi-lattice. Such a structure is called *Iceberg lattice* [Stumme *et al.*, 2002] and is formally defined as follows:

Definition 24 (Iceberg lattice)
Let $\mathcal{FCI}$ be the set of frequent CI*s of a context $\mathcal{K}$. When the set $\mathcal{FCI}$ is partially ordered w.r.t. set inclusion, the resulting structure only preserves the Join operator [Ganter and Wille, 1999]. This structure is called a `join semi-lattice` or an `upper semi-lattice` [Mephu Nguifo, 1994], and is hereafter referred to as "Iceberg lattice" [Stumme* et al.*, 2002].*

Since the Iceberg lattice only maintains frequent CIs, the top of the lattice can be pruned due to the infrequency of the corresponding CI. In this case, a new top element is added which covers all frequent CIs, which makes it a lattice again [Stumme *et al.*, 2002]. Hereafter, we will only consider the current frequent CIs. When necessary to be added (due to its infrequency), the top element simply contains the set of items $\mathcal{I}$. This latter is ensured to be closed and covering all frequent CIs.

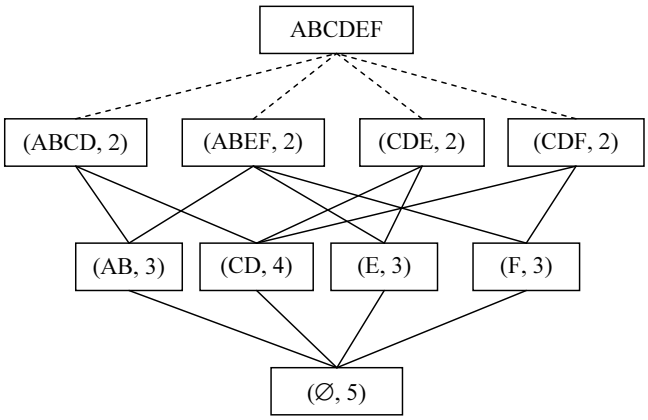


Figure 2.2: For *minsupp* = **2**, the Iceberg lattice associated to the context of Table 2.1.

Example 14 *For minsupp* = **2**, *Figure 2.2 shows the Iceberg lattice corresponding to our context depicted in Table 2.1. Each node contains the couple composed by a frequent* CI *and its support. The extent part of each concept is reduced to the associated support which corresponds to the cardinality of the extent part. In this case, the top element contains the* CI `ABCDEF`. *Since the latter is infrequent, its support is not shown in the figure. In addition, the precedence links between* `ABCDEF` *and the* CI*s of its lower cover are drawn with dashed lines.*

Each node (or equivalently, CI) in the Iceberg lattice has an *upper cover*, formally defined as follows:

Definition 25 (Upper cover)
The upper cover of a CI f *(denoted* $Cov^u(f)$*) consists of the* CI*s that immediately cover* f *in the Iceberg lattice. The set* $Cov^u(f)$ *is given as follows:* $Cov^u(f) = \{f_1 \in \mathcal{FCI} \mid f \subset f_1$ *and* $\nexists\ f_2 \in \mathcal{FCI}$ *s.t.* $f \subset f_2 \subset f_1\}$.

Example 15 *Let us consider the* CI `AB` *of the Iceberg lattice depicted by Figure 2.2. Then, we have:* $Cov^u($`AB`$) =$ {`ABCD`, `ABEF`}.

Dually, a node in the Iceberg lattice has a *lower cover*, formally defined as follows:

Definition 26 (Lower cover)
The lower cover of a CI f *(denoted* $Cov_l(f)$*) consists of the* CI*s that are immediately covered by* f *in the Iceberg lattice. The set* $Cov_l(f)$ *is given as follows:* $Cov_l(f) = \{f_1 \in \mathcal{FCI} \mid f_1 \subset f$ *and* $\nexists\ f_2 \in \mathcal{FCI}$ *s.t.* $f_1 \subset f_2 \subset f\}$.

Example 16 *Consider the* CI `ABEF` *of the Iceberg lattice of Figure 2.2. Then, we have:* $Cov_l($`ABEF`$) =$ {`E`, `F`, `AB`}.

2.4 Association Rule Extraction

2.4.1 Association Rule Framework

As an important topic in data mining, association rule mining research [Ceglar and Roddick, 2006] has progressed in various directions since its inception. The formalization of the association rule extraction problem was initially introduced by Agrawal *et al.* [Agrawal *et al.*, 1993]. The derivation of association rules is achieved starting from the set $\mathcal{FI}$ of frequent itemsets extracted from a context $\mathcal{K}$, for a minimal support threshold *minsupp*. The next definitions introduce the association rule framework.

Definition 27 (Association rule)
An association rule R is a relation between itemsets and is of the form R: $X \Rightarrow (Y \backslash X)$, such that X and Y are two itemsets, and $X \subset Y$. The itemsets X and $(Y \backslash X)$ are, respectively, called the `premise` *(or* `antecedent`*) and the* `conclusion` *(or* `consequent`*) of the association rule R.*

Definition 28 (Support, Confidence of an association rule)
Let R: $X \Rightarrow (Y \backslash X)$ be an association rule. The support of R, $Supp(R)$, is equal to $Supp(Y)$. While its confidence is equal to $Conf(R) = \frac{Supp(Y)}{Supp(X)}$.

Note that the confidence of R is always greater than or equal to its frequency: $Conf(R) \geq Freq(R) = \frac{Supp(R)}{|\mathcal{O}|}$. Indeed, we have $Supp(X) \leq |\mathcal{O}|$.

Definition 29 (Valid, Exact, Approximate association rule)
An association rule R is said to be `valid` *(or* `strong`*) if:*

- *its support value $Supp(R)$ is greater than or equal to the user-specified threshold, minsupp, and,*
- *its confidence value $Conf(R)$ is greater than or equal to a user-specified threshold, denoted minconf.*

If $Conf(R) = \mathbf{1}$, then R is called `exact` *association rule, otherwise it is called* `approximate` *association rule.*

Notation 2 *As for the minsupp threshold, the value of minconf can be sketched in percentage in the tables and figures presenting experimental results.*

Given user-specified minimum support and confidence, the problem of association rule mining can be split into two steps as follows [Agrawal *et al.*, 1993]:

- Extract all frequent itemsets, *i.e.*, having the support value greater than or equal to *minsupp*.
- Generate valid association rules from frequent itemsets. This generation is limited to rules having the confidence value greater than or equal to *minconf*.

These steps are solved by a pioneer algorithm in the data mining field, namely Apriori [Agrawal and Srikant, 1994]. The second step is relatively straightforward. However, the first one presents a great challenge because the set of frequent itemsets may grow exponentially with $|\mathcal{I}|$. The problem of discovering association rules is an exponential one (in the length of the longest frequent itemset). In

fact, from a frequent itemset I, $\mathbf{2}^{|I|}$ - $\mathbf{1}$ non trivial association rules can be generated. However, these rules can be generated in a straightforward manner, *i.e.*, without accessing the disk resident context. Therefore, the cost of this step is found to be low compared to that of the frequent itemsets extraction. An efficient algorithm, called GEN-RULES, is proposed in [Agrawal *et al.*, 1993] for generating association rules starting from the set of frequent itemsets. Nevertheless, the number of association rules generated may grow up to several millions [Stumme *et al.*, 2001, Zaki, 2004]. In addition, it was proven that a large number of rules are *redundant* in the sense that they convey the same information as others [Ashrafi *et al.*, 2007, Ben Yahia *et al.*, 2009b].

2.4.2 Generic Bases of Association Rules

In the literature related to association rule mining, the problem of the relevance and the usefulness of association rules is of paramount importance. Indeed, an overwhelming quantity of association rules can be extracted even from small real-life datasets, among which a large number is *redundant* (*i.e.*, conveying the same information) [Bastide *et al.*, 2000a, Stumme *et al.*, 2001, Zaki, 2004]. This is always a real hamper towards their effective exploitation by the end-users. The survival of the association rule extraction technique is thus owed to the retrieval of compactly sized with added-value knowledge. Many approaches were hence proposed for reducing large sized sets of association rules, while preserving the most interesting ones. For example, some works relied on the use of other quality measures, in addition to the support and the confidence, like *lift*, *conviction*, *dependency*, etc. [Geng and Hamilton, 2006, Guillet and Hamilton, 2007], while others introduced user-defined constraints during the mining process or as a post-processing step [Bonchi and Lucchese, 2006, Boulicaut and Jeudy, 2001, Lee *et al.*, 2006, Srikant *et al.*, 1997]. Another interesting approach consists in extracting *generic bases*. Such an approach is mainly based on the closure operators of the Galois connection used in Formal Concept Analysis (FCA) [Ganter and Wille, 1999].

The FCA-based approach focuses on extracting irreducible nuclei of all association rules – generic bases – from which the remaining *redundant* association rules can be derived without information loss. These generic bases hence define a compact set of relevant association rules that is easier to interpret for the end-user. Since their size is reduced and as they are generating sets, they also constitute efficient solutions for the long-term storage on secondary memories and the computer-aided management of a set of valid association rules [Pasquier, 2009]. According to the approach based on FCA, the redundancy within association rules is defined as follows [Bastide *et al.*, 2000a]:

Definition 30 (ASSOCIATION RULE REDUNDANCY)
Let $\mathcal{AR}$ be the set of valid association rules that can be drawn from a context $\mathcal{K}$ for a minimum support threshold minsupp and a minimum confidence threshold minconf. An association rule R_1: $X_1 \Rightarrow Y_1 \in \mathcal{AR}$ is said redundant with respect to (or derivable from) a rule R_2: $X_2 \Rightarrow Y_2 \in \mathcal{AR}$ iff:

1. *$Supp(R_1) = Supp(R_2)$ and $Conf(R_1) = Conf(R_2)$, and,*
2. *$X_2 \subseteq X_1$ and $Y_1 \subset Y_2$.*

Example 17 *Suppose we have both rules R_1: `ABC` $\Rightarrow$ `D` and R_2: `ABC` $\Rightarrow$ `DE`. If `ABCD` and `ABCDE` belong to the same γ-equivalence class, then $Supp($`ABCD`$) = Supp($`ABCDE`$)$. Hence, $Supp(R_1) = Supp(R_2)$. Moreover, $Conf(R_1) = Conf(R_2)$, since they have the same premise. Indeed, $Conf(R_1) = \frac{Supp(R_1)}{Supp(\texttt{ABC})} =$*

$\frac{Supp(R_2)}{Supp(ABC)} = Conf(R_2)$. *Consequently,* R_1 *is redundant w.r.t.* R_2 *since they share the same support and confidence values while* $D \subset DE$.

Based on Definition 30, for an association rule $X_1 \Rightarrow Y_1$, if there is no other rule $X_2 \Rightarrow Y_2$ such that $Supp(R_1) = Supp(R_2)$, $Conf(R_1) = Conf(R_2)$, $X_2 \subseteq X_1$, and $Y_1 \subset Y_2$, then $X_1 \Rightarrow Y_1$ is said *minimal non-redundant* [Bastide *et al.*, 2000a]. Note that this definition ensures that non-redundant association rules will hence have minimal premises and maximal conclusions.

Example 18 *Let* R_1: $ABC \Rightarrow DE$ *be an association rule and suppose that* $Supp(ABC) = Supp(AB)$. *Then,* R_1 *is clearly redundant w.r.t.* R_2: $AB \Rightarrow CDE$.

In addition, suppose that we have a rule R_3: $BC \Rightarrow DE$ *and* $Supp(BCDE) = Supp(ABCDE)$. *Then,* R_3 *is redundant w.r.t. the rule* R_4: $BC \Rightarrow ADE$.

In both cases, the premise part is required to be minimal, while the conclusion part to be maximal.

In fact, the correctness of Definition 30 relies on the fact that R_1 can be derived starting from R_2 without information loss. Since the appearance of the approach adapting the FCA to association rule mining through the extraction of frequent closed itemsets, several generic association rule bases were introduced [Kryszkiewicz, 2002]. Definition 31 describes the properties that characterize a generic basis once it is extracted without loss of information.

Definition 31 (Generic basis properties)
A generic basis $\mathcal{B}$, *to which is associated an appropriate inference mechanism, is said to fulfill the ideal properties of an association rule representation if it is [Kryszkiewicz, 2002]:*

1. ***lossless**: $\mathcal{B}$ must enable the derivation of all valid association rules,*
2. ***sound**: $\mathcal{B}$ must forbid the derivation of association rules that are not valid, and,*
3. ***informative**: $\mathcal{B}$ must allow to exactly retrieve the support and confidence values of each derived association rule.*

The generic basis $\mathcal{B}$ *is said to verify the property of derivability if it is lossless and sound.*

If a generic basis fulfills the aforementioned properties, then it ensures the regeneration of redundant valid association rules without information loss. It also makes it possible the derivation of their exact support and confidence values. The majority of generic bases convey association rules presenting implications between minimal generators and closed itemsets [Ashrafi *et al.*, 2007, Bastide *et al.*, 2000a, Ben Yahia *et al.*, 2009b, Kryszkiewicz, 2002, Li, 2006, Pasquier, 2009]. This ensures obtaining association rules with minimal premise part and maximal conclusion part. Such rules convey the maximum of information, and are hence qualified as the most informative association rules [Bastide *et al.*, 2000a, Kryszkiewicz, 1998, Pasquier, 2009]. Indeed, they offer the maximum of information (conveyed by the items in the conclusion part) using the minimum of conditions (through the items in the premise part). In this respect, these rules are the preferred ones *w.r.t.* the minimum description length principle (MDLP) since patterns with the shortest description are privileged [Grunwald, 2007, Rissanen, 1978].

In [Zaki, 2004], the authors propose rules with minimal premise part and minimal conclusion part. There are also other forms of rules using pseudo-closed itemsets or closed itemsets in premise, like the

Guigues-Duquenne basis [Guigues and Duquenne, 1986] and the Luxenburger basis [Luxenburger, 1991] respectively. Note that the minimum description length principle can also be used in the context of association rules to evaluate the conciseness of a rule set [Geng and Hamilton, 2006]. The shortest premises, *i.e.*, those composed by MGs, are thus the most preferred ones according to this principle.

2.4.3 Extraction of Informative Association Rules

In the following, we present a couple of association rule subsets that are sufficient to derive the whole set of redundant association rules. This couple is defined as follows [Bastide *et al.*, 2000a]:

1. The *generic basis for exact association rules* is defined as follows:

   Definition 32 (Generic basis for exact association rules)
   Let $\mathcal{FCI}$ be the set of frequent closed itemsets extracted from a context $\mathcal{K}$. For each entry f in $\mathcal{FCI}$, let MG_f be the set of its minimal generators. The generic basis for exact association rules $\mathcal{GB}$ is given by: $\mathcal{GB} = \{R\text{: } g \Rightarrow (f \setminus g) \mid f \in \mathcal{FCI} \text{ and } g \in \mathrm{MG}_f \text{ and } g \neq f\}$. [1]

2. The transitive reduction of the informative basis [Bastide *et al.*, 2000a], which is a cover of all approximate association rules, is defined as follows:

   Definition 33 (Transitive reduction of the informative basis)
   Let $\mathcal{FMG}$ be the set of frequent minimal generators extracted from a context $\mathcal{K}$. The transitive reduction $\mathcal{RI}$ is given by: $\mathcal{RI} = \{R \mid R\text{: } g \Rightarrow (f \setminus g) \mid f \in \mathcal{FCI} \text{ and } g \in \mathcal{FMG} \text{ and } \gamma(g) \in Cov^u(f) \text{ and } Conf(R) \geq minconf\}$.

In comparison to the other generic bases, the basis ($\mathcal{GB}$, $\mathcal{RI}$) fulfills the ideal properties of an association rule representation (summarized by Definition 31) [Kryszkiewicz, 2002]. The association rules forming this couple are minimal non-redundant ones [Bastide *et al.*, 2000a] *w.r.t.* Definition 30. They are also informative, since having minimal premises and maximal conclusions. This property simplifies the interpretation by the end-user, as association rules with smaller premises are easier to interpret, and the information in each rule is maximized to minimize their number [Pasquier, 2009]. In this respect, this couple also offers interesting compactness rates *vs.* the whole set of association rules when compared to the remaining representations [Bastide *et al.*, 2000a, Ben Yahia *et al.*, 2009b].

Given an *Iceberg lattice* – in which each frequent closed itemset is decorated by its list of minimal generators – the derivation of the basis ($\mathcal{GB}$, $\mathcal{RI}$) can be straightforwardly performed [Hamrouni *et al.*, 2005b]. Indeed, approximate generic association rules represent "inter-node" implications, assorted with the confidence value, between two adjacent comparable γ-equivalence classes, *i.e.*, from a frequent closed itemset to another frequent closed itemset immediately covering it. For example, referring to the *Iceberg lattice* depicted by Figure 2.2 and Table 2.2, the approximate generic association rule $\mathtt{C} \overset{0.50}{\Rightarrow} \mathtt{ABD}$ is generated from both γ-equivalence classes topped respectively by the frequent closed itemsets `CD`, having `C` for minimal generator, and `ABCD`. Conversely, exact generic association rules are "intra-node" implications, with a confidence value equal to **1**, extracted from each node in the partially ordered structure. For example, from the closed itemset `CDE`, this exact generic association rule is derived: `CE`$\Rightarrow$`D`.

[1] The condition $g \neq f$ ensures discarding non-informative association rules of the form $g \Rightarrow \emptyset$.

Therefore, the problem of mining association rules may be reformulated under the redundancy removal point of view as follows [Ben Yahia *et al.*, 2006]:

1. Discover frequent CIs and their associated MGs. Also, the upper cover of each frequent CI should be available.
2. From the information discovered in the first step, derive generic bases of association rules (from which all remaining rules can be derived).

2.5 Conclusion

A traditional problem of the data mining field is the search for association rules in databases, introduced by Agrawal *et al.* [Agrawal *et al.*, 1993]. Given the high number of frequent itemsets, and consequently the high number of the (redundant) association rules that can be drawn even from small amount of data, a new approach relying on the mathematical background of Formal Concept Analysis was proposed. The main purpose is to reduce the number of extracted rules without information loss.

In the next chapter, we will give a thorough survey of the main concise representations of frequent itemsets proposed in the literature. A comparative study of these representations will also be presented.

Chapter 3

Main Concise Representations of Frequent Itemsets

3.1 Introduction

The growth of the interest in the frequent itemset mining is owed to the usefulness of frequent itemsets in many important fields. Indeed, thanks to these itemsets, human experts can obtain pertinent correlations between dataset items. Therefore, they can acquire a deeper understanding thanks to the mined patterns. In real-life datasets, carried out experiments showed that the number of frequent itemsets is huge when the dataset is dense or the minimum support threshold is set too low [Calders *et al.*, 2005]. This phenomenon makes the exploitation and the handling of such amount of extracted knowledge very difficult. In order to offer a manageable set of elements from which the derivation of all frequent itemsets is possible, the notion of exact concise representation was introduced. In the literature, many exact concise representations were proposed, amongst which the main ones are those based on:

1. The frequent closed itemsets [Pasquier *et al.*, 1999b],
2. The frequent minimal generators [Boulicaut *et al.*, 2003, Liu *et al.*, 2007, Phan Luong, 2002],
3. The frequent non-derivable itemsets [Calders and Goethals, 2007],
4. The frequent closed non-derivable itemsets [Muhonen and Toivonen, 2006], and,
5. The frequent essential itemsets [Casali *et al.*, 2005a].

The exact representations based on the set of frequent minimal generators are not described hereafter since they were shown in the literature to be always of larger size than that based on frequent closed itemsets [Calders and Goethals, 2003]. Nevertheless, we will establish the link of each described representation with minimal generators. Indeed, these key itemsets will be proved to be at the roots of the different representations.

In this chapter, we thus focus on exact concise representations of frequent itemsets. It is however important to mention that many approximate concise representations were proposed in the literature, like maximal frequent itemsets [Bayardo, 1998] (and their dual, *i.e.*, minimal infrequent itemsets), δ-free sets [Boulicaut *et al.*, 2003], condensed frequent pattern bases [Pei *et al.*, 2004], δ-clusters [Xin

et al., 2007], and δ-tolerance closed frequent itemsets [Cheng *et al.*, 2006]. Although, they offer very high compactness rates, we did not treat them since they do not allow deriving the exact frequency of itemsets. Moreover, their accuracy closely depends on the tolerated error bound. The use of concise representations was also extended to many pattern classes. For example, they are at the roots of different works aiming at concisely representing pattern classes such as association rules [Ceglar and Roddick, 2006], associative classification rules [Baralis and Chiusano, 2004], sequential patterns [Balcázar and Casas-Garriga, 2007, Lo *et al.*, 2008, Raïssi *et al.*, 2008], graphs [Yan and Han, 2003], trees [Balcázar *et al.*, 2007], minimal transversals [Hébert *et al.*, 2007], multidimensional patterns [Casali *et al.*, 2009, Pei *et al.*, 2006], etc.

In the remainder, we describe the main exact concise representations of frequent itemsets that were proposed in the literature. Then, we carry out a critical comparative study of the surveyed representations.

3.2 Frequent Closed Itemset-based Representation

3.2.1 Description

The concise representation based on frequent closed itemsets (CIs) was introduced by Pasquier *et al.* [Pasquier *et al.*, 1999b]. The set of frequent CIs is defined as follows:

Definition 34 (Set of frequent closed itemsets)
Consider a context $\mathcal{K}$ and the closure operator γ. The set of frequent closed itemsets that can be drawn from $\mathcal{K}$, denoted $\mathcal{FCI}$, is defined as follows: $\mathcal{FCI} = \{I \subseteq \mathcal{I} \mid \gamma(I) = I$ and $Supp(I) \geq minsupp\}$.

The smallest closed itemset, *w.r.t.* set inclusion, containing an itemset I is obtained by applying γ on I. Since I and its closure belong to the same γ-equivalence class, then we have $Supp(I) = Supp(\gamma(I))$.

Theorem 1 states that the set of frequent CIs represents an exact concise representation of the set $\mathcal{FI}$ of frequent itemsets.

Theorem 1 *The set $\mathcal{FCI}$ of frequent closed itemsets, associated to their respective supports, is an exact concise representation of the set $\mathcal{FI}$ [Pasquier* et al.*, 1999b].*

Indeed, given an itemset I, thanks to $\mathcal{FCI}$, we are able to guess whether I is frequent or not. In the affirmative case, its exact support can be derived from $\mathcal{FCI}$. In the remainder, the representation based on $\mathcal{FCI}$ will be denoted FCIs_rep.

3.2.2 Mining Algorithm

Thanks to the large success of this representation, many algorithms were proposed to extract the set of frequent CIs (*e.g.* Close [Pasquier *et al.*, 1999b], LCM [Uno *et al.*, 2004], and Prince [Hamrouni *et al.*, 2005b]). Many comparative studies were hence carried out in the literature on these algorithms [Ben Yahia *et al.*, 2006, Ceglar and Roddick, 2006, Goethals and Zaki, 2003, Zheng *et al.*, 2001]. In this respect, the frequent itemset mining implementations (FIMI) repository [Bayardo *et al.*, 2004] offers efficient implementations dedicated to the extraction of frequent CIs.

Example 19 *Consider the context given by Table 2.1 (cf. page 12) and consider minsupp* $= \mathbf{1}$*. The application of the* A-Close *algorithm [Pasquier* et al.*, 1999a] for example allows the extraction of the set of frequent* CIs*. This algorithm mines in a first step the set* $\mathcal{FMG}$ *of frequent minimal generators thanks to a levelwise traversal of the search space. For each candidate, it checks whether it is a frequent minimal generator using Definition 7 (cf. page 20). For example,* AC *is a frequent minimal generator since* $Supp(AC) = \mathbf{2} \geq minsupp$ *and* $Supp(AC) < min\{Supp(\emptyset), Supp(A), Supp(C)\} = \mathbf{3}$*. It also benefits from the key property of the set* $\mathcal{FMG}$ *being an order ideal (cf. Proposition 6, page 20). For example, since the itemset* $CD \notin \mathcal{FMG}$*, its super-set* ACD *cannot be a frequent minimal generator.*

Once the set $\mathcal{FMG}$ *extracted,* A-Close *delves in the context to get the closure of each frequent minimal generator. For each element,* A-Close *computes of the transaction in which it appears. For example, the closure of* AC *results from the intersection of transactions* **1** *and* **4***, i.e.,* $ABCD$ *and* $ABCDEF$*, and is hence equal to* $ABCD$*.*

The obtained set $\mathcal{FCI}$ *is sketched by Table 2.2 (cf. page 21). Given this set, we are able to derive the conjunctive support of any frequent itemset. Suppose we are interested in deriving the support of* AB*. Since the latter is a frequent* CI*, then its support is equal to* **3***. Suppose now we are interested in deriving the support of the itemset* ABC*. This latter is not a frequent* CI*. We then search for the smallest frequent* CI *containing* ABC*, i.e., the frequent* CI $ABCD$*. Since an itemset has the same support as its closure, we have* $Supp(ABC) = Supp(ABCD) = \mathbf{2}$*.*

3.2.3 Discussion

The cardinality of the representation based on frequent closed itemsets cannot exceed the cardinality of the whole set of frequent itemsets. It is hence a *perfect cover* [Pasquier *et al.*, 1999b]. Moreover, this representation was extensively used as a starting point for extracting generic bases of rules of association. Such bases allow to concisely represent valid association rules while allowing the faithful retrieval of their respective support and confidence values. Unfortunately, this representation is not very attractive whenever handling weakly correlated (or sparse) contexts. Indeed, in these contexts, each itemset is often equal to its closure. Thus, the reduction ratio of this concise representation becomes very poor.

3.2.4 Link with Minimal Generators

As described in Chapter 2 (*cf.* page 19), closed itemsets and minimal generators are closely related. Indeed, once the Galois closure operator γ applied on the power-set of items, it partitions itemsets in γ-equivalence classes. In each class, a CI is the unique maximal element *w.r.t.* set inclusion while at least a MG is a minimal one. MGs are hence the first reachable elements within each class which explains why many algorithms rely on, for efficiently extracting CIs.

From a concise representation point of view, frequent MGs do not constitute by themselves an exact concise representation of frequent itemsets. They hence must be augmented by other itemsets, like the infrequent [Kryszkiewicz, 2001] or frequent [Liu *et al.*, 2007] part of the associated negative border. It is then clear that the FCIs_rep is always smaller than frequent MG-based representations. Nevertheless, the order ideal property of the frequent MG set motivated the efficient extraction of such representations [Hamrouni *et al.*, 2006, Hamrouni *et al.*, 2005b, Liu *et al.*, 2007].

The next section proposes a generalization of frequent minimal generators leading to the definition of the concise representation based on frequent non-derivable itemsets.

3.3 Frequent Non-Derivable Itemset-based Representation

3.3.1 Description

The notion of non-derivable itemset was introduced in [Calders and Goethals, 2002]. In order to present the non-derivability property, we need to recall the notion of deduction rule, presented in the following definition.

Definition 35 (Deduction rule)
Let I, $J \subseteq \mathcal{I}$ be two itemsets s.t. $I \subseteq J$. The deduction rule, linking the support of J to that of I, denoted by $\mathcal{R}_I(J)$, is one of the following inequalities [Calders and Goethals, 2007]:

$$\begin{aligned} Supp(J) &\leq \sum_{I \subseteq J' \subset J} (\text{-}1)^{|J \setminus J'|+1} Supp(J') \quad \text{if } |J \setminus I| \text{ is odd,} \\ Supp(J) &\geq \sum_{I \subseteq J' \subset J} (\text{-}1)^{|J \setminus J'|+1} Supp(J') \quad \text{if } |J \setminus I| \text{ is even} \end{aligned}$$

The definition of a frequent non-derivable itemset is then as follows:

Definition 36 (Frequent non-derivable itemset)
Let $J \subseteq \mathcal{I}$ be an itemset. J is said to be `non-derivable` if there is no couple of deduction rules from which the support of J can exactly be retrieved. J is a `frequent non-derivable` itemset if J is also `frequent` [Calders and Goethals, 2007].

Roughly speaking, an itemset is said *non-derivable* if the combination of the supports of its subsets does not allow to exactly derive its support. Otherwise, it is said *derivable*. Thus, deduction rules partition the set of frequent itemsets into two disjoint subsets: the first contains non-derivable itemsets, *i.e.*, those for which an access to the context is required for computing their exact supports. These itemsets will be retained in the representation. The second subset contains derivable itemsets, whose respective supports can be exactly derived given those of their associated proper subsets. The next theorem states an important result about deduction rules.

Theorem 2 *Let $J \subseteq \mathcal{I}$. The deduction rules associated to J, i.e., $\{\mathcal{R}_I(J) \mid I \subseteq J\}$, are sound and complete for deducing tight upper and lower bounds on the support of J [Calders, 2004].*

Example 20 *Consider the context depicted by Table 2.1 (cf. page 12). In order to simplify the notations, we will use in this example the notation S_I to denote the support of I. Let us determine the support of the itemset `ABC` using deduction rules. The set of deduction rules associated to `ABC` is given by the following*

system:

$$\begin{cases} S_{ABC} \leq S_\emptyset - S_A - S_B - S_C + S_{AB} + S_{AC} + S_{BC} & (\mathcal{R}_\emptyset) \\ S_{ABC} \geq -S_A + S_{AB} + S_{AC} & (\mathcal{R}_A) \\ S_{ABC} \geq -S_B + S_{AB} + S_{BC} & (\mathcal{R}_B) \\ S_{ABC} \geq -S_C + S_{AC} + S_{BC} & (\mathcal{R}_C) \\ S_{ABC} \leq S_{AB} & (\mathcal{R}_{AB}) \\ S_{ABC} \leq S_{AC} & (\mathcal{R}_{AC}) \\ S_{ABC} \leq S_{BC} & (\mathcal{R}_{BC}) \\ S_{ABC} \geq 0 & (\mathcal{R}_{ABC}) \end{cases}$$

Let us detail the method for obtaining these rules. Let $\mathcal{R}_I$ be a given deduction rule, and $\sigma(I,J)$ be the right part of $\mathcal{R}_I$ (for example, $\sigma(B, ABC)$ = - S_B + S_{AB} + S_{BC}). Each deduction rule is hence in the form of $S_J \lessgtr \sigma(I,J)$. The sign of the inequality is obtained from the cardinality of $|J \backslash I|$. Indeed, according to Definition 35, if $|J \backslash I|$ is even (resp. odd), then $\sigma(I,J)$ is a lower bound (resp. upper bound) of S_J. Consequently, $\mathcal{R}_I \equiv S_J \geq \sigma(I,J)$ (resp. $\mathcal{R}_I \equiv S_J \leq \sigma(I,J)$).

Consider for example the rule $\mathcal{R}_\emptyset$. Since $|ABC \backslash \emptyset| = \mathbf{3}$, then $\sigma(\emptyset, ABC)$ is an upper bound and, hence, we have $\mathcal{R}_\emptyset \equiv S_{ABC} \leq \sigma(\emptyset, ABC)$. Moreover, since $\sigma(\emptyset, ABC) = S_\emptyset - S_A - S_B - S_C + S_{AB} + S_{AC} + S_{BC}$, we obtain $\mathcal{R}_I \equiv S_{ABC} \leq S_\emptyset - S_A - S_B - S_C + S_{AB} + S_{AC} + S_{BC}$.

By numerically assessing the above system of deduction rules, we obtain the following system:

$$\begin{cases} S_{ABC} \leq 2 & (\mathcal{R}_\emptyset) \\ S_{ABC} \geq 2 & (\mathcal{R}_A) \\ S_{ABC} \geq 2 & (\mathcal{R}_B) \\ S_{ABC} \geq 0 & (\mathcal{R}_C) \\ S_{ABC} \leq 3 & (\mathcal{R}_{AB}) \\ S_{ABC} \leq 2 & (\mathcal{R}_{AC}) \\ S_{ABC} \leq 2 & (\mathcal{R}_{BC}) \\ S_{ABC} \geq 0 & (\mathcal{R}_{ABC}) \end{cases}$$

From rules $\mathcal{R}_\emptyset$ and $\mathcal{R}_A$, we deduce that $S_{ABC} = \mathbf{2}$. Let us remark that in this case, using these deduction rules, we are able to exactly compute the support of S_{ABC} without accessing the extraction context. This is nevertheless conditioned by the fact that the associated supports of all proper subsets of ABC must be known.

The next theorem states that the set $\mathcal{NDI}$ of frequent non-derivable itemsets is an exact representation of frequent itemsets.

Theorem 3 *The set $\mathcal{NDI}$ of frequent non-derivable itemsets, associated to their respective supports, is an exact concise representation of the set $\mathcal{FI}$ [Calders and Goethals, 2007].*

In the remainder, the representation based on $\mathcal{NDI}$ will be denoted NDI*s_rep*.

3.3.2 Mining Algorithm

The frequent non-derivable itemsets fulfill the order ideal property. Indeed, "to be non-derivable" was shown to be an anti-monotone constraint [Calders and Goethals, 2007]. Since the conjunction of two

anti-monotone constraints, namely "to be non-derivable" and "to be frequent", is also anti-monotone, then the set $\mathcal{NDI}$ forms an order ideal as stated by the following proposition:

Proposition 8 *The set $\mathcal{NDI}$ of frequent non-derivable itemsets is an order ideal in $(\mathcal{P}(\mathcal{I}), \subseteq)$.*

Such a constraint is used as an efficient way for pruning candidates. In this respect, Calders and Goethals proposed a breadth-first algorithm, called NDI [Calders and Goethals, 2007], to extract the frequent non-derivable itemsets. Note that in [Calders and Goethals, 2005], they also proposed a depth-first algorithm, called dfNDI, for mining this representation. The search space is traversed right-to-left, ensuring to reach an itemset after all its proper subsets were already treated.

Example 21 *Consider the context depicted by Table 2.1 (cf. page 12). For minsupp = 1, the set $\mathcal{NDI}$ of frequent non-derivable itemsets is shown by Table 3.1.*

Frequent non-derivable itemset	Support
$\emptyset$	5
A	3
B	3
C	4
D	4
E	3
F	3
AB	3
CD	4

Table 3.1: The set $\mathcal{NDI}$ for *minsupp* = 1.

3.3.3 Discussion

The main advantage of this representation is that it has a reduced cardinality for the majority of the real-life contexts. Moreover, it has made possible to define a new way for characterizing association rules through the so-called *non-derivable association rules* [Goethals *et al.*, 2005]. Nevertheless, non-derivable itemsets do not have particular semantics being able to bring the end-users with further information about the mined context. Indeed, they are based on a numerical reduction without any added-value from the structural point of view, contrary to minimal generators and closed itemsets. In addition, the regeneration process of frequent itemsets starting from this representation is very expensive [Liu *et al.*, 2007, Mielikäinen *et al.*, 2006]. Indeed, for a derivable itemset of size n, the computation process of its support is performed in two steps: the first consists in checking whether all its proper subsets are frequent, otherwise it will be infrequent and the process will stop. The second step consists in evaluating $\mathbf{2}^n$ deduction rules. The evaluation of $\mathbf{2}^n$ deduction rules is also usually required during the extraction process for a candidate itemset having all its proper subsets frequent non-derivable itemsets, *i.e.*, verifying the anti-monotone constraint of being frequent non-derivable. Note however that some optimizations were proposed by the authors trying to optimise the evaluation cost of deduction rules.

3.3.4 Link with Minimal Generators

It is worth noting that the representation based on frequent non-derivable itemsets is basically a generalization of minimal generators with respect to the considered subsets of a given itemset. Indeed, for testing if an arbitrary itemset is a minimal generator, its support is only compared with those of its immediate subsets. On the other hand, the frequent non-derivable itemsets rely on a larger *neighborhood exploration* (also called *depth* in [Calders and Goethals, 2003]) since using all proper subsets of an itemset. This explains why the representation based on frequent non-derivable itemsets is smaller than that based on frequent minimal generators. Indeed, the former requires by far more comparisons through deduction rules than the latter. For example, for an itemset of size n, $\mathbf{2}^n$ deduction rules need to be evaluated to check whether it is non-derivable or not, while only n are used for checking its minimality status within the associated γ-equivalence class. Note however that the determination of the lower and upper support bound may be stopped when their current temporary values become equal. On the other hand, the regeneration process of frequent itemsets starting from frequent non-derivable itemsets is awfully costly [Liu *et al.*, 2007, Mielikäinen *et al.*, 2006] compared to that starting from frequent minimal generator-based representations [Liu *et al.*, 2007].

Noteworthily, other exact concise representations can be categorized in the same pool as non-derivable itemsets being also different generalizations of minimal generators with respect to the used subset neighborhood. These representations are those based on disjunction-free sets [Bykowski and Rigotti, 2001, Bykowski and Rigotti, 2003] and (generalized) disjunction-free generators [Kryszkiewicz, 2002], etc. A unified view of most of these representations was proposed in [Calders and Goethals, 2003].

The next section presents an exact concise representation of frequent itemsets resulting from a combination of the concept of non-derivability and the Galois closure operator.

3.4 Frequent Closed Non-Derivable Itemset-based Representation

3.4.1 Description

The frequent closed non-derivable itemsets have been introduced by Muhonen and Toivonen [Muhonen and Toivonen, 2006]. This representation combines the concise representations presented above, namely those based on frequent closed itemsets and frequent non-derivable itemsets, respectively. Indeed, its basic idea consists in applying the Galois closure operator on frequent non-derivable itemsets in order to generate a more compact representation than the set of frequent non-derivable itemsets. The next theorem states that the obtained set preserves the exactness of the regeneration process of frequent itemsets.

Theorem 4 *The set $\mathcal{CNDI}$ of frequent closed non-derivable itemsets, associated to their respective supports, is an exact concise representation of the set $\mathcal{FI}$ [Muhonen and Toivonen, 2006].*

In the remainder, we will denote this concise representation by CNDIs_rep.

3.4.2 Mining Algorithm

For determining the set $\mathcal{CNDI}$, the first step consists in extracting the set $\mathcal{NDI}$ using one of the dedicated algorithms to this task (*cf.* previous section). Then, the second step is devoted to the computation of the respective closures of the elements of $\mathcal{NDI}$ by means of an additional access to the extraction context. [1]

Example 22 *Consider the context given by Table 2.1 (cf. page 12). For minsupp* $= \mathbf{1}$*, the set* $\mathcal{CNDI}$ *of frequent closed non-derivable itemsets is depicted by Table 3.2. For example, consider the frequent non-derivable itemset* A*. Since its closure is* AB*, then* AB $\in \mathcal{CNDI}$*.*

Frequent closed non-derivable itemset	Support
$\emptyset$	5
E	3
F	3
AB	3
CD	4

Table 3.2: The set $\mathcal{CNDI}$ for *minsupp* $= \mathbf{1}$.

3.4.3 Discussion

Since the concise representation CNDIs_rep simply gathers the conjunctive closures of frequent non-derivable itemsets, its cardinality is always smaller than or equal to those of NDIs_rep and FCIs_rep. Nevertheless, its extraction is quite complicated since relying on the extraction of frequent non-derivable itemsets. An additional access to the context is also required to compute closures. The regeneration of frequent itemsets from this representation inherits the same difficulty when using frequent non-derivable itemsets. Finally, the representation based on frequent closed non-derivable itemsets also lacks a semantic part since it is simply resulting from taking closure of numerically retained itemsets, *i.e.*, the non-derivable ones.

3.4.4 Link with Minimal Generators

The link of the frequent closed non-derivable itemset-based representation with minimal generators derives from those of its "ancestors" with this important concept. Indeed, this representation is in fact the result of taking closures of a generalization of minimal generators *w.r.t.* the used neighborhood, namely non-derivable itemsets.

Moreover, the computation of frequent closed non-derivable itemsets can be optimized if we further concentrate on minimal generators. Indeed, each closed non-derivable itemset can easily be shown to be the closure of at least a non-derivable minimal generator. By "non-derivable minimal generator", we indicate an itemset which is both "non-derivable" and "minimal generator". Hence, instead of computing

[1] The authors offer such an algorithm, called FIRM, whose source code is available on Muhonen's website at: *http://www.cs.helsinki.fi/u/jomuhone/firm/firm-3-3-3.tar.gz*.

the whole set of frequent non-derivable itemsets for which the associated closures must be computed, we can only use the set of frequent non-derivable minimal generators. This set being the result of three anti-monotone constraints, namely "to be frequent", "to be non-derivable" and "to be minimal generator", will give rise to an order ideal. Indeed, the conjunction of the three aforementioned anti-monotone constraints will also give an anti-monotone constraint.

To get out the set of frequent non-derivable minimal generators, a slight modification of algorithms dedicated to frequent non-derivable itemset mining has to be performed. Its main aim is to only retain the itemsets fulfilling the minimal generator constraint among the set of frequent non-derivable itemsets. A comparison between the actual support of a frequent non-derivable itemset and the minimum of its immediate subsets supports is hence sufficient. The detection of minimal generators within these algorithms will hence optimize the candidate generation and closure computation steps. Indeed, the number of frequent non-derivable minimal generators is lower than that of frequent non-derivable itemsets.

The next section presents the unique concise representation for frequent itemsets relying on the disjunctive support, in addition to the conjunctive one.

3.5 Frequent Essential Itemset-based Representation

3.5.1 Description

The representation based on frequent essential itemsets was introduced by Casali et *al.* [Casali *et al.*, 2005a]. The notion of essential itemset is based on the inclusion-exclusion principle [Galambos and Simonelli, 2000, Narushima, 1982]. In this respect, this representation offers the possibility to retrieve all kinds of support sketched in Definition 2 (*cf.* page 12).

In this subsection, we will mainly concentrate on the conjunctive and the disjunctive supports of an itemset. The negative support can easily be derived from the disjunctive support as shown in Definition 2. Hence, to avoid confusion between both supports, we will explicitly mention the nature of the support: conjunctive or disjunctive. Nevertheless, the use of the term "*frequent*" is restricted to the conjunctive support in the sense that a *frequent* itemset have a conjunctive support greater than or equal to *minsupp*.

Let us begin by introducing the definition of frequent essential itemset.

Definition 37 (Frequent essential itemset)
An itemset $I \subseteq \mathcal{I}$ is `essential` *if $Supp(\vee I) > max\{Supp(\vee(I \setminus \{i\})) \mid i \in I\}$. I is a* `frequent essential` *itemset if it is simultaneously* `frequent` *and* `essential`*.*

Example 23 *Consider the context given by Table 2.1 (cf. page 12) for minsupp = 1. The itemset AB is not an essential itemset since $Supp(\vee AB) = Supp(\vee A) = 3$. Whereas AC is an essential itemset since $Supp(\vee AC) = 5$, in conjunction with the facts that $Supp(\vee AC) \neq Supp(\vee A)$ (since $Supp(\vee A) = 3$) and $Supp(\vee AC) \neq Supp(\vee C)$ (since $Supp(\vee C) = 4$). The itemset AC is also frequent since $Supp(AC) = 2 \geq$ minsupp.*

Remark 1 *It is important to note that Definition 37 is slightly modified w.r.t. the original one given in [Casali* et al.*, 2005a]. Indeed, it also include the implicit consideration of the empty set as an essential*

itemset although the disjunctive support is not defined on this pattern since it does not contain any item. This consideration is argued by the fact that this will allow ensuring that the set of essential itemsets is an order ideal in $(\mathcal{P}(\mathcal{I}), \subseteq)$ *– useful property for the efficient mining of these patterns (cf. next subsection). The same process has been recently highlighted in [Kryszkiewicz, 2009]. In addition, we affect to the empty set the cardinality of the set of objects (i.e.,* $|\mathcal{O}|$*) as support what ensures the correct regeneration of its conjunctive support during the expansion process to the whole set of frequent itemsets starting from the representation based on frequent essential itemsets. Note that, this assignation does not have any effect on the support of the other itemsets since the support of the empty set is not used in the inclusion-exclusion identities.*

In the remainder, we will denote by $\mathcal{FEI}$ the set of frequent essential itemsets that can be extracted from an extraction context $\mathcal{K}$. The following lemma shows how we can obtain the disjunctive support of a frequent itemset given the set $\mathcal{FEI}$.

Lemma 2 *Let* $I \in \mathcal{FI}$. $Supp(\vee I) = max\{Supp(\vee I_1) \mid I_1 \subseteq I$ *and* $I_1 \in \mathcal{FEI}\}$ *[Casali* et al.*, 2005a].*

The following definition presents the set *Argmax* associated to a frequent itemset I. This set contains the frequent essential itemsets contained in I and having the maximum disjunctive support among those of the subsets of I.

Definition 38 (ARGMAX)
Let $I \in \mathcal{FI}$. $J \in Argmax(I)$ *if* $J \subseteq I$, $J \in \mathcal{FEI}$ *and* $Supp(\vee J) = max\{Supp(\vee I_1) \mid I_1 \subseteq I\}$.

To derive the conjunctive support of a frequent itemset I, a straightforward manner is to use the equality shown in Lemma 1. However, an optimized way of the computation can be performed using an element of the set *Argmax* associated to I. The following lemma [Casali *et al.*, 2005a] shows how this can be done.

Lemma 3 *Let* $I \in \mathcal{FI}$ *and* $J \in Argmax(I)$. *We then have:*

$$Supp(I) = \sum_{\emptyset \subset I_1 \subseteq I} (\text{-}1)^{|I_1|-1} \begin{cases} Supp(\vee J) & \text{if } J \subseteq I_1 \\ Supp(\vee I_1) & \text{elsewhere} \end{cases}$$

Proof. The proof of this formula is based on the inclusion-exclusion identities (*cf.* Lemma 1, page 13), and on the fact that $\forall\ I_1$ *s.t.* $J \subseteq I_1 \subseteq I$: $Supp(\vee I_1) = Supp(\vee J)$ (*cf.* Lemma 2). $\diamondsuit$

The following theorem indicates how to derive the conjunctive support of a frequent itemset once the set of frequent essential itemsets is extracted.

Theorem 5 *Let* $I \in \mathcal{FI} \backslash \mathcal{FEI}$ *and* $J \in Argmax(I)$. *We then have:*

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

Proof. If we apply the inclusion-exclusion identities, we get:

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

$$\sum_{\substack{\emptyset \subset I_1 \subseteq I \\ J \not\subseteq I_1}} (\text{-}1)^{|I_1|-1} Supp(\vee I_1) + \sum_{J \subseteq I_1 \subseteq I} (\text{-}1)^{|I_1|-1} Supp(\vee I_1)$$

There are $\mathbf{2}^{|I|-|J|}$ itemsets encompassed between J and I. These itemsets have $Supp(\vee J)$ for disjunctive support (according to Lemma 3). Among this set of itemsets, there is the same number of itemsets with odd cardinality than those with even cardinality. Hence, the second part of the sum is equal to $\mathbf{0}$. $\diamondsuit$

It is worth noting that the optimization offered through Theorem 5 does not apply for an essential itemset I since in this case $J = Argmax(I) = I$. Thus, there is no support to be pruned within the inclusion-exclusion identity associated to I.

Remark 2 *A remark about Theorem 5 concerns the fact that in [Casali* et al.*, 2005a], the authors considered that the itemset J should not be a superset of I_1. Indeed, the formula was given as follows:*

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

In the proof they gave [Casali et al.*, 2005b], they claim that the latter formula results from the following decomposition:*

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

$$\sum_{\substack{\emptyset \subset I_1 \subseteq I \\ I_1 \not\subseteq J}} (\text{-}1)^{|I_1|-1} Supp(\vee I_1) \; + \sum_{J \subseteq I_1 \subseteq I} (\text{-}1)^{|I_1|-1} Supp(\vee I_1)$$

However, if we consider that $I = ABCD$, $J = AB$ and $I_1 = ABC$, then I_1 belongs to the first part of the sum ($\emptyset \subset ABC \subseteq ABCD$ and $ABC \not\subseteq AB$) as well as to the second part ($AB \subseteq ABC \subseteq ABCD$). Hence, both parts of the sum are not disjoint. The formula is then erroneous.

Although their usefulness, the set of frequent essential itemsets suffers from a main limitation. Indeed, having only the information offered by $\mathcal{FEI}$, we are not able to decide whether a given itemset I is frequent or not. Since the application of Theorem 5 to derive the conjunctive support of I mainly relies on this information and to overcome this limitation, Casali *et al.* augment the set $\mathcal{FEI}$ with the set of maximal frequent itemsets $\mathcal{B}d^+(\mathcal{FI})$. The latter will be used to check whether I is frequent or not. The following theorem summarizes the concise representation based on frequent essential itemsets.

Theorem 6 *The set $\mathcal{FEI}$ of frequent essential itemsets, associated to their respective disjunctive supports, augmented by the set $\mathcal{B}d^+(\mathcal{FI})$ of maximal frequent itemsets is an exact concise representation of the set $\mathcal{FI}$ of frequent itemsets [Casali* et al.*, 2005a].*

In the remainder, the frequent essential itemset-based representation will be denoted FEIs_rep.

3.5.2 Mining Algorithm

To extract the frequent essential itemset-based representation, Casali *et al.* proposed a levelwise algorithm, called MEP [2] [Casali *et al.*, 2005a]. This algorithm benefits from the fact that the set of frequent essential itemsets fulfills the interesting property of being an order ideal in $(\mathcal{P}(\mathcal{I}), \subseteq)$. This is stated by Proposition 9.

Proposition 9 *The set of frequent essential itemsets is an order ideal in* $(\mathcal{P}(\mathcal{I}), \subseteq)$.

Hence, if I is a frequent essential itemset, then each subset of I is also a frequent essential itemset. In a dual way, if I is not a frequent essential itemset, then each superset of I cannot be a frequent essential itemset. This interesting property helps levelwise algorithms to efficiently be adapted to extract this set. In this respect, the unique algorithm allowing to extract this set, *i.e.*, the MEP algorithm, is an adaptation of the well known levelwise APRIORI algorithm [Agrawal and Srikant, 1994].

Example 24 *Consider the context given by Table 2.1 (cf. page 12) for minsupp* $= \mathbf{1}$. *The application of the* MEP *algorithm gives the exact concise representation composed by:*

- $\mathcal{B}d^+(\mathcal{FI}) = \{ABCDEF\}$,
- *The set* $\mathcal{FEI}$ *which is summarized in Table 3.3.*

Note that we give the conjunctive support of frequent essential itemsets only for the sake of exhaustivity. Indeed, the MEP *algorithm does not extract the exact conjunctive supports of frequent essential itemsets. It only checks their frequency status by testing their inclusion in at least an element of* $\mathcal{B}d^+(\mathcal{FI})$ *[Casali* et al., *2005a]. Thus, during the regeneration process, in the case where an itemset belongs to* $\mathcal{FEI}$, *its conjunctive support must also be computed using an inclusion-exclusion identity. Table 3.3 also contain the couple* $(\emptyset, \mathbf{5})$ *added as aforementioned to ensure the exact regeneration of the empty set conjunctive support.*

Given the set $\mathcal{FEI}$ *augmented with* $\mathcal{B}d^+(\mathcal{FI})$, *we are able to derive the conjunctive support of any frequent itemset. Suppose we are interested in computing the conjunctive support of the itemset* DF *starting from the disjunctive supports of its subsets, i.e., using Theorem 5. Since* DF *is a frequent essential itemset,* $\text{Argmax}(DF) = \{DF\}$. *Then, the formula proved in the theorem is reduced to the application of the inclusion-exclusion identities given in Lemma 1 (cf. page 13). Hence,* $Supp(DF) = -Supp(\vee DF) + Supp(\vee D) + Supp(\vee F) = -\mathbf{5} + \mathbf{4} + \mathbf{3} = \mathbf{2}$.

Suppose now that we have to compute the conjunctive support of the itemset CEF. *The latter is a frequent itemset since it is included in the single element forming* $\mathcal{B}d^+(\mathcal{FI})$, *namely* $ABCDEF$. *According to Theorem 5,* $Supp(CEF)$ *can be computed starting from the disjunctive supports of its subsets that do not include* CE, *since* $CE \in \text{Argmax}(CEF)$. *Note that* CF *also belongs to* $\text{Argmax}(CEF)$. *Hence, the conjunctive support of the itemset* CEF *is computed as follows:* $Supp(CEF) = (-1)^{|CF|-1} Supp(\vee CF) + (-1)^{|EF|-1} Supp(\vee EF) + (-1)^{|C|-1} Supp(\vee C) + (-1)^{|E|-1} Supp(\vee E) + (-1)^{|F|-1} Supp(\vee F) = -Supp(\vee CF) - Supp(\vee EF) + Supp(\vee C) + Supp(\vee E) + Supp(\vee F) = -\mathbf{5} - \mathbf{4} + \mathbf{4} + \mathbf{3} + \mathbf{3} = \mathbf{1}$.

3.5.3 Discussion

To the best of our knowledge, this representation is the unique one in the literature relying on the disjunctive support as a way for characterizing its elements. For several real-life contexts and for given

[2] MEP is the acronym of Mining Essential Patterns.

Frequent essential itemset	Disjunctive support	Conjunctive support	Frequent essential itemset	Disjunctive support	Conjunctive support
$\emptyset$	5	5	A	3	3
B	3	3	C	4	4
D	4	4	E	3	3
F	3	3	AC	5	2
AD	5	2	AE	4	2
AF	4	2	BC	5	2
BD	5	2	BE	4	2
BF	4	2	CE	5	2
CF	5	2	DE	5	2
DF	5	2	EF	4	2
AEF	5	2	BEF	5	2

Table 3.3: The set $\mathcal{FEI}$ for *minsupp* = **1**.

settings of the *minsupp* threshold, the cardinality of the representation based on frequent essential itemsets is lower than that of the representation based on frequent closed itemsets [Casali *et al.*, 2005a]. Moreover, it makes it possible to efficiently determine the disjunctive and negative supports of frequent itemsets. Nevertheless, this representation suffers from its augmentation by the positive border of frequent itemsets, which leads to three main limitations:

1. **A heterogeneity within the concise representation:** indeed, the elements of $\mathcal{B}d^{+}(\mathcal{FI})$ are characterized by the conjunctive support and the maximum size of the itemsets within the associated γ-equivalence classes. While the frequent essential itemsets are characterized by their disjunctive supports. This obliges storing a membership flag of each itemset of this representation to the associated set. Noteworthily, an itemset can simultaneously be a frequent essential itemset and a maximal frequent itemset, which constitutes a first form of redundancy.

2. **An increase in the size of the representation:** this becomes clear especially for dense contexts where the size of $\mathcal{B}d^{+}(\mathcal{FI})$ can even exceed that of $\mathcal{FEI}$. In addition, this representation does not take into account the fact that several essential itemsets can characterize the same set of objects, which constitutes a second form of redundancy.

3. **An external algorithmic dependency:** The dependence of any algorithm extracting this concise representation to another one dedicated to the extraction of maximal frequent itemsets, like the MAX-MINER algorithm [Bayardo, 1998].

3.5.4 Link with Minimal Generators

Here, we will establish the link between two spaces: the conjunctive search space and the disjunctive search space. In each one, itemsets are characterized using the corresponding support. To understand

where are localized minimal generators and essential itemsets in their respective search space, let us recall their associated characterization. Let $I \subseteq \mathcal{I}$:

- I is a minimal generator *iff* $Supp(I) < \min\{Supp(I') \mid I' \subseteq \mathcal{I} \text{ and } I' \subset I\}$.
- I is an essential itemset *iff* $Supp(\vee I) > \max\{Supp(\vee I') \mid I' \subseteq \mathcal{I} \text{ and } I' \subset I\}$.

Thus, minimal generators and essential itemsets are dually defined *w.r.t.* conjunctive and disjunctive supports respectively. This derives from the fact that to be frequent *w.r.t. a minimum conjunctive support threshold* (like *minsupp* in our case) induces an order ideal. While to be frequent *w.r.t. a minimum disjunctive support threshold* induces an order filter. Indeed, the conjunctive (*resp.* disjunctive) support is a decreasing (*resp.* increasing) function of the size of itemsets.

Let us now split the disjunctive/conjunctive itemsets into equivalence classes *w.r.t.* the associated support in the sense that two itemsets belong to the same equivalence class if they have the same support. With respect to the aforementioned characterization, minimal generators and essential itemsets are hence the minimal itemsets, *w.r.t.* set inclusion, within their associated classes.

Since two itemsets can have the same supports while not being present in the same objects (*i.e.*, do not have the same extent), we can refine the aforementioned classes by dividing them into smaller ones according to a tougher constraint: two itemsets belong to the same equivalence class if they verify the same set of objects. This constraint obviously covers the previous one, *i.e.*, the equality of itemset supports. Then, we will for example obtain the γ-equivalence classes *w.r.t.* the conjunctive support. In both cases, minimal generators and essential itemsets are the minimal elements of their respective classes while their associated closure is unique in the associated equivalence class.

To summarize, the set of minimal generators and the set of essential itemsets have common structural properties in their associated search space. From the point of view of concise representations of frequent itemsets, none of them constitutes an exact representation by itself. They must hence be augmented to ensure the exactness of the regeneration process.

The next section presents a critical comparative study of the surveyed concise representations.

3.6 Comparative Study of Concise Representations for Frequent Itemsets

In the light of what was previously presented in this chapter, we notice that the exact concise representations have main differences that we organize according to the following axes:

1. **Composition:** This axis describes the nature of the itemsets contained in a given representation.
2. **Main features characterizing the itemsets composing the representation:** This axis sheds light on how are characterized the itemsets of the representation, such as the associated measures (*i.e.*, conjunctive/disjunctive support), deduction rules, and the size of the itemsets.
3. **Mining algorithms:** In order to extract an exact concise representation, some algorithms are proposed. Here, we present the most known ones.
4. **Link with minimal generators:** This axis summarizes how a given concise representation is linked to minimal generators.

5. **Regeneration mechanism:** Each concise representation has its proper regeneration mechanism allowing to derive the whole set of frequent itemsets associated to their respective supports.
6. **Advantages:** This axis presents the main advantages of the concise representations proposed in the literature. Here, we distinguish four major advantages:
   (a) `Efficient derivation of the disjunctive and negative supports:` This feature is very important since avoiding to evaluate inclusion-exclusion identities to derive disjunctive and negative supports starting from conjunctive ones. The evaluation of these identities can be costly when the number of itemsets frequent is large (especially for dense contexts).
   (b) `Homogeneity:` Thanks to this feature, it is possible to know if the associated representation is composed of itemsets of the same search space (*w.r.t.* the associated supports), etc.
   (c) `Derivation of generic association rules:` Generic association rules are lossless subsets of the whole set of association rules [Pasquier, 2009]. They hence allow to only manipulate a reduced number of rules and thus to remove redundancy within association rules.
   (d) `Efficient derivation of generalized association rules:` Contrary to classic association rules having positive items in both premise and conclusion parts, generalized association rules offer richer knowledge. Indeed, they convey various forms of connectors between items, like conjunction, disjunction and negations. They hence reveal the finest items correlations to the end-users. Thus, they help them to obtain a deeper analysis about the mined context, which improves their decisions.

Table 3.4 summarizes the results of our critical study on the main exact concise representations of the literature. Note that we did not include frequent closed non-derivable itemset-based representation in the summarizing table since it is a combination of closed and non-derivable itemsets. The main observations are as follows:

1. The concept of minimal generator plays a key role in the setting of each concise representation.
2. The derivation of the conjunctive supports of frequent itemsets is straightforward starting from the frequent closed itemset-based representation. Nevertheless, the efficient derivation of their disjunctive and negative supports is not straightforwardly possible.
3. The representation based on frequent closed itemsets is the unique one exploited in the literature towards extracting generic association rules. Note however that the frequent non-derivable itemsets were extended to association rules through the introduction of non-derivable association rules in [Goethals *et al.*, 2005].
4. The regeneration of the frequent itemsets starting from the frequent non-derivable itemsets is expensive in running time. Indeed, the computation of the conjunctive support of a derivable itemset requires the evaluation of $\mathbf{2}^{|I|}$ deduction rules. While the representation based on frequent essential itemsets only requires the evaluation of one inclusion-exclusion identity in order to determine the conjunctive support of the frequent itemsets.
5. The concise representation based on frequent essential itemsets is the unique one offering the possibility for an efficient derivation of various generalized association rule forms. Indeed, deriving disjunctive and negative supports starting from this representation is straightforward.

Axis	Representation based on frequent closed itemsets	Representation based on frequent non-derivable itemsets	Representation based on frequent essential itemsets
1- Composition	frequent closed itemsets	frequent non-derivable itemsets	frequent essential + maximal itemsets
2- Main features	Galois closure operator + conjunctive support	deduction rules + conjunctive support	disjunctive support + conjunctive support + size of itemsets
3- Mining algorithms	CLOSE, LCM, PRINCE, etc.	NDI and dfNDI	MEP
4- Link with minimal generators	closures of frequent minimal generators	generalizations of frequent minimal generators *w.r.t.* the explored neighborhood	dual of minimal generators *w.r.t.* the associated search space
5- Regeneration mechanism	the support of an itemset is equal to that of the smallest frequent closed itemset containing it	evaluation of $\mathbf{2}^n$ deduction rules for each itemset of size n	evaluation an inclusion-exclusion identity for each itemset
6- Advantages			
(a) Efficient derivation of the disjunctive and negative supports	No	No	Yes
(b) Homogeneity	Yes	Yes	No
(c) Derivation of generic association rules	Yes	No	No
(d) Efficient derivation of generalized association rules	No	No	Yes

Table 3.4: Comparison of the main exact concise representations proposed in the literature.

3.7 Conclusion

In this chapter, we presented the main exact concise representations proposed in the literature. We also highlighted their main advantages and limitations. Our critical study of these representations sheds light on the following observations:

1. Minimal generators play an important role in almost all concise representations of the literature. Indeed, they are used as an efficient mean for computing representations, like the frequent closed itemset-based one. In addition, they are at the basis of various generalizations leading to different concise representations, like the frequent non-derivable itemset-based one. Nevertheless, the existence in general of more than one minimal generator per γ-equivalence class augments the redundancy ratio within the extracted knowledge. Indeed, a same part of knowledge will be redundantly conveyed by the distinct patterns (for example, the itemsets and their uses in association rule mining).

   In this respect, one of the main issues addressed in this thesis consists in exploring efficient methods for a lossless reduction of the minimal generator family (*cf.* Chapter 4). This is extended to

the association rule framework through redundancy removal within generic association rules (*cf.* Chapter 5).

2. The representation based on frequent essential itemsets constitutes an interesting alternative to represent in an exact way the set of the frequent itemsets. Indeed, it makes it possible to efficiently determine the various forms of support of the frequent itemsets. Thus, it constitutes for example a powerful starting point for obtaining a more reduced concise representation for frequent itemsets. In addition, it can give rise to some interesting association rule forms, to go beyond the classic ones. Nevertheless, it mainly suffers from the necessity to add the positive border of frequent itemsets to make it an exact representation. In this situation, such an addition necessarily augments the size of the representation and, more dramatically, leads to a dependency on algorithms for mining maximal frequent itemsets. This representation also suffers from the heterogeneity of its elements with respect to their support – disjunctive *vs.* conjunctive.

   In order to palliate these limitations, we propose a new exact concise representation based on a new closure operator dedicated to the disjunctive search space and, hence, to essential itemsets in particular (*cf.* Chapter 6). Moreover, we show how this new representation is able to overcome the limits of the representation based on frequent essential itemsets. The obtained results will also be applied for extracting generalized association rules (*cf.* Chapter 7).

Part II

Exploration of the Conjunctive Search Space

Chapter 4

Lossless Reductions of the Minimal Generator Family of an Extraction Context

4.1 Introduction

Standing at the "antipodes" of closed itemsets (CIs) within their respective γ-equivalence classes induced by the Galois closure operator, minimal generators (MGs) [Bastide *et al.*, 2000a] are the minimal elements of a class while the CIs [Pasquier *et al.*, 1999b] are the largest. They hence help delimit the classes and ease their detection/traversal. Although their study grasped little interest compared to that paid to CIs, MGs appear to be at the crossroads of many theoretical and practical problem settings related to closure systems. Indeed, they are used in graph theory (as minimal transversals [Berge, 1989]), database design (as minimal keys [Maier, 1983]), and data mining, to cite but a few. The computational complexity of some decision and counting problems related to MGs of closed sets were investigated in [Hermann and Sertkaya, 2008].

Practically speaking, it has been shown in [Li *et al.*, 2006] that, in applications related to inductive inference, model selection and classification, MGs are highly instrumental and even preferable to CIs *w.r.t.* the minimal description length principle (MDLP) [Grunwald, 2007, Rissanen, 1978]. Indeed, they are usually strictly smaller in size terms than their CIs (unless themselves closed), and hence offer minimal combinations of conditions necessary to identify a class of situations. Simultaneously used with their CIs, MGs also offered a concise representation of odds ratio and relative risk patterns of a binary context [Li *et al.*, 2005], and of mined frequent itemsets from data streams [Xie *et al.*, 2006]. In addition, they made possible a structural localization of statistically important γ-equivalence classes *w.r.t.* some measures [Li *et al.*, 2007]. Moreover, MGs are used for mining complex pattern classes, like sequential patterns [Balcázar and Casas-Garriga, 2007, Lo *et al.*, 2008], etc.

In general, many MGs belong to the same γ-equivalence class which leads to one-to-one correspondence between the knowledge pieces involving these MGs. Consequently, some redundancy clearly still exists for real-life contexts. In this respect, a study of intra-class redundancies in MGs was initiated by Dong

et al., who proposed a way to derive MGs from other ones in the same γ-equivalence class [Dong *et al.*, 2005]. The overall reduction principle may be roughly summarized as follows: an arbitrary total order is defined on the itemset family and the unique irreducible members are kept. This results in a split of the global MG family into *succinct* and *redundant* parts. Thus, the succinct system of minimal generators was introduced as a concise representation from which the entire MG family can be retrieved without any information loss. However, contrary to the authors' claims in [Dong *et al.*, 2005], we prove that the original succinct system of minimal generators is loss-prone since some *redundant* MGs are impossible to derive in some cases. Furthermore, for a given context, the systems resulting from different imposed total order relations on itemsets do not necessarily share the same size, again contradicting what was stated in [Dong *et al.*, 2005].

In this chapter, we carry out a thorough study of the succinct system of minimal generators. This allows us to clarify several aspects and to highlight interesting properties of this system, not mentioned in [Dong *et al.*, 2005]. We then propose a second approach that overcomes the flaws of the original system. Our redefined succinct system of minimal generators allows to obtain an *exact* representation of the MG family. In addition, for a given extraction context, the new definition leads to equal-size families *w.r.t.* the total order relation. Unfortunately, this definition leads to the loss of the order ideal structure which greatly complicates its practical extraction. As a hybrid approach between the first and second ones, we introduce a third system that overcomes their worst limitations. We present its definition and show that it preserves the precious order ideal property together with further structural properties that underly a lossless reduction mechanism. Finally, an experimental evaluation illustrates the benefits of our approach towards offering to the end-users a redundancy-free set of minimal generators.

The main contribution of this chapter is thus to show how to reach, without information loss, the case where each γ-equivalence class only contains *irreducible* minimal generators. This is helpful for real-life contexts. Indeed, an antimatroid closure space, which corresponds to the case where each γ-equivalence class contains a unique minimal generator [Pfaltz and Taylor, 2002], is unlikely to happen in real-life contexts which leads to a highly combinatorial redundancy. Reducing such combinatorial variations within the MG family is hence a central issue for its results in smaller-size storage and for making easier further manipulations. Indeed, in applications based on MGs such as association rule mining [Ceglar and Roddick, 2006] and association rule-based classification [Baralis and Chiusano, 2004], reducing the number of MGs without information loss will help saving spaces on which results will be stored and will ease their interpretation for the end-users. This is argued by the fact that only MGs bringing further knowledge are maintained while being able to losslessly derive redundant ones.

The chapter is organized as follows: Section 4.2 is a detailed study of the succinct system of minimal generators as defined by Dong *et al.* Section 4.3 expands on our first solution towards a lossless reduction of the MG family through its definition as well as its structural properties. Section 4.4 presents the main features of our hybrid approach as well as a regeneration mechanism allowing deriving the whole set of MGs. Section 4.5 proposes a discussion of the proposed MG families. In Section 4.6, an algorithm for extracting the hybrid family is proposed, followed by a study of its properties. Section 4.7 discusses the main related work. The empirical evidences about the soundness of our work are shown in Section 4.8.

4.2 Original Succinct System of Minimal Generators

In this section, we will study the main characteristics of the original succinct system of minimal generators (OSSMG) [Dong *et al.*, 2005]. We then clarify the aspects of the definition that remained unclear and show its flaws. Please note that we mainly refer to the SSMG_MINER algorithm proposed by the authors [Dong *et al.*, 2005]. In fact, the concrete examples related to SSMG_MINER are the unique source of precise information about several aspects of the target structure.

4.2.1 Description

In [Dong *et al.*, 2005], Dong *et al.* showed that the minimal generator (MG) set may contain redundant information. In fact, some MGs associated to a closed itemset (CI) can be derived from other ones by a process based on subsets substitution. They hence tried to remove the redundancy within the MG set and to achieve a succinct representation of MGs. Thus, Dong *et al.* introduced the succinct system of minimal generators as a concise representation of the MG set. The main idea was then to remove the redundant information by choosing one (*e.g.*, the smallest *w.r.t.* a given total order) MG of a CI, to elect it as its *representative* MG, and discarding those containing at least a non-representative MG [Dong *et al.*, 2005]. In each γ-equivalence class induced by the Galois closure operator, the purpose is only to retain those MGs that cannot be derived from other ones of the same γ-equivalence class. The authors hence proposed to set up a relation between itemsets. This relation is defined as follows [Dong *et al.*, 2005]:

Definition 39 (ITEMSET RELATION)
Let f be a closed itemset. Let X and Y be two itemsets. X and Y are called `f-equivalent`*, denoted $X \approx_f Y$, if:*
(i) X and Y are two minimal generators of a closed itemset f_1 s.t. $f_1 \subset f$.
(ii) X can be obtained from Y by replacing a subset Z_1 of X ($Z_1 \subset X$) by a subset Z_2 of Y ($Z_2 \subset Y$) s.t. $Z_1 \approx_f Z_2$.

Example 25 *To illustrate this definition, consider the context given by Table 2.1 (cf. page 12). The associated list of closed itemsets and their respective minimal generators is given in Table 4.1 (cf. page 54). The relation between itemsets given in the case (i) is fulfilled by* `A` *and* `B` *w.r.t. the* CI `ABCD`*. Indeed, both are* MG*s of* `AB` *which is included in* `ABCD`*. Hence,* `A` $\approx_{ABCD}$ `B`*. While the relation given in the case (ii) is fulfilled by* `AC` *and* `BC` *also w.r.t.* `ABCD`*. Indeed, by replacing* `A` *by* `B` *in the* MG `AC`*, we obtain* `BC`*. This replacement is sound since* `A` $\approx_{ABCD}$ `B`*.*

Surprisingly enough, $\approx_f$ is not an equivalence relation since the transitivity property is not fulfilled, as this will be shown in Subsection 4.2.3. Dong *et al.* aimed at using this relation to split the MGs, associated to a given CI, into disjoint equivalence classes. To avoid confusion with the γ-equivalence classes induced by the Galois closure operator γ, the latter will be denoted *σ-equivalence classes*. The achievement of the goal of deriving a minimal non-redundant subset of MGs is carried out by only maintaining a unique MG for each σ-equivalence class. The choice of the representative member of a σ-equivalence class is of paramount importance. Dong *et al.* proposed to freely choose a *representative* MG for the minimal CIs. For the other CIs, the authors proposed to choose one of the *canonical* MGs,

i.e., those that do not contain any non-representative MG of a subsumed CI. Even though the authors do not give a precise way to choose the *representative* MG, the illustrative examples of their paper hint that shorter sets are considered as smaller and are hence favored [Dong *et al.*, 2005]. In other words, the cardinality of MGs constitutes the first criterion when choosing a *representative* MG among a set of canonical ones belonging to the same σ-equivalence class.

4.2.2 Clarification of Imprecise Aspects

Several aspects remain unclear in the presentation of the OSSMG in [Dong *et al.*, 2005]. For instance, the selection of the representative itemset for each γ-equivalence class seems to be defined procedurally rather than analytically: a climb in the Boolean lattice is used to guide the choice which is some way randomly performed for the minimal CIs. On upper levels of the CI lattice, the representative is chosen among the sets that are canonical *w.r.t.* already fixed part of the *representative* MG set (thus enforcing the order ideal structure of the target OSSMG). After cross-checking with the algorithmic description, it becomes clear that a global order on items is used which makes all the choices on the lowest level deterministic. Moreover, the choice among canonicals on upper levels is fixed by a preference for smaller-size sets. The following definition of a total order on itemsets summarizes this:

Definition 40 (Total order relation)
Let $\preceq$ be a total order over the set of items, i.e., $\forall\ i, j \in \mathcal{I}$ s.t. $i \neq j$, either $i \prec j$ or $j \prec i$ holds. This relation is extended to itemsets as follows: let X and Y be two distinct itemsets sorted w.r.t. $\preceq$. Let $|X|$ (resp. $|Y|$) be the cardinality of X (resp. Y) and x_k (resp. y_k) be its k^{th} item. We then distinguish two cases:

- $|X| < |Y|$: $X \prec Y$.
- $|X| = |Y|$: *if there is an integer t s.t. $x_t \prec y_t$ and $\forall\ k \in \{1, \ldots, (t\text{ - }1)\}$, $x_k = y_k$, then $X \prec Y$. Otherwise, $Y \prec X$.*

Example 26 *Consider the alphabetic order on items as the basis for the total order relation $\preceq$ on itemsets.* [1] *For example:*

- $D \prec BE$ *since* $|D| < |BE|$.
- $ABD \prec ABE$ *since* $|ABD| = |ABE|$, *and the third item of* ABD, *namely* D, *is smaller w.r.t. $\preceq$ than that of* ABE, *namely* E, *while their respective first two items are the same.*

The cardinality-based criterion used in the definition of the total order relation preserves the spirit of MGs. Indeed, the smallest itemset, *w.r.t.* $\preceq$, in each γ-equivalence class will necessarily be a MG. Three categories of MGs emerge [Dong *et al.*, 2005] which we formalize as follows:

Definition 41 (Minimal generators categories)
The set MG_f, *of the* MG*s associated to a* CI f, *can be portioned into three distinct subsets as follows:*

[1] In the remainder, we will only mention the criterion used to order items (*e.g.*, alphabetic order). The latter order is then extended to be a total order relation on itemsets, as shown in Definition 40.

(*i*) $\mathbf{MG}\mathtt{rep}_f = \{g \in \mathrm{MG}_f \mid \nexists\ g_1 \in \mathrm{MG}_f \text{ s.t. } g_1 \prec g\}$: *the* $\mathrm{MG}\boldsymbol{rep}_f$ *set contains the smallest* MG, *given a total order relation* $\preceq$, *which constitutes the* ***representative*** MG *of* f.

(*ii*) $\mathbf{MG}\mathtt{can}_f = \{g \in \mathrm{MG}_f \mid (g \notin \mathrm{MG}\boldsymbol{rep}_f) \wedge (\forall\ g_1 \subset g, \exists\ f_1 \text{ s.t. } f_1 = \gamma(g_1) \text{ and } g_1 \in \mathrm{MG}\boldsymbol{rep}_{f_1})\}$: *the* $\mathrm{MG}\,\boldsymbol{can}_f$ *set contains the* ***canonical*** MG*s of* f. *A canonical* MG *is not the smallest one in* MG_f, *and hence is not the representative* MG *of* f. *Nevertheless, all its subsets are the representative* MG*s of their respective* γ*-equivalence classes.*

(*iii*) $\mathbf{MG}\mathtt{red}_f = \{g \in \mathrm{MG}_f \mid \exists\ g_1 \subset g, \exists\ f_1 \text{ s.t. } f_1 = \gamma(g_1) \text{ and } g_1 \notin \mathrm{MG}\boldsymbol{rep}_{f_1}\} = \mathrm{MG}_f \setminus (\mathrm{MG}\boldsymbol{rep}_f \cup \mathrm{MG}\,\boldsymbol{can}_f)$: *the* $\mathrm{MG}\boldsymbol{red}_f$ *set contains the* ***redundant*** MG*s of* f. *A redundant* MG *contains at least a subset which is not a representative* MG.

Definition 42 introduces the set of *succinct* MGs according to the approach of Dong *et al.*

Definition 42 (Set of succinct minimal generators)
A MG *is said to be* ***succinct*** *if it is either a representative or a canonical one. The set* $\mathrm{MG}\boldsymbol{suc}_f$ *of succinct* MG*s associated to the* CI f *is then equal to the union of* $\mathrm{MG}\boldsymbol{rep}_f$ *and* $\mathrm{MG}\,\boldsymbol{can}_f$: $\mathrm{MG}\boldsymbol{suc}_f = \mathrm{MG}\boldsymbol{rep}_f \cup \mathrm{MG}\,\boldsymbol{can}_f$.

Example 27 *Consider the context* $\mathcal{K}$ *depicted by Table 2.1 (cf. page 12) for minsupp* = **1**. *Let the alphabetic order be the total order relation* $\preceq$. *This relation is used to sort the* MG*s associated to the* CI*s shown in Table 4.1. Note that for* **10** CI*s, there are* **30** MG*s, from which only* **14** *are* `succinct` *ones. There are as many* CI*s as* `representative` MG*s, i.e.,* **10**, *and only* **4** `canonical` *ones (which are underlined in the table). The* MG `AC` *is a* `representative` *one since it is the smallest* MG *w.r.t.* $\preceq$, *among those of* `ABCD`. *Indeed,* $\mathtt{AC} \prec \mathtt{AD}$, $\mathtt{AC} \prec \mathtt{BC}$ *and* $\mathtt{AC} \prec \mathtt{BD}$. *The* MG `B` *is not the* `representative` *of its* CI `AB` *since* $\mathtt{A} \prec \mathtt{B}$. *Nevertheless, its unique subset (i.e.,* $\emptyset$*) is a* `representative` MG. *Hence,* `B` *is a* `canonical` MG. *Finally, the* MG `DF` *is a* `redundant` *one since at least one of its subsets is not a* `representative` MG (`D`, *for example*).

The definition of a succinct system of minimal generators according to Dong *et al.* is as follows [Dong *et al.*, 2005]:

Definition 43 (Original Succinct System of Minimal Generators)
A succinct system of minimal generators w.r.t. a total order relation $\preceq$, *consists of, for each closed itemset, the representative minimal generator and a possibly empty set of canonical minimal generators.*

Noteworthily, for a given context, there may be several OSSMGs depending on the choice of the total order relation $\preceq$. The context, shown in Table 4.2 (Top) page 58, represents such an example and is discussed later in Subsection 4.2.3. It is also important to point out that the OSSMG is clearly a generalization of the clone items framework which only focuses on items playing symmetric roles within CIs [Gély *et al.*, 2005] (*cf.* Section 4.7, page 76). Indeed, instead of single items, the OSSMG considers subsets of items.

Proposition 10 states the relation between the number of *representative* MGs and that of CIs, while Proposition 11 provides an interesting property of the set of *canonical* MGs.

#	CI	MGs	Succinct MGs	Support
1	∅	∅	∅	5
2	E	E	E	3
3	F	F	F	3
4	AB	A, B	A, B	3
5	CD	C, D	C, D	4
6	CDE	CE, DE	CE	2
7	CDF	CF, DF	CF	2
8	ABCD	AC, AD, BC, BD	AC	2
9	ABEF	AE, AF, BE, BF, EF	AE, AF, EF	2
10	ABCDEF	ACE, ACF, ADE, ADF, BCE, BCF, BDE, BDF, CEF, DEF	ACE	1

Table 4.1: The list of closed itemsets, and for each one, the corresponding minimal generators, *succinct* minimal generators and support.

Proposition 10 *The cardinality of the set of representative* MG*s is equal to that of* CI*s.*

Proof. There is only one CI per γ-equivalence class, which is also the case of representative MGs. Indeed, the total order relation $\preceq$ ensures the uniqueness of the *representative* MG associated to a given CI. $\diamondsuit$

Proposition 11 *The set $\mathcal{MG}_{\texttt{can}}$ contains incomparable elements w.r.t. set inclusion.*

Proof. According to Definition 41, two distinct *canonical* MGs are necessarily incomparable *w.r.t.* set inclusion. Indeed, the contrary would lead to a contradiction with the status of the largest one. $\diamondsuit$

Remark 3 *If the set $\mathcal{MG}_{\texttt{can}}$ is empty, then the set $\mathcal{MG}_{\texttt{red}}$ is necessarily empty. However, the reverse is not always true.*

Proposition 13 states that the subsets of a *representative* MG are also representative ones. This proposition is required to show that the set $\mathcal{MG}_{\texttt{suc}}$ is an order ideal in $(\mathcal{P}(\mathcal{I}), \subseteq)$. Using Lemma 4 and Proposition 12, the proof of this proposition stresses on the fact that the admission of the contrary, *i.e.*, the existence of a subset which is not a representative, would lead to a contradiction with the "smallest" status of a *representative* MG, *w.r.t.* $\preceq$.

Lemma 4 *Let $X, Y \subseteq \mathcal{I}$. If $\gamma(X) = \gamma(Y)$, then $\forall\ Z \subseteq \mathcal{I}$, $\gamma(X \cup Z) = \gamma(Y \cup Z)$ [Pasquier, 2000].*

In our context, with $X \cup Y$, we will indicate the ordered set of items, *w.r.t.* the total order relation $\preceq$, contained in X or in Y.

Proposition 12 *Let X, Y, Z be three itemsets s.t. $X \cap Z = \emptyset$ and $Y \cap Z = \emptyset$. If $X \preceq Y$, then $(X \cup Z) \preceq (Y \cup Z)$.*

Proof. The proof straightforwardly derives from the definition of the total order relation (*cf.* Definition 40) when $|X| < |Y|$, and from the properties of the lexicographic order when $|X| = |Y|$. $\diamondsuit$

Example 28 *Let the alphabetic order be the total order relation $\preceq$:*
- Since $D \preceq BE$, then $(D \cup AF) \preceq (BE \cup AF)$ (i.e., $ADF \preceq ABEF$).
- Since $ABD \preceq ABE$, then $(ABD \cup CF) \preceq (ABE \cup CF)$ (i.e., $ABCDF \preceq ABCEF$).

Proposition 13 *All subsets of a representative* MG *are also representative ones.*

Proof. Let g be a *representative* MG and f its closure. Suppose, we have $g_1 \subset g$ and $g_1 \notin \text{MG}\mathtt{rep}_{f_1}$ with $f_1 = \gamma(g_1)$. Let g_2 be the *representative* MG of f_1. Consequently, $g_2 \prec g_1$. Since $\gamma(g_1) = \gamma(g_2)$, then, according to Lemma 4, we have $\gamma(g_1 \cup (g \setminus g_1)) = \gamma(g_2 \cup (g \setminus g_1))$, and hence $\gamma(g) = \gamma(g_2 \cup (g \setminus g_1))$. Let g_3 be equal to $(g_2 \cup (g \setminus g_1))$. According to the second case in Definition 40 and to Proposition 12, we have $g_3 \prec g$ since $g_2 \prec g_1$, $g_2 \cap (g \setminus g_1) = \emptyset$ and $g_1 \cap (g \setminus g_1) = \emptyset$. Note that $g_2 \cap (g \setminus g_1)$ is ensured to be empty since otherwise g will not be a MG, which is in contradiction with the initial assumption that g is a *representative* MG. Indeed, if $g_2 \cap (g \setminus g_1) = q \neq \emptyset$, then $\gamma(g_1 \cup q) = \gamma(g_1)$, and hence $(g_1 \cup q)$ is not a MG. Since $(g_1 \cup q) \subseteq g$, then g is also not a MG since the MG set is an order ideal *w.r.t.* set inclusion.

Two situations need then to be distinguished :

1. If g_3 is a MG, then g cannot be a *representative* MG, which is also in contradiction with the initial assumption that g is a *representative* MG.
2. If g_3 is not a MG, then there is a MG g_4 *s.t.* $g_4 \subset g_3$ and $\gamma(g_4) = \gamma(g_3)$. Since $|g_4| < |g_3|$, then $g_4 \prec g_3$ (according to the first case in Definition 40), and hence $g_4 \prec g$. This result is also in contradiction with the starting assumption.

Thus, we can conclude that each subset of g is necessarily a *representative* MG. $\diamondsuit$

Hence, according to Proposition 13, if f is a CI, then $\text{MG}\mathtt{suc}_f = \text{MG}\mathtt{rep}_f \cup \text{MG}\mathtt{can}_f = \{g \in \text{MG}_f | \forall\ g_1 \subset g,\ g_1 \in \text{MG}\mathtt{rep}_{f_1}$ with $f_1 = \gamma(g_1)\}$. Thanks to Proposition 14, we show that the *succinctness* of MGs is an anti-monotone constraint. Hence, the set $\mathcal{MG}\mathtt{suc}$ is an order ideal in $(\mathcal{P}(\mathcal{I}), \subseteq)$.

Proposition 14 *Let g be an itemset. g fulfills the following two properties:*

1. *If $g \in \mathcal{MG}\mathtt{suc}$, then $\forall\ g_1$ s.t. $g_1 \subset g$, $g_1 \in \mathcal{MG}\mathtt{suc}$.*
2. *If $g \notin \mathcal{MG}\mathtt{suc}$, then $\forall\ g_1$ s.t. $g \subset g_1$, $g_1 \notin \mathcal{MG}\mathtt{suc}$.*

Proof.

1. $g \in \mathcal{MG}\mathtt{suc} \Longrightarrow \forall\ g_1$ *s.t.* $g_1 \subset g$, $g_1 \in \text{MG}\mathtt{rep}_{f_1}$ with $f_1 = \gamma(g_1)$ (according to Definition 41) $\Longrightarrow$ $\forall\ g_1$ *s.t.* $g_1 \subset g$, $g_1 \in \text{MG}\mathtt{suc}_{f_1}$ (since $\text{MG}\mathtt{rep}_{f_1} \subseteq \text{MG}\mathtt{suc}_{f_1}$) $\Longrightarrow \forall\ g_1$ *s.t.* $g_1 \subset g$, $g_1 \in \mathcal{MG}\mathtt{suc}$ (since $\text{MG}\mathtt{suc}_{f_1} \subseteq \mathcal{MG}\mathtt{suc}$).

2. $g \notin \mathcal{MG}\texttt{suc} \Longrightarrow \forall\ g_1$ *s.t.* $g \subset g_1$, $g_1 \in \text{MG}\texttt{red}_{f_1}$ with $f_1 = \gamma(g_1)$ (indeed, g_1 has at least a *non-representative* subset, namely g, since the latter is not a *succinct* MG, and hence is not a *representative* one) $\Longrightarrow \forall\ g_1$ *s.t.* $g \subset g_1$, $g_1 \notin \text{MG}\texttt{suc}_{f_1}$ (according to Definition 41, g_1 cannot be *redundant* and *succinct* at the same time) $\Longrightarrow \forall\ g_1$ *s.t.* $g \subset g_1$, $g_1 \notin \mathcal{MG}\texttt{suc}$ (we have $g_1 \notin \text{MG}\texttt{suc}_{f_1}$. In addition, $g_1 \notin (\mathcal{MG}\texttt{suc} \backslash \text{MG}\texttt{suc}_{f_1})$ since the closure of g_1 is unique and is equal to f_1).

$\diamondsuit$

Proposition 15 *The set $\mathcal{FMG}\texttt{suc}$, of the succinct frequent* MG*s extracted from the context $\mathcal{K}$, is an order ideal in $(\mathcal{P}(\mathcal{I}), \subseteq)$.*

Proof. The proof is based on Proposition 14 and on the fact that the frequency constraint is also an anti-monotone constraint. Since the conjunction of two anti-monotone constraints (*i.e.*, to be simultaneously *succinct* and *frequent*) results in an anti-monotone constraint, the set $\mathcal{FMG}\texttt{suc}$ is an order ideal in $(\mathcal{P}(\mathcal{I}), \subseteq)$. $\diamondsuit$

This interesting property allows us to propose an efficient algorithm to extract the OSSMG according to the definition of Dong *et al.* Indeed, to check whether a given MG g of size k is a *succinct* one or not, the proposed algorithm takes advantage of this property by limiting the test to its subsets of size (k - 1).

4.2.3 Unveiling Problems in the OSSMG

As mentioned in Subsection 4.2.1, the application of the relation $\approx_f$ does not induce an equivalence relation on the MG set of a CI. Indeed, let us consider Table 4.1 (*cf.* page 54). Let us concentrate on the γ-equivalence class having ABCD for CI. We have AC $\approx_{\texttt{ABCD}}$ BC $\approx_{\texttt{ABCD}}$ BD but AC $\not\approx_{\texttt{ABCD}}$ BD. Hence, this relation is not an equivalence one, since the transitivity property is not fulfilled. In their work [Dong *et al.*, 2005], the authors also made the following claims:

Claim 1: The OSSMG is a *lossless representation* of the MG set, *i.e.*, if g is a *redundant* MG, then g can be inferred from the OSSMG without loss of information.

Claim 2: The cardinality of the OSSMG is insensitive *w.r.t.* the total order relation $\preceq$.

To infer the *redundant* MGs of each γ-equivalence class, Dong *et al.* proposed to replace the subsets (one or more) of its *succinct* MGs by *non-representative* MGs having, respectively, the same closures as those of the replaced subsets [Dong *et al.*, 2005]. For example, the *redundant* MG ADE can be inferred from the *succinct* MG ACE by replacing its subset CE by DE. Indeed, both MGs CE and DE have the same closure as shown in Table 4.1.

To be fulfilled, both claims closely rely on how maintaining representative members of the different σ-equivalence classes, from where the remainder can be derived using the relation $\approx_f$. However, localizing such members by pruning redundant elements, containing non-representative subsets, can lead to:

• **A σ-equivalence class without a representative member:** It is the case of the σ-equivalence class $\mathcal{S}_1$ = {ECF, EDF, ACB, ABD, ABF, CBF, BDF} associated to the closed itemset EACBDF (*cf.* Table 4.2

(Bottom) for the ascending support order). Indeed, each element of this σ-equivalence class contains at least a non-representative MG. Hence, such σ-equivalence class will not be taken into account and all its elements will then not be derived, which presents a loss of information, in the contrary to the statement of **Claim 1**.

• **A σ-equivalence class with more than one candidate for being the representative member:** It is the case of the σ-equivalence class $\mathcal{S}_2$ = {BDF, BDA, BFA, BFC, BAC, DFE, FCE} associated to the closed itemset BDFACE (*cf.* Table 4.2 (Bottom) for the descending support order). Indeed, BDF, BDA and BFA have all their subsets as *representative* MGs. Even one can choose the smallest candidate and elect it as the representative member, the definition given by Dong *et al.* lacks the important part allowing to delete the remaining candidates.

Noteworthily, the elements of $\mathcal{S}_1$ and $\mathcal{S}_2$ are exactly the same while sorted according to two different order relations. In $\mathcal{S}_1$, there is no representative member while in $\mathcal{S}_2$, there are three possible ones. This fact shows that the cardinality of the OSSMG closely depends on the selected total order relation, in the contrary to the statement of **Claim 2**. It is however important to note that this claim seemed to hold when confronted to the extraction context depicted by Table 2.1, page 12. Indeed, for different total order relations (*e.g.*, the alphabetic order, the ascending/descending support order, etc.), we obtain the same number of *succinct* MGs (*cf.* Table 4.1). It is the same for the running example used in the proper paper of Dong *et al.* [Dong *et al.*, 2005]. Nevertheless, if we consider the context sketched by Table 4.2 (Top), we find that their claim is clearly erroneous. Indeed, as shown by Table 4.2 (Bottom), the total number of *succinct* MGs is equal to **23** if the alphabetic order is of use, whereas it is equal to **22** in the case of the ascending support order, and **25** in the case of the descending support order. The difference occurs within the γ-equivalence class number **11**.

Even if the second flaw (*i.e.*, that related to the size of the different OSSMGs associated to a given extraction context) can be regarded as not having a dramatic consequence, fixing the first one (*i.e.*, the loss of information) is a compelling issue since the need for *exact* concise representations is always conditioned by the ability to discover *all redundant* information without consulting the initial extraction context. Hence, aiming to ensure the completeness of the derivation of *all redundant* MGs, we introduce, in the next section, new definitions allowing to palliate the flaws that we revealed in the work proposed by Dong *et al.* Before that, let us analyze why the original definition failed in the regeneration process of redundant MGs.

In fact, in their work, Dong *et al.* looked for a classic way to reduce a set, *i.e.*, by breaking it into equivalence classes that are further shrunk to a unique representative element. Thus, they based their system definition on a substitution-based $\approx_f$ relation. Although $\approx_f$ was wrongfully assumed to be an equivalence relation, this did not result in a major flaw in the construction of the OSSMG. Indeed, the authors put, implicitly, the requirement for it being an order ideal (through the definition of a canonical element). We have shown above that this ideal can be assimilated to a total order on itemsets, itself induced by an order on items. The exact composition of the ideal, however, strongly depends on the chosen total order relation: different orders could result in different sets becoming representative and canonical.

What Dong *et al.* seem to have miscalculated is the interplay between the ideal and the partition of the Boolean lattice into substitution-based σ-equivalence classes. Indeed, they hastily concluded that

	A	B	C	D	E	F
1	×	×				
2	×		×			×
3	×			×		
4		×	×	×	×	
5	×	×	×	×	×	×
6			×	×		
7		×				×
8				×		×
9		×			×	×

	alphabetic *order*		***ascending support*** *order*		***descending support*** *order*	
#	**CI**	**MGs**	**CI**	**MGs**	**CI**	**MGs**
1	∅	∅	∅	∅	∅	∅
2	A	**A**	A	**A**	A	**A**
3	B	**B**	B	**B**	B	**B**
4	C	**C**	C	**C**	C	**C**
5	D	**D**	D	**D**	D	**D**
6	BE	**E**	EB	**E**	BE	**E**
7	F	**F**	F	**F**	F	**F**
8	AB	**AB**	AB	**AB**	BA	**BA**
9	ACF	**AC**, **AF**, **CF**	ACF	**AC**, **AF**, **CF**	FAC	**FA**, **FC**, **AC**
10	AD	**AD**	AD	**AD**	DA	**DA**
11	ABCDEF	**AE**, **ABC**, ABD, ABF, **ACD**, ADF, BCF, BDF, CDF, CEF, DEF	EACBDF	**EA**, ECF, EDF, ACB, **ACD**, ABD, ABF, ADF, CBF, CDF, BDF	BDFACE	**AE**, **BDF**, **BDA**, **BFA**, BFC, BAC, **DFA**, DFC, DFE, DAC, FCE
12	BCDE	**BC**, **BD**, **CE**, **DE**	ECBD	**EC**, **ED**, **CB**, **BD**	BDCE	**BD**, **BC**, **DE**, **CE**
13	BF	**BF**	BF	**BF**	BF	**BF**
14	CD	**CD**	CD	**CD**	DC	**DC**
15	DF	**DF**	DF	**DF**	DF	**DF**
16	BEF	**EF**	EBF	**EF**	BFE	**FE**

Table 4.2: (**Top**) An extraction context. (**Bottom**) The list of closed itemsets, and for each one, the corresponding minimal generators for different total order relations. The *succinct* MGs, according to the definition of Dong *et al.*, are indicated with bold letters.

whatever the order, there will always be at least one canonical element per σ-equivalence class, which is of course wrong. Moreover, their claim of invariance for the OSSMG size upon the choice of the total order relation seems to come from either (i) belief that there will be a unique canonical element in each σ-equivalence class, also wrong, or (ii) a discrepancy between the analytical description and the SSMG_MINER algorithm which clearly keeps all the canonical elements that are found.

One main observation is that three different phenomena interact in the original definition:

1. The substitution-based relation,
2. The minimal generator status of a set within its γ-equivalence class which induces an order ideal of its own,
3. An additional ordering on the Boolean lattice of itemsets which induces a different, yet somehow connected to the previous one, order ideal composed of the representative itemsets. The latter ideal is completed with its "shell" of canonical elements which constitute its outside frontier, or *negative border* (actually a subset of it), and the result is another, a bit larger ideal combining both sorts of MGs.

It is however impossible to always have all three constructs "aligned", *i.e.*, that all substitution-based classes of MGs intersect the second order ideal to a unique element. In practice, it may happen that several from such elements belong to the same substitution class, just as there could be none in some classes.

4.3 New Succinct System of Minimal Generators

In this section, we will introduce a new lossless reduction of the MG set that repairs the flaws pointed out in the proposal of Dong *et al.*

4.3.1 Description

In our attempt to fix the main flaws of the original succinct system of minimal generators, we propose a relation allowing to divide the set of MGs associated to a given CI f into correctly defined σ-equivalence classes. The intuition behind such a relation is that if Y and Z are two itemsets having the same closure (*i.e.*, $\gamma(Y) = \gamma(Z)$), then their respective supersets, obtained by adding the same items to Y and Z, will also have the same closure. We can hence derive the supersets of Z thanks to those of Y by substituting Y by Z. Note that in the case of the MG family, such supersets are obtained by adding items not already belonging to Y and Z, otherwise they will not be MGs. This relation hence uses a substitution operator denoted *Subst* allowing to replace a subset Y of an itemset X by another itemset Z belonging to the same γ-equivalence class as Y. This operator is then defined as follows:

Definition 44 (SUBSTITUTION OPERATOR)
Let X, Y and Z be three itemsets s.t. $Y \subset X$, $\gamma(Y) = \gamma(Z)$, and $(X \backslash Y) \cap Z = \emptyset$. The substitution operator Subst, w.r.t. X, Y and Z, is defined as follows: $Subst(X, Y, Z) = (X \backslash Y) \cup Z$.

To prove that X and $Subst(X, Y, Z)$ have the same closure, we need the following lemma.

Lemma 5 *Let X and Y be two itemsets. X and Y fulfill the following property: $\gamma(X \cup Y) = \gamma(\gamma(X) \cup \gamma(Y))$ [Pasquier, 2000].*

Proposition 16 *X and Subst(X, Y, Z) belong to the same γ-equivalence class.*

Proof. Let W be the result of *Subst*(X, Y, Z), *i.e.*, $W = (X \backslash Y) \cup Z$. We will show that X and W have the same closure. Using Lemma 5, we have: $\gamma(X) = \gamma((X \backslash Y) \cup Y) = \gamma(\gamma(X \backslash Y) \cup \gamma(Y))$. Since $\gamma(Y) = \gamma(Z)$, then $\gamma(X) = \gamma(\gamma(X \backslash Y) \cup \gamma(Y)) = \gamma(\gamma(X \backslash Y) \cup \gamma(Z)) = \gamma((X \backslash Y) \cup Z) = \gamma(W)$. Hence, $\gamma(X) = \gamma(W)$. Thus, we can conclude that X and W necessarily belong to the same γ-equivalence class. $\diamondsuit$

For each γ-equivalence class $\mathcal{C}$ (or equivalently, for each CI f), the substitution operator induces an equivalence relation on the set MG_f portioning it into σ-equivalence classes. The definition of a σ-equivalence class requires that we redefine the notion of *redundant* MG under the point of view of the substitution operator *Subst*. Indeed, according to the definition given by Dong *et al.* (see Definition 41), *redundant* MGs are blindly pruned according to purely syntactic properties, that only consist in checking the order of their subsets *w.r.t.* $\preceq$, in their respective γ-equivalence classes. Hence, we propose to incorporate a semantic part based on the actual concept of redundancy. This is captured by the following definition.

Definition 45 (Minimal generators redundancy)
Let g and g_1 be two MG*s belonging to the same γ-equivalence class.*

- *g is said to be a **direct redundant** (resp. derivable) with respect to (resp. from) g_1, denoted $g_1 \vdash g$, iff $\exists\ g_2 \subset g_1$, $\exists\ g_3 \in \mathcal{MG}$ s.t. $\gamma(g_2) = \gamma(g_3) \Longrightarrow Subst(g_1, g_2, g_3) = g$.*
- *g is said to be a **transitive redundant** with respect to g_1, denoted $g_1 \vDash g$, if there is a sequence of n* MG*s $gen_1, gen_2, \ldots, gen_n$, such that $gen_i \vdash gen_{(i+1)}$ $(i \in \{1, \ldots, (n - 1)\})$ where $gen_1 = g_1$ and $gen_n = g$.*

Remark 4 *It is worth noting that the substitution relation $\vdash$ can be considered as a special case of the well known Armstrong axiom of pseudo-transitivity [Armstrong, 1974]:*

$$\frac{X \rightarrow Y; WY \rightarrow Z}{WX \rightarrow Z}.$$

In the substitution Subst(WY, Y, X) $= WX$ we formalized, the following constraints on the above general rule apply: (i) X and Y belong to the same γ-equivalence class (hence $Y \rightarrow X$ is also true), (ii) X, Y, WX and WY are MG*s, and (iii) Z is the closure of WY.*

The next proposition states that it is sufficient to use *immediate*, and not *all*, subsets of MGs to get redundant ones.

Proposition 17 *Let g and g_1 be two* MG*s belonging to the same γ-equivalence class. If $g_1 \vdash g$, then there are two* MG*s g_2 and g_3 s.t. $|g_2| = |g_1|$ - **1**, $|g_3| = |g|$ - **1**, and Subst(g_1, g_2, g_3) $= g$ holds.*

Proof. Suppose we have $g_1 \vdash g$. There are then two MGs g'_2 and g'_3 *s.t.* *Subst*(g_1, g'_2, g'_3) $= g$, where $g'_2 \subset g_1$, $g'_3 \in \mathcal{MG}$ and $\gamma(g'_2) = \gamma(g'_3)$.

Let $W = g_1 \backslash g'_2$. The MG g is then equal to $W \cup g'_3$. Let $W_1 \subset W$ *s.t.* $|W_1| = |W|$ - **1**, $g_2 = W_1 \cup g'_2$, and $g_3 = W_1 \cup g'_3$. We then have g_2 (*resp.* g_3) as an immediate subset of g_1 (*resp.* g). Being respective subsets of two MGs, g_2 and g_3 are also two MGs.

We have to prove that we can obtain $g_1 \vdash g$ using g_2 and g_3 instead of g'_2 and g'_3. This requires proving that *Subst*(g_1, g_2, g_3) = g holds. We have $g_2 \subset g_1$. We also have $(g_1 \backslash g_2) \cap g_3 = \emptyset$, since otherwise g_1 is not a MG, which it is in contradiction with the initial hypothesis. It remains to be proven that $\gamma(g_2) = \gamma(g_3)$.

We have $\gamma(g_2) = \gamma(W_1 \cup g'_2) = \gamma(\gamma(W_1) \cup \gamma(g'_2))$. According to the definition of the substitution operator *Subst* (*cf.* Definition 44), we have $\gamma(g'_2) = \gamma(g'_3)$. Thus, $\gamma(g_2) = \gamma(\gamma(W_1) \cup \gamma(g'_3)) = \gamma(W_1 \cup g'_3) = \gamma(g_3)$. We then have $\gamma(g_2) = \gamma(g_3)$.

It follows that *Subst*(g_1, g_2, g_3) = g holds. Hence, the MG g is then redundant *w.r.t.* the MG g_1 using their respective immediate subsets, namely g_2 and g_3, within the substitution operation. $\diamondsuit$

According to the previous proposition, to check whether $g_1 \vdash g$, it is sufficient to use their respective *immediate* subsets and not *all*. This constitutes an important optimization for computing redundant MGs.

Proposition 18 presents the properties fulfilled by both substitution relations.

Proposition 18 *The substitution relations* $\vdash$ *and* $\vDash$ *respectively fulfill the following properties:*

- *The substitution relation* $\vdash$ *is reflexive, symmetric.*
- *The substitution relation* $\vDash$ *is reflexive, symmetric and transitive.*

Proof. Let g, g' and g'' be three MGs belonging to the same γ-equivalence class.

- The relation $\vdash$ is:
  - **reflexive:** According to the definition of the substitution relation $\vdash$ (*cf.* Definition 45), if $g_2 = g_3$, we straightforwardly have $g \vdash g$.
  - **symmetric:** Suppose we have $g \vdash g'$. According to Definition 45, there is then g_2 and g_3 *s.t.* *Subst*(g, g_2, g_3) = g'. Hence, according to Definition 44, *Subst*(g', g_3, g_2) = g, which leads to $g' \vdash g$.
- The relation $\vDash$ is:
  - **reflexive:** For n = **2**, the operator $\vDash$ is simply reduced to the operator $\vdash$. Since $g \vdash g$, then we have $g \vDash g$.
  - **symmetric:** Suppose we have $g \vDash g'$. According to Definition 45, there is then a sequence of MGs, constructed by successive direct substitution using $\vdash$, that starts in g and finish in g'. It is hence sufficient to permute, in each application of $\vdash$, the substituted itemset by the corresponding substitute and vice versa to get $g' \vDash g$.
  - **transitive:** Suppose we have $g \vDash g'$ and $g' \vDash g''$. This means the existence of a first sequence of substitutions starting from g and reaching g', and a second one from g' to g''. The union of both sequences while considering g' as a connection point leads to a third sequence starting in g and leading to g''. We thus obtain $g \vDash g''$.

$\diamondsuit$

According to Proposition 18, the $\vDash$ operator fulfills the reflexive, symmetric and transitive properties. Hence, it induces an equivalence relation on the MGs of a given CI, portioning them into σ-equivalence classes. The formal definition of a σ-equivalence class is as follows:

Definition 46 (**σ-Equivalence class**)
Let f be a CI. *If $g \in \mathrm{MG}_f$, then the σ-equivalence class of g, denoted by $\sigma\text{-EC}_g$, is the subset of MG_f consisting of all elements that are transitively redundant w.r.t. g. In other words, we have: $\sigma\text{-EC}_g = \{g_1 \in \mathrm{MG}_f \mid g \vDash g_1\}$.*

To uniquely define a representative MG for each σ-equivalence class, we adopt the same total order relation between itemsets used in the original approach (*cf.* Definition 40, page 52). Once this relation established, we can define succinct and redundant MGs as given by Definition 47.

Definition 47 (**Succinct and redundant minimal generators**)
Let $\preceq$ be a total order relation and σ-EC be a σ-equivalence class. The smallest MG *in σ-EC, w.r.t. $\preceq$, is called* ***succinct*** MG, *while the remaining ones are tagged as* ***redundant*** *ones.*

Definition 47 makes it possible, for each σ-equivalence class, to only maintain a representative MG – the succinct one – and, hence, eliminate the remaining ones since they are redundant *w.r.t.* the maintained one according to Definition 45. Obviously, each MG having all its immediate subsets unique in their γ-equivalence classes will constitute a σ-equivalence class by itself. Indeed, no substitution is possible in this case. Algorithm 1 offers a straightforward method for extracting the different σ-equivalence classes associated to a CI f.

Algorithm 1: σ-Equivalence_Classes_Miner

Input: The set MG_f of the MGs associated to f.
Output: The σ-equivalence classes and the set $\mathrm{MG}\texttt{suc}_f$ of succinct MGs associated to f.

1 **Begin**
2 $\quad\mathcal{S} := \mathrm{MG}_f$;
3 $\quad\mathrm{MG}\texttt{suc}_f = \emptyset$;
4 $\quad i := \mathbf{0}$;
5 $\quad$**While** $(\mathcal{S} \neq \emptyset)$ **Do**
6 $\quad\quad i := i + \mathbf{1}$;
7 $\quad\quad g_s := min_{\preceq}(\mathcal{S})$; /*$g_s$ is the smallest MG in $\mathcal{S}$ *w.r.t.* $\preceq$.*/
8 $\quad\quad \mathrm{MG}\texttt{suc}_f = \mathrm{MG}\texttt{suc}_f \cup \{g_s\}$;
9 $\quad\quad \sigma\text{-EC}_i := \{g_s\} \cup \{g \in \mathcal{S} \mid g_s \vDash g\}$;
10 $\quad\quad \mathcal{S} := \mathcal{S} \backslash \sigma\text{-EC}_i$;
11 **End**

Remark 5 *The different σ-equivalence classes associated to a given* CI f *are a partition of* MG_f*. They hence verify the following properties:*

1. $\forall\, i \in \{1, \ldots, |\mathrm{MG}\,\mathit{suc}_f|\}$, σ-$\mathrm{EC}_i \neq \emptyset$,
2. $\forall\, i, j \in \{1, \ldots, |\mathrm{MG}\,\mathit{suc}_f|\}$ *s.t.* $i \neq j$, σ-$\mathrm{EC}_i \cap \sigma$-$\mathrm{EC}_j = \emptyset$, *and,*
3. $\cup_{i=1}^{i \leq |\mathrm{MG}\,\mathit{suc}_f|}\ \sigma$-$\mathrm{EC}_i = \mathrm{MG}_f$.

Example 29 *Let us consider the extraction context depicted by Table 4.2 (cf. page 58), the ascending support order as a total order relation $\preceq$ and the γ-equivalence class having for* CI *EACBDF. Using Algorithm 1, the* MG*s associated to EACBDF are divided as follows:*

1. *First, $\mathcal{S} = \mathrm{MG}_{EACBDF} = \{EA, ECF, EDF, ACB, ACD, ABD, ABF, ADF, CBF, CDF, BDF\}$ and $i = \mathbf{1}$. EA is the smallest* MG *in $\mathcal{S}$. Hence, σ-$\mathrm{EC}_1 = \{EA\} \cup \{g \in \mathcal{S} \mid EA \vDash g\}$. However, none* MG *can be deduced from EA. Thus, σ-$\mathrm{EC}_1 = \{EA\}$.*

2. *Second, $\mathcal{S} = \mathcal{S} \backslash \sigma$-$\mathrm{EC}_1 = \{EA, ECF, EDF, ACB, ACD, ABD, ABF, ADF, CBF, CDF, BDF\} \backslash \{EA\} = \{ECF, EDF, ACB, ACD, ABD, ABF, ADF, CBF, CDF, BDF\}$ and $i = \mathbf{2}$. ECF is the smallest one in $\mathcal{S}$. Hence, σ-$\mathrm{EC}_2 = \{ECF\} \cup \{g \in \mathcal{S} \mid ECF \vDash g\} = \{ECF\} \cup \{EDF, ACB, ABD, ABF, CBF, BDF\}$. Indeed, Subst$(ECF, EC, ED) = EDF \in \mathrm{MG}_{EACBDF}$ ($ECF \vdash EDF$, and hence $ECF \vDash EDF$), Subst$(ECF, EC, CB) = CBF \in \mathrm{MG}_{EACBDF}$ ($ECF \vdash CBF$, and hence $ECF \vDash CBF$), Subst$(CBF, CF, AC) = ACB \in \mathrm{MG}_{EACBDF}$ ($ECF \vDash ACB$ since $ECF \vdash CBF$ and $CBF \vdash ACB$), etc.*

3. *Finally, $\mathcal{S} = \mathcal{S} \backslash \sigma$-$\mathrm{EC}_2 = \{ECF, EDF, ACB, ACD, ABD, ABF, ADF, CBF, CDF, BDF\} \backslash \{ECF, EDF, ACB, ABD, ABF, CBF, BDF\} = \{ACD, ADF, CDF\}$ and $i = \mathbf{3}$. ACD is the smallest* MG *in $\mathcal{S}$. Hence, σ-$\mathrm{EC}_3 = \{ACD\} \cup \{g \in \mathcal{S} \mid ACD \vDash g\} = \{ACD\} \cup \{ADF, CDF\}$ since Subst$(ACD, AC, AF) = ADF$ ($ACD \vdash ADF$, and hence $ACD \vDash ADF$) and Subst$(ACD, AC, CF) = CDF$ ($ACD \vdash CDF$, and hence $ACD \vDash CDF$).*

Thus, MG_{EACBDF} is divided into three σ-equivalence classes as follows (succinct MG*s are marked with bold letters): $\mathrm{MG}_{EACBDF} = \{EA\} \cup \{ECF, EDF, ACB, ABD, ABF, CBF, BDF\} \cup \{ACD, ADF, CDF\}$. Note that ECF was not considered as a succinct* MG *according to the original definition that was introduced by Dong et al., since its subset CF is not the representative* MG *of its* CI *ACF. Hence, all* MG*s belonging to σ-EC_2 cannot be inferred according to their definition, contrary to ours.*

Remark 6 *For the same context, if we consider the descending support order as a total order relation $\preceq$, then we will note that the* OSSMG*, as formerly defined by Dong et al., can even contain redundancy in comparison to our definition. Indeed, thanks to the substitution operator* Subst, MG_{BDFACE} *is divided as follows: $\mathrm{MG}_{BDFACE} = \{AE\} \cup \{BDF, BDA, BFA, BFC, BAC, DFE, FCE\} \cup \{DFA, DFC, DAC\}$. The storage of the* MG*s BDA and BFA is then redundant and useless since they can simply be inferred starting from the succinct* MG *BDF* ($BDF \vDash BDA$ *and* $BDF \vDash BFA$). *Indeed,* Subst$(BDF, BD, BC) = BFC$, Subst$(BFC, FC, FA) = \underline{BFA}$, Subst$(BFA, FA, AC) = BAC$, *and finally,* Subst$(BAC, BC, BD) = \underline{BDA}$.

Using the new definitions of both *succinct* and *redundant* MGs (*cf.* Definition 47), we can now introduce the redefined succinct system of minimal generators (RSSMG) as follows:

Definition 48 (Redefined Succinct System of Minimal Generators)
Given a total order relation $\preceq$, the redefined succinct system of minimal generators is the set of all succinct MG*s of the* CI*s.*

According to Proposition 19, the number of *succinct* MGs associated to each CI f (*i.e.*, $|\mathrm{MG}\,\mathrm{suc}_f|$) is equal to the number of σ-equivalence classes induced by the substitution relation $\vDash$, independently of

the chosen total order relation. Hence, the cardinality of the set $\mathcal{MG}_{\texttt{suc}}$ remains unchanged even if we change the total order relation. In other words, the different RSSMGs associated to an extraction context have the same size, whatever the inherent total order relation.

Proposition 19 *Whatever the used total order relation $\preceq$, the substitution operator Subst maintains unchanged the elements belonging to each σ-equivalence class.*

Proof. Let $\preceq_1$ and $\preceq_2$ be two different total order relations. Let f be a CI and MG_f be the set of its associated MGs. Using $\preceq_1$, MG_f will be divided into σ-equivalence classes. Let $\sigma\text{-EC}_{\preceq_1}$ be one of them and g_{s_1} be its *succinct* MG (*i.e.*, the smallest one in $\sigma\text{-EC}_{\preceq_1}$ *w.r.t.* $\preceq_1$). $\sigma\text{-EC}_{\preceq_1}$ can be represented by a tree, denoted $\mathrm{T}_{\preceq_1}$. The root of $\mathrm{T}_{\preceq_1}$ contains the *succinct* MG g_{s_1}. In this tree, a node N, which represents a MG g, points to a node N_1, which represents a MG g_1, if $g \vdash g_1$. Hence, from whatever node in $\mathrm{T}_{\preceq_1}$, we can access the remaining nodes as follows: we move downward from the node N to the node N_1 using the relation $g \vdash g_1$ and conversely, from N_1 to N using the dual relation $g_1 \vdash g$. Indeed, if $\mathit{Subst}(g, g_2, g_3) = g_1$ where $g_2 \subset g$ and $g_3 \in \mathcal{MG}$ *s.t.* $\gamma(g_3) = \gamma(g_2)$, then we also have $\mathit{Subst}(g_1, g_3, g_2) = g$ since the relation $\vdash$ is symmetric (*cf.* Proposition 18, page 61).

Now, consider the set $\sigma\text{-EC}_{\preceq_1}$ ordered *w.r.t.* the second total order relation $\preceq_2$. The obtained new set will be denoted $\sigma\text{-EC}_{\preceq_2}$ and its associated *succinct* MG will be denoted g_{s_2}. Hence, if we transform the tree $\mathrm{T}_{\preceq_1}$ in a new one, denoted $\mathrm{T}_{\preceq_2}$ and rooted in g_{s_2}, then we are able to reach *all* remaining MGs contained in $\sigma\text{-EC}_{\preceq_2}$ thanks to the substitution-based operations as explained above. Thus, the modification of the total order relation does not affect the content of the $\sigma\text{-EC}_{\preceq_1}$ since it does not involve the deletion of any node in $\mathrm{T}_{\preceq_1}$.

Furthermore, this modification does not augment the $\sigma\text{-EC}_{\preceq_2}$ size by an additional *redundant* MG. Indeed, suppose that a MG denoted g_{new}, not already belonging to $\sigma\text{-EC}_{\preceq_1}$, will be added to $\sigma\text{-EC}_{\preceq_2}$ once we shift the total order relation from $\preceq_1$ to $\preceq_2$ (*i.e.*, $g_{s_2} \vDash g_{new}$ but $g_{s_1} \nvDash g_{new}$). Since $g_{s_1} \vDash g_{s_2}$ ($g_{s_2} \in \sigma\text{-EC}_{\preceq_1}$) and $g_{s_2} \vDash g_{new}$, then $g_{s_1} \vDash g_{new}$. Indeed, starting from the fact that the relation $\vDash$ is transitive (*cf.* Proposition 18), then g_{new} should belong to $\sigma\text{-EC}_{\preceq_1}$ (according to Definition 46). This result is in contradiction with the starting assumption ($g_1 \nvDash g_{new}$). Thus, $g_2 \nvDash g_{new}$.

Therefore, we can conclude that the elements belonging to $\sigma\text{-EC}_{\preceq_2}$ are exactly the same as those contained in $\sigma\text{-EC}_{\preceq_1}$, ordered *w.r.t.* $\preceq_2$ instead of $\preceq_1$. $\diamondsuit$

Example 30 *If we review both Example 29 and Example 6, we note that $\sigma\text{-EC}_1$, $\sigma\text{-EC}_2$ and $\sigma\text{-EC}_3$ are exactly the same for both examples, even though they are sorted according to the ascending support order and to the descending support order, respectively.*

4.3.2 Regenerating All Minimal Generators

According to Proposition 20 given hereafter, the succinct system of minimal generators, as redefined in Definition 48, becomes an *exact* concise representation of the MG set.

Proposition 20 *The definition of the* RSSMG *ensures the inference of each redundant* MG g.

Proof. Since g is a *redundant* MG, then g is not the smallest one in its σ-equivalence class. Hence, according to the definition of a σ-equivalence class (see Definition 46), there is necessarily a *succinct* MG

g_s belonging to the RSSMG whose substitution process certainly leads to g ($g_s \vDash g$), since the number of MGs belonging to each σ-equivalence class is finite. $\diamondsuit$

In conclusion, both Proposition 19 and Proposition 20 allow to correct the claims of Dong *et al.* [Dong *et al.*, 2005] thanks to the new consideration of the concept of redundancy within the MG set. In addition, the RSSMG is a *perfect cover* of the MG set. Proposition 21 shows this interesting property of the RSSMG.

Proposition 21 *The redefined succinct system of minimal generators is a perfect cover of the* MG *family.*

Proof. The proof is ensured by the fact that the RSSMG is only composed by MGs and that the cardinality of $\mathcal{MG}_{\texttt{suc}}$ is always smaller than that of $\mathcal{MG}$. $\diamondsuit$

Finally, we note that the cardinality of the RSSMG is intrinsic to the associated context, in the sense that it is independent of all the (subjective) constraints/preferences, like the order relation choice, end-user's preferences *w.r.t.* the items, etc. In this respect, we showed that this system can be an interesting mean for a formal characterization of the extraction contexts sparseness [Hamrouni *et al.*, 2009a].

4.3.3 Problems in the RSSMG

As shown in the previous section, this new definition of the succinct system of minimal generators constitutes a lossless reduction of the MG family of constant size *w.r.t.* the total order relation $\preceq$. Nevertheless, this approach still presents a main limitation. Indeed, the interesting order ideal property – which is usually exploited as an efficient pruning of the search space – is not preserved. In fact, the choice of the unique class member to keep in the RSSMG has been disconnected from any order. For example, if we consider the extraction context given in Table 4.2 and the ascending support order as a total order relation, the MG `ECF` will be characterized as a *succinct* MG since it is the smallest one in its σ-equivalence class. However, its subset `CF` is not the smallest one in its γ-equivalence class (or equivalently, is not a representative MG *w.r.t.* the original definition given by Dong *et al.*). Hence, additional tests have to be performed to guess whether a MG is a *succinct* one or not. Thus, the compaction of the RSSMG is conditioned by the necessity of more extensive computation effort in the construction of the system, in particular, for testing reducibility between MGs.

To overcome this limitation, the next section offers a lossless reduction of the MG family while preserving the interesting order ideal property of the obtained system. We start with a summary of the relative merits of both proposed succinct systems of minimal generators, which motivates the developments presented in the remainder of the chapter.

It is worth pointing out that the picture gets more regular on the higher granularity level, *i.e.*, within a γ-equivalence class. Indeed, whatever the used item order for its generation, the ideal of representative/canonical MGs has at least one element in each γ-equivalence class. This fact admits an immediate proof based on the same induction employed in the completeness proof for our expansion procedure (*cf.* Subsection 4.4.2). Moreover, one can easily show that given a γ-equivalence class, the *representative*, *i.e.*, minimal set *w.r.t.* any linear extension of the $\subseteq$-induced order, which is the case here, is necessarily a

MG. As canonical elements from the border are also required to be MGs, one might (too) easily conclude that the entire border is in the MG family. Unsurprisingly, this does not hold in the general case: there will be non-MG elements whose every subset is representative. The existence of these elements seems to have been missed by Dong *et al.*, although they play the same role in the substitution mechanism as the canonical MGs. In fact, this is the main reason for their expansion mechanism to be incomplete, *i.e.*, to fail in the recovery of some of the redundant MGs from the OSSMG.

One may now question the interplay between the substitution and the total order, *i.e.*, in what sense the representative/canonical sets are *irreducible* for substitution? After all, the substitution is a reversible operator, so that any MG within a σ-equivalence class could have been chosen as its distinguished element to be kept. Although [Dong *et al.*, 2005] says little on that point, our analysis shows that the representative/canonical order ideal structure is crucial. Indeed, it works like a magnetic nucleus for substitution in the sense that when properly performed, *i.e.*, in the *right* direction, it transforms an arbitrary itemset into a member of the representative ideal or of its (complete) border. Here, the right direction is the substitution of a subset Z_1 in the argument X by the representative Z_2 in the γ-equivalence class of Z_1. We prove below that this inevitably "attracts" the result within the aforementioned set where such substitutions can no more be performed.

Our proposal is about completing the succinct representation with all those non-MGs from the border of the order ideal of representative MGs, as in many cases they are the unique point from which some of the redundant MGs can be reached by substitution. The details of our approach come in the following paragraphs.

4.4 Directed Substitution-Free Sets

The idea behind our hybrid approach is to "repair" the flaws in the proposal of Dong *et al.* while preserving the exactness label of the obtained representation, as in the proposed system in the previous section.

4.4.1 Description

Here is an illustration of the above arguments: Assume the ascending support order on items in Table 4.2 and consider the MGs of `EACBDF`. As pointed out above, it is impossible to derive `ECF` from the resulting original succinct system of minimal generators (OSSMG). Indeed, its subset `CF` is a non-representative MG (`AC` is the representative in its γ-equivalence class). Hence, it remains outside the system, whereas neither `EA` nor `ACD` have a derivation chain that ends at `ECF`. If we look the case other way round, the only sensible substitution from `CF` backwards is `CF`/`AC`. This produces `EAC`, a curious set whose every subset is a representative (hence it belongs to the border of the corresponding ideal) without the set itself being even a MG. Clearly, adding `EAC` to the OSSMG would restore its completeness. This leads to a larger definition of the canonicity which we provide below.

Definition 49 (Negative border of representative minimal generators)
Let $\mathcal{MG}_{\texttt{rep}}$ be the set of the representative MG*s that can be extracted from a context $\mathcal{K}$. The negative border of $\mathcal{MG}_{\texttt{rep}}$ is: $\mathcal{B}d^{-}(\mathcal{MG}_{\texttt{rep}}) = \{X \subseteq \mathcal{I} \mid \forall\, Y \subset X, Y \in \mathcal{MG}_{\texttt{rep}}$ and $X \notin \mathcal{MG}_{\texttt{rep}}\}$.*

Since canonical itemsets form the negative border of the representative ideal, the old canonical MGs of Dong *et al.* are obviously included in it ($\mathcal{MG}_{\texttt{can}} \subseteq \mathcal{B}d^{-}(\mathcal{MG}_{\texttt{rep}})$), together with the canonical non-MG

elements. Moreover, the frequency constraint further splits it into four subsets.

In order to formalize the irreducibility status of the above sets, we rely on a constrained substitution operator. Actually, we distinguish two complementary "directions" for the substitution depending on the status of the involved sets. Thus, a positive (*resp.* negative) substitution for an itemset X amounts to replace a subset Z_1 of X by a set Z_2 of the same closure as Z_1 which is larger (*resp.* smaller) *w.r.t.* the itemset order. We focus on the relations between the set X and the set Y induced by one of the substitutions defined through Definition 50 and Definition 51.

Definition 50 (Positive substitution)
Let $X, Y \subseteq \mathcal{I}$, $Z_1 \subset X$ and $Z_2 \subseteq \mathcal{I}$ s.t. $\gamma(Z_1) = \gamma(Z_2)$. The positive direct and transitive substitution operators, denoted respectively $\vdash^+$ and $\models^+$, are defined as follows:

- *Y is said to be a **positive direct redundant** (derivable) with respect to (from) X, denoted $X \vdash^+ Y$, if Subst$(X, Z_1, Z_2) = Y$ and $Z_1 \preceq Z_2$.*
- *Y is said to be a **positive transitive redundant** w.r.t. X, denoted $X \models^+ Y$, if there is a sequence of n itemsets $I_1, I_2, \ldots, I_n$, such that $I_i \vdash^+ I_{(i+1)}$ $(i \in \{1, \ldots, (n\text{ - }1)\})$ with $I_1 = X$ and $I_n = Y$.*

Definition 51 (Negative substitution)
Let $X, Y \subseteq \mathcal{I}$, $Z_1 \subset X$ and $Z_2 \subseteq \mathcal{I}$ s.t. $\gamma(Z_1) = \gamma(Z_2)$. The negative direct and transitive substitution operators, denoted respectively $\vdash^-$ and $\models^-$, are defined as follows:

- *Y is said to be a **negative direct redundant** (derivable) with respect to (from) X, denoted $X \vdash^- Y$, if Subst$(X, Z_1, Z_2) = Y$ and $Z_2 \preceq Z_1$.*
- *Y is said to be a **negative transitive redundant** w.r.t. X, denoted $X \models^- Y$, if there is a sequence of n itemsets $I_1, I_2, \ldots, I_n$, such that $I_i \vdash^- I_{(i+1)}$ $(i \in \{1, \ldots, (n\text{ - }1)\})$ with $I_1 = X$ and $I_n = Y$.*

The operator $\vdash^+$ (*resp.* $\vdash^-$) is reflexive, symmetric but not necessarily transitive, while the operator $\models^+$ (*resp.* $\models^-$) fulfills the three properties.

It is noteworthy that each substitution is either positive or negative, *i.e.*, there is no neutral substitution. Moreover, positive substitutions produce results that are larger *w.r.t.* $\preceq$ than the initial sets and hence have bigger ranks in the order ($X \vdash^+ Y$ implies $X \preceq Y$), while the negative ones have the opposite effect. In particular, if the replaced set is a representative, then the substitution is necessarily positive, while, conversely, if a representative replaces another set, then it is negative. This is formally shown using Definition 52 and Proposition 22 as follows.

The following definition introduces the function ρ which associates to each itemset its rank *w.r.t.* the total order relation $\preceq$.

Definition 52 (Rank function)
Given the total order relation $\preceq$ and an itemset X, the rank function ρ is defined as follows:

$$\begin{aligned} \rho : \mathcal{P}(\mathcal{I}) &\rightarrow \mathbb{N} \\ X &\mapsto \rho(X) \end{aligned}$$

such that by default, $\rho(\emptyset) = \mathbf{0}$, and $\forall\, Y, Z \subseteq \mathcal{I}$, $\rho(Y) < \rho(Z)$ iff $Y \prec Z$.

It is worth noting that, once the total order relation adopted, the rank of each itemset is immediately set. Proposition 22 shows the effect of a positive/negative substitution on the rank of the obtained itemset.

Proposition 22 *Let X and Y be two itemsets s.t. $X \neq Y$.*
- If $X \vdash^{+} Y$, then $\rho(X) < \rho(Y)$.
- If $X \vdash^{-} Y$, then $\rho(X) > \rho(Y)$.

Consider now the irreducible elements for the negative substitution, *i.e.*, elements for which such substitution could not be applied. We call them *directed substitution-free* sets (denoted DSFSs).

Definition 53 (Directed Substitution-Free Sets)
Let $\mathcal{DSFS}$ be the collection of the directed substitution-free sets that can be extracted from a context $\mathcal{K}$. $\mathcal{DSFS} = \{I \subseteq \mathcal{I} \mid \forall\ I_1 \subset I, \forall\ I_2 \subseteq \mathcal{I},\ (\gamma(I_1) = \gamma(I_2) \Longrightarrow I_1 \preceq I_2)\}$.

Example 31 *Consider the context in Table 4.2 (cf. page 58) with the ascending support order on items. The itemset* `EAC` *is a* DSFS*, as mentioned above, whereas the family comprises* `EA` *and* `ACD`*, but not* `ECF`*.*

Clearly, the set of DSFSs equals the union of representative MGs and their negative border ($\mathcal{DSFS} = \mathcal{MG}_{\texttt{rep}} \cup \mathcal{B}d^{-}(\mathcal{MG}_{\texttt{rep}})$). The next proposition is therefore immediate.

Proposition 23 *The set $\mathcal{DSFS}$ is an order ideal of $(\mathcal{P}(\mathcal{I}), \subseteq)$.*

Given its structure, the DSFS family can be easily constructed by a levelwise algorithm that, additionally, enumerates itemsets in the order $\preceq$. Thus, all the DSFSs at a particular level are easily recognizable since all their subsets (in particular the maximal ones) belong to the already discovered part of the family. An additional effort is necessary to identify the representative itemsets among all the family members. To that end, the order properties are exploited. In fact, a representative is the first itemset to be examined within its γ-equivalence class. Hence, to establish that a DSFS is a representative, it is enough to check that its closure has not been produced by a previously extracted DSFS.

4.4.2 Regenerating All Minimal Generators

So far, we have established that any total order on itemsets generates a core ideal in the Boolean lattice that works as an irreducible nucleus for swapping subsets with equivalent ones. On the reverse side of the question, there is the expansion process: it starts with the DSFSs and retrieves the entire MG family. Unsurprisingly, the positive substitution is used to that end. Moreover, as for each negative substitution there is a reverse positive one, and vice versa, every itemset from the Boolean lattice is necessarily reachable by at least one chain of positive substitutions starting from a DSFS. In particular, redundant MGs are reachable in this way.

Following to the above arguments, we claim that every redundant MG can be derived from a DSFS of the same closure, using positive substitutions. More specifically, starting from the DSFS X, and operating successive substitutions of a representative subset Z_1 by a non-representative set Z_2 from the same γ-equivalence class will necessarily result in the generation of the entire MG family. Hence we can assert that the above retrieval mechanism, in its most general form is a complete mean for computing the MGs. Now we will prove that the expansion process starting from the set $\mathcal{DSFS}$ is complete.

Proposition 24 *The expansion process is* ***complete****. Indeed, given the set $\mathcal{DSFS}$ of the* DSFS*s that can be drawn from an extraction context $\mathcal{K}$, we are able to derive all minimal generators:*

$$\forall\ g \in \mathcal{MG},\ \exists\ \overline{g} \in \mathcal{DSFS}\ s.t.\ \overline{g} \vDash^{+} g.$$

Proof. Let g be a minimal generator. We show by induction on the rank of g how it can be reached starting from an element of $\mathcal{DSFS}$.

Base case: $g = \emptyset$ is trivial, since $\emptyset$ always belongs to $\mathcal{DSFS}$ because it is always the smallest element in $\mathcal{P}(\mathcal{I})$.

General case: We will mainly concentrate on redundant minimal generators. Indeed, if g is not a redundant one then it is contained in $\mathcal{DSFS}$. Hence, it is trivial that g can be derived from an element of $\mathcal{DSFS}$ (obviously, itself). Let g be a redundant minimal generator. $\rho(g)$ is necessarily greater than $\mathbf{0}$ ($\rho(g) > \mathbf{0}$) since the empty set can never be a redundant minimal generator.

Inductive hypothesis: $\forall\ \widetilde{g} \subseteq \mathcal{I}$ with $\rho(\widetilde{g}) < \rho(g)$, $\exists\ \widehat{g} \in \mathcal{DSFS}\ s.t.\ \widehat{g} \vDash^{+} \widetilde{g}$.

Let Y be the subset of g having the maximal rank among the non-representative subsets of g. Such an itemset Y necessarily exists since otherwise g would not be redundant. Furthermore, Y is necessarily a direct subset of g. Indeed, the existence of a smaller subset Z which is a redundant minimal generator implies that all direct subsets of g including Z are also redundant. Since the itemsets size is a dominant factor for determining $\preceq$, then Y (which we supposed to be maximal) is necessarily the biggest (for $\preceq$) among the direct subsets of g containing Z.

Let T be the itemset *s.t.* $T = min_{\preceq}\{I \subseteq \mathcal{I} \mid \gamma(I) = \gamma(Y)\}$. Let $W = \mathit{Subst}(g, Y, T)$. We necessarily have $\rho(W) < \rho(g)$ since $g \vdash^{-} W$ (according to Proposition 22).

Since $\rho(W) < \rho(g)$, by hypothesis, $\exists\ \widehat{g} \in \mathcal{DSFS}\ s.t.\ \widehat{g} \vDash^{+} W$. Since $W \vdash^{+} g$, we also have $\widehat{g} \vDash^{+} g$. Hence, g is derivable starting from $\mathcal{DSFS}$. $\diamondsuit$

Another concern with the retrieval is the correctness of the mechanism, *i.e.*, the warranty that only MGs will be retrieved. To that end, we employ a straightforward support test: An itemset is a MG whenever its support is strictly lower than the support of all its proper subsets. For efficiency reasons, this test is limited to maximal subsets only. Obviously, such a test requires a levelwise traversal of the Boolean lattice, which is a classic approach of frequent itemset mining. Consequently, we may assert that the expansion is correct as well.

Proposition 25 *The expansion process is* ***correct****.*

Proof. By construction, all derived elements from a DSFS are explicitly checked for being MGs. $\diamondsuit$

Proposition 25 states that the expansion process is correct *w.r.t.* the derivation of minimal generators (useful in the case where only these latter itemsets are looked for by the regeneration process). Theorem 7 states the adequacy of our global approach.

Theorem 7 *The set $\mathcal{DSFS}$ of the directed substitution-free sets* (DSFS*s*) *is a lossless representation of the minimal generator set.*

Proof. The proof straightforwardly derives from Proposition 24 and Proposition 25. $\diamondsuit$

Interestingly, the DSFS family is not only a lossless reduction of the MG family but also that of the *whole set of frequent itemsets*. This is stated thanks to the following theorem.

Theorem 8 *The set $\mathcal{DSFS}$ of the directed substitution-free sets* (DSFS*s*) *is a lossless representation of the set of frequent itemsets.*

Proof. As shown by the proof of Proposition 24, each frequent itemset contracts to a DSFS by negative substitutions. It is hence simply derivable from a DSFS using positive ones. $\diamondsuit$

4.5 Analysis and Comparison of the Proposed Systems

To sum up, the DSFS family is the complete structure necessary to ensure that every MG can be reached by a substitution-based expansion process that is well directed and hence cycle-free. The DSFSs can be efficiently mined, due to the order ideal form of the family as it does not even require the discovery of all MGs. Despite the significant progress with respect to the previous two studies, there are issues with our framework that are yet to be clarified.

First, while both succinct systems of minimal generators only rely on MGs, our construct of the DSFSs involves further sets from (yet laying not too far in) the Boolean lattice. The impact of these elements on the size of the representation needs to be examined. Some clues on how many non-MG DSFSs could appear are provided in Section 4.8.

Another issue, somewhat related to the previous one, concerns the expansion mechanism. An important feature thereof would be to limit all substitutions to MG subsets. In other terms, it would be much simpler and more efficient always to replace a representative MG Z_1 by a non-representative one Z_2, and not any arbitrary set from the same γ-equivalence class.

Finally, the minimality of the DSFS family is an important issue as well. Whereas it is definitely minimal for the entire Boolean lattice, it could be in some cases that a proper subset of the DSFSs suffices to generate all the MGs. For instance, if there are more than one DSFS in the same σ-equivalence class, then clearly only the smallest of them *w.r.t.* the total order is of need. Moreover, some non-MG DSFSs may not be of use for the expansion towards all MGs, so it could be useful to remove them from the effective representation. Provided a method for eliminating unnecessary DSFSs is designed, the trade-off between reduction rate and cost should also be looked at.

We now sketch the main properties of the MG families respectively associated to the three proposed approaches. We also compare them *w.r.t.* set inclusion. In the remainder of this section, we simply denote by DSFS the set of DSFSs.

Table 4.3 presents the properties fulfilled by each family *w.r.t.* the following axes:

1. `Lossless`: is the family extracted without information loss?
2. `Perfect`: is the family a subset of the MG family? [2]
3. `Total order relation` $\preceq$: Can the elements composing the family change *w.r.t.* $\preceq$? and, does its size depend on $\preceq$?
4. `Order ideal`: does the family constitute an order ideal structure?

[2] It is unnecessary to check this feature if the family is extracted with information loss.

Approach	Lossless	Perfect	Total order $\preceq$		Order ideal
			w.r.t. content	*w.r.t.* size	
OSSMG	No	–	Yes	Yes	Yes
RSSMG	Yes	Yes	Yes	No	No
DSFS	Yes	No	Yes	Yes	Yes

Table 4.3: Properties of the proposed minimal generator families.

Family$_1$ *vs.* Family$_2$	Comparison
OSSMG *vs.* RSSMG	• OSSMG $\not\subseteq$ RSSMG (*cf.* Table 4.2 (*cf.* page 58) for the ascending support order, `BDA` $\in$ OSSMG but `BDA` $\notin$ RSSMG) • RSSMG $\not\subseteq$ OSSMG (*cf.* Table 4.2 for the descending support order, `ECF` $\in$ RSSMG but `ECF` $\notin$ OSSMG)
OSSMG *vs.* DSFS	• OSSMG $\subseteq$ DSFS
RSSMG *vs.* DSFS	• RSSMG $\not\subseteq$ DSFS (*cf.* Table 4.2 for the descending support order, `ECF` $\in$ RSSMG but `ECF` $\notin$ DSFS) • DSFS $\not\subseteq$ RSSMG (RSSMG is a perfect cover of the MG family and, hence, cannot contain non-MG DSFSs)

Table 4.4: Comparison of the proposed minimal generator families *w.r.t.* set inclusion.

On the other hand, Table 4.4 allows to compare the different approaches *w.r.t.* set inclusion. It results from both tables that:

1. OSSMG, RSSMG, and DSFS respective contents depend on the total order relation $\preceq$. Nevertheless, only the size of OSSMG and DSFS can change once $\preceq$ is modified.
2. RSSMG and DSFS are lossless reductions of the MG set, contrary to OSSMG.
3. RSSMG is a perfect cover of the MG family, contrary to DSFS.
4. DSFS offers a lossless reduction of the MG family while preserving the interesting order ideal

Notation	Description
c	: A candidate itemset.
c.Supp	: The support of c.
c.FCI	: The closure of c.
c.Direct_subsets	: The list of immediate subsets of c.

Table 4.5: Notations used by the DSFS_Miner algorithm.

property, contrary to RSSMG.

5. OSSMG is a subset of DSFS. The difference is the set of non-MG DSFSs.
6. RSSMG is incomparable (*w.r.t.* set inclusion) with OSSMG and DSFS.

From an algorithmic point of view, aiming at exploiting the key property of order ideal, we propose in the next section a new algorithm, called DSFS_Miner, for an efficient extraction of the DSFS family.

4.6 The DSFS_Miner Algorithm

4.6.1 Description

Now, we sketch the key ideas related to an algorithm allowing the extraction of the DSFSs. This algorithm, called DSFS_Miner, uses a breadth-first (or levelwise) browsing of the search space. In each iteration, it hence treats minimal generator (MG) candidates by ascending size. For a given size, the associated candidates are sorted *w.r.t.* the total order relation $\preceq$. This is naturally obtained as soon as items are ordered *w.r.t.* $\preceq$. Indeed, the procedure we use to generate candidates of size $(i + 1)$, using those of size i, respects the total order relation since it combines each time two itemsets X and Y, *s.t.* $X \prec Y$, sharing their first $(i - 1)$ items. The latter items will be augmented by the remaining one in X and, then, by the remaining one in Y. Hence, the total order relation will always be respected.

The pseudo-code of the DSFS_Miner algorithm is given by Algorithm 2. While Table 4.5 summarizes the attributes characterizing a candidate. In the pseudo-code, the acronym $\mathcal{FDSFS}$ denotes the Frequent Directed Substitution-Free Sets that can be extracted from the extraction context $\mathcal{K}$. While the set of frequent closed itemsets, which can be extracted from $\mathcal{K}$, is denoted $\mathcal{FCI}$. The set of candidates, to be tested during the i^{th} iteration whether they are representative frequent MGs or not, is denoted $\mathcal{FMG}_{\texttt{rep}_i}$.

Since by definition, the representative MG is the smallest one in its γ-equivalence class, *w.r.t.* $\preceq$, c is a representative if it is the first one to produce the associated closure of its γ-equivalence class (*cf.* Algorithm 2, lines 12-13). To generate candidates of size $(i + 1)$ starting from representative frequent MGs of size i, DSFS_Miner uses the Gen-Representative procedure whose pseudo-code is given by Algorithm 3. The running of the latter is illustrated by Example 32.

Example 32 *Consider the context given by Table 4.2 (Top), page 58. Let minsupp be equal to* **1** *and the total order relation be the ascending support one. We will mainly focus on how our new definition is able to take in consideration an itemset such as* EAC *thanks to the tests used in the* Gen-Representative

Algorithm 2: DSFS_Miner

Input: - An extraction context $\mathcal{K}$ where items are sorted *w.r.t.* the total order relation $\preceq$, and the threshold of support *minsupp*.

Output: - The set $\mathcal{FDSFS}$.

```
1  Begin
2    FDSFS := {∅};
3    FCI := {γ(∅)};
4    FMG_rep1 := {{j} | j ∈ I \ γ(∅)};
5    ForEach (i = 1; FMG_rep_i ≠ ∅ ; i++) Do
6      FMG_rep_i := Gen-Closure(FMG_rep_i); /*The Gen-Closure procedure
       produces closures as done in [Pasquier et al., 1999b]. It also computes the
       candidates supports.*/
7      ForEach (c ∈ FMG_rep_i) Do
8        If (c.Supp < minsupp) Then
9          FMG_rep_i := FMG_rep_i \ {c};
10       Else
11         FDSFS := FDSFS ∪ {c};
12         If (c.FCI ∉ FCI) Then
13           FCI := FCI ∪ {c.FCI};
14         Else
15           FMG_rep_i := FMG_rep_i \ {c};
16     FMG_rep_(i+1) := Gen-Representative(FMG_rep_i);
17   Return FDSFS;
18 End
```

procedure (cf. Algorithm 3). Indeed, when generating the set of representative MG *candidates of size* **3**, *we have the* **2**-*representative frequent* MG*s* `EA` *and* `EC` *that have their first item in common. Hence, by composing them we obtain the candidate* `EAC` *(cf. lines 2-6). After that,* `EAC` *will be tested to check whether all its subsets are representative frequent* MG*s. It is the case. Hence, the value of the variable* `is-deleted` *remains equal to* **0**. *While that of* `is-covered` *will change and become equal to* **1** *since* `EAC` *is included in the closure of its subset* `EA`, *equal to* `EACBDF` *(cf. lines 7-17). After this test, we have the information that* `EAC` *has all its subsets as representative frequent* MG*s but is not a* MG *since it has the same closure than one of its subsets. Hence, it is a frequent non-*MG DSFS. *Thus,* `EAC` *belongs to the frequent part of the negative border. It will then be retained as an element of the representation (cf. lines 18-20).*

Algorithm 3: GEN-REPRESENTATIVE

Input: - The set $\mathcal{FMG}\texttt{rep}_i$.

Output: - The set $\mathcal{FMG}\texttt{rep}_{(i+1)}$.

Begin

/*The combinatorial phase of APRIORI-GEN [Agrawal and Srikant, 1994] *w.r.t.* the total order relation $\preceq$*/

insert into $\mathcal{FMG}\texttt{rep}_{(i+1)}$

select p[1], p[2], ..., p[i - **1**], p[i], q[i]

from $\mathcal{FMG}\texttt{rep}_i$ p, $\mathcal{FMG}\texttt{rep}_i$ q

where p[1] = q[1], p[2] = q[2], ..., p[i - **1**] = q[i - **1**], p[i] $\prec$ q[i];

ForEach ($c \in \mathcal{FMG}\texttt{rep}_{(i+1)}$) **Do**

`is-deleted` := **0**; /*This variable checks whether c is deleted because one of its immediate subsets is not a representative frequent MG of its γ-equivalence class.*/

`is-covered` := **0**; /*This variable checks whether c is covered by the closure of one of its immediate subsets.*/

ForEach ($c_1 \in c.\texttt{Direct\_subsets}$) **Do**

If ($c_1 \notin \mathcal{FMG}\texttt{rep}_i$) **Then**

$\mathcal{FMG}\texttt{rep}_{(i+1)} := \mathcal{FMG}\texttt{rep}_{(i+1)} \setminus \{c\}$;

`is-deleted` := **1**;

break;

Else

If ($c \subseteq c_1.\texttt{FCI}$) **Then**

`is-covered` := **1**;

If (*is-deleted* = **0** *and* *is-covered* = **1**) **Then**

$\mathcal{FDSFS} := \mathcal{FDSFS} \cup \{c\}$;

$\mathcal{FMG}\texttt{rep}_{(i+1)} := \mathcal{FMG}\texttt{rep}_{(i+1)} \setminus \{c\}$;

Return $\mathcal{FMG}\texttt{rep}_{(i+1)}$;

End

4.6.2 Correctness and Complexity

The next theorem states the soundness and the correctness of the DSFS_MINER algorithm.

Theorem 9 *The* DSFS_MINER *algorithm is sound and correct. It exactly extracts all elements belonging to the set* $\mathcal{DSFS}$.

Proof. The conjunction of two anti-monotone constraints, namely "to be frequent" and "to be minimal generator" is also anti-monotone. Thus, a levelwise algorithm guarantees that all frequent minimal generators are extracted as well as the associated negative border [Mannila and Toivonen, 1997]. In the case of the DSFS_Miner algorithm, only candidates having all their subsets representative frequent minimal generators will be maintained. The other ones will be pruned since they do not fulfill the condition of the DSFS set being an order ideal (*cf.* Proposition 23). During this candidate generation step, non-MG DSFSs will be retained in $\mathcal{DSFS}$. These latter, belonging to $\mathcal{B}d^{-}(\mathcal{MG}_{\texttt{rep}})$, are checked using the closure of their subsets. In addition, since DSFS_Miner also computes the closure of each frequent minimal generator g, this allows testing whether g is the first one that gives rise to a given closure. This means that g is a representative minimal generator of its γ-equivalence class. In this case, g will be used as a seed for generating candidates of larger size. In the case where g is a frequent minimal generator but not the first generating a closure, g will be added to $\mathcal{DSFS}$. However, g will not be used for generating next candidates. Indeed, g is a canonical minimal generator. Thus, DSFS_Miner is sound and correct. $\diamondsuit$

Proposition 26 shows the complexity of the DSFS_Miner algorithm.

Proposition 26 *In the worst case, the theoretical complexity of* DSFS_Miner *is in* $O((n^2 + m \times n) \times \mathbf{2}^n)$, *where* $n = |\mathcal{I}|$ *and* $m = |\mathcal{O}|$.

Proof. In the following, we suppose that items are sorted according to a total order relation. Thus, inclusion, intersection, union and difference operations between two itemsets are in $O(n)$. In the worst case, any set of items appears at least once in the context, and each candidate is a frequent MG, equal to its closure. Consequently, there is a unique element within each γ-equivalence class, namely the associated representative MG. Thus, there are no redundant MGs. We have $\mathbf{2}^n$ frequent itemsets such that each one forms its proper γ-equivalence class. Then, DSFS_Miner has to perform the following tasks:

1. Initializing sets which is in $O(m \times n)$ (*cf.* Algorithm 2, page 73, lines 2-4),
2. Incrementing i from 1 to n is in $O(n)$ (*cf.* line 5),
3. Computing the closures and supports of candidates which is in $O((m \times n) \times \mathbf{2}^n)$ (*cf.* line 6),
4. Verifying if candidates are representative frequent MGs or not. This is done in $O(n \times \mathbf{2}^n)$ (*cf.* lines 7-15),
5. Generating candidates in $O(\mathbf{2}^n - n)$, since $\mathbf{2}^n - n - \mathbf{1}$ candidates are to be generated in the worst case (*cf.* Algorithm 3, page 74, lines 2-6),
6. Testing whether candidates are MGs or non-MG DSFSs. This is carried out in $O(n^2 \times (\mathbf{2}^n - n))$ (*cf.* lines 7-20 in Algorithm 3).

Hence, the total cost is in $O(m \times n + n + (m \times n) \times \mathbf{2}^n + n \times \mathbf{2}^n + \mathbf{2}^n - n + n^2 \times \mathbf{2}^n - n^3) = O((m \times n + n^2) \times \mathbf{2}^n)$.

Thus, the complexity in the worst case of the DSFS_Miner algorithm is bounded by $O((m \times n + n^2) \times \mathbf{2}^n)$. $\diamondsuit$

4.7 Related Work and Discussion

In this part, we will mainly concentrate on the concept of *clone items* [Gély *et al.*, 2005, Medina *et al.*, 2006] since it is closely related to our work. Clone items can be roughly considered as a *restriction* of our approach to itemsets of size one (*i.e.*, items). Indeed, the authors only concentrated on items playing symmetric roles within implications premises of the Guigues-Duquenne basis [Guigues and Duquenne, 1986]. This can be considered as equivalent to our approach for γ-equivalence classes having two or more *items* as MGs (like the couple (A, B) and the couple (C, D) of Table 4.1 (*cf.* page 54)). The authors [Gély *et al.*, 2005, Medina *et al.*, 2006] show that, for a couple like (A, B), items A and B present symmetries which can be seen as redundant information since for *all* implications containing A in the premise there exists the same implications where A is replaced by B [Medina *et al.*, 2006]. Thus, they propose to ignore *all* implications containing B but not A without loss of information [Medina *et al.*, 2006]. This reduction process was applied to the Guigues-Duquenne basis [Guigues and Duquenne, 1986]. This basis presents implications between pseudo-closed itemsets and their associated closed itemsets. Note that clone items when applied to pseudo-closed itemsets are called *P-clone items* [Gély *et al.*, 2005].

4.8 Experimental Results

In these experiments, we compare the cardinality of $\mathcal{FDSFS}$ to that of the succinct frequent MGs as defined by Dong *et al.* (denoted $\mathcal{FMG}_{\texttt{suc}}$) and to that of the whole set of frequent MGs (denoted $\mathcal{FMG}$). For the sake of clarity, we give the cardinality of $\mathcal{FDSFS}$ as a sum of those of its components, *i.e.*, the representative frequent MGs (denoted $\mathcal{FMG}_{\texttt{rep}}$), the canonical frequent MGs (denoted $\mathcal{FMG}_{\texttt{can}}$) and the canonical frequent non-MG elements. The latter set represents the difference between $\mathcal{FDSFS}$ and $\mathcal{FMG}_{\texttt{suc}}$. It will hence be denoted $\mathcal{DIFF}_{\mathcal{K}}$. Since a representative frequent MG is *unique* in its γ-equivalence class, the cardinality of the set $\mathcal{FCI}$ of frequent closed itemsets (CIs) is equal to that of $\mathcal{FMG}_{\texttt{rep}}$. It will hence not be given. Note that this cardinality is insensitive *w.r.t.* the total order relation $\preceq$.

Hereafter, we will give some representative results obtained from the Pumsb, Mushroom, Chess, Connect, and T40I10D100K datasets (*cf.* Appendix A for their detailed description). The ascending support order is chosen as an example of a total order relation $\preceq$. Obtained results are summarized in Table 4.6, and graphically sketched in Figure 4.1.

For the Pumsb and Mushroom datasets, we notice an important lossless reduction reaching a peak of **2.70** and **1.75** times, respectively, when comparing the number of frequent MGs to that of frequent DSFSs (*cf.* the seventh column in Table 4.6). Indeed, a large part of the frequent MGs proves to be *redundant*. It is important to mention that this ratio increases proportionally to the decrease of the *minsupp* value. This can be explained by the fact that once *minsupp* is lowered, γ-equivalence classes become larger which augments the number of the associated MGs and, hence, redundant ones. For the Pumsb dataset, the number of canonical elements is too small. Hence, we can assume that there is an average of **1** frequent DSFS per γ-equivalence class. For the Mushroom dataset, this number is larger than that of the first dataset. Nevertheless, *w.r.t.* the number of frequent CIs, that of canonical elements is still very low which makes possible to get, in average, only **1.17** frequent DSFS per γ-equivalence

minsupp (%)	$\lvert\mathcal{FMG}\rvert$	$\lvert\mathcal{FMG}_{\texttt{rep}}\rvert$	$\lvert\mathcal{FMG}_{\texttt{can}}\rvert$	$\lvert\mathcal{DIFF}\rvert$	$\lvert\mathcal{FDSFS}\rvert$	$\frac{\lvert\mathcal{FMG}\rvert}{\lvert\mathcal{FDSFS}\rvert}$	$\frac{\lvert\mathcal{FDSFS}\rvert}{\lvert\mathcal{FMG}_{\texttt{rep}}\rvert}$
PUMSB							
90	2, 032	1, 467	3	**0**	1, 470	**1.38**	**1.00**
85	13, 795	8, 514	5	**7**	8, 526	**1.62**	**1.00**
80	67, 860	33, 308	5	**8**	33, 321	**2.04**	**1.00**
75	248, 406	101, 083	5	**8**	101, 096	**2.48**	**1.00**
70	658, 565	241, 259	6	**11**	241, 276	**2.70**	**1.00**
MUSHROOM							
10	7, 631	4, 897	471	**545**	5, 913	**1.29**	**1.21**
5	21, 160	12, 854	1, 207	**1, 251**	15, 312	**1.38**	**1.19**
3	37, 973	22, 230	1, 943	**1, 911**	26, 084	**1.46**	**1.17**
2	57, 728	31, 767	2, 644	**2, 479**	36, 890	**1.56**	**1.16**
1	103, 517	51, 672	3, 818	**3, 576**	59, 066	**1.75**	**1.14**
CHESS							
90	504	504	0	**2**	506	**1.00**	**1.00**
80	5, 114	5, 114	0	**6**	5, 120	**1.00**	**1.00**
70	23, 992	23, 992	0	**16**	24, 008	**1.00**	**1.00**
60	98, 804	98, 778	1	**28**	98, 807	**1.00**	**1.00**
50	372, 604	369, 451	2	**63**	369, 516	**1.01**	**1.00**
CONNECT							
90	3, 487	3, 487	0	**22**	3, 509	**0.99**	**1.01**
80	15, 112	15, 112	0	**43**	15, 155	**1.00**	**1.00**
70	35, 881	35, 881	0	**54**	35, 935	**1.00**	**1.00**
60	68, 350	68, 350	0	**71**	68, 421	**1.00**	**1.00**
50	130, 112	130, 112	0	**80**	130, 192	**1.00**	**1.00**
T40I10D100K							
5.00	317	317	0	**0**	317	**1.00**	**1.00**
2.50	1, 222	1, 222	0	**0**	1, 222	**1.00**	**1.00**
2.00	2, 294	2, 294	0	**0**	2, 294	**1.00**	**1.00**
1.50	6, 540	6, 540	0	**0**	6, 540	**1.00**	**1.00**
1.00	65, 237	65, 237	0	**0**	65, 237	**1.00**	**1.00**

Table 4.6: Size of the different sets for benchmark contexts.

class. Noteworthily, the number of canonical elements which are MGs is nearly equal to that of non-MG DSFSs (*i.e.*, the canonical elements which are not MGs). It is also interesting to note that the ratio between the cardinality of the set $\mathcal{FDSFS}$ and that of frequent CIs (or equivalently $\mathcal{FMG}_{\texttt{rep}}$) decreases proportionally to the decrease of *minsupp* values (*cf.* the last column in Table 4.6). The efficiency of our approach hence increases once the *minsupp* value decreases.

It is important to mention that for the PUMSB context, the redundancy is mainly caused by the fact that there are some couples of items having the same closure (like `A` and `B` of Table 4.1, page 54).

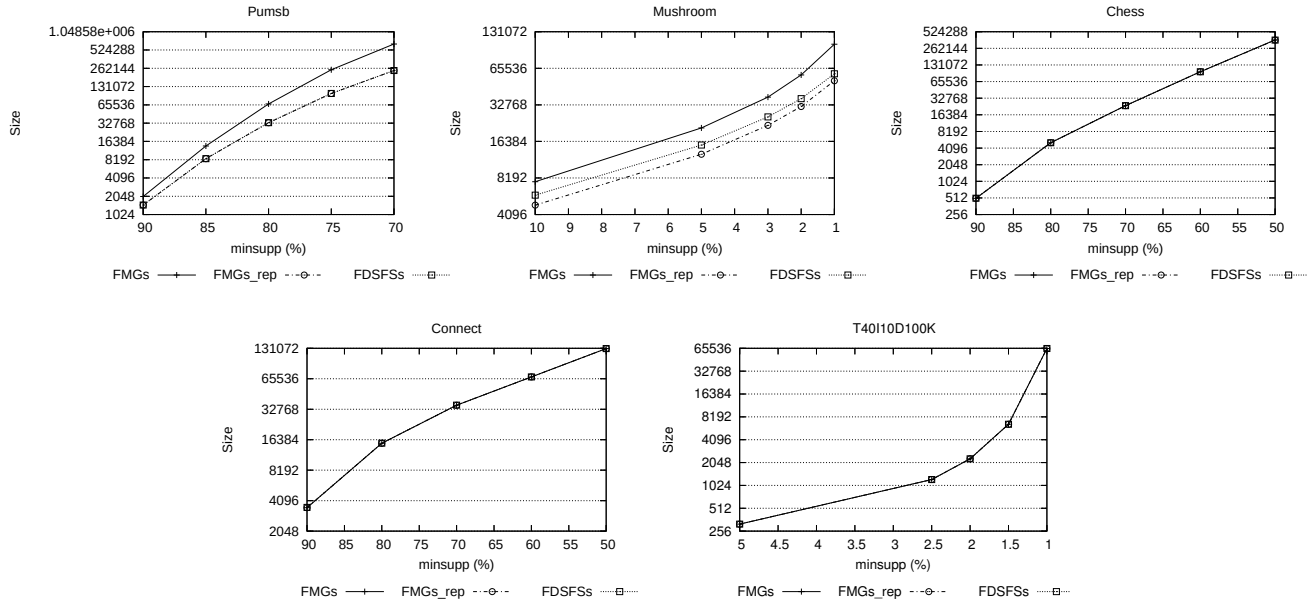


Figure 4.1: Size of the sets $\mathcal{FMG}$, $\mathcal{FMG}_{\texttt{rep}}$, and $\mathcal{FDSFS}$ for benchmark contexts.

Hence, using only an item, instead of both items forming each couple, was sufficient to eliminate all the redundancy. This is not the same for the MUSHROOM context where detecting such couples is not sufficient to completely remove the redundancy. This clearly proves the advantage of our approach as a generalization of redundancy removal within MGs independently from their sizes.

The case of the CHESS dataset is also very interesting. At a glance, for *minsupp* values greater than **60**%, statistics are far from indicating that CHESS is a dense dataset. Indeed, each γ-equivalence class only contains a unique frequent MG, *i.e.*, the representative one. Nevertheless, the latter is in general different from its closure which leads to the existence of some canonical elements in $\mathcal{FDSFS}$. These elements are necessarily non-MGs (*i.e.*, belong to $\mathcal{DIFF}$). Hence, for these values of *minsupp*, the size of $\mathcal{FDSFS}$ is slightly greater than that of $\mathcal{FMG}$. For *minsupp* = **50**%, the size of $\mathcal{FDSFS}$ becomes lower than that of $\mathcal{FMG}$ since, in this case, there are redundant frequent MGs whose number (equal to ($|\mathcal{FMG}|$ - $|\mathcal{FMG}_{\texttt{rep}}|$ - $|\mathcal{FMG}_{\texttt{can}}|$), *i.e.*, **3, 531**) is by far greater than the cardinality of $\mathcal{DIFF}$ (equal to **63**). It is important to note that we got the same behavior for the CONNECT dataset. For this latter and although it is also known to be a "dense" one, each frequent CI extracted from this dataset has only a unique frequent MG, and hence there are no *redundant* ones.

For the sparse dataset T40I10D100K, each itemset is equal to its closure. Hence, the set $\mathcal{FDSFS}$ is simply equal to the set $\mathcal{FMG}_{\texttt{rep}}$. Indeed, T40I10D100K behaves as a "worst case" dataset [Hamrouni *et al.*, 2005a] where each γ-equivalence class is reduced to a unique element. Consequently, for this context, the respective curves representing the size of the sets $\mathcal{FMG}$, $\mathcal{FMG}_{\texttt{rep}}$ and $\mathcal{FDSFS}$ collapse (*cf.* Figure 4.1). The same behavior is also noted when dealing with the KOSARAK and RETAIL datasets. The same applies for the T10I4D100K unless we set the *minsupp* support too low. In this latter case, some redundant MGs appeared.

It is worth noting that when we adopted other total order relations (*e.g.*, descending support order, lexicographic order, etc.), the cardinalities of $\mathcal{FMG}_{\texttt{can}}$ and $\mathcal{DIFF}_{\mathcal{K}}$ are almost unchanged. In addition, when a change occurred, the difference is too tiny. On the other hand, the compactness rates obtained using the RSSMG [Hamrouni *et al.*, 2008a] are similar to those offered by the DSFSs. Obtained results prove that our approach allows to almost reach the ideal case: a *unique irreducible* MG per γ-equivalence class. Noteworthily, the reduction ratio from the number of all frequent MGs to that of succinct ones can be considered as a new measure for a finer contexts classification. Indeed, according to this new classification we investigated in [Hamrouni *et al.*, 2009a], we obtained for example that the PUMSB and CHESS do not belong to the same pool, while CHESS and CONNECT share the same one.

4.9 Conclusion

Exploring real-life contexts is a difficult task due to the large number of frequent itemsets (and hence association rules) that can be extracted, even for high *minsupp* values. The simultaneous use of both concepts – MG and CI – can help augmenting the added value of such an exploration. Nevertheless, the fact that a unique CI can be associated to many MGs augments the combinatorial redundancy within these latter. The removal of such redundancy is hence an important challenge that motivated a push towards deeper understanding of their structural properties, computational behavior, connections to other constructs, etc.

In this chapter, we studied the main properties of the succinct system of minimal generators as formerly defined by Dong *et al.* to be a representation of the MG set. Once the limitations of the current definition pointed out, we introduced a new one aiming to make of, on the one hand, the associated family an *exact* representation of the minimal generator (MG) set and, on the other hand, its size independent from the adopted total order relation. Unfortunately, the new succinct system of minimal generators causes the loss of the interesting order ideal structure. In this situation, we introduced a generation operator for MGs and a family of irreducible elements for the operator, called *directed substitution-free sets*, which jointly constitute a concise yet lossless representation of the entire MG family, preserving the order ideal structure. Our work follows an original idea from the literature that was developed to a theoretically sound construct and provided with both deeper structural results and computational means. Empirical evidences for the benefits of our approach have been obtained as well. They confirmed that the proposed redundancy removal techniques makes it possible getting, in average, almost as many CIs as irreducible MGs, thanks to the elimination of an important number of *redundant* ones.

The next chapter presents an approach relying on the different kinds of minimal generators we detailed in this chapter. The purpose is to efficiently extract a lossless subset of association rules only containing succinct and informative ones.

Chapter 5

Succinct and Informative Association Rules

5.1 Introduction

Benefiting from the mathematical framework of closure operators used in Formal Concept Analysis [Ganter and Wille, 1999], generic bases of association rules were the first milestone towards losslessly reducing redundancy within association rules. In this context, they were flagged as irreducible nuclei of association rules from which *redundant* ones can be derived without any loss of information [Bastide *et al.*, 2000a].

The study we have made in the previous chapter has shown that the MG set still contains some redundancy (*cf.* Chapter 4, page 51) through the original succinct system of minimal generators. We also proved that the redefined system (*cf.* chapter 4, page 59) allows to overcome the limitations of the original one. Indeed, it offers a perfect cover of the MG family. This motivated us to extend the proposed system to the association rule framework to dramatically eliminate redundancy even within generic bases of association rules. This mainly relies on avoiding generic rules based on redundant MGs. Indeed, the inherent absence of a unique MG associated to a given CI offers an "ideal" gap towards a tougher redundancy removal.

In the remainder, to avoid confusion, we adopt the same notations used in the previous chapter. Hence, we will denote by RSSMG the redefined succinct system of minimal generators, while OSSMG denotes the original one. The main claim of this chapter is thus to mark a novel milestone towards a trilogy: "efficiency, effectiveness, meeting the end-user's needs". For this purpose, we present an approach towards extracting a succinct and informative set of association rules for pushing further the compactness of mined knowledge beyond the limits tagged by generic bases of association rules. The retained set of rules will also be lossless in the sense that redundant ones will be derivable without information loss if desired. Our approach relies on the different kinds of MGs, detailed in the previous chapter. These kinds are *representative*, *canonical* and *redundant* MGs according to OSSMG, and *succinct* ones according to RSSMG. This will be ensured as follows:

1. We incorporate the RSSMG into the framework of generic bases to reduce as far as possible the redundancy within generic association rules. Thus, after a thorough study of the best known

generic bases, we apply the RSSMG to the basis ($\mathcal{GB}$, $\mathcal{RI}$) formerly proposed by Bastide *et al.* [Bastide *et al.*, 2000a], since it gathers many interesting properties (*cf.* page 26). We then study the obtained generic rules - once the RSSMG applied - to check whether they are extracted without loss of information. For this reason, we give a thorough formal study of the related inference mechanisms allowing to derive *all redundant* association rules starting from the retained ones.

2. We propose an original algorithm, called IMG_Extractor, [1] for efficiently mining succinct association rules based on minimal generators as a starting point. The IMG_Extractor algorithm amortizes the prohibitive cost of the precedence relation determination by avoiding the itemset closure computation "pitfall" and the subset-superset tests between the frequent CIs. Indeed, by only comparing succinct frequent MGs belonging to the OSSMG, IMG_Extractor allows a shrewd construction of a partially ordered structure called the *minimal generator lattice* ($\mathcal{MGL}$). This structure is an isomorphic structure to an Iceberg lattice in which each γ-equivalence class is reduced to the corresponding set of frequent MGs. From this structure, frequent CIs are simply derived, jointly with generic association rules based on RSSMG. These generic association rules form the succinct generic association rule bases.

To prove the soundness of the proposed approach, an extensive performance study was conducted. In this respect, practical performances of the IMG_Extractor algorithm have been compared to those of the SSMG_Miner algorithm [Dong *et al.*, 2005] which is to the best of our knowledge the unique existing algorithm allowing the OSSMG extraction. The SSMG_Miner algorithm allows the extraction of the succinct frequent MGs belonging to the OSSMG, and the list of the frequent CIs. Nevertheless, it does not bear the cost of the retrieval of the precedence relations between frequent CIs. Hence, it does not allow a straightforward extraction of generic association rules without associating it with another algorithm, contrary to IMG_Extractor. Our experiments were carried out on benchmark datasets, dense and sparse. Obtained results are very encouraging. Indeed, on the one hand, they show that our approach makes it possible to eliminate without information loss an important number of redundant generic association rules and thus, to only present succinct and informative ones to the end-users. On the other hand, although our IMG_Extractor algorithm performs the partial order construction task, it largely outperforms the SSMG_Miner algorithm.

The organization of the chapter is as follows: Section 5.2 is devoted to the presentation of the succinct generic association rules. In order to derive all redundant association rules that can be extracted from a context, an axiomatic system and a study of its main properties are also provided in Section 5.3. Section 5.4 offers a detailed description of the IMG_Extractor algorithm. In section 5.5, several experiments illustrate the utility of the proposed approach.

5.2 Succinct and Informative Generic Bases

In this section, we put the focus on the integration of the redefined succinct system of minimal generators (RSSMG) within the framework of generic bases of association rules [Pasquier, 2009]. This integration aims at further reducing the number of extracted rules, through exploiting the redundancy within MGs (*cf.* Definition 45, page 60). In this respect, it is important to mention that although the directed

[1] IMG_Extractor stands for Irreducible Minimal Generators Extractor.

substitution-free sets (DSFSs) can also be used, the choice of RSSMG is argued by the fact that it is a *perfect cover* of the MG family. This mainly ensures always obtaining rules with minimal premises and maximal conclusions. Moreover, the size of the obtained rule set is independent from the adopted total order relation $\preceq$.

In this respect, our purpose is to obtain, without information loss, a more compact set of association rules from which the remaining *redundant* ones can be faithfully generated if desired. Thus, only a small set of rules needs to be presented to the end-user, which can later selectively derive other rules of interest. Succinct MGs are well suited for such a task, since they offer the minimal possible premises. They are also the most interesting ones since correlations in each *succinct* MG cannot be predicted given correlations of its subsets and those of the other (redundant) MGs. The definition of a succinct association rule is hence as follows:

Definition 54 (Succinct association rule)
Let $\mathcal{AR}$ be the set of valid association rules that can be drawn from a context $\mathcal{K}$ for a minimum support threshold minsupp and a minimum confidence threshold minconf. Given a total order relation $\preceq$, an association rule R_1: $X_1 \Rightarrow Y_1 \in \mathcal{AR}$ is said to be succinct iff $\nexists$ R_2: $X_2 \Rightarrow Y_2 \in \mathcal{AR}$ such that $Supp(R_1) = Supp(R_2)$ and $Conf(R_1) = Conf(R_2)$ with either $X_2 \subseteq X_1$ and $Y_1 \subset Y_2$, or $X_2 \vDash X_1$.

In other words, succinct association rules are non-redundant ones according to the classic definition (*cf.* Definition 30, page 24) which ensures obtaining rules with minimal premises and maximal conclusions. In addition, they must fulfill the condition that their associated premises are composed by non-redundant MGs (or equivalently, by succinct MGs).

The basis ($\mathcal{SGB}$, $\mathcal{SRI}$) of succinct generic association rules that we introduce is then defined as follows [Hamrouni *et al.*, 2008a]:

Definition 55 (Succinct generic basis for exact association rules)
Let $\mathcal{FCI}$ be the set of frequent CI*s extracted from a context $\mathcal{K}$. For each entry f in $\mathcal{FCI}$, let* MGsuc_f *be the set of its* succinct MG*s. The succinct generic basis $\mathcal{SGB}$ for exact association rules is given by: $\mathcal{SGB}$ = {R: $g \Rightarrow (f \backslash g) \mid f \in \mathcal{FCI}$ and $g \in$* MGsuc_f *and $g \neq f$}.*

Definition 56 (Succinct transitive reduction for approximate association rules)
Let $\mathcal{FMG}\mathit{suc}$ be the set of the succinct frequent MG*s extracted from a context $\mathcal{K}$. The succinct transitive reduction $\mathcal{SRI}$ for approximate association rules is given by: $\mathcal{SRI}$ = {R: $g \Rightarrow (f \backslash g) \mid f \in \mathcal{FCI}$ and $g \in \mathcal{FMG}\mathit{suc}$ and $f \in Cov^u(f_1)$ with $f_1 = \gamma(g)$ and $Conf(R) \geq minconf$}.*

Example 33 *Consider the context $\mathcal{K}$ given by Table 2.1 (cf. page 12). For the sake of simplicity, the alphabetic order is considered as total order relation $\preceq$ among itemsets, although any other total order could be obviously used. In this case, the* OSSMG *and the* RSSMG *are the same. In this respect, Table 4.1 (cf. page 54) shows, for each* CI*, the following information: its* MG*s, its succinct* MG*s and its support. For example, the* MG *AC is succinct, since it is the smallest one w.r.t. $\preceq$ among those of ABCD. Indeed, AC $\preceq$ AD, AC $\preceq$ BC and AC $\preceq$ BD, and AD, BC and BD are transitive redundant starting from AC.*

Being given an Iceberg lattice - in which each frequent CI *is accompanied by its succinct frequent* MG*s - the derivation of these generic association rules is straightforwardly performed. Indeed, consider our*

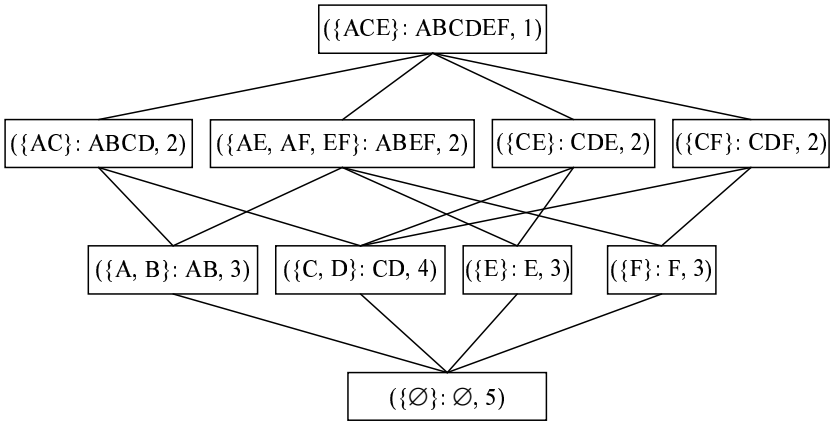


Figure 5.1: For *minsupp* = **1**, the Iceberg lattice associated to the extraction context given by Table 2.1.

running context for a minsupp value equal to **1**. *The associated Iceberg lattice is depicted by Figure 5.1. Each one of its* γ*-equivalence classes contains a frequent* CI f *accompanied by the set of its set* MG$\mathbf{suc}_f$ *of succinct frequent* MG*s, and its support, in the form* (MG$\mathbf{suc}_f$: f, $Supp(f)$).

A succinct exact generic association rule is an "intra-node" implication, with a confidence value equal to **1**, *within a* γ*-equivalence class of the Iceberg lattice. The use of the* RSSMG *allows, for example, to only extract the succinct exact generic association rule* $ACE \Rightarrow BDF$ *from the* γ*-equivalence class having* $ABCDEF$ *for frequent* CI, *instead of* **10** *if redundant frequent* MG*s were of use* (*as indicated by the last entry in Table 4.1, page 54*). *While, a succinct approximate generic association rule represents an "inter-node" implication, assorted with the confidence measure, between a* γ*-equivalence class and another belonging to its upper cover. For example, for minconf* = **0.40**, *only the association rule* $AC \overset{0.50}{\Rightarrow} BDEF$ *is extracted from both* γ*-equivalence classes having, respectively,* $ABCD$ *and* $ABCDEF$ *for frequent* CI *instead of* **4** *if redundant frequent* MG*s were of use* (*as indicated by the seventh entry in Table 4.1, page 54*). *The complete set of succinct generic association rules is reported in Table 5.1. The cardinality of* $\mathcal{SGB}$ (*resp.* $\mathcal{GB}$) *is equal to* **11** (*resp.* **27**), *while that of* $\mathcal{SRI}$ (*resp.* $\mathcal{RI}$) *is equal to* **18** (*resp.* **25**). *Hence, using the* RSSMG, *we were able to discard* **16** (*resp.* **7**) *redundant exact* (*resp. approximate*) *generic association rules, which constitutes a reduction of* **59.25**% (*resp.* **28.00**%). *Note that the total number of association rules, which can be retrieved from* $\mathcal{K}$, *is no less than* **665**. *This clearly shows the important compactness rate offered by* (*succinct*) *generic bases.*

Remark 7 *In [Deogun and Jiang, 2005], the authors baptized* `succinct association rules` *those obtained using a pruning strategy based on a model called* `maximal potentially useful` (Max PUF) *association rules. However, such reduction is done with information loss since the capability to regenerate the whole set of valid association rules is not ensured. It is important to mention that this approach and ours can be easily combined towards a more reduced set of association rules.*

The $\mathcal{SGB}$ basis		
R_1: A⇒B	R_2: B⇒A	R_3: C⇒D
R_4: D⇒C	R_5: AC⇒BD	R_6: AE⇒BF
R_7: AF⇒BE	R_8: EF⇒AB	R_9: CE⇒D
R_{10}: CF⇒D	R_{11}: ACE⇒BDF	
The $\mathcal{SRI}$ basis		
R_1: $\emptyset \overset{0.60}{\Rightarrow}$AB	R_2: $\emptyset \overset{0.80}{\Rightarrow}$CD	R_3: $\emptyset \overset{0.60}{\Rightarrow}$E
R_4: $\emptyset \overset{0.60}{\Rightarrow}$F	R_5: A$\overset{0.67}{\Rightarrow}$BCD	R_6: B$\overset{0.67}{\Rightarrow}$ACD
R_7: A$\overset{0.67}{\Rightarrow}$BEF	R_8: B$\overset{0.67}{\Rightarrow}$AEF	R_9: E$\overset{0.67}{\Rightarrow}$ABF
R_{10}: E$\overset{0.67}{\Rightarrow}$CD	R_{11}: F$\overset{0.67}{\Rightarrow}$ABE	R_{12}: F$\overset{0.67}{\Rightarrow}$CD
R_{13}: AC$\overset{0.50}{\Rightarrow}$BDEF	R_{14}: AE$\overset{0.50}{\Rightarrow}$BCDF	R_{15}: AF$\overset{0.50}{\Rightarrow}$ BCDE
R_{16}: EF$\overset{0.50}{\Rightarrow}$ ABCD	R_{17}: CE$\overset{0.50}{\Rightarrow}$ABDF	R_{18}: CF$\overset{0.50}{\Rightarrow}$ABDE

Table 5.1: The complete set of succinct generic association rules.

5.3 Derivation of Redundant Association Rules

In the following, we study the structural properties of the new generic bases introduced in the previous subsection. The study requires checking the *ideal* properties of an association rule representation (*cf.* Definition 31, page 25). Since it was shown in [Kryszkiewicz, 2002] that the basis ($\mathcal{GB}$, $\mathcal{RI}$) is extracted without loss of information, it is sufficient to show that it is possible to derive without loss of information *all* association rules that belong to the basis ($\mathcal{GB}$, $\mathcal{RI}$) starting from the basis ($\mathcal{SGB}$, $\mathcal{SRI}$). Thus, *all redundant* association rules can be straightforwardly derived from ($\mathcal{SGB}$, $\mathcal{SRI}$).

Association rules belonging to the basis ($\mathcal{SGB}$, $\mathcal{SRI}$) are implications between *succinct frequent* minimal generators (MGs) and *frequent* closed itemsets (CIs). Hence, to derive the basis ($\mathcal{GB}$, $\mathcal{RI}$), redundant frequent MGs need to be deduced since they form the premises of *redundant* generic association rules, *i.e.*, those belonging to ($\mathcal{GB}$, $\mathcal{RI}$) and discarded from ($\mathcal{SGB}$, $\mathcal{SRI}$). In order to derive *all* association rules belonging to ($\mathcal{GB}$, $\mathcal{RI}$), we propose a new axiom called the **substitution axiom**. Thus, from each association rule R: $X \Rightarrow (Y \backslash X)$ of ($\mathcal{SGB}$, $\mathcal{SRI}$) where $X \in \mathcal{FMG}_{\texttt{suc}}$ and $Y \in \mathcal{FCI}$, we propose to derive, using the substitution axiom, the set of *redundant* generic association rules given by: $Red\_Gen\_Assoc\_Rules_{R:\ X \Rightarrow (Y \backslash X)} = \{R'\colon Z \Rightarrow (Y \backslash Z) \mid Z \in \mathcal{FMG}_{\texttt{red}}\ s.t.\ X \vDash Z\}$, where $\mathcal{FMG}_{\texttt{red}}$ denotes the set of redundant frequent minimal generators associated to a context $\mathcal{K}$.

The **substitution axiom** proceeds according to the following steps:

Step 1: The set $\mathcal{GB}$ (*resp.* $\mathcal{RI}$) is firstly initialized to $\mathcal{SGB}$ (*resp.* $\mathcal{SRI}$).

Step 2: Association rules belonging to ($\mathcal{GB}$, $\mathcal{RI}$) are processed in an ascending order of their respective sizes. [2] Thus, for an association rule R: $X \Rightarrow (Y \backslash X) \in (\mathcal{GB}, \mathcal{RI})$ where $X \in \mathcal{FMG}_{\texttt{suc}}$ and $Y \in$

[2] The size of an association rule $X \Rightarrow Y$ is equal to the cardinality of $X \cup Y$.

$\mathcal{FCI}$, the set of *redundant* generic association rules associated to each association rule R_1: $X_1 \Rightarrow (Y_1 \backslash X_1)$, such that $X_1 \subset X$ and $Y_1 \subset Y$, were already derived.

Step 2.1: For each association rule R: $X \Rightarrow (Y \backslash X) \in \mathcal{GB}$, derive the set of *redundant* generic association rules $Red\_Gen\_Assoc\_Rules_{R:\ X \Rightarrow (Y \backslash X)} = \{R'$: $Z \Rightarrow (Y \backslash Z) \mid Z$ is the result of the substitution of a subset of X, say V, by T such that $\{R_1$: $V \Rightarrow (I \backslash V)$, R_2: $T \Rightarrow (I \backslash T)\} \in \mathcal{GB}$ where $I \in \mathcal{FCI}$ and $\nexists\ Z_1 \subseteq Z$ such that $Z_1 \Rightarrow (Y \backslash Z_1) \in \mathcal{GB}\}$.

Step 2.2: For each association rule R: $X \Rightarrow (Y \backslash X) \in \mathcal{RI}$, derive the set of *redundant* generic association rules $Red\_Gen\_Assoc\_Rules_{R:\ X \Rightarrow (Y \backslash X)} = \{R'$: $Z \Rightarrow (Y \backslash Z) \mid Z$ is the result of the substitution of a subset of X, say V, by T such that $\{R_1$: $V \Rightarrow (I \backslash V)$, R_2: $T \Rightarrow (I \backslash T)\} \in \mathcal{GB}$ where $I \in \mathcal{FCI}$ and $\nexists\ Z_1 \subseteq Z$ such that $Z_1 \Rightarrow (Y \backslash Z_1) \in \mathcal{RI}\}$. ♦

Note that comparing Z to Z_1 ensures discarding the case where a substitution leads to an already existing association rule or to a one having a *non-minimal* generator as a premise.

Example 34 *From the association rule R:* $AC \Rightarrow BD$ *belonging to* $\mathcal{SGB}$*, we will show how to derive association rules belonging to* $\mathcal{GB}$ *which are redundant w.r.t. R. Before that the rule R is processed, all association rules whose respective sizes are lower than that of R (i.e., lower than* **4***) were handled and redundant association rules were derived from such association rules. Among the handled association rules, we find those having for premises the* **1***-subsets of* AC*, i.e.,* $A \Rightarrow B$ *and* $C \Rightarrow D$*. To derive the redundant generic association rules associated to R, the first* **1***-subset of* AC*, i.e.,* A*, is replaced by the frequent* MG *having its closure, i.e.,* B*. Then, we augment* $\mathcal{GB}$ *by the following association rule:* $BC \Rightarrow AD$*. The second* **1***-subset of* AC*, i.e.,* C*, is then replaced by* D*. Thus, we add the association rule* $AD \Rightarrow BC$ *to* $\mathcal{GB}$*. The same process is applied to* $BC \Rightarrow AD$*. We hence obtain both association rules:* $AC \Rightarrow AD$ *and* $BD \Rightarrow AD$ *and only the latter one will be added to* $\mathcal{GB}$*. Indeed, there is already an association rule in* $\mathcal{GB}$ *s.t.* $Z_1 \Rightarrow (ABCD \backslash Z_1)$ *and* $Z_1 \subseteq AC$ *(*Z_1 *being itself equal to* AC*). From the association rule* $BD \Rightarrow AD$*, no other will be determined and the treatments come to an end.*

Now, we prove that the substitution axiom allows the basis ($\mathcal{SGB}$, $\mathcal{SRI}$) to be *lossless* and *sound*. Then, we show that this couple is also *informative*.

Proposition 27 *The basis* ($\mathcal{SGB}$, $\mathcal{SRI}$) *is* ***lossless****:* $\forall$ *R:* $X \Rightarrow (Y \backslash X) \in$ ($\mathcal{SGB}$, $\mathcal{SRI}$)*, the set* $Red\_Gen\_Assoc\_Rules_R = \{R'$*:* $Z \Rightarrow (Y \backslash Z) \mid Z \in \mathcal{FMG}_{\mathbf{red}}$ *s.t.* $X \vDash Z\}$ *of the redundant generic association rules with respect to R, is completely derived thanks to the proposed substitution axiom.*

Proof. The sorting imposed in Step **2** of the substitution axiom ensures that, before a rule R is processed, all association rules whose respective sizes are lower than that of R were handled, and *redundant* generic association rules were then derived from such association rules. Hence, all information required to derive association rules belonging to $Red\_Gen\_Assoc\_Rules_R$ are gathered thanks to the different sets $Red\_Gen\_Assoc\_Rules_{R_1}$ such that R_1:$X_1 \Rightarrow (Y_1 \backslash X_1)$, $X_1 \in \mathcal{FMG}_{\mathsf{suc}}$, $Y_1 \in \mathcal{FCI}$ and $Y_1 \subset Y$. Using these sets, *all* redundant frequent MGs, with respect to X, are straightforwardly derived. Indeed, for each subset X_1 of X, the different frequent MGs belonging to its equivalence class are already known as they are the premises of association rules belonging to the sets $Red\_Gen\_Assoc\_Rules_{R_1}$ defined above. Hence, *all* association rules belonging to ($\mathcal{GB}$, $\mathcal{RI}$) can be deduced from ($\mathcal{SGB}$, $\mathcal{SRI}$) using the

substitution axiom. Therefore, the basis ($\mathcal{SGB}$, $\mathcal{SRI}$) is lossless. $\diamondsuit$

Proposition 28 *The basis* ($\mathcal{SGB}$, $\mathcal{SRI}$) *is* ***sound****:* $\forall\, R'$*:* $Z \Rightarrow (Y \backslash Z) \in Red\_Gen\_Assoc\_Rules_{R:X \Rightarrow (Y \backslash X)}$, $Supp(R') = Supp(R)$ *and* $Conf(R') = Conf(R)$.

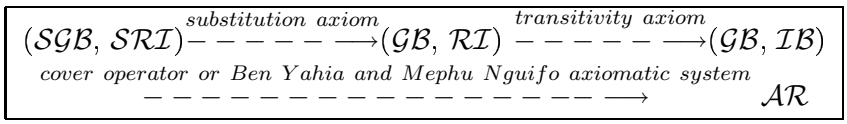
Proof. On the one hand, $Supp(R)$ is equal to $Supp(Y)$. It is also the case for $Supp(R')$. Hence, $Supp(R')$ = $Supp(R)$. On the other hand, X and Z are two frequent MGs belonging to the same γ-equivalence class. Hence, $Supp(X)$ is equal to $Supp(Z)$. Thus, $Conf(R') = \frac{Supp(Y)}{Supp(Z)} = \frac{Supp(Y)}{Supp(X)} = Conf(R)$. Therefore, the basis ($\mathcal{SGB}$, $\mathcal{SRI}$) is sound. $\diamondsuit$

The property of derivability is fulfilled by the basis ($\mathcal{SGB}$, $\mathcal{SRI}$) since it is lossless and sound. Now, we show that this couple allows the retrieval of the exact values of the support and the confidence associated to each derived association rule.

Proposition 29 *The basis* ($\mathcal{SGB}$, $\mathcal{SRI}$) *is* ***informative****: the support and the confidence of all derived association rules can exactly be retrieved from* ($\mathcal{SGB}$, $\mathcal{SRI}$).

Proof. Association rules belonging to the basis ($\mathcal{SGB}$, $\mathcal{SRI}$) are of the following form: $g \Rightarrow (f \backslash g)$ where $g \in \mathcal{FMG}_{\mathtt{suc}}$ and $f \in \mathcal{FCI}$. Therefore, we are able to reconstitute all necessary frequent CIs by concatenation of the premise and the conclusion parts of the generic association rules belonging to ($\mathcal{SGB}$, $\mathcal{SRI}$). Since the support of a frequent itemset I is equal to the support of the smallest frequent CI containing it [Pasquier *et al.*, 1999b], then the support of I and its closure can be straightforwardly derived from ($\mathcal{SGB}$, $\mathcal{SRI}$). Hence, the respective support and confidence values of *all redundant* association rules can exactly be retrieved. Thus, the basis ($\mathcal{SGB}$, $\mathcal{SRI}$) is informative. $\diamondsuit$

The substitution axiom is proved to be lossless, sound and informative; allowing to derive *all* association rules forming ($\mathcal{GB}$, $\mathcal{RI}$) as well as their *exact* support and confidence values. Since the basis ($\mathcal{GB}$, $\mathcal{RI}$) is shown to be extracted without loss of information [Kryszkiewicz, 2002], we can deduce that the basis ($\mathcal{SGB}$, $\mathcal{SRI}$) is also extracted without information loss. In order to find the complete set of *valid redundant* association rules, which can be extracted from a context $\mathcal{K}$, the axiom of transitivity proposed by Luxenburger [Luxenburger, 1991] should be applied to the $\mathcal{RI}$ basis to derive association rules forming the informative basis $\mathcal{IB}$ for the approximate association rules [Bastide *et al.*, 2000a]. Then, the cover operator proposed by Kryszkiewicz [Kryszkiewicz, 2002] or the lossless and sound axiomatic system proposed by Ben Yahia and Mephu Nguifo [Ben Yahia and Mephu Nguifo, 2004] makes it possible to derive *all valid redundant* association rules starting from the couple ($\mathcal{GB}$, $\mathcal{IB}$). The complete process allowing to derive *all valid* (*redundant*) association rules (denoted $\mathcal{AR}$), starting from the basis ($\mathcal{SGB}$, $\mathcal{SRI}$), is hence as follows:



5.4 The IMG_Extractor Algorithm

In this section, we introduce a new algorithm, called IMG_Extractor, allowing to efficiently extract the frequent CI set and, for each frequent CI, its associated succinct frequent MGs as well as the succinct generic association rules belonging to the basis ($\mathcal{SGB}$, $\mathcal{SRI}$). Hence, the knowledge offered to the end-users becomes by far more useful since an important amount of redundancy is removed. IMG_Extractor takes as input a total order relation $\preceq$, an extraction context $\mathcal{K}$, the minimum threshold of support *minsupp* and the minimum threshold of confidence *minconf*. It operates in three successive steps:

1. Determination of the MG set,
2. Construction of the minimal generator lattice ($\mathcal{MGL}$), and,
3. Extraction of the succinct generic association rule bases.

Note that the IMG_Extractor algorithm inherits from the Prince algorithm [Hamrouni *et al.*, 2005b] its first step. However, as explained in the remainder, it has for advantage *w.r.t.* this latter a reduction of the lattice construction cost thanks to the localization of redundant MGs. In addition, IMG_Extractor is able to extract generic rules as well as succinct ones while Prince is only dedicated to generic association rules.

5.4.1 Determination of the Minimal Generator Set

Following a breadth-first (or levelwise) strategy, the IMG_Extractor algorithm traverses the search space by level in a bottom-up manner. It determines the set $\mathcal{FMG}$ of the frequent MGs, extracted from the extraction context $\mathcal{K}$ and sorted by decreasing support values. IMG_Extractor also keeps track of the infrequent part of the negative border of the frequent minimal generator set, denoted $\mathcal{GB}\mathrm{d}^-$ [Kryszkiewicz, 2001]. [3] In the second step, the set $\mathcal{FMG}$ will serve as a backbone to construct the $\mathcal{MGL}$. As shown by the following proposition, the union of the sets $\mathcal{FMG}$ and $\mathcal{GB}\mathrm{d}^-$ will be used, in the second step, as an exact concise representation of frequent itemsets:

Proposition 30 *Let X be an itemset. If $\exists\ Z \in \mathcal{GB}d^-$ and $Z \subseteq X$, then X is infrequent. Otherwise, X is frequent and $Supp(X) = min\ \{Supp(g) \mid g \in \mathcal{FMG}$ and $g \subseteq X\}$ [Kryszkiewicz, 2001].*

Beyond the usual pruning based on support (*w.r.t. minsupp*), IMG_Extractor uses pruning strategies exploiting the order ideal shape of the $\mathcal{FMG}$ set. In this respect, a candidate c that does not have all its immediate subsets as frequent minimal generator will be pruned. In addition, c must not have the same support as one of its subsets, otherwise it will not be a minimal generator.

It is important to mention that in the next section, we will see the importance of redundant MGs in the construction of the partially ordered structure. Indeed, without them, some precedence relations between γ-equivalence classes can be missed. This explains why we do not prune them in this step.

The pseudo-code of the procedure, called Gen-MGs, covering this step is given by Algorithm 4. This procedure requires an access to the extraction context as long as the set of minimal generator candidates is not empty. This access is performed through the Gen-Next-MGs procedure (*cf.* Algorithm 5) whose

[3] An itemset belongs to $\mathcal{GB}\mathrm{d}^-$ if it is an infrequent MG and all its subsets are frequent MGs.

Notation	Description
$\mathcal{MGC}_k$ (*resp.* $\mathcal{FMG}_k$)	: Set of k-candidate (*resp.* k-frequent) minimal generators.
c	: Element of $\mathcal{MGC}_k$ or $\mathcal{FMG}_k$.
c.Direct_subsets	: List of (k - 1)-subsets of c.
c.Actual_Supp	: Actual support of c.
c.Estimated_Supp	: Estimated support of c, which will contain the minimum support of its direct subsets.
c.Upper_cover	: List of the immediate successors of the γ-equivalence class of c.
c.FCI	: Frequent closed itemsets of c.

Table 5.2: Notations used by the IMG_Extractor algorithm.

goal is to extract the set of frequent minimal generators of larger size and to affect infrequent candidates to the negative border. The notations used by the aforementioned procedures as well as the remaining steps of IMG_Extractor are summarized in Table 5.2.

A unique *trie* is sufficient to store the MG set. This choice is argued by the fact that this set is an order ideal. This key property allows optimizing the memory space since the path from the root to each node represents a MG and, hence, there are no useless nodes.

5.4.2 Construction of the Minimal Generator Lattice

In this step, the frequent MG set will form a $\mathcal{MGL}$, and this *without performing any access to the extraction context*. Let us begin by defining the minimal generator lattice.

Definition 57 (Minimal generator lattice)
A minimal generator lattice ($\mathcal{MGL}$) *is an isomorphic structure to an Iceberg concept lattice, i.e., the precedence relation is that amongst frequent closed itemsets. For each equivalence class of the* $\mathcal{MGL}$*, its label is reduced to the associated frequent* MG*s.*

The main idea is how to construct the partially ordered structure without computing itemset closures, *i.e.*, how guessing the precedence relations by only comparing the succinct frequent MGs? To achieve this goal, the list of the immediate successors of each γ-equivalence class will be incrementally constructed. Hereafter, by the term "immediate successor", we indicate a representative frequent MG.

The processing of $\mathcal{FMG}$ is done according to the order imposed in the first step (*i.e.*, by decreasing support values). It is worth noting that this sorting does not infringe any adopted total order relation $\preceq$. Indeed, for each γ-equivalence class having a given support, the representative frequent MG will necessarily have the smallest size since the browsing of the search space was carried out in a levelwise manner. Furthermore, if two frequent MGs or more belong to the same γ-equivalence class and have the same size, then the total order relation $\preceq$ clearly indicates which one precedes the other *w.r.t.* $\preceq$.

Before describing in detail this step, let us take an example recalling the different types of MGs according to the OSSMG.

Algorithm 4: GEN-MGS

Input: - An extraction context $\mathcal{K}$, and the threshold of support *minsupp*.

Output: - The set $\mathcal{FMG}$ of frequent minimal generators, the $\mathcal{GB}\mathrm{d}^-$ border, and the closure of the empty set.

```
1  Begin
2    MGC_1 := I ;
3    COMPUTE-SUPPORT (MGC_1) /*Computation of item supports*/ ;
4    ∅.Actual_Supp := |O|;
5    FMG_0 := {∅};
6    ForEach (c ∈ MGC_1) Do
7      If (c.Actual_Supp = |O|) Then
8        ∅.FCI := ∅.FCI ∪ c;
9      Else
10       If (c.Actual_Supp ≥ minsupp) Then
11         c.Direct_subsets := {∅};
12         FMG_1 := FMG_1 ∪ c;
13       Else
14         GBd⁻ := GBd⁻ ∪ c;
15   ForEach (k = 1 ; FMG_k ≠ ∅ ; k++) Do
16     FMG_(k+1) := GEN-NEXT-MGS(FMG_k);
17   FMG := ∪ {FMG_i | i = 0 ... k };
18 End
```

Example 35 *Let us consider once again Table 4.1 (cf. page 54), summarizing for each* CI *its (succinct)* MG*s. The* MG *AC is a representative one, since it is the smallest w.r.t.* $\preceq$ *among those of the* CI *ABCD. Indeed,* $AC \preceq AD$, $AC \preceq BC$ *and* $AC \preceq BD$. *The* MG *EF is not the representative of its* CI *ABEF, since* $AE \preceq EF$. *Nevertheless, its* **1**-*subsets (i.e., E and F) are the representative* MG*s of their respective* CI*s. Hence, EF is a canonical* MG. *Finally, the* MG *ADE is a redundant one, since at least one of its subsets is not a representative* MG (*DE, for example*).

For each frequent MG g of size k ($k \geq 1$), the treatments closely depend whether g is a succinct frequent MG or a *redundant* one (or equivalently, whether all its (k - 1)-subsets are representative frequent MGs or not). Recall that during the previous step, the links to the (k - 1)-subsets of g were stored when checking for the order ideal property. We hence distinguish the following two cases:

1. If g is a succinct frequent MG: g will be introduced into the $\mathcal{MGL}$ by only comparing it to the immediate successor lists of its (k - 1)-subsets. This is based on the *isotony* property of the closure operator γ [Davey and Priestley, 2002]. Indeed, let g_1 be one of the (k - 1)-subsets of g, $g_1 \subset g \Rightarrow \gamma(g_1) \subset \gamma(g)$. Thus, the γ-equivalence class $\mathcal{C}_g$ to which belongs g is a successor (not necessarily an immediate

Algorithm 5: Gen-Next-MGs

Input: - The set $\mathcal{FMG}_k$.

Output: - The set $\mathcal{FMG}_{(k+1)}$.

```
1  Begin
2    /* Generating candidates using the Apriori-Gen procedure [Agrawal and Srikant,
     1994] */
3    MGC_(k+1) := Apriori-Gen(FMG_k)
4    /* Testing the order ideal property of frequent minimal generators */
5    ForEach (c ∈ MGC_(k+1)) Do
6      c.Estimated_Supp := |O|; /* maximal possible support */
7      ForEach (c1 such that |c1| = k and c1 ⊂ c) Do
8        If (c1 ∉ FMG_k) Then
9          MGC_(k+1) := MGC_(k+1) \ c;
10         break;
11         End For;
12       c.Estimated_Supp := min(c.Estimated_Supp, c1.Actual_Supp);
13       c.Direct_subsets := c.Direct_subsets ∪ c1;
14   /* Computation of candidates supports and pruning infrequent ones */
15   Compute-Support (MGC_(k+1));
16   ForEach (c ∈ MGC_(k+1)) Do
17     If (c.Actual_Supp ≠ c.Estimated_Supp and c.Actual_Supp ≥ minsupp) Then
18       FMG_(k+1) := FMG_(k+1) ∪ c;
19     Else
20       If (c.Actual_Supp < minsupp) Then
21         GBd⁻ := GBd⁻ ∪ c;
22   Return FMG_(k+1)
23 End
```

one) of the γ-equivalence class $\mathcal{C}_{g_1}$ to which belongs g_1. Let us denote by L the immediate successor list of g_1. If L is still empty when g is compared to it, then g is simply added to L. Otherwise, g is compared to the elements already belonging to L using Proposition 31, by replacing the itemsets X and Y respectively by g and one of the elements of L. The processing order has the advantage of restricting comparisons to the case where $Supp(X)$ is lower than or equal to $Supp(Y)$. The following lemma will be used in the proof of Proposition 31.

Lemma 6 *Let $X, Y \subseteq \mathcal{I}$, $(X \subseteq Y \wedge Supp(X) = Supp(Y)) \Rightarrow (\gamma(X) = \gamma(Y))$ [Stumme* et al.*, 2002].*

Proposition 31 *Let $X, Y \subseteq \mathcal{I}$, $\mathcal{C}_X$ and $\mathcal{C}_Y$ be their respective γ-equivalence classes:*

1. *If $Supp(X \cup Y) = min\ \{Supp(X), Supp(Y)\}$, then $\mathcal{C}_X$ and $\mathcal{C}_Y$ are comparable:*
   (a) *X and Y belong to the same γ-equivalence class if $Supp(X) = Supp(Y)$.*
   (b) *$\mathcal{C}_X$ (resp. $\mathcal{C}_Y$) is a successor (resp. predecessor) of $\mathcal{C}_Y$ (resp. $\mathcal{C}_X$) if $Supp(X) < Supp(Y)$.*
2. *If $Supp(X \cup Y) \neq min\ \{Supp(X), Supp(Y)\}$, then $\mathcal{C}_X$ and $\mathcal{C}_Y$ are incomparable.*

Proof.

1. (a) $(X \subseteq (X \cup Y) \wedge Supp(X) = Supp(X \cup Y)) \Rightarrow (\gamma(X) = \gamma(X \cup Y))$ (according to Lemma 6) (1)
   $(Y \subseteq (X \cup Y) \wedge Supp(Y) = Supp(X \cup Y)) \Rightarrow (\gamma(Y) = \gamma(X \cup Y))$ (according to Lemma 6) (2)
   According to (1) and (2), $\gamma(X) = \gamma(Y)$ and thus X and Y belong to the same γ-equivalence class (*i.e.*, $\mathcal{C}_X$ and $\mathcal{C}_Y$ are identical).
   (b) $(X \subseteq (X \cup Y) \wedge Supp(X) = Supp(X \cup Y)) \Rightarrow (\gamma(X) = \gamma(X \cup Y))$ (according to Lemma 6) (1)
   $(Y \subseteq (X \cup Y) \wedge Supp(Y) \neq Supp(X \cup Y)) \Rightarrow (\gamma(Y) \subset \gamma(X \cup Y))$. However, according to (1), $\gamma(X) = \gamma(X \cup Y)$ and, thus, $\gamma(Y) \subset \gamma(X)$. Hence, $\mathcal{C}_X$ (*resp.* $\mathcal{C}_Y$) is a successor (*resp.* predecessor) of $\mathcal{C}_Y$ (*resp.* $\mathcal{C}_X$).
2. Suppose that $\mathcal{C}_X$ and $\mathcal{C}_Y$ are comparable. This means either $\gamma(Y) \subseteq \gamma(X)$ or $\gamma(X) \subseteq \gamma(Y)$. The treatment of both cases is similar. Let us look at the first one: $\gamma(Y) \subseteq \gamma(X) \Rightarrow Y \subseteq \gamma(Y) \subseteq \gamma(X)$. Thanks to the extensivity property of a closure operator, we have $X \subseteq \gamma(X)$, and hence $(X \cup Y) \subseteq \gamma(X)$. Thus, $X \subseteq (X \cup Y) \subseteq \gamma(X)$. By the isotony property of a closure operator, we have $\gamma(X) \subseteq \gamma(X \cup Y) \subseteq \gamma(\gamma(X))$, and thanks to the idempotency property, we have $\gamma(X) \subseteq \gamma(X \cup Y) \subseteq \gamma(X)$. Thus, $\gamma(X) = \gamma(X \cup Y)$. This result is in contradiction with the fact that $\text{Supp}(X) > \text{Supp}(X \cup Y)$. Consequently, $\mathcal{C}_X$ et $\mathcal{C}_Y$ are incomparable.

$\diamondsuit$

In our case, the itemsets X and Y are both minimal generators. Let Z be equal to $(X \cup Y)$. The computation of the support of Z is performed in a direct manner if Z belongs to $\mathcal{FMG} \cup \mathcal{GB}\text{d}^-$. Indeed, in this former case, we can directly access the support of Z being known since the first step. $\mathcal{C}_X$ and $\mathcal{C}_Y$ are then incomparable since $Supp(X) \neq Supp(Z)$ (otherwise, Z should not be a minimal generator). If Z does not belong to $\mathcal{FMG} \cup \mathcal{GB}\text{d}^-$, then Proposition 30 will be applied. It is worth noting that the support computation stops, in this latter case, as soon as we find a minimal generator W that is included in Z and having a support strictly lower than that of X (as mentioned above $Supp(X) \leq Supp(Y)$). Hence, $Supp(Z) \leq Supp(W) < Supp(X)$ and then $Supp(X) \neq Supp(Z)$. $\mathcal{C}_X$ and $\mathcal{C}_Y$ are thus incomparable.

To maintain the concept of *equivalence class* throughout the processing of the $\mathcal{FMG}$ list, IMG_EXTRACTOR uses the MANAGE-EQUIVALENCE-CLASS function. Indeed, this function is used if g is compared to the representative frequent MG of its γ-equivalence class, say $\mathcal{R}$. The MANAGE-EQUIVALENCE-CLASS function then replaces all occurrences of g by $\mathcal{R}$ in the immediate successor lists where g was added. Then,

comparisons to carry out with g will be made only with $\mathcal{R}$. Thus, for each γ-equivalence class, *only* its representative frequent MG appears in the immediate successor lists. Hence, this function allows to optimize the management of the γ-equivalence classes by dramatically reducing unnecessary comparisons.

2. If g is a redundant frequent MG: g will not be introduced into the $\mathcal{MGL}$. However, it is necessary to take care to compare the representative frequent MG of $\mathcal{C}_g$ with the immediate successor lists of *only* the *representative* (k - 1)-subsets of g. Otherwise, some links in the $\mathcal{MGL}$ can be lost. Indeed, consider the example given by Table 4.1 (*cf.* page 54) associated to the context of Table 2.1 (*cf.* page 12). If we do not consider the redundant frequent MGs of the equivalence class having ABCDEF for frequent CI, then the precedence relation between this equivalence class and that having CDF for frequent CI will be lost (*cf.* Figure 5.1, page 84). It would be then the same for some succinct generic approximate association rules (*cf.* Table 5.1, page 85). The reason is that CF, the representative frequent MG of CDF, is not a subset of ACE, ACE being the unique succinct frequent MG of ABCDEF.

Nevertheless, to optimize the treatments, we will use a second function called Locate-Equivalence-Class. Indeed, if g is a redundant frequent MG, this means that at least one of its (k - 1)-subsets is a redundant frequent MG. Let us note by g_1 this subset, by c the item appearing in g and not in g_1 (*i.e.*, $\{c\} = (g \backslash g_1)$) and by $\mathcal{R}_1$ the representative frequent MG of $\mathcal{C}_{g_1}$. A call to the Locate-Equivalence-Class function makes it possible to locate the equivalence class to which belongs g and hence, the representative frequent MG of $\mathcal{C}_g$, say $\mathcal{R}$. This function avoids comparing g to the immediate successor lists of *all* its (k - 1)-subsets. Indeed, *only* the comparison of $\mathcal{R}$ with the immediate successor lists, of the *representative* (k - 1)-subsets of g, is needed. Using Definitions 40 (*cf.* page 52) and 41 (*cf.* page 52) and Lemma 5 (*cf.* page 60), we will show that a call to the Locate-Equivalence-Class function correctly computes $\mathcal{R}$ since the latter already exists in the $\mathcal{MGL}$. In our context, with $X \cup Y$, we will indicate the ordered sequence of items contained in X or in Y.

Proposition 32 *The* Locate-Equivalence-Class *function correctly locates* $\mathcal{R}$.

Proof. Using Lemma 5, we have:

$$\gamma(g) = \gamma(g_1 \cup \{c\}) = \gamma(\gamma(g_1) \cup \gamma(\{c\}))$$

Since $\gamma(g_1) = \gamma(\mathcal{R}_1)$,

$$\gamma(\gamma(g_1) \cup \gamma(\{c\})) = \gamma(\gamma(\mathcal{R}_1) \cup \gamma(\{c\})) = \gamma(\mathcal{R}_1 \cup \{c\}) = \gamma(W) \text{ } s.t. \text{ } W = \mathcal{R}_1 \cup \{c\}.$$

Since $\gamma(g) = \gamma(W)$, g and W necessarily belong to the same equivalence class. We then distinguish the following two cases:

• If W is a frequent MG, then $W \prec g$ since $\mathcal{R}_1 \prec g_1$ (first property in Definition 40). From the definition of a *representative* MG (Definition 41), $\mathcal{R} \preceq W$. Since $\mathcal{R} \preceq W$ and $W \prec g$, then $\mathcal{R} \prec g$.

• If W is not a frequent MG, then there is a frequent MG Z *s.t.* $Z \subset W$ and $\gamma(Z) = \gamma(W)$. Since $|Z| < |W|$ and $|W| \leq |g|$, $|Z| < |g|$ and thus $Z \prec g$ (second property in Definition 40). From Definition 41, $\mathcal{R} \preceq Z$. Consequently, $\mathcal{R} \prec g$ since $\mathcal{R} \preceq Z$ and $Z \prec g$.

We conclude that in both cases, $\mathcal{R}$ was already treated and was thus correctly introduced into the $\mathcal{MGL}$.

$\diamondsuit$

It is worth noting that the performed treatments in this step allow to manage only one immediate successor list for *all* the succinct frequent MGs belonging to the same γ-equivalence class. Hence, this optimizes both runtime and memory consumption.

The pseudo-code of the second step is given by the GEN-ORDER procedure (Algorithm 6). In this algorithm, we use the denotation `Supp` to indicate the actual support of g since we will not have to distinguish any more between this latter and the estimated support of g. At the end of the execution of the GEN-ORDER procedure, g.`Upper_cover` is empty if g is not the representative frequent MG of $\mathcal{C}_g$ or if the latter is a maximal γ-equivalence class, *i.e.*, not covered by any other. Otherwise, this list will only contain representative frequent MGs.

5.4.3 Extraction of the Succinct Generic Association Rule Bases

In this step and for each γ-equivalence class $\mathcal{C}$ of the $\mathcal{MGL}$, two main tasks are performed: deriving the corresponding frequent closed itemset (CI) f and then extracting the associated succinct generic association rules. These tasks are far from being time-consuming as sketched out in the remainder. Indeed, IMG_EXTRACTOR efficiently derives frequent CIs using Proposition 33. The proof of this latter proposition relies on the links between minimal generators (MGs) and the important concepts in lattice construction of *face* and *minimal blocker*, studied in [Pfaltz and Taylor, 2002]. These concepts are presented through the following definitions and Theorem 10 [Pfaltz and Taylor, 2002].

Definition 58 (FACE)
Let f and $f_1 \in \mathcal{FCI}$. If f immediately covers f_1 in the Iceberg lattice, then the face of f compared to f_1 corresponds to $f \backslash f_1$.

Definition 59 (MINIMAL BLOCKER)
Let $G = \{G_1, \ldots, G_n\}$ be a family of n sets. A blocker B of the family G is a set such that its intersection with all the sets $G_i \in G$ is not empty. B is said to be minimal if there is no blocker B_1 of G included in B.

Theorem 10 *Let $f \in \mathcal{FCI}$ and MG_f be the set of its frequent* MG*s. If $f_1 \in \mathcal{FCI}$ such that f immediately covers f_1 in the Iceberg lattice, then the face of f compared to f_1 is a minimal blocker of MG_f.*

Proposition 33 *Let f, $f_1 \in \mathcal{FCI}$ such that f immediately covers f_1 in the Iceberg lattice. Let MG_f be the set of the frequent* MG*s of f. The closure f can be obtained as follows: $f = \cup \{g \mid g \in \mathrm{MG}_f\} \cup f_1$.*

Proof. Let B be the set resulting from the union of the frequent MGs of f (*i.e.*, $B = \cup\{g | g \in \mathrm{MG}_f\}$). Since B is a blocker of MG_f, then the face of f compared to f_1 (*i.e.*, $f \setminus f_1$), which is a minimal blocker of MG_f according to Theorem 10, is included in B. Thus, it is sufficient to compute the union of f_1 with B to derive the frequent CI f. $\diamondsuit$

It is worth noting that the derivation of f is performed in a straightforward manner since the γ-equivalence class, to which belongs a given frequent MG g, is necessarily located in the second step even

Algorithm 6: Gen-Order

Input: - The set $\mathcal{FMG}$ of frequent minimal generators, and the total order relation $\preceq$.

Output: - The partially ordered structure $\mathcal{MGL}$.

```
1  Begin
2    ForEach (g ∈ FMG) Do
3      S1 := ∅;
4      S2 := ∅;
5      ForEach (g1 ∈ g.Direct_subsets) Do
6        If (g1 is a representative frequent MG) Then
7          S1 := S1 ∪ {g1};
8        Else
9          S2 := S2 ∪ {g1};
10     If (S2 = ∅) Then
11       ForEach (g1 ∈ S1) Do
12         ForEach (g2 ∈ g1.Upper_cover) Do
13           If (g.Supp = g2.Supp = Supp(g ∪ g2)) Then
14             Manage-Equivalence-Class(g, g2);
15           Else If (g.Supp < g2.Supp and g.Supp = Supp(g ∪ g2)) Then
16             g will be compared with g2.Upper_cover;
17             For the remaining elements of g1.Upper_cover, g will only be compared
               with each MG g3 s.t. g3.Supp > g.Supp;
18         If (∀ g2 ∈ g1.Upper_cover, Cg and Cg2 are incomparable) Then
19           g1.Upper_cover := g1.Upper_cover ∪ {g};
20     Else
21       R := Locate-Equivalence-Class(g, g1);
22       ForEach (g1 ∈ S1) Do
23         ForEach (g2 ∈ g1.Upper_cover and g2.Supp > R.Supp) Do
24           If (R.Supp = Supp(R ∪ g2)) Then
25             R will be compared with g2.Upper_cover;
26         If (∀ g2 ∈ g1.Upper_cover, CR and Cg2 are incomparable) Then
27           g1.Upper_cover := g1.Upper_cover ∪ {R};
28 End
```

if g is a redundant frequent MG (*cf.* Algorithm 6, line 21). Hence, a simple bottom-up sweeping of the $\mathcal{MGL}$ is sufficient to completely derive f and to extract the associated succinct generic association rules. Another advantage of such a way of the lattice traversal is that only the storage of the upper cover of each equivalence class is needed. Indeed, the storage of the lower cover is redundant and useless. Note that for each γ-equivalence class, the derivation of the associated succinct frequent MGs belonging to

RSSMG is performed thanks to the σ-EQUIVALENCE_CLASSES_MINER function we proposed in the previous chapter (*cf.* Algorithm 1, page 62).

The traversal of the $\mathcal{MGL}$ is carried out from the bottom of the lattice until reaching maximal γ-equivalence classes. The closure of the empty set was computed at the beginning of the first step by simply collecting, if there are, the items belonging to all objects of the context. If it is not empty, the succinct generic exact association rule, having the empty set for premise and its closure for conclusion, is extracted. Then, IMG_EXTRACTOR extracts the succinct generic approximate association rules between $\mathcal{C}_\emptyset$ and the γ-equivalence classes belonging to the upper cover of $\mathcal{C}_\emptyset$. Their respective closures are derived thanks to Proposition 33, using the associated frequent MGs and the closure of the empty set. These γ-equivalence classes are then stored which makes it possible to apply the same process to them. By the same manner, IMG_EXTRACTOR treats higher levels of the $\mathcal{MGL}$ until reaching the maximal γ-equivalence class(es).

The pseudo-code of this step is given by the GEN-SGRB procedure (Algorithm 7). For each frequent MG g, the FCI attribute allows storing the frequent CI corresponding to $\mathcal{C}_g$ if g is its *representative*. In the GEN-SGRB procedure, L_1 indicates the list of γ-equivalence classes from which are extracted the valid succinct generic association rules. By L_2, we note the list of γ-equivalence classes which immediately cover those forming L_1. A test is carried out to check whether a γ-equivalence class does not belong to L_2. This test consists in checking if the corresponding frequent CIs were already computed (*cf.* line 8 in Algorithm 7).

Example 36 *Consider the context $\mathcal{K}$ given by Table 2.1 (cf. page 12). Let the minsupp and minconf values be, respectively, equal to* **1** *and* **0.4**. *We consider the lexicographic order among items as a total order relation $\preceq$. The first step allows the determination of the closure of the empty set, equal to the empty set, the sorted set $\mathcal{FMG}_\mathcal{K}$, given by Table 4.1 (cf. page 54), and the negative border of* MG*s $\mathcal{GBd}^-$, equal to the empty set. During the second step,* IMG_EXTRACTOR *processes the elements of $\mathcal{FMG}$ to construct the $\mathcal{MGL}$. Consider for example the* **2***-frequent* MG *AE. The latter is a succinct frequent* MG *since all its* **1***-subsets are the representative frequent* MG*s of their respective frequent* CI*s (as depicted by Table 4.1). Hence, AE will be compared to the immediate successor lists of both A and E (cf. Algorithm 6, lines 11-19). Since $\mathcal{C}_A$ has $\mathcal{C}_{AC}$ as an immediate successor, AE is then compared to AC: $Supp(AE \cup AC) = Supp(ACE) \neq min\ \{AE.Supp, AC.Supp\}$. Hence, $\mathcal{C}_{AE}$ and $\mathcal{C}_{AC}$ are incomparable. AE is then added to the immediate successor list of A. The immediate successor list of the second subset E is still empty and AE is simply added to it. If we consider the case of the* **3***-frequent* MG *ACF, the latter is a redundant frequent* MG *since AF is not the representative frequent* MG *of its γ-equivalence class. Instead of performing unnecessary treatments by comparing ACF to the immediate successor lists of all its* **2***-subsets, the representative frequent* MG *of $\mathcal{C}_{ACF}$ will be found (cf. Algorithm 6, line 21). Since AE is the representative frequent* MG *of $\mathcal{C}_{AF}$, the representative frequent* MG *of $\mathcal{C}_{ACF}$ is that of $\mathcal{C}_{(AE \cup (ACF \setminus AF))}$ (i.e., $\mathcal{C}_{ACE}$) and is equal to ACE. The latter will only be compared to the immediate successor lists of the representative* **2***-subsets of ACF, i.e., AC and CF (cf. Algorithm 6, lines 22-27). ACE was already compared to the immediate successor list of AC since the latter is one of its direct subsets. Hence, this new comparison is redundant and is thus not performed. ACE will then be compared to the immediate successor list of CF. Since the latter is still empty, ACE is simply added to it. At the end of this second step, the $\mathcal{MGL}$ is built. During the third step, an ascending sweeping is carried out from $\mathcal{C}_\emptyset$. Since $\gamma(\emptyset) = \emptyset$, no*

Algorithm 7: Gen-SGRB

Input: - The structure $\mathcal{MGL}$, and the minimum threshold of confidence *minconf*.

Output: - The frequent CI associated to each equivalence class, the succinct generic basis of exact rules (denoted $\mathcal{SGB}$) and the transitive reduction of approximate rules (denoted $\mathcal{SRI}$).

```
1  Begin
2    SGB := ∅; SRI := ∅; L1 := {∅}; L2 := ∅;
3    While (L1 ≠ ∅) Do
4      ForEach (g ∈ L1) Do
5        If (g.FCI ≠ g) Then
6          SGB := SGB ∪ {(t ⇒ (g.FCI \ t), g.Supp) | t ∈ FMGsuc and t ∈ Cg};
7        ForEach (g1 ∈ g.Upper_cover) Do
8          If (g1.FCI = ∅) Then
9            g1.FCI := ∪ {t ∈ FMG | t ∈ Cg1 } ∪ g.FCI;
10           L2 := L2 ∪ {g1};
11         If (g1.Supp / g.Supp ≥ minconf) Then
12           SRI := SRI ∪ {(t ⇒ (g1.FCI \ t), g1.Supp, g1.Supp / g.Supp) | t ∈ FMGsuc
             and t ∈ Cg};
13     L1 := L2;
14     L2 := ∅;
15 End
```

exact association rule is extracted from $C_\emptyset$. $\emptyset$.Upper_cover = {A, C, E, F}. The frequent CI *associated to C_A is then found and is equal to AB. The succinct generic approximate association rule $\emptyset \Rightarrow AB$, of a support value equal to* **3** *and a confidence value equal to* **0.6**, *is then extracted. It is the same for C_C, C_E and C_F. Using the same process and from the upper cover of $C_\emptyset$,* IMG_Extractor *performs a bottom-up traversal of the $\mathcal{MGL}$ until reaching the non covered γ-equivalence class having ABCDEF for frequent* CI. *The complete set of succinct generic association rules, sketched by Table 5.1, is thus straightforwardly extracted.*

5.4.4 Correctness and Complexity

The following theorem proves the soundness and the correctness of the IMG_Extractor algorithm.

Theorem 11 *The* IMG_Extractor *algorithm is sound and correct. It extracts all succinct frequent* MG*s and derives all frequent* CI*s and all valid succinct generic association rules.*

Proof. During the first step, a candidate MG c is pruned only if its estimated support is equal to its actual support or if it does not verify the order ideal property of MGs. Otherwise, c is a MG and by comparing its actual support to *minsupp*, the IMG_Extractor algorithm adds it to the frequent MG set $\mathcal{FMG}$ or to the negative border of MGs $\mathcal{GB}\mathrm{d}^-$. Thus, at the end of the first step of IMG_Extractor, all frequent MGs are extracted in addition to the negative border of MGs.

During the second step, IMG_Extractor takes care to introduce all succinct frequent MGs into the minimal generator lattice ($\mathcal{MGL}$). Indeed, based on Proposition 14 (*cf.* page 55), IMG_Extractor checks whether a frequent MG g is a *succinct* one or not. After that, treatments depend on the nature of g: *succinct* or *redundant*. In the former case, g will simply be compared to the immediate successor list of all its (k - 1)-subsets. This is based on the isotony property of the closure operator γ. In this case, both properties sketched by Proposition 31 are treated in Algorithm 6. The Manage-Equivalence-Class function allows to manage each γ-equivalence class once g is compared to the representative frequent MG $\mathcal{R}$ of its γ-equivalence class. Then, comparisons will be done using $\mathcal{R}$ instead of g without affecting the correctness of the algorithm. Indeed, $\mathcal{R}$ and g share the same properties since they belong to the same γ-equivalence class. In the latter case – g is *redundant* – the Locate-Equivalence-Class function allows to find the representative frequent MG $\mathcal{R}$ of $\mathcal{C}_g$ as shown by Proposition 32. Then, $\mathcal{R}$ will only be compared to the immediate successor having supports greater than the support of $\mathcal{R}$. Indeed, $\mathcal{R}$ is the unique succinct frequent MG belonging to $\mathcal{C}_g$ that appears in the different immediate successor lists. Hence, it is useless to compare it to those having the same support since none of them will be added to the γ-equivalence class of $\mathcal{R}$. At the end of this step, the minimal generator lattice is completely built.

During the third step, all γ-equivalence classes are taken in consideration when deriving frequent CIs as well as valid succinct generic association rules. Indeed, each equivalence class $\mathcal{C}$, except $\mathcal{C}_\emptyset$, has *at least one* immediate predecessor. Hence, the *representative* of $\mathcal{C}$ belongs at least to one immediate successor list of another γ-equivalence class, say $\mathcal{C}_1$. When treating $\mathcal{C}_1$, the frequent CI is completely derived using Proposition 33. $\mathcal{C}$ is also added to the list of γ-equivalence classes from which valid succinct generic association rules will be derived in the next iteration. Thus, at the end of this step, all frequent CIs and all valid succinct generic association rules are entirely derived. $\diamondsuit$

Proposition 34 evaluates the complexity of the IMG_Extractor algorithm.

Proposition 34 *In the worst case, the theoretical complexity of* IMG_Extractor *is in* $O((n^3+m)\times 2^n)$, *where* $n = |\mathcal{I}|$ *and* $m = |\mathcal{O}|$.

Proof. The worst case is obtained whenever any set of items appears at least once in the context, and each extracted itemset is a frequent closed minimal generator. Thus, the frequent itemset lattice strictly overlaps both the Iceberg lattice and the minimal generator lattice. The number of frequent closed minimal generators is hence equal to 2^n. Each frequent MG is equal to its closure and is hence the representative frequent MG of its γ-equivalence class. We assume that each object contains n distinct items.

During the first step (*cf.* Algorithm 4, page 90, and Algorithm 5, page 91), IMG_Extractor performs two main tasks. The first task consists in the candidate support computations and is bounded by $O(m \times 2^n)$. The second task consists in pruning non-MG candidates. This is done in $O(n^2 \times 2^n)$. The cost of the first step is then bounded by $O((n^2 + m) \times 2^n)$.

During the second step (*cf.* Algorithm 6, page 95), and for each frequent MG g of size k ($\leq n$), IMG_Extractor verifies whether g is a succinct frequent MG or not. This is carried out in $O(n)$. Since in the worst case g is a *representative* and hence a *succinct* generator, IMG_Extractor performs $(k \times (n-k))$ comparisons which will be bounded by n^2. Indeed, g has k immediate subsets (*i.e.*, those of size (k - 1)). Each (k - 1)-subset g_1 has, in the worst case, $(n-k)$ immediate successors when comparing g with g_1.`Upper_cover`. Each comparison is performed by making the union of g with an element of g_1.`Upper_cover`. The union cost is $O(n)$. The search of the support of the resulting itemset costs $O(n)$ since it is a frequent MG. The cost of the second step is then bounded by $O((n + ((n+n) \times n^2)) \times 2^n)$, *i.e.*, $O(n^3 \times 2^n)$.

During the third step (*cf.* Algorithm 7, page 97), and for each γ-equivalence class $\mathcal{C}$, IMG_Extractor performs two complementary tasks. The first consists in deriving the corresponding frequent CI. This is carried out by performing the union of the frequent MG of $\mathcal{C}$ and the frequent CI associated to an equivalence class which is immediate predecessor of $\mathcal{C}$. The first task then costs $O(n)$ in the worst case. The second task consists in deriving valid informative association rules. As each frequent MG is also closed, there is no exact succinct generic association rule. However, by fixing *minconf* to **0**, there are k approximate succinct generic association rules, for a γ-equivalence class whose frequent closed minimal generator is of size k. To derive each approximate succinct generic association rule, IMG_Extractor computes the difference between the frequent CI of $\mathcal{C}$ and the corresponding premise. This is performed in $O(n)$. The second task then costs $O(k \times n)$ (k will be bounded by n). Hence, the cost of the third step is bounded by $O((n + n^2) \times 2^n)$, *i.e.*, $O(n^2 \times 2^n)$.

Thus, in the worst case, the theoretical complexity of IMG_Extractor is bounded by the sum of the costs of its three steps which is in $O((n^3 + m) \times 2^n)$. $\diamondsuit$

It is important to mention that although IMG_Extractor builds the Iceberg lattice, its theoretical complexity remains of the same order of magnitude as that of algorithms only dedicated to the extraction of frequent CIs [Kuznetsov and Obiedkov, 2002, Pasquier, 2000].

5.5 Experimental Results

In this section, we shed light on the compactness rate obtained through the proposed generic bases of association rules. After that, we lead a thorough analysis of the performances of the IMG_Extractor algorithm compared to those of the SSMG_Miner algorithm [Dong *et al.*, 2005]. Note that SSMG_Miner does not allow a straightforward extraction of generic association rules without associating it with another algorithm. The source code of the SSMG_Miner algorithm, kindly provided by its authors, is implemented using the ascending support order as a total order relation $\preceq$. Hence, in order to allow a fair comparison, the IMG_Extractor algorithm also uses this order. [4]

All experiments were carried out on a PC equipped with a 2.4GHz Pentium IV and 512MB of main memory (with 2GB of swap space) and running the GNU/Linux distribution S.u.S.E 9.0. To rate the different behaviors of the considered algorithms, we ran experiments on benchmark datasets (*cf.* Appendix A for their detailed description). Hereafter, we use a logarithmically scaled ordinate axis in all

[4] The source code of the IMG_Extractor algorithm is available at: *http://www.cck.rnu.tn/sbenyahia/software_release.htm.*

figures.

The next subsection describes the compactness rate brought by our approach. Then, we compare through a detailed analysis the performances of IMG_EXTRACTOR to those of SSMG_MINER.

5.5.1 Extracted Rule Compactness

We compared both bases ($\mathcal{SGB}$, $\mathcal{SRI}$) and ($\mathcal{GB}$, $\mathcal{RI}$) using the couple size as an evaluation criterion, for a fixed *minsupp* value. Representative results we obtained are graphically sketched by Figure 5.2. The associated experiments were carried out for the PUMSB (*resp.* CONNECT, MUSHROOM and T40I10D100K) context for a *minsupp* value equal to **70**% (*resp.* **50**%, **0.01**% and **1**%). Note that the choice of each *minsupp* value is performed according to the context density. For each context, the *minconf* value varies between the aforementioned *minsupp* value and **100**%.

Figure 5.2 points out that removing redundancy within the frequent MG set offers an interesting lossless reduction of the number of the extracted generic association rules. Indeed, our approach allows to remove in average **63.03**% (*resp.* **49.46**%) of the *redundant* generic association rules extracted from the PUMSB (*resp.* MUSHROOM) context. The maximum rate of redundancy reaches **68.11**% (*resp.* **53.84**%) for the PUMSB (*resp.* MUSHROOM) context, for a *minconf* value equal to **100**% (*resp.* **20**%).

For the CONNECT and T40I10D100K contexts, the respective curves representing the size of the basis ($\mathcal{SGB}$, $\mathcal{SRI}$) and those representing the size of the basis ($\mathcal{GB}$, $\mathcal{RI}$) collapse. Indeed, these two contexts do not generate redundant frequent MGs, and hence there are no *redundant* generic association rules. Furthermore, for the T40I10D100K context, no *exact* association rule is generated since each frequent MG is itself a CI.

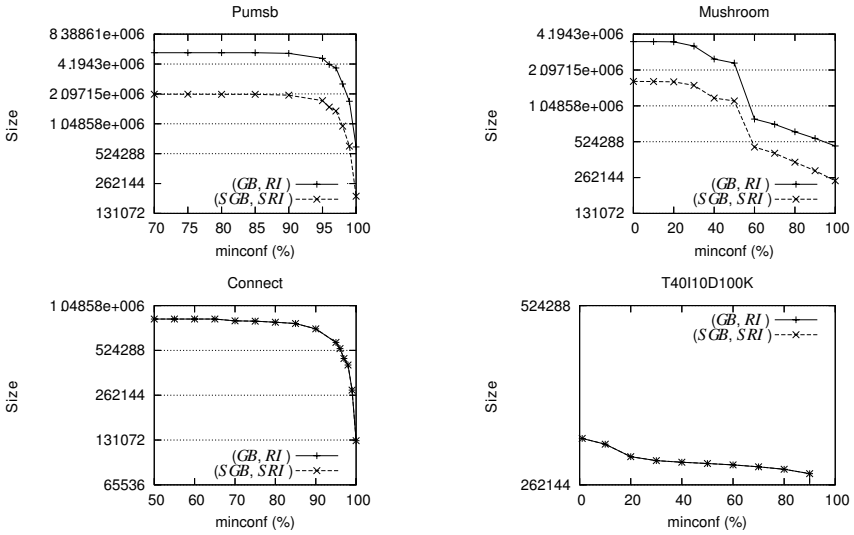


Figure 5.2: For a fixed *minsupp* value, the size of the basis ($\mathcal{GB}$, $\mathcal{RI}$) *vs.* that of the basis ($\mathcal{SGB}$, $\mathcal{SRI}$) for benchmark contexts.

Obviously, once the *minsupp* value fixed, the size of both sets $\mathcal{GB}$ and $\mathcal{SGB}$ remains unchanged along with the variation of *minconf* values. In this respect, our next experiment studies the variation of the size

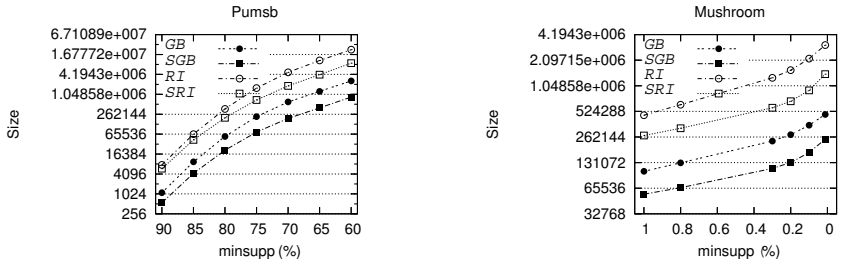


Figure 5.3: For a fixed *minconf* value, the size of the generic basis $\mathcal{GB}$ (*resp.* $\mathcal{RI}$) *vs.* that of the *succinct* generic basis $\mathcal{SGB}$ (*resp.* $\mathcal{SRI}$) for benchmark contexts.

of the generic bases ($\mathcal{SGB}$, $\mathcal{SRI}$) and ($\mathcal{GB}$, $\mathcal{RI}$) *w.r.t.* *minsupp*. For this purpose, we set the *minconf* value to **0**%, while the *minsupp* value varies between **60**% and **90**% (*resp.* **0.01**% and **1**%) for the PUMSB (*resp.* MUSHROOM) context. Our aim is to evaluate the reduction rate within *valid exact* generic association rules (*i.e.*, the generic basis $\mathcal{GB}$) compared to that within *approximate* ones (*i.e.*, the $\mathcal{RI}$ basis).

Figure 5.3 shows that, for the PUMSB context, in average **62.46**% (*resp.* **49.11**%) of the exact (*resp.* approximate) generic association rules are *redundant*, and the maximum rate of redundancy reaches **68.46**% (*resp.* **62.65**%) for a *minsupp* value equal to **65**% (*resp.* **65**%). For the MUSHROOM context, in average **50.55**% (*resp.* **52.65**%) of the exact (*resp.* approximate) generic association rules are *redundant*, and the maximum rate of redundancy reaches **53.23**% (*resp.* **57.86**%) for a *minsupp* value equal to **0.20**% (*resp.* **0.10**%).

Please note that we used very low support thresholds in our experiments. This explains the important number of mined valid association rules. That is because, for very low support values, the number of frequent itemsets increases exponentially and, consequently, the number of induced association rules also increases exponentially. In this respect, it is worth indicating the benefit brought by generic bases in general and the basis ($\mathcal{SGB}$, $\mathcal{SRI}$) in particular towards helping the end-users browsing interesting rules only. For example, for the MUSHROOM context and *minsupp* = **10**%, the size of ($\mathcal{SGB}$, $\mathcal{SRI}$) is **25, 609** while that of the complete set of valid association rules is **380, 791, 946** which constitutes a reduction rate equal to **1, 486.95**. [5] Our experiments hence clearly indicate that our approach can advantageously be used to eliminate, without loss of information, a large number of *redundant* (generic) association rules.

5.5.2 Runtime

Figure 5.4 shows the runtime of the IMG_EXTRACTOR algorithm compared to those of the SSMG_MINER algorithm. In these experiments, the *minconf* value used in IMG_EXTRACTOR is set to **0**. Thus, our algorithm is in the worst case *w.r.t.* the number of mined rules since, for each *minsupp* value, it extracts *all* valid succinct generic association rules.

In almost all experiments, our algorithm turned out to be faster. For example, for the MUSHROOM

[5] The set of valid association rules is provided by the implementation of Bart Goethals available at: *http://www.adrem.ua.ac.be/~goethals/software/*.

(*resp.* T10I4D100K) dataset, IMG_EXTRACTOR is, in average, **39.74** (*resp.* **30.95**) times faster than SSMG_MINER and the difference between both algorithms reaches **68.82** (*resp.* **49.47**) times for a *minsupp* value equal to **0.01**% (*resp.* **0.10**%). Furthermore, due to a lack of memory space, SSMG_MINER executions were not able to come to an end for some datasets and for low *minsupp* values. This happened for the PUMSB (*resp.* RETAIL and T40I10D100K) dataset for a *minsupp* value equal to **60**% (*resp.* **0.01**% and **0.50**%) after more than **4** (*resp.* **7** and **8**) hours of execution. For these three datasets and for the other tested *minsupp* values, IMG_EXTRACTOR largely outperforms SSMG_MINER. For the CONNECT (*resp.* CHESS) dataset, IMG_EXTRACTOR outperforms SSMG_MINER for *minsupp* values greater than or equal to **60**% (*resp.* **70**%). However, SSMG_MINER gets the better for low *minsupp* values even if the difference only reaches **1.58** (*resp.* **3.39**) times for a *minsupp* value equal to **50**% (*resp.* **60**%).

It is important to mention that on average the time spent by the third step of IMG_EXTRACTOR does not exceed **0.08**% of the total time. This clearly shows that once the partially ordered structure built, the derivation of succinct association rules becomes straightforward.

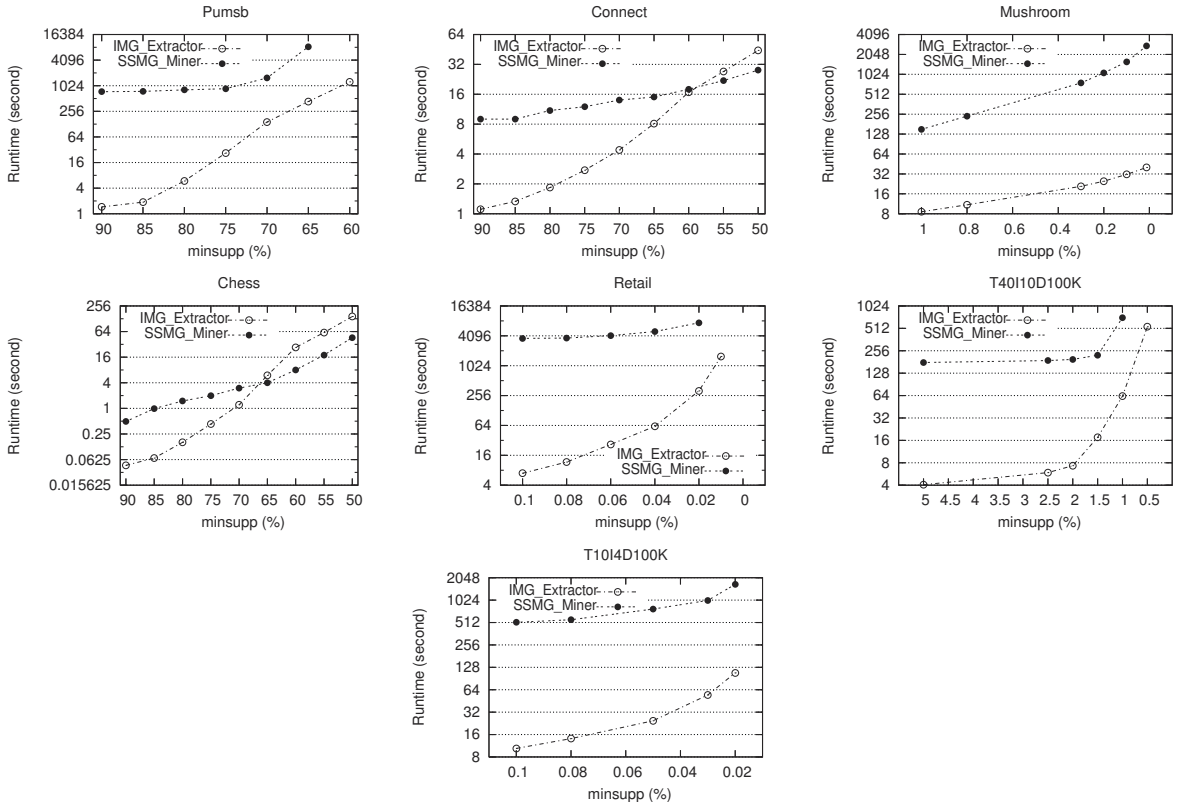


Figure 5.4: Performances of IMG_EXTRACTOR compared to those of SSMG_MINER for benchmark contexts.

We believe that the obtained results are mainly due to the following observations. The SSMG_MINER algorithm uses a depth-first left-to-right order to traverse the search space. Hence, it extracts from each visited node a *potential* frequent CI f and a *potential* frequent MG g. A highly time-consuming subsumption checking is then of paramount importance to check whether a superset of f, with the

same support, was already extracted. If it is the case, then g is compared to the *potential* frequent MGs, already belonging to its γ-equivalence class, to remove the *non-minimal* ones. After that, g is compared to the actual representative frequent MG of its γ-equivalence class, say $\mathcal{R}$, and can take its place if $g \prec \mathcal{R}$. The modification of the representative frequent MG is a consequence of the depth-first traversal since the latter does not respect the total order relation $\preceq$. If such a modification occurs, then SSMG_Miner performs an expensive treatment to remove redundant frequent MGs. Indeed, all previously extracted frequent MGs having $\mathcal{R}$ as a subset are removed since $\mathcal{R}$ became a *potential canonical* one. Hence, this removal also requires a costly search of such supersets of $\mathcal{R}$. On its side, IMG_Extractor straightforwardly extracts the frequent MG set and does not compute closure but simply derives them. In addition, the removal of a redundant frequent MG is performed by only checking if all its direct subsets are representative frequent MGs or not. Furthermore, each representative frequent MG of a given γ-equivalence class is found once for all and will not be replaced by another frequent MG. Nevertheless, it should be recalled that the task of IMG_Extractor is much more involved, than that of SSMG_Miner, since it bears the construction of the partially ordered structure. Thanks to the use of efficient functions and optimizations (described in Section 5.4), IMG_Extractor presents very interesting performances since it reduces the cost of such a construction as much as possible by avoiding unnecessary and redundant comparisons.

5.6 Conclusion

In this chapter, we described an approach aiming at extracting a lossless subset of association rules. This approach relied on the different types of MGs we presented in the previous chapter. Our approach has two main features covering the two complementary axes "effectiveness" and "efficiency". On the one hand, the proposed concise representation of association rules is only composed by succinct and informative ones based on the redefined succinct system of minimal generators (RSSMG). In addition, it is lossless in the sense that non-retained rules, *i.e.*, redundant ones, are derivable without information loss if desired. In this respect, we proposed a complete axiomatic mechanism allowing deriving valid redundant rules starting from succinct generic bases. On the other hand, aiming at offering an efficient tool for extracting the proposed representation, we designed a fast algorithm, called IMG_Extractor, which has for main originality its shrewd construction of the precedence relation between closed itemsets (CIs). Indeed, this is carried out simply using the set of frequent minimal generators (MGs). The distinction between the different types of MGs as presented in the original succinct system of minimal generators (OSSMG) was sharply exploited for reducing the cost of such a construction. Once the partially ordered structure built, deriving CIs as well as retained rules becomes an easy task. Finally, an experimental study confirms that relying on RSSMG allows to eliminate, as much as possible, *redundant* (generic) association rules, and hence to offer to the end-users a more interesting knowledge. In addition, since IMG_Extractor exploits several optimizations, obtained results show that, although it constructs the partially ordered structure, it largely outperforms the SSMG_Miner algorithm.

In the next part, we will explore the disjunctive search space by proposing new concise representations of frequent itemsets based on a disjunctive closure operator. We also generalize association rules through disjunctions.

Part III

Exploration of the Disjunctive Search Space

Chapter 6

Disjunctive Closure and Associated Exact Concise Representations of Frequent Itemsets

6.1 Introduction

Many concise representations were proposed in the literature, like those based on frequent closed itemsets [Pasquier *et al.*, 1999b], minimal generators [Liu *et al.*, 2007], disjunction-free sets [Bykowski and Rigotti, 2001, Bykowski and Rigotti, 2003], (generalized) disjunction-free generators [Kryszkiewicz, 2002], (closed) non-derivable itemsets [Calders and Goethals, 2007, Muhonen and Toivonen, 2006], and essential itemsets [Casali *et al.*, 2005a]. Considering the set of frequent itemsets as data, all those representations follow the minimum description length principle (MDLP) [Rissanen, 1978] which is based on the following insight: any regularity in the data can be used to describe the data using fewer symbols than the number of symbols needed to describe the data literally [Grunwald, 2007]. In practice, they were used in various applications where frequent itemsets and their associated supports are useful [Calders *et al.*, 2005, Mielikäinen *et al.*, 2006].

The exact concise representation based on frequent essential itemsets presents a noteworthily singularity: it explores the *disjunctive search space*. In this space, itemsets are characterized by their respective disjunctive supports. Thus, an itemset verifies an element of a context (or object) if one of its items belongs to this object. Various applications of disjunctive patterns are possible in the contexts of market basket analysis [Nanavati *et al.*, 2001], medical data analysis [Ralbovský and Kuchar, 2007], social network analysis and bioinformatics [Zhao *et al.*, 2006], etc.

In the disjunctive search space, an essential itemset contains a minimal, *w.r.t.* set inclusion, set of items among those itemsets characterizing a common set of objects. To bridge both disjunctive and conjunctive search spaces, the inclusion-exclusion identities [Galambos and Simonelli, 2000] are of use to deduce the conjunctive supports of itemsets starting from their disjunctive supports. Hence, this representation offers a basis for straightforwardly deriving the conjunctive, disjunctive and negative frequencies of a pattern [Casali *et al.*, 2003, Casali *et al.*, 2005a]. From a structural point of view, the set of frequent essential

itemsets offers an interesting structural property thanks to its order ideal shape. This makes it useful, for example, for assessing a given context density as performed in [Flouvat *et al.*, 2005].

In spite of such interesting structural and compactness properties, this exact concise representation presents two major limitations:

1. It is not self-contained in the sense that the set of frequent essential itemsets does not make it possible by itself to decide whether an itemset is frequent or not. Hence, to get out this information, this set has to be burdened by the positive border of frequent itemsets, composed by the frequent maximal itemsets [Bayardo, 1998];
2. Several essential itemsets may characterize the same set of data and, therefore, they present a certain form of redundancy.

In this respect, a compelling and thriving issue is to find a closure operator related to essential itemsets in the sake of getting a more reduced concise representation, following the minimum description length principle. Indeed, a gain in compactness terms can be reached thanks to the non-injectivity property of the closure operator since many essential itemsets will be mapped into a single element within the disjunctive search space.

Furthermore, the simultaneous use of essential itemsets and disjunctive closed itemsets can also ease the detection of their respective disjunctive equivalence classes and, hence, the traversal of the disjunctive search space. This can intensively be explored in many applications as done within the conjunctive search space thanks to their correspondences; *minimal generators* [Bastide *et al.*, 2000a] and closed itemsets [Pasquier *et al.*, 1999b], respectively. Indeed, these particular itemsets are structurally localized within the associated lattice, which gives them more semantics, contrary to other itemsets numerically retained (like non-derivable itemsets) independently from their localization. A scrutiny of the dedicated related work also highlighted the importance of essential and disjunctive closed itemsets as well as their close links with important pattern classes (see Section 6.7).

In this chapter, our main contributions are threefold:

1. We introduce a new closure operator associated to the disjunctive search space as well as its theoretical properties.
2. We show that the set of the disjunctive closures of frequent essential itemsets does not constitute by itself an exact concise representation of frequent itemsets.
3. We lead a thorough study of the finest sets of elements that can be added to gain the exactness label. The correctness of the associated exact representations is then proved as well as a description of an algorithm, called DCPR_MINER, for their mining. The targeted representation aims at palliating the limitations of that based on frequent essential itemsets as follows:
   (a) Getting out a **more compact** representation than that based on frequent essential itemsets by exploiting the non-injectivity property of the introduced closure operator. In fact, we have to only retain the disjunctive closed itemsets that ensure to exactly recovering the whole set of frequent itemsets.
   (b) Ensuring the **homogeneity** of the obtained concise representation by only keeping itemsets characterized by their disjunctive support.

Exhaustive experiments, focusing on the compactness aspect, show the effectiveness of the concise representation uniquely composed by disjunctive closed itemsets compared to the pioneering ones of the literature. Here again, the minimum description length principle allows for an objective comparison of alternative models regardless of their form or number of parameters in case the interest is in model selection [Rissanen, 1978]. In addition, to the best of our knowledge, our work is the first one allowing the extraction of such a cover thanks to a disjunctive closure operator.

The chapter is organized as follows: Section 6.2 details the disjunctive closure operator and its main properties. Then, Section 6.3 describes the structural properties of the disjunctive search space. New disjunctive closure-based representations of (frequent) itemsets are then introduced in Section 6.4. We then propose, in Section 6.5, an algorithm for extracting the proposed disjunctive itemset-based representations of frequent itemsets. The empirical evidences about the utility of our approach are provided in Section 6.6. We also discuss related work in Section 6.7.

6.2 Disjunctive Connection and Compound Operators

6.2.1 Description

The basic idea of our new concise representations is to apply a closure operator on frequent essential itemsets to obtain a more compact representation while preserving their interesting properties. As this will be structurally characterized in the next section, the disjunctive itemsets will be divided into subsets, and each subset simply represented by a *unique* element: the disjunctive closed itemset. This relies on the non-injectivity property of any closure operator. The application of this operator makes it possible to reduce the number of itemsets to be retained in the representation while being able to regenerate the whole set of frequent itemsets without information loss.

The targeted operator is different from that applied in the case of conjunctively closed itemsets [Pasquier *et al.*, 1999b]. Indeed, essential itemsets are characterized within the "*disjunctive search space*" and no more within the "*conjunctive one*". Thus, as shown by Definition 37 (*cf.* page 37), they are characterized by their disjunctive supports and no more by their conjunctive ones. Hence, a new *disjunctive* closure operator has to be devised.

The presentation of the new disjunctive closure requires that we define the corresponding applications ensuring the link from the power-set of items $\mathcal{P}(\mathcal{I})$ to that of objects $\mathcal{P}(\mathcal{O})$ and vice versa.

Definition 60 (Disjunctive connection)
Let $\mathcal{K} = (\mathcal{O}, \mathcal{I}, \mathcal{M})$ be an extraction context. The operators ensuring the connection between the $\mathcal{P}(\mathcal{I})$ and $\mathcal{P}(\mathcal{O})$ are as follows [Hamrouni et al.*, 2009b]:*

$f : \mathcal{P}(\mathcal{O}) \rightarrow \mathcal{P}(\mathcal{I})$

$$O \mapsto f(O) = \{i \in \mathcal{I} | (\exists\, o \in O)\ (((o,i) \in \mathcal{M}) \wedge ((\forall\, o_1 \in \mathcal{O} \backslash O)((o_1,i) \notin \mathcal{M})))\}$$

$g : \mathcal{P}(\mathcal{I}) \rightarrow \mathcal{P}(\mathcal{O})$

$$I \mapsto g(I) = \{o \in \mathcal{O} | (\exists\, i \in I)((o,i) \in \mathcal{M})\}$$

Let us semantically explain these operators. With respect to set inclusion, $f(O)$ is the maximal set of items which ***only*** appear in the objects of O. Dually, $g(I)$ is the largest set of objects which contain ***at least*** an item of I.

Example 37 *In this chapter, we will consider the context depicted by Table 6.1. Indeed, this latter will constitute a key example for illustrating our second contribution indicated in the introductory section, namely the limitation of disjunctive closures associated to frequent essential itemsets w.r.t. the exactness label of a concise representation. For this context, we have:* $f(\{4\}) = \emptyset$, $f(\{2, 3, 5, 6, 7\}) = \{B, C\}$, $g(\{A, C\}) = \{1, 2, 3, 4, 5, 6, 7\}$, *and* $g(\{B, D\}) = \{2, 4, 5, 6, 7\}$.

	A	B	C	D
1	×			
2	×	×		
3	×		×	
4	×			×
5	×	×	×	
6	×	×		×
7	×		×	×

Table 6.1: An extraction context.

Based on the operators introduced in Definition 60, we present the compound operators $f \circ g$ and $g \circ f$.

Definition 61 (Disjunctive compound operators)
Let $\mathcal{K} = (\mathcal{O}, \mathcal{I}, \mathcal{M})$ *be an extraction context. Let* f *and* g *be the operators as introduced in Definition 60. We define the resulting compound operators as follows [Hamrouni* et al., *2009b]:*

$h = f \circ g : \mathcal{P}(\mathcal{I}) \rightarrow \mathcal{P}(\mathcal{I})$

$$I \mapsto h(I) = \{i \in \mathcal{I} | (\forall\, o \in \mathcal{O})\ (((o,i) \in \mathcal{M}) \Rightarrow (\exists\, i_1 \in I)((o,i_1) \in \mathcal{M}))\}$$

$h' = g \circ f : \mathcal{P}(\mathcal{O}) \rightarrow \mathcal{P}(\mathcal{O})$

$$O \mapsto h'(O) = \{o \in \mathcal{O} | (\exists\, i \in \mathcal{I})\ (((o,i) \in \mathcal{M}) \wedge ((\forall\, o_1 \in \mathcal{O} \backslash O)\ ((o_1,i) \notin \mathcal{M})))\}$$

Let us semantically explain these compound operators. Let I be an itemset, $h(I) = f \circ g(I)$ is equal to the largest set of items which *only* appear in the objects that contain at least an item of I. Let O be a set of objects, $h'(O) = g \circ f(O)$ is equal to the set of objects that contain at least an item *only* appearing in the objects of O.

Example 38 *Consider the context given by Table 6.1. We have:* $h(AC) = f \circ g(AC) = f(\{1, 2, 3, 4, 5, 6, 7\}) = ABCD$, $h(BC) = f \circ g(BC) = f(\{2, 3, 5, 6, 7\}) = BC$, *and* $h'(\{4\}) = g \circ f(\{4\}) = g(\emptyset) = \emptyset$, $h'(\{4, 6, 7\}) = g \circ f(\{4, 6, 7\}) = g(D) = \{4, 6, 7\}$.

Using itemset supports, we can also characterize the disjunctive closure of an arbitrary itemset as shown by the following definition.

Definition 62 *The disjunctive closure of an itemset* I *is equal to:* $h(I) = I \cup \{i \in \mathcal{I} \backslash I |\ Supp(\vee I) = Supp(\vee(I \cup \{i\}))\}$.

Thus, $h(I)$ is the maximal itemset, *w.r.t.* set inclusion, containing I and having the same disjunctive support. It can be obtained incrementally if we have the disjunctive support of the proper supersets of I by considering items that do not change the disjunctive supports of I. The appearance of these items in the context is consequently dependent on that of a nonempty subset of I.

Example 39 *Consider the context given by Table 6.1. Let us look once again for the disjunctive closure of* `AC`*. We have* $Supp(\vee$`AC`$) = Supp(\vee$`ABC`$)$ *and* $Supp(\vee$`AC`$) = Supp(\vee$`ACD`$)$*. Indeed,* `B` *and* `D` *appear each time* `A` *or* `C` *appear. Thus,* `B` *and* `D` *belong to the closure of* `AC` *since their presence does not affect its disjunctive support. Consequently,* $h($`AC`$) =$ `ABCD`*.*

6.2.2 Properties

In the following, we present and prove the main theoretical properties of the (compound) operators we introduced.

Proposition 35 *The following properties hold for all* I, I_1, $I_2 \in \mathcal{P}(\mathcal{I})$ *and* O, O_1, $O_2 \in \mathcal{P}(\mathcal{O})$*:*

(1) $O_1 \subseteq O_2 \Rightarrow f(O_1) \subseteq f(O_2)$ (1') $I_1 \subseteq I_2 \Rightarrow g(I_1) \subseteq g(I_2)$

(2) $I \subseteq h(I)$ (2') $h'(O) \subseteq O$

(3) $I_1 \subseteq I_2 \Rightarrow h(I_1) \subseteq h(I_2)$ (3') $O_1 \subseteq O_2 \Rightarrow h'(O_1) \subseteq h'(O_2)$

(4) $f(O) = f(h'(O))$ (4') $g(I) = g(h(I))$

(5) $h(I) = h(h(I))$ (5') $h'(O) = h'(h'(O))$

(6) $g(I) \subseteq O \Leftrightarrow I \subseteq f(O)$

Proof.

- *Property (1)* $O_1 \subseteq O_2 \Rightarrow f(O_1) \subseteq f(O_2)$**.**

• *Suppose that* $O_1 \subseteq O_2$*. If* $i \in f(O_1)$*, then from Definition 60, we have* $(\exists\, o \in O_1)\ (((o,i) \in \mathcal{M}) \wedge ((\forall\, o_1 \in \mathcal{O}\backslash O_1)((o_1,i) \notin \mathcal{M})))$*. Since by hypothesis, we have* $O_1 \subseteq O_2$*, then* $(\exists\, o \in O_2)\ ((o,i) \in \mathcal{M})$*. Let us show that* i *verifies the second clause. Since we have* $(\forall\, o_1 \in \mathcal{O}\backslash O_1)((o_1,i) \notin \mathcal{M})$*, then* $(\forall\, o_1 \in \mathcal{O}\backslash O_2)((o_1,i) \notin \mathcal{M})$ *also holds. Hence,* $(\exists\, o \in O_2)\ (((o,i) \in \mathcal{M}) \wedge ((\forall\, o_1 \in \mathcal{O}\backslash O_2)((o_1,i) \notin \mathcal{M})))$ *is true. This implies that* $i \in f(O_2)$*. We conclude that* $f(O_1) \subseteq f(O_2)$*.*

- *Property (1')* $I_1 \subseteq I_2 \Rightarrow g(I_1) \subseteq g(I_2)$**.**

• *Suppose that* $I_1 \subseteq I_2$ *and let* $o \in g(I_1)$*. According to Definition 60, we have the veracity of the clause* $(\exists\, i \in I_1)((o,i) \in \mathcal{M})$*. Since* $I_1 \subseteq I_2$*, then* $(\exists\, i \in I_2)((o,i) \in \mathcal{M})$ *is also true. Thus,* $o \in g(I_2)$*. We conclude that* $g(I_1) \subseteq g(I_2)$*.*

- *Property (2)* ($f \circ g$ ***is extensive***) $I \subseteq f \circ g(I)$**.**

• *Let* $i \in I$*. By definition (cf. Definition 61), we have* $f \circ g(I) = \{i \in \mathcal{I} | (\forall\, o \in \mathcal{O})\ (((o,i) \in \mathcal{M}) \Rightarrow (\exists\, i_1 \in I)((o,i_1) \in \mathcal{M}))\}$*. If we take* $i_1 = i$*, then* $i \in f \circ g(I)$*. We then conclude that* $I \subseteq f \circ g(I)$*.*

- *Property (2')* ($g \circ f$ *is contractive*) $g \circ f(O) \subseteq O$.

• *Let* $o \in g \circ f(O)$. *According to the definition of* $g \circ f$ (*cf. Definition 61*), *we deduce that* o *verifies* $(\exists\, i \in \mathcal{I})\ (((o,i) \in \mathcal{M}) \wedge ((\forall\, o_1 \in \mathcal{O} \backslash O)\ ((o_1,i) \notin \mathcal{M})))$. *Now, suppose that* $o \notin O$. *We will have* $((o,i) \notin \mathcal{M})$ *which is in contradiction with the fact that* $o \in g \circ f(O)$. *Hence,* $o \in O$. *We can thus conclude that* $g \circ f(O) \subseteq O$.

- *Property (3)* ($f \circ g$ *is isotone*) $I_1 \subseteq I_2 \Rightarrow f \circ g(I_1) \subseteq f \circ g(I_2)$.

• *We have* $I_1 \subseteq I_2$.
$\Rightarrow g(I_1) \subseteq g(I_2)$ (*according to Property (1')*).
$\Rightarrow f \circ g(I_1) \subseteq f \circ g(I_2)$ (*according to Property (1)*).

- *Property (3')* ($g \circ f$ *is isotone*) $O_1 \subseteq O_2 \Rightarrow g \circ f(O_1) \subseteq g \circ f(O_2)$.

• *We have* $O_1 \subseteq O_2$.
$\Rightarrow f(O_1) \subseteq f(O_2)$ (*according to Property (1)*).
$\Rightarrow g \circ f(O_1) \subseteq g \circ f(O_2)$ (*according to Property (1')*).

- *Property (4)* $f(O) = f \circ g \circ f(O)$.

• *We will prove this property by proving the inclusion in both directions.*
$(\subseteq)$
We have $g \circ f(O) \subseteq O$ (*according to Property (2')*). *Hence,* $f \circ g \circ f(O) \subseteq f(O)$ (*according to Property (1)*).
$(\supseteq)$
We have $I \subseteq f \circ g(I)$ (*according to Property (2)*). *For the particular case where* $I = f(O)$ *and by replacing* I *by* $f(O)$, *we obtain* $f(O) \subseteq f \circ g \circ f(O)$.
We can then conclude that $f(O) = f \circ g \circ f(O)$.

- *Property (4')* $g(I) = g \circ f \circ g(I)$.

• *We will prove this property by proving the inclusion in both directions.*
$(\subseteq)$
We have $I \subseteq f \circ g(I)$ (*according to Property (2)*). *Hence,* $g(I) \subseteq g \circ f \circ g(I)$ (*according to Property (1')*).
$(\supseteq)$
We have $g \circ f(O) \subseteq O$ (*according to Property (2')*). *In particular, for* $O = g(I)$ *and by replacing* O *by* $g(I)$, *we obtain* $g \circ f \circ g(I) \subseteq g(I)$.
We can then conclude that $g(I) = g \circ f \circ g(I)$.

- *Property (5)* ($f \circ g$ *is idempotent*) $f \circ g(I) = f \circ g \circ f \circ g(I)$.

• *We have* $g(I) = g \circ f \circ g(I)$ (*according to Property (4')*). *By applying* f *on both sides of the equality, we obtain* $f \circ g(I) = f \circ g \circ f \circ g(I)$.

- *Property (5')* ($g \circ f$ *is idempotent*) $g \circ f(O) = g \circ f \circ g \circ f(O)$.

• *We have* $f(O) = f \circ g \circ f(O)$ (*according to Property (4)*). *By applying* g *on both sides of the equality,*

we obtain $g \circ f(O) = g \circ f \circ g \circ f(O)$.

- ***Property (6)*** $g(I) \subseteq O \Leftrightarrow I \subseteq f(O)$.
- *We will prove this equivalence by proving that both implications hold.*

($\Rightarrow$)
Suppose that $g(I) \subseteq O$. *Then, we have* $f \circ g(I) \subseteq f(O)$ (*according to Property (1)*). *Since we also have* $I \subseteq f \circ g(I)$ (*according to Property (2)*), *we conclude by transitivity that* $I \subseteq f(O)$.
($\Leftarrow$)
Suppose that $I \subseteq f(O)$. *Then, we have* $g(I) \subseteq g \circ f(O)$ (*according to Property (1')*). *Since we also have* $g \circ f(O) \subseteq O$ (*according to Property (2')*), *we conclude by transitivity that* $g(I) \subseteq O$.
Hence, $g(I) \subseteq O \Leftrightarrow I \subseteq f(O)$. $\diamondsuit$

The following two propositions straightforwardly derive from Proposition 35.

Proposition 36 *The operator* h *is a closure operator.*

Proof. According to Proposition 35, h fulfills the conditions required by the definition of a closure operator (*cf.* Definition 14, page 17). Indeed, it is extensive (*cf.* Property (2)), isotone (*cf.* Property (3)) and idempotent (*cf.* Property (5)). $\diamondsuit$

Proposition 37 *The operator* h' *is a kernel operator.*

Proof. According to Proposition 35, h' is contractive (*cf.* Property (2')), isotone (*cf.* Property (3')) and idempotent (*cf.* Property (5')). It is hence a kernel operator according to Definition 14 (*cf.* page 17). $\diamondsuit$

6.3 Structural Characterization of the Disjunctive Search Space

The definition of the closure operator h structurally characterizes the disjunctive closure of any itemset I. This allows to straightforwardly compute disjunctive closed itemsets freely either using a breadth-first or a depth-first traversal of the search space. Thanks to this operator, the disjunctive search space is partitioned into so-called *disjunctive equivalence classes w.r.t.* the relation "has the same disjunctive closure". In each class, the associated disjunctive closed itemset is the unique maximal element *w.r.t.* set inclusion, while the corresponding essential itemsets are the minimal ones. The elements of each equivalence class share the same set of objects and, hence, have the same disjunctive support and disjunctive closure. The exploration of the disjunctive search space using the operator h will be at the origin of the new concise representations for the set of frequent itemsets as explained in the remainder.

Now, we begin by presenting the definition of a disjunctive closed itemset.

Definition 63 (Disjunctive closed itemset)
An itemset I *is said to be disjunctive closed if* $h(I) = I$. *Equivalently,* I *is disjunctive closed iff* $Supp(\vee I) < min\{Supp(\vee(I \cup \{i\}))|\ i \in \mathcal{I} \backslash I\}$.

A disjunctive closed itemset is then the maximal set of items only contained in the set of objects where at least an item of I appears, and nowhere else. Since the disjunctive support augments proportionally to itemset sizes, *i.e.*, $Supp(\vee I_1) \leq Supp(\vee I_2)$ if $I_1 \subseteq I_2$, it is sufficient to only compare the disjunctive support of I with those of its *immediate* supersets, instead of *all*, to check whether it is a disjunctive closed itemset or not. Let us give some examples of the closure operator h that will be at the roots of the concise representations we will introduce.

Example 40 *Given the context depicted by Table 6.1, the itemset* BC *is a disjunctive closed itemset, since it is equal to the largest set of items only contained in the set of objects where* B *or* C *appears, i.e., {2, 3, 5, 6, 7}. Hence,* $h(BC) = BC$. *Using disjunctive supports, we have* $Supp(\vee BC) = \mathbf{5} < min\{Supp(\vee ABC), Supp(\vee BCD)\} = \mathbf{6}$. *While* ACD *is not a disjunctive closed itemset since* B *only appears in the set of objects, equal to* $\mathcal{O}$, *where at least one item of* ACD *appears. Actually,* $h(ACD) = ABCD$.

Note that as for the frequent essential itemset-based representation (*cf.* Remark 1, page 37), we need to add the couple $(\emptyset, |\mathcal{O}|)$ to the representation to make it lossless *w.r.t.* the empty set. This does not affect the structural properties of disjunctive closed itemsets nor the homogeneity of the representations we will propose in the remained. The set of *all* disjunctive closed itemsets that can be drawn from a context $\mathcal{K}$ will be denoted $\mathcal{DCI}$.

The following proposition shows how to select the disjunctive closure of an arbitrary itemset I among those belonging to $\mathcal{DCI}$.

Proposition 38 *Let* $I \subseteq \mathcal{I}$. *The itemset* $h(I)$ *is the smallest disjunctive closure containing* I: $h(I) = min_{\subseteq}\{I_1 \in \mathcal{DCI} \mid I \subseteq I_1\}$.

Proof. The proof straightforwardly derives from the definition of a disjunctive closed itemset. $\diamondsuit$

Proposition 39 establishes the link between the disjunctive support of an itemset and that of its closure.

Proposition 39 *Let* $I \subseteq \mathcal{I}$. $Supp(\vee I) = Supp(\vee h(I))$.

Proof. According to Property (4') (*cf.*, Proposition 35), we have $g(I) = g \circ f \circ g(I)$. Hence, $g(I) = g(h(I))$. We then have: $|g(I)| = |g(h(I))|$. It follows that $Supp(\vee I) = Supp(\vee h(I))$. $\diamondsuit$

Proposition 40 shows that it is possible to deduce the disjunctive closure of an itemset thanks to one of its subsets.

Proposition 40 *Let* $I, I_1 \subseteq \mathcal{I}$ *be two itemsets. We then have:*

$$(I_1 \subseteq I \subseteq h(I_1)) \Rightarrow (h(I) = h(I_1)).$$

Proof. We have $I_1 \subseteq I \subseteq h(I_1)$. Since h is isotone as being a closure operator, we obtain $h(I_1) \subseteq h(I) \subseteq h(h(I_1))$. Thanks to the idempotency property, we get $h(I_1) \subseteq h(I) \subseteq h(I_1)$. Thus, we can conclude that $h(I) = h(I_1)$. $\diamondsuit$

Thanks to Proposition 41, we establish the link between disjunctive closed itemsets and essential itemsets.

Proposition 41 *Let $\mathcal{EI}$ be the set of all essential itemsets that can be extracted from a context $\mathcal{K}$.*

$$\forall\ (I \subseteq \mathcal{I}),\ \exists\ (I_1 \in \mathcal{DCI} \text{ and } I_2 \in \mathcal{EI}) \text{ such that } h(I_2) = h(I) = I_1 \text{ and } I_2 \subseteq I.$$

Proof. Let $X \in \mathcal{EI}$ be a maximal subset of I such that $Supp(\vee X) = Supp(\vee I)$. Hence, $g(X) = g(I)$. By applying f, we have: $f \circ g(X) = f \circ g(I)$. Hence, $h(X) = h(I)$. Since $I \subseteq h(I)$, then $I \subseteq h(X)$. We can then conclude that there is a disjunctive closed itemset $I_1 = h(X)$ associated to an essential itemset, namely X, that contains I. It is hence sufficient to take $I_2 = X$. $\diamondsuit$

It is important to mention that Proposition 39 and Proposition 40 offer a new characterization of essential itemsets. Indeed, recall that their original characterization was based on their associated supports as follows: $I \subseteq \mathcal{I}$ is an essential itemset if $Supp(\vee I) > max\{Supp(\vee I \backslash \{i\}) \mid i \in I\}$ (*cf.* Definition 37, page 37). The new characterization, based on disjunctive closed itemsets, is as follows:

Proposition 42 *Let $I \subseteq \mathcal{I}$. I is an essential itemset if $\forall\ I_1 \subset I,\ I \nsubseteq h(I_1)$.*

Proof. Suppose that $\exists\ I_1 \subset I$ *s.t.* $I \subseteq h(I_1)$. According to Proposition 40, we have $h(I) = h(I_1)$. Thanks to Proposition 39, we have $Supp(\vee I_1) = Supp(\vee h(I_1)) = Supp(\vee h(I)) = Supp(\vee I)$. Since $Supp(\vee I_1) = Supp(\vee I)$, then I is not an essential itemset. Thus, if I is an essential itemset, then $\forall\ I_1 \subset I,\ I \nsubseteq h(I_1)$. $\diamondsuit$

It is also worth noting that some essential itemsets as well as disjunctive closed itemsets can be characterized using disjunctive rules, as done in [Bykowski and Rigotti, 2001, Bykowski and Rigotti, 2003] for the disjunctive-free sets. Indeed, an itemset I is an essential of size greater than or equal to **2** if there is no item $i \in I$ *s.t.* $i \Rightarrow \vee(I \backslash \{i\})$ is always satisfied. This means that i must appear in an object in which no item of $(I \backslash \{i\})$ appears. On the other hand, an itemset I is a disjunctive closed of size greater than or equal to **1** if there is no item $i \in \mathcal{I} \backslash I$ *s.t.* $i \Rightarrow \vee I$ is always satisfied. This means that i must appear in an object in which no item of I appears.

The following proposition ensures that it is possible to derive the disjunctive support of each subset of an arbitrary itemset starting from $\mathcal{DCI}$.

Proposition 43 *Let $I \subseteq \mathcal{I}$. $\forall\ I_1 \subseteq I$, the disjunctive support of I_1 can be exactly derived from $\mathcal{DCI}$.*

Proof. The set $\mathcal{DCI}$ contains all the disjunctive closed itemsets that can be drawn from a context $\mathcal{K}$. Hence, $\forall\ I_1 \subseteq I,\ h(I_1) \in \mathcal{DCI}$. We can thus retrieve the exact disjunctive support of I_1 thanks to Proposition 39. $\diamondsuit$

6.4 Disjunctive Closure-based Concise Representations of Frequent Itemsets

6.4.1 New Concise Representation for All Itemsets

Let us begin by introducing a concise representation of the *whole* set of itemsets based on disjunctive closed itemsets. This is stated in Theorem 12.

Theorem 12 *The set $\mathcal{DCI}$ of disjunctive closed itemsets, associated to their respective disjunctive supports, is an exact concise representation of the whole set of itemsets.*

Proof. Let $I \subseteq \mathcal{I}$. It was proven through Proposition 43 that the disjunctive support of I and those of its subsets can be exactly derived from $\mathcal{DCI}$. Then, by applying an inclusion-exclusion identity using the obtained disjunctive supports (*cf.* Lemma 1, page 13), we are able to obtain the exact conjunctive support of I. $\diamondsuit$

The set $\mathcal{DCI}$ is thus not only a concise representation of *frequent* itemsets but also that of the *whole* set of itemsets that can be drawn from a context (*i.e.*, even the associated supports of *infrequent* itemsets can be derived using $\mathcal{DCI}$).

Example 41 *Consider the context given by Table 6.1. The associated disjunctive lattice is sketched by Figure 6.1, where each node contains a disjunctive itemset along with its disjunctive support. Different sets of itemsets are also indicated. The essential itemsets are shown with bold letters, while the itemsets belonging to $\mathcal{DCI}$ are underlined. The set $\mathcal{FEI}$ induces an order ideal, as shown in Figure 6.1 for minsupp = 1. The elements belonging to the negative border of $\mathcal{FEI}$, denoted $\mathcal{B}d^{-}(\mathcal{FEI})$, are in italics. An example of a disjunctive equivalence class, induced by the disjunctive closure operator, is also sketched. Its minimal element is the essential itemset A and its largest one is the disjunctive closed itemset ABCD. Please note that if, for example, an itemset is in bold letters and is also underlined, then this means that it is both an essential itemset and a disjunctive closed one, e.g., the itemset BC. As an indication, the itemsets belonging to $\mathcal{B}d^{+}(\mathcal{FI})$ are marked by dashed circles.*

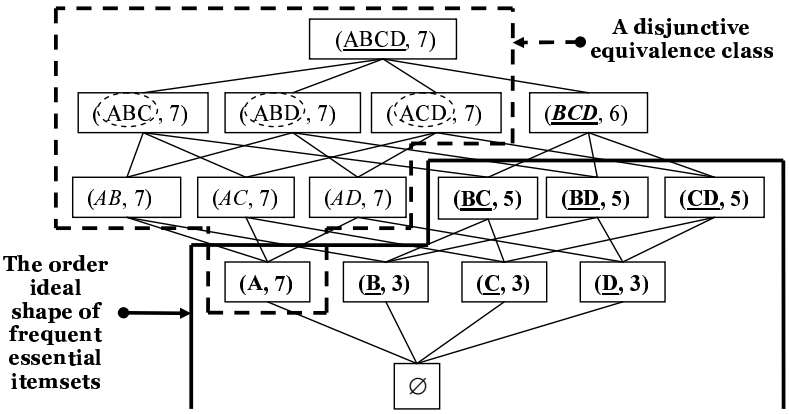


Figure 6.1: The disjunctive lattice associated to the context depicted by Table 6.1.

6.4.2 Effect of Setting the Conjunctive Frequency Constraint

In practice, the end-users are mainly interested in *frequent* itemsets and not in *all* itemsets. The selection of frequent itemsets can be done as a post-treatment by comparing the obtained supports with *minsupp*. Nevertheless, it is more advantageous to restrict the representation to only the required elements while preserving the exact regeneration of frequent itemsets. Among these elements, disjunctive closed itemsets

having at least a *frequent* essential itemset as a generator should obviously be maintained. Indeed, they cover at least a frequent itemset, namely the associated frequent essential itemset. These closed sets, along with their associated disjunctive supports, will hence constitute the key information allowing to derive the *exact* disjunctive and, hence, conjunctive supports of frequent itemsets. The subset of $\mathcal{DCI}$ containing these closures will be denoted $\mathcal{EDCI}$. [1] This set is then as follows:

Definition 64 (Set of essential disjunctive closed itemsets)
The set $\mathcal{EDCI}$ *is equal to:* $\mathcal{EDCI} = \{h(I) \in \mathcal{DCI} | \ I \in \mathcal{FEI}\}$.

Example 42 *Consider Figure 6.1. For minsupp* $= \mathbf{1}$, $ABCD \in \mathcal{EDCI}$, *since it has* A *for frequent essential itemset.*

The next lemma compares the size of $\mathcal{EDCI}$ with that of $\mathcal{FEI}$.

Lemma 7 *The cardinality of* $\mathcal{EDCI}$ *is at most equal to that of* $\mathcal{FEI}$.

Proof. To each frequent essential itemset is associated a unique element in $\mathcal{EDCI}$. Hence, the size of the set $\mathcal{EDCI}$ will be lower than or equal to that of $\mathcal{FEI}$. $\diamondsuit$

Thanks to Lemma 8, we can correctly derive the disjunctive supports of *frequent* itemsets from the elements of $\mathcal{EDCI}$. Once disjunctive supports derived, Lemma 1 (*cf.* page 13) will then be used when desired to deduce their conjunctive and negative supports.

Lemma 8 *Let* $\mathcal{FI}$ *be the set of frequent itemsets,* $I \subseteq \mathcal{I}$ *and* $I_{min} = \min_{\subseteq}\{I_1 \in \mathcal{EDCI} \mid I \subseteq I_1\}$ *if it exists. We then have:*

$$\forall\ I \in \mathcal{FI},\ (\exists\ I_{min}) \ \wedge\ (Supp(\vee I) \ = \ Supp(\vee I_{min})).$$

Proof. The proof straightforwardly derives from that of Proposition 41 (*cf.* page 115) and the fact that the disjunctive closure of a *frequent* itemset I is the smallest one covering it among those of $\mathcal{EDCI}$. $\diamondsuit$

As mentioned above, a concise representation of frequent itemsets based on disjunctive closed itemsets must contain the elements of $\mathcal{EDCI}$. Nevertheless, is this set sufficient to offer an exact concise representation?

During the regeneration process of frequent itemsets, the *minimal infrequent* itemsets are also checked since they have all their subsets frequent. Let I be such an itemset. If I is not covered by any closure of $\mathcal{EDCI}$, then it is infrequent according to Lemma 8. However, the itemset I can be covered by an element belonging to $\mathcal{EDCI}$, while having its proper closure not in $\mathcal{EDCI}$. Indeed, recall that the set $\mathcal{EDCI}$ results from combining two constraints of different types, namely a monotone one through the disjunctive support and an anti-monotone constraint using *minsupp*. Some key disjunctive closed itemsets for a correct regeneration process may thus be pruned since they have all their essential itemsets infrequent. This leads to affecting to I a wrong disjunctive support which, in some cases, will incorrectly make I frequent instead of infrequent. Let us take a concrete example.

[1] Stands for frequent Essential-based Disjunctive Closed Itemsets

Example 43 *Let us consider the context shown in Table 6.1. According to Figure 6.1, for minsupp* $= \mathbf{1}$*, we have* $\mathcal{EDCI} = \{B, C, D, BC, BD, CD, ABCD\}$*. The first six closed itemsets of* $\mathcal{EDCI}$ *are equal to their respective frequent essential itemsets. While the last one has the frequent essential itemset* A *as a generator. Note that* $BCD \notin \mathcal{EDCI}$ *since its generator, namely itself, is an infrequent essential itemset. Indeed, the conjunctive support of* BCD *is equal to* $\mathbf{0}$ *(cf. Figure 6.1 where it is shown not to belong to the order ideal induced by setting minsupp to* $\mathbf{1}$*).*

Let us regenerate the set of frequent itemsets starting from $\mathcal{EDCI}$*. We begin by* $\mathbf{1}$*-itemsets, i.e.,* A*,* B*,* C *and* D*. The smallest closure containing* A *is* $ABCD$*. Hence, its disjunctive support is equal to* $\mathbf{7}$*, which also corresponds to its conjunctive support. It is the same for the remaining* $\mathbf{1}$*-itemsets. Thus, we find that their associated conjunctive supports are respectively equal to* $\mathbf{7}$*,* $\mathbf{3}$*,* $\mathbf{3}$ *and* $\mathbf{3}$*. We hence have the four candidates as frequent.*

We then handle candidate $\mathbf{2}$*-itemsets. Thanks to the anti-monotone property of the frequency, an arbitrary itemset will only be treated if all its subsets were already proved to be frequent itemsets. This can be ensured thanks to a levelwise [Agrawal* et al.*, 1996] or a depth-first right-to-left [Calders and Goethals, 2005] traversal of the search space. Consider the case of* AB *whose subsets* A *and* B *are shown to be frequent. The smallest closure in* $\mathcal{EDCI}$ *containing* AB *is* $ABCD$*. The disjunctive support of* AB *is then equal to* $\mathbf{7}$*. By applying an inclusion-exclusion equality (cf. Lemma 1), we have* $Supp(AB) = -Supp(\vee AB) + Supp(\vee A) + Supp(\vee B) =$ $\mathbf{-\ 7\ +\ 7\ +\ 3\ =\ 3}$*. The itemset* AB *is hence frequent. The same process is applied for the remaining candidate* $\mathbf{2}$*-itemsets.*

Let us now focus on the candidate $\mathbf{3}$*-itemset* BCD *whose all subsets are frequent and which hence must be checked. When we retrieve the disjunctive support of* BCD *from* $\mathcal{EDCI}$*, we will assign to* BCD *the disjunctive support of the smallest element (w.r.t. set inclusion) in* $\mathcal{EDCI}$ *subsuming it, i.e.,* $ABCD$*. When computing the conjunctive support of* BCD*, we obtain* $Supp(BCD) = Supp(\vee BCD)$ - $Supp(\vee BC)$ - $Supp(\vee BD)$ - $Supp(\vee CD)$ + $Supp(\vee B)$ + $Supp(\vee C)$ + $Supp(\vee D) =$ $\mathbf{7\ -\ 5\ -\ 5\ -\ 5\ +\ 3\ +\ 3\ +\ 3\ =\ 1}$*. However, the actual disjunctive support of* BCD *is equal to* $\mathbf{6}$ *and not* $\mathbf{7}$*. Its actual conjunctive support is then equal to* $\mathbf{0}$ *and not* $\mathbf{1}$*. Consequently, the obtained result will falsify the frequency status of* BCD *w.r.t. minsupp since it will be wrongly classified as frequent instead of infrequent. The flaw is due to the pruning of the disjunctive closed itemset* BCD *whose unique generator is the infrequent essential itemset* BCD *(i.e., itself).*

The previous example clearly shows that $\mathcal{EDCI}$ cannot constitute by itself an exact concise representation of frequent itemsets. We thus need to retain some closures, in addition to $\mathcal{EDCI}$, that ensure correctly flagging the frequency status of itemsets whenever a wrong computation can arise. The following subsection explores this issue.

6.4.3 New Concise Representations of Frequent Itemsets

We propose, in this section, new concise representations of frequent itemsets. These representations are homogeneous in the sense that they are *only* composed by disjunctive itemsets. They hence require *only* exploring the disjunctive search space while offering the direct retrieval of the different types of support of frequent itemsets. These representations hence avoid the exploration of the conjunctive search space since they do not require supplementary information from the conjunctive search space in order to check whether an itemset is frequent or not.

A. Disjunctive Search Space-based Representation

The first representation consists in a straightforward solution to ensure the exactness of the representation based on $\mathcal{EDCI}$. This is carried out thanks to the set $\mathcal{FEI}$ of frequent essential itemsets. The exactness of this representation is stated by the following theorem.

Theorem 13 *The set $\mathcal{EDCI} \cup \mathcal{FEI}$ of disjunctive itemsets, associated to their respective disjunctive supports, is an exact representation of the set of frequent itemsets $\mathcal{FI}$.*

Proof. Let I be an arbitrary itemset. If there is an itemset I_1 *s.t.* $I_1 \in \mathcal{FEI}$ and $I_1 \subseteq I \subseteq h(I_1)$, then $h(I) = h(I_1)$ since h is isotone as being a closure operator. Hence, $Supp(\vee I) = Supp(\vee I_1)$. Since the disjunctive support of I is correctly derived, then its conjunctive support can be exactly computed thanks to Lemma 1 (*cf.* page 13), and then compared *vs.* *minsupp* to retrieve its frequency status. If there is not such an itemset I_1, then I is necessarily encompassed between an *infrequent* essential itemset and its closure. Consequently, I is infrequent since the set of frequent itemsets is an order ideal. $\diamondsuit$

The proof of Theorem 13 can be treated as a naive algorithm for determining frequent itemsets and their supports. Indeed, this can straightforwardly be done in a levelwise manner that regenerates **1**-frequent itemsets, **2**-frequent itemsets, and so forth. As shown in the proof, this representation ensures the easy derivation of the disjunctive support of each frequent itemset, and hence the negative one using De Morgan's law. Since it is composed by particular elements within the disjunctive search space, namely essential and disjunctive closed itemsets, then it will be denoted $\mathcal{DSSR}$, which stands for Disjunctive Search Space-based Representation.

B. Disjunctive Closed Itemset-based Representations

Now, we propose to only add some disjunctive closed itemsets to $\mathcal{EDCI}$, instead of $\mathcal{FEI}$. This will ensure obtaining the same kind of itemsets – disjunctive closed – within the resulting representation [Hamrouni *et al.*, 2007a]. In this respect, the added itemsets constitute the set $\mathcal{BDCI}$. [2] This set contains disjunctive closures of odd-sized infrequent seeds belonging to the negative border of $\mathcal{FEI}$. Having all their respective subsets as frequent itemsets, such seeds need to be checked during the regeneration process *w.r.t.* the constraint "to be frequent". In this situation, whenever their closures not retained in the representation, a wrong derivation of their exact disjunctive supports can be misleading *w.r.t.* their infrequency (*cf.* the case of the infrequent itemset `BCD` detailed in Example 43). The set $\mathcal{BDCI}$ is formally defined as follows:

Definition 65 (Set of all added disjunctive closed itemsets)
Let $\mathcal{EI}$ be the set of all essential itemsets that can be extracted from a context $\mathcal{K}$. The set $\mathcal{BDCI}$ is defined as follows: $\mathcal{BDCI} = \{h(I) \in \mathcal{DCI} |\ (I \in \mathcal{B}d^{-}(\mathcal{FEI}) \cap \mathcal{EI}) \wedge (|I| \text{ is odd})\}$

Example 44 *For minsupp = 1, `BCD` $\in \mathcal{BDCI}$. Indeed, its unique essential itemset is itself. Moreover, the essential itemset `BCD` is an odd-sized infrequent itemset, having all its proper subsets frequent. Hence, it belongs to the negative border of frequent essential itemsets.*

[2] Stands for Border-based Disjunctive Closed Itemsets

It is important to mention that in the definition of the set $\mathcal{BDCI}$, we did not consider the disjunctive closures of *even-sized infrequent* essential itemsets belonging to the border of $\mathcal{FEI}$. This is argued by the fact that the absence of such closures does not affect the exactness of the regeneration process as proved in the following. *Infrequent non*-essential itemsets belonging to $\mathcal{B}d^-(\mathcal{FEI})$ were also omitted since they are already included in $\mathcal{EDCI}$ (*cf.* Proposition 40). Note however that both sets $\mathcal{EDCI}$ and $\mathcal{BDCI}$ are not necessarily disjoint in the sense that a same closure can belong to both sets. Indeed, a disjunctive closure can simultaneously have as seeds a frequent essential itemsets (and hence belong to $\mathcal{EDCI}$) and an odd-sized infrequent essential itemsets (and hence belong to $\mathcal{EDCI}$). However, since in the representation $\mathcal{EDCI} \cup \mathcal{BDCI}$ we take the union of both sets, such a redundancy is necessarily removed. The exactness of the representation based on $\mathcal{EDCI}$ and $\mathcal{BDCI}$ is provided by Theorem 14.

Theorem 14 *The set $\mathcal{EDCI} \cup \mathcal{BDCI}$ of disjunctive closed itemsets, associated to their respective disjunctive supports, is an exact concise representation of the set $\mathcal{FI}$ of frequent itemsets.*

Proof. Let $I \subseteq \mathcal{I}$. If $\exists\ I_1 \subset I$ *s.t.* I_1 is infrequent, then I is also infrequent. Otherwise (*i.e.*, $\forall\ I_1 \subset I$, $I_1 \in \mathcal{FI}$), we need to show that the frequency status of I is correctly retrieved starting from $\mathcal{EDCI} \cup \mathcal{BDCI}$. In addition, its conjunctive support must be exactly computed if it is frequent. Two cases have to be distinguished:

1. If I `is frequent`, then its disjunctive support will be correctly derived thanks to Lemma 8 (*cf.* page 117). Indeed, I is either a frequent essential itemset and its closure is in $\mathcal{EDCI}$, or encompassed between a frequent essential itemset and its closure, obviously the latter belonging to $\mathcal{EDCI}$. Once its disjunctive support derived, the computation of the conjunctive support becomes then straightforward thanks to an inclusion-exclusion identity.
2. If I `is infrequent`, then two cases arise:
   (a) If I ***is not*** `an essential itemset`, then it is contained in the disjunctive closure of one of its subsets. By hypothesis, this latter is frequent and hence its closure belongs to $\mathcal{EDCI}$. Also in this case, the disjunctive support of I will be correctly derived and hence its conjunctive support. By comparing the conjunctive support of I with *minsupp*, we get the information that I is infrequent.
   (b) If I ***is*** `an essential itemset`, then necessarily $I \in \mathcal{B}d^-(\mathcal{FEI}) \cap \mathcal{EI}$. Let h_s be the smallest disjunctive closed itemset in $\mathcal{EDCI}$ containing I. If h_s does not exist then I is immediately guessed to be infrequent (thanks to Lemma 8). Otherwise, from Formula (1) of Lemma 1, we have:
   $$Supp(I) = \sum_{\emptyset \subset I_1 \subseteq I} (\text{-}1)^{|I_1|-1} Supp(\vee I_1) = (\text{-}1)^{|I|-1} Supp(\vee I) + \sum_{\emptyset \subset I_1 \subset I} (\text{-}1)^{|I_1|-1} Supp(\vee I_1).$$
   Hence, according to the size of I we have:
      i. If $|I|$ `is even`, then $Supp(I) = -Supp(\vee I) + \sum_{\emptyset \subset I_1 \subset I} (\text{-}1)^{|I_1|-1} Supp(\vee I_1) < minsupp$ (since I is infrequent). Since $I \subset h_s$, we have $Supp(\vee I) \leq Supp(\vee h_s)$. Hence, $-Supp(\vee h_s) + \sum_{\emptyset \subset I_1 \subset I} (\text{-}1)^{|I_1|-1} Supp(\vee I_1) \leq -Supp(\vee I) + \sum_{\emptyset \subset I_1 \subset I} (\text{-}1)^{|I_1|-1} Supp(\vee I_1) < minsupp$. This inequality points out that even if h_s is not necessarily the disjunctive closure of I, we can detect that I is infrequent.

ii. If $|I|$ `is odd`, then by applying the same process as for the previous case, we are not able to detect in all the cases the frequency status of I. Indeed, in this case, $(\text{-}1)^{|I|-1}Supp(\vee I)$ is a positive quantity and not a negative one as in the case where $|I|$ is even. Hence, if h_s is not the correct closure of I, then $h(I)$ has all its essential itemsets infrequent. It then belongs to $\mathcal{BDCI}$ (*cf.* Definition 65) and its addition to the representation is necessary to ensure the correct detection of the status of I.

Thus, the set $\mathcal{EDCI} \cup \mathcal{BDCI}$ is an exact concise representation of $\mathcal{FI}$. $\diamondsuit$

The proof of Theorem 14 can easily be transformed to a naive algorithm for deriving frequent itemsets and their associated supports starting from our representation. In the remainder, the concise representation $\mathcal{EDCI} \cup \mathcal{BDCI}$ will be denoted BDCIs_rep.

According to Lemma 8 (*cf.* page 117), we can further reduce the cardinality of the obtained representation. This is carried out by only retaining, in the set $\mathcal{BDCI}$, each closure not already belonging to $\mathcal{EDCI}$ which is covered by at least a disjunctive closed itemset from $\mathcal{EDCI}$. Hereafter, the resulting subset of $\mathcal{BDCI}$ after this pruning will be denoted $\mathcal{ADCI}$, [3] and is formally introduced by the following definition.

Definition 66 (Set of added disjunctive closed itemsets)
The set $\mathcal{ADCI}$ is defined as follows: $\mathcal{ADCI} = \{I \in \mathcal{BDCI} \mid (I \notin \mathcal{EDCI})$ and $(\exists\ I' \in \mathcal{EDCI}$ s.t. $I \subset I')\}$.

Both sets $\mathcal{EDCI}$ and $\mathcal{ADCI}$ are thus *ensured to be disjoint* (*i.e.*, $\mathcal{EDCI} \cap \mathcal{ADCI} = \emptyset$). Indeed, each closure of $\mathcal{EDCI}$ has *at least* a *frequent* essential itemset as a seed, while *all* the essential itemsets of a closure belonging to $\mathcal{ADCI}$ are *infrequent*. The next theorem states the correctness of the representation $\mathcal{EDCI} \cup \mathcal{ADCI}$.

Theorem 15 *The set $\mathcal{EDCI} \cup \mathcal{ADCI}$ of disjunctive closed itemsets, associated to their respective disjunctive supports, is an exact concise representation of the set $\mathcal{FI}$ of frequent itemsets.*

Proof. The proof is based on that of Theorem 14 and on Lemma 8. Indeed, if an itemset I is not covered by any element of $\mathcal{EDCI}$, then we can directly assert that I is infrequent (*cf.* Lemma 8). Therefore, thanks to the extensivity property of any closure operator, $h(I)$ cannot be subsumed by any element of $\mathcal{EDCI}$. Thus, it can be pruned from $\mathcal{BDCI}$ while ensuring the correctness of the regeneration mechanism (*cf.* proof of Theorem 14). The set $\mathcal{EDCI} \cup \mathcal{ADCI}$ is then an exact concise representation of $\mathcal{FI}$. $\diamondsuit$

In the remainder, the concise representation $\mathcal{EDCI} \cup \mathcal{ADCI}$ will be denoted DCIs_rep.

6.4.4 Features of the Proposed Representations

In addition to the exact retrieval of frequent itemsets as well as their different kinds of support, the proposed concise representations present several interesting properties.

[3] Stands for Added Disjunctive Closed Itemsets.

A. Case of the DCIs_rep Representation

The DCIs_rep representation offers the following main advantages:

1. **Homogeneity:** The DCIs_rep set overcomes the heterogeneity problem since it only involves disjunctive itemsets (*vs.*, for example, $\mathcal{FEI} \cup \mathcal{B}d^{+}(\mathcal{FI})$). Its elements have the same structural properties. Indeed, they are the top elements of their associated equivalence classes within the disjunctive search space. This ensures the homogeneity of the representation since all its elements are also provided with the same type of support, *i.e.*, the disjunctive support.

2. **Redundancy free:** Redundancy is due to the fact that a set of disjunctive itemsets can characterize the same set of objects. This is avoided in our case since such a set is simply represented by a unique disjunctive closed itemset, thanks to the proposed disjunctive closure operator.

3. **Small size:** $\mathcal{EDCI}$ is the smallest set that concisely represents the equivalence classes containing at least a frequent itemset, since only a unique element is maintained per class. In addition, the size of $\mathcal{ADCI}$ is expected to be very small compared to $\mathcal{B}d^{+}(\mathcal{FI})$, since its elements must fulfill many easy-to-check constraints. This will be confirmed by experiments where DCIs_rep is shown to provide very interesting compactness rates.

4. **Low regeneration cost:** It is worth mentioning that our concise representation allows retrieving the conjunctive support faster than from frequent non-derivable itemsets. Indeed, for an itemset I of size n, the retrieval process of $Supp(I)$ from this representation requires the costly evaluation of $\mathbf{2}^n$ deduction rules based on Bonferroni-inequalities [Mielikäinen *et al.*, 2006]. The computation cost for inferring supports is then awfully high which makes this representation not very easy to use [Liu *et al.*, 2007, Mielikäinen *et al.*, 2006]. Note also that taking closures of frequent non-derivable itemsets to obtain the closed non-derivable representation complicates both the extraction process of this latter, as well as the regeneration process of frequent itemsets. On its side, the frequent closed itemset-based representation [Pasquier *et al.*, 1999b] allows retrieving the conjunctive support of I by searching for the smallest closure containing it. However, it does not allow the straightforward derivation of its disjunctive and negative supports. While the retrieval of $Supp(I)$ from our concise representation only needs to evaluate a unique inclusion-exclusion identity. Moreover, given at hand $Supp(I_1)$ such that I_1 is an immediate subset of I and $I \backslash I_1 = i$, we can straightforwardly deduce the support of I. Indeed, it derives from Formula (1) in Lemma 1 that:

$$Supp(I) = Supp(I_1) + \sum_{i \subseteq I_2 \subseteq I} (\text{-}1)^{|I_2|\ \text{-}\ 1}\ Supp(\vee I_2)$$

   The regeneration process can also be further optimized as follows. Let us suppose that the smallest closure covering an itemset I belongs to $\mathcal{ADCI}$. Since this latter set gathers closures whose associated equivalence classes only contain infrequent itemset, the itemset I is hence necessarily infrequent. Thus, we do not need to compute its conjunctive support. Nevertheless, for the sake of homogeneity, the closures belonging to $\mathcal{EDCI}$ and those belonging to $\mathcal{ADCI}$ were included in DCIs_rep without distinguishing their membership.

   In this situation, a solution is to assign a common support for the closures belonging to $\mathcal{ADCI}$, for example **0**, during the mining process. The choice of this value is interesting since a disjunctive

support cannot be equal to **0**, what allows distinguishing the membership of a closure. Indeed, if its support is different from **0**, then it belongs to $\mathcal{EDCI}$. Otherwise, it belongs to $\mathcal{ADCI}$. This also does not affect the correctness of the regeneration process since the associated supports of closures belonging to $\mathcal{ADCI}$ will not be used for computing conjunctive supports. They will only ensure checking whether an itemset is infrequent when the smallest closure covering it has a support equal to **0**. Consequently, this solution allows to avoid the computation of its conjunctive support since ensured to be infrequent. This is carried out without affecting the homogeneity of the proposed representation.

B. Case of the $\mathcal{DSSR}$ Representation

In addition to some common properties with the DCIs_rep representation, the $\mathcal{DSSR}$ representation (*cf.* page 119) presents the following interesting features:

1. **Structural characterization of disjunctive equivalence classes:** This is carried out thanks to the elements contained in this representation, namely essential itemsets and their disjunctive closures. This also ensures the **homogeneity** of the representation *w.r.t.* the explored search space.
2. **Low regeneration cost:** In this respect, it offers the same advantage as the DCIs_rep representation (see above).
3. **Optimized storage:** This representation can be stored in a very compact way and without information loss. This is carried out as follows: $\mathcal{DSSR} = \{(e,\ f\backslash e,\ \mathit{Supp}(\vee e)) \mid e \in \mathcal{FEI}$ and $f = h(e) \in \mathcal{EDCI}\}$. Each disjunctive closed pattern, like f, is then simply derivable by getting the union between e and $f\backslash e$.

6.5 The DCPR_Miner Algorithm

In this section, we introduce a new algorithm, called DCPR_Miner, [4] dedicated to the extraction of the disjunctive closed itemset-based representations, namely BDCIs_rep and DCIs_rep. Note that a slight modification of this algorithm makes it possible mining the $\mathcal{DSSR}$ representation. Indeed, this algorithm will be shown to extract both sets composing $\mathcal{DSSR}$, namely $\mathcal{EDCI}$ and $\mathcal{FEI}$. It is hence sufficient to omit the extraction of both sets $\mathcal{ADCI}$ and $\mathcal{BDCI}$.

6.5.1 Description

The disjunctive closed itemsets composing both representations have, for associated seeds, the set $\mathcal{FEI}$ of frequent essential itemsets and a subset of the infrequent part of the associated negative border. Interestingly, these latter seeds form a downward closed set. Thus, a levelwise traversal of the search space is indicated for localizing them without overhead *w.r.t.* those of the negative border [Mannila and Toivonen, 1997]. Indeed, the negative border consists of exactly those itemsets which, on the basis of other information, could be frequent essential, and on which the constraint "to be frequent essential"

[4] DCPR_Miner is the acronym of Disjunctive Closed Pattern-based Representation Miner.

should therefore be checked. The DCPR_MINER algorithm is thus designed to adopt such a traversal technique for localizing the required seeds. Once located, their disjunctive closure will be efficiently derived as explained hereafter. In this respect, the computation of the disjunctive closures of equal-size itemsets can be performed using a unique pass over the extraction context.

According to Definition 62 (*cf.* page 110), a naive method for obtaining the closure of I is to augment it by the items maintaining its disjunctive support unchanged. However, this requires knowing beforehand the disjunctive support of $(I \cup \{i\})$ for each item $i \in \mathcal{I} \backslash I$, which can be very costly. In this situation, DCPR_MINER relies on an efficient method based on an exploitation of the complementary of an itemset *w.r.t.* the set of items of the context. Indeed, the disjunctive closure $h(I)$ of an itemset I is the maximal set of items that *only* appear in the transactions having at least an item of I (*cf.* Definition 61, 110). Hence, we firstly compute the set $\overline{h(I)}$ of items that appear in the objects that does not contain any item of an essential itemset I. Then, by evaluating the set $\mathcal{I} \backslash \overline{h(I)}$, we simply obtain $h(I)$.

Example 45 *Consider the extraction context given by Table 6.1 (cf. page 110). Let us compute the disjunctive closure of the essential itemset* BC. *The item* A *appears in a transaction that does not contain neither* B *nor* C *(cf. transaction* **1***, for example). It is the same for the item* D *(cf. transaction* **4***, for example). Then,* $\overline{h(BC)} = AD$. *Consequently,* $h(BC) = \mathcal{I} \backslash \overline{h(BC)} = ABCD \backslash AD = BC$.

By definition, essential itemsets are the minimal elements in the associated disjunctive equivalence classes (*cf.* Definition 37, page 37). Therefore, they are the first elements from which the disjunctive closures are computed whenever a levelwise traversal of the search space is adopted. Dually, the disjunctive closures can be used to efficiently detect essential itemsets. Indeed, an essential itemset must not be covered by the closure of one of its immediate subsets (*cf.* Proposition 42, page 115). This new characterization of essential itemsets, adopted by DCPR_MINER, allows the detection of essential itemsets without computing their disjunctive supports. Indeed, we only need to have at hand the disjunctive closures of the immediate subsets of an itemset to guess whether it is essential or not.

The pseudo-code of DCPR_MINER is depicted by Algorithm 8, while Table 6.2 summarizes the associated notations. The mining of the disjunctive closures of $\mathcal{EDCI}$ and $\mathcal{BDCI}$, associated to their supports, is carried out by means of the COMPUTE_SUPPORTS_CLOSURES procedure (*cf.* line 4 in Algorithm 8). The pseudo-code of this procedure is given by Algorithm 9. Thanks to one pass over the extraction context, this procedure computes the conjunctive and disjunctive supports of i-candidates as well as the complementary, *w.r.t.* the set of items $\mathcal{I}$, of their associated disjunctive closed itemsets. Then, it deduces the disjunctive closures of frequent candidates from their complementary and inserts them in $\mathcal{EDCI}$ (*cf.* line 16 in Algorithm 9). While the disjunctive closures of odd-sized infrequent itemsets are added to $\mathcal{BDCI}$ (*cf.* line 20).

The generation of $(i + \mathbf{1})$-candidates is performed by the APRIORI-GEN procedure [Agrawal and Srikant, 1994], applied on the retained i-frequent essential itemsets (*cf.* line 5 in Algorithm 8). The next instruction (*cf.* line 6) ensures that each element of $\mathcal{C}_{(i+1)}$ has all its immediate subsets as frequent essential itemsets. For this purpose, a candidate having an immediate subset which is not a frequent essential itemset is withdrawn. While pruning non-essential itemsets from $\mathcal{C}_{(i+1)}$ is performed thanks to the characterization of essential itemsets using disjunctive closures. Indeed, Proposition 42 allows the pruning of a candidate which is included in the disjunctive closure of one of its immediate subsets, since

Notation	Description
$\mathcal{C}_i$ (*resp.* $\mathcal{L}_i$)	: Set of candidate (*resp.* frequent) essential itemsets of size i.
X_i	: Itemset of size i.
$X_i.h$	: Disjunctive closure of X_i.
$X_i.\overline{h}$	: Complementary of $X_i.h$ *w.r.t.* $\mathcal{I}$ (*i.e.*, $X_i.\overline{h} = \mathcal{I} \setminus X_i.h$).
$X_i.Conj\_Supp$ (*resp.* $X_i.Disj\_Supp$)	: Conjunctive (*resp.* disjunctive) support of X_i.

Table 6.2: Notations used by the DCPR_Miner algorithm.

Algorithm 8: DCPR_Miner

Input: - An extraction context $\mathcal{K}$, and the minimum threshold of support *minsupp*.

Output: - The exact concise representation DCI*s_rep* = $\mathcal{EDCI} \cup \mathcal{ADCI}$.

```
1  Begin
2    EDCI := {(∅, |O|)}; BDCI := ∅; i := 1; C1 := I;
3    While (Ci ≠ ∅) Do
4      Compute_Supports_Closures(K, minsupp, Ci, Li, EDCI, BDCI);
5      C(i+1) := Apriori-Gen(Li);
6      C(i+1) := {X(i+1) ∈ C(i+1) | ∀ Yi ⊂ X(i+1), Yi ∈ Li and X(i+1) ⊄ Yi.h};
7      i := i + 1;
8    ADCI := {X ∈ BDCI | (X ∉ EDCI) and (∃ Y ∈ EDCI, X ⊂ Y)};
9    Return DCIs_rep;
10 End
```

it is necessarily not an essential itemset.

Finally, the set $\mathcal{ADCI}$ is derived from both sets $\mathcal{BDCI}$ and $\mathcal{EDCI}$, as shown by line 8 in Algorithm 8.

Example 46 *Consider the context depicted by Table 6.1 and let minsupp* = **1**. *First,* DCPR_Miner *initializes* $\mathcal{EDCI}$ *to the couple* $\{(\emptyset, 7)\}$ *added to ensure the exact regeneration of the empty set conjunctive support, while* $\mathcal{BDCI}$, i *and* $\mathcal{C}_1$ *are respectively set to* $\emptyset$, 1 *and* $\mathcal{I}$. *Then, it iterates over the context to compute the conjunctive and disjunctive supports as well as disjunctive closures of* **1***-essential candidates,* **2***-essential candidates, and so on. The iteration process stops whenever the candidate set is found to be empty.*

Initially, DCPR_Miner *considers the set* $\mathcal{C}_1$ *of* **1***-essential candidates. For these itemsets, the conjunctive and disjunctive supports as well as the complementary of their associated disjunctive closures are computed by the* Compute_Supports_Closures *procedure thanks to an access to the context. The result of this access is shown in Table 6.3 (Left). Then, this procedure constructs the set* $\mathcal{L}_1$ *containing frequent* **1***-essential itemsets. It also deduces their closures starting from their respective complementary. These closures will be included in* $\mathcal{EDCI}$, *since all items are frequent. Table 6.3 (Right) shows the result of this step.*

Algorithm 9: COMPUTE_SUPPORTS_CLOSURES

Input: - A context $\mathcal{K}$, the minimum support threshold *minsupp*, and the set of candidate essential itemsets $\mathcal{C}_i$.

Output: - The set of frequent essential itemsets $\mathcal{L}_i$, and the updated sets $\mathcal{EDCI}$ and $\mathcal{BDCI}$.

```
1  Begin
2    L_i := ∅;
3    ForEach (o ∈ O) Do
4      ForEach (X_i ∈ C_i) Do
5        Ω := X_i ∩ I /*I denotes the items associated to the object o.*/;
6        If (Ω = ∅) Then
7          X_i.h̄ := X_i.h̄ ∪ I;
8        Else
9          X_i.Disj_Supp := X_i.Disj_Supp + 1;
10         If (Ω = X_i) Then
11           X_i.Conj_Supp := X_i.Conj_Supp + 1;
12   ForEach (X_i ∈ C_i) Do
13     If (X_i.Conj_Supp ≥ minsupp) Then
14       L_i := L_i ∪ {X_i};
15       X_i.h := I\X_i.h̄;
16       EDCI := EDCI ∪ {(X_i.h, X_i.Disj_Supp)};
17     Else
18       If (i is odd) Then
19         X_i.h := I\X_i.h̄;
20         BDCI := BDCI ∪ {(X_i.h, X_i.Disj_Supp)};
21 End
```

Thus, $\mathcal{L}_1 = \mathcal{C}_1 =$ {A, B, C, D}, $\mathcal{EDCI}$ = {(∅, 7), (B, 3), (C, 3), (D, 3), (ABCD, 7)} *and* $\mathcal{BDCI}$ = ∅. *Then,* DCPR_MINER *generates the set* $\mathcal{C}_2$ *of the next iteration thanks to the* APRIORI-GEN *procedure. After this step,* $\mathcal{C}_2$ = {AB, AC, AD, BC, BD, CD}. *In order to only retain essential itemsets in* $\mathcal{C}_2$, *the instruction of line 6 is executed to prune the candidates included in the disjunctive closure of one of their respective immediate subsets. The candidates* AB, AC *and* AD *will hence be pruned from* $\mathcal{C}_2$ *since included in the closure of* A, *namely* ABCD. *This latter set is then reduced to* {BC, BD, CD}, *and the* COMPUTE_SUPPORTS_CLOSURES *procedure will then handle its elements. The result of the access step is shown in Table 6.4 (Left). After this step, the construction of the sets* $\mathcal{L}_2$, $\mathcal{EDCI}$ *and* $\mathcal{BDCI}$ *is performed. Since the size of these candidates is even, the set* $\mathcal{BDCI}$ *remains unchanged. However, the*

sets $\mathcal{L}_2$ and $\mathcal{EDCI}$ will be updated. This step is shown in Table 6.4 (Right). Thus, $\mathcal{L}_2 = \{$BC, BD, CD$\}$. The set $\mathcal{EDCI}$ is augmented by $\{$(BC, 5), (BD, 5), (CD, 5)$\}$.

After that, the APRIORI-GEN *procedure is called in order to generate the set $\mathcal{C}_3$ equal to $\{$BCD$\}$. Since BCD is not included in any closure of its immediate subsets, then it is an essential itemset. A third access to the context is then required. The access output is given by Table 6.5. The set $\mathcal{EDCI}$ remains unchanged, while the set $\mathcal{BDCI} = \{$(BCD, 6)$\}$.*

	Access step			**Construction step**			
X_1	$Conj\_Supp$	$Disj\_Supp$	$\overline{h}$	$X_1 \in \mathcal{FI}$?	h	$X_1.h \in \mathcal{EDCI}$?	$X_1.h \in \mathcal{BDCI}$?
A	7	7	$\emptyset$	yes	ABCD	yes	no
B	3	3	ACD	yes	B	yes	no
C	3	3	ABD	yes	C	yes	no
D	3	3	ABC	yes	D	yes	no

Table 6.3: The access (Left) and the construction (Right) steps for the first iteration.

	Access step			**Construction step**			
X_2	$Conj\_Supp$	$Disj\_Supp$	$\overline{h}$	$X_2 \in \mathcal{FI}$?	h	$X_2.h \in \mathcal{EDCI}$?	$X_2.h \in \mathcal{BDCI}$?
BC	1	5	AD	yes	BC	yes	no
BD	1	5	AC	yes	BD	yes	no
CD	1	5	AB	yes	CD	yes	no

Table 6.4: The access (Left) and the construction (Right) steps for the second iteration.

	Access step			**Construction step**			
X_3	$Conj\_Supp$	$Disj\_Supp$	$\overline{h}$	$X_3 \in \mathcal{FI}$?	h	$X_3.h \in \mathcal{EDCI}$?	$X_3.h \in \mathcal{BDCI}$?
BCD	0	6	A	no	BCD	no	yes

Table 6.5: The access (Left) and the construction (Right) steps for the third iteration.

The iteration process ends since there is no size **4** *candidate. The construction of the set $\mathcal{ADCI}$ then begins starting from $\mathcal{BDCI}$. Since BCD is covered by an element of $\mathcal{EDCI}$, namely ABCD, then $\mathcal{ADCI} = \{$(BCD, 3)$\}$.*

Finally, the DCPR_MINER *algorithm outputs the exact representation* DCI$s_rep = \{$(∅, 7), (B, 3), (C, 3), (D, 3), (BC, 5), (BD, 5), (CD, 5), (BCD, 6), (ABCD, 7)$\}$.

By analyzing the sets built by the DCPR_MINER algorithm, we can point out that it operates in two generic steps. The first one, called EXTRACTION, consists in extracting the elements of the sets $\mathcal{EDCI}$ and $\mathcal{BDCI}$ associated to their disjunctive supports from the extraction context (*cf.* lines 2-7). The second step, called COVER_TEST, consists in constructing the set $\mathcal{ADCI}$ by only maintaining the elements of $\mathcal{BDCI}$ covered by a disjunctive closure of $\mathcal{EDCI}$ (*cf.* line 8). Consequently, DCPR_MINER can easily be adapted either to the extraction of the representation BDCIs_rep or DCIs_rep. For this purpose, we only need to omit the COVER_TEST step to return the former representation, while this step should be retained to get the latter one. Thus, users interested in obtaining better performances can omit the

COVER_TEST step, while users that prefer high compactness rates can execute the whole algorithm.

6.5.2 Correctness and Complexity

The following theorem ensures the soundness and the correctness of the DCPR_MINER algorithm.

Theorem 16 *The* DCPR_MINER *algorithm is sound and correct. It exactly extracts all the closures belonging to* DCI*s_rep, associated to their disjunctive supports.*

Proof. The conjunction of two anti-monotone constraints, namely "to be frequent" and "to be essential", is also anti-monotone. Hence, a levelwise algorithm like DCPR_MINER guarantees that all frequent essential itemsets are extracted as well as the associated negative border [Mannila and Toivonen, 1997]. For each candidate essential itemset, the algorithm also computes the items that cannot belong to its disjunctive closures. It thus allows a correct derivation of the required disjunctive closed itemsets respectively forming $\mathcal{EDCI}$ and $\mathcal{BDCI}$. After that, a cover test allows only retaining in $\mathcal{BDCI}$ the closures which are at least covered by an element of $\mathcal{EDCI}$. This step gives the set $\mathcal{ADCI}$. Thus, DCPR_MINER is sound and correct. $\diamondsuit$

Proposition 44 gives the theoretical complexity of the DCPR_MINER algorithm.

Proposition 44 *The worst case complexity of the* EXTRACTION *step is bounded by* $O((n^2+m\times n)\times 2^n)$, *while the worst case complexity of the* COVER_TEST *step is bounded by* $O(n \times \binom{n}{\lceil \frac{n}{2} \rceil} \times 2^n)$, *where* $n = |\mathcal{I}|$ *and* $m = |\mathcal{O}|$. *The theoretical complexity of* DCPR_MINER *is bounded by the sum of those of its two steps.*

Proof. First of all, let us recall the respective role of both distinct steps of the DCPR_MINER algorithm. The first, namely EXTRACTION, mines the closures belonging respectively to $\mathcal{EDCI}$ and $\mathcal{BDCI}$. While the second step, namely COVER_TEST, derives $\mathcal{ADCI}$ from $\mathcal{BDCI}$.

- **Complexity of the EXTRACTION step:** The theoretical complexity of this step is equal to those of its associated instructions which are as follows:
  1. The cost of the initializations carried out in line 2 (*cf.* Algorithm 8, page 125) is in $O(1)$.
  2. The worst case complexity of the COMPUTE_SUPPORTS_CLOSURES procedure (*cf.* line 4) is reached whenever any set of items appears at least once in the context, and each candidate is a frequent essential itemset. There are hence $\mathbf{2}^n$ - **1** frequent essential itemsets, equal to their respective closures. The cost of this procedure is then as follows:
     (a) The cost of the initialization of the set $\mathcal{L}_i$ for $i = 1..n$ is in $O(n)$ (*cf.* line 2 in Algorithm 9, page 126).
     (b) The cost of the computation of the disjunctive and conjunctive supports of itemset candidates X_i as well as the complementary of $X_i.h$, namely $X_i.\overline{h}$, is bounded by $O((m \times n) \times \mathbf{2}^n)$ (*cf.* Algorithm 9, lines 3-11).
     (c) The cost of the pruning of candidates *w.r.t.* the minimum support threshold *minsupp* as well as the construction of disjunctive closures starting from their complementary is in $O(n \times \mathbf{2}^n)$ (*cf.* Algorithm 9, lines 12-20).

3. There are, in the worst case, $\mathbf{2}^n$ - n - $\mathbf{1}$ candidates to be generated using the APRIORI-GEN procedure (*cf.* Algorithm 8, line 5). The cost of this step is in $O(\mathbf{2}^n$ - $n)$.
4. The cost of pruning candidates through the characterization of essential itemsets using disjunctive closures (*cf.* Algorithm 8, line 6) is in $O(n^2 \times (\mathbf{2}^n$ - $n))$.
5. The cost of the incrementation of i from 1 to n is in $O(n)$ (*cf.* Algorithm 8, line 7).

Consequently, the cost of this step is bounded by $O(n + (m \times n) \times \mathbf{2}^n + n \times \mathbf{2}^n + \mathbf{2}^n$ - $n + n^2 \times (\mathbf{2}^n$ - $n) + n) = O((m \times n + n^2 + n + \mathbf{1}) \times \mathbf{2}^n$ - $n^3 + n) = O((n^2 + m \times n) \times \mathbf{2}^n)$.

In the worst case, the complexity of the EXTRACTION step is then bounded by $O((n^2 + m \times n) \times \mathbf{2}^n)$.

- **Complexity of the COVER_TEST step:** In this step, each closure of $\mathcal{BDCI}$ is checked whether it is covered by at least an element of $\mathcal{EDCI}$ (*cf.* line 8 in Algorithm 8). In the affirmative case, it will belong to $\mathcal{ADCI}$. The complexity of this step is then at most equal to $O(n \times |\mathcal{BDCI}| \times |\mathcal{EDCI}|)$. We will now assess the size of both sets, *i.e.*, $\mathcal{EDCI}$ and $\mathcal{BDCI}$, in the worst case.

  The set $\mathcal{BDCI}$ gathers the closures of odd-sized infrequent essential itemsets belonging to $\mathcal{B}d^-(\mathcal{FEI})$. Its size is hence bounded by that of $\mathcal{B}d^-(\mathcal{FEI})$. The cardinality of this latter border is at most equal to $\binom{n}{\lceil \frac{n}{2} \rceil}$.

  Now, we assess the size in the worst case of the set $\mathcal{EDCI}$. For this purpose we suppose that each essential itemset is equal to its closure. Hence, we have at most $\mathbf{2}^n$ - $\mathbf{1}$ disjunctive closures. The size of $\mathcal{EDCI}$ is thus at most equal to $\mathbf{2}^n$ - $\mathbf{1}$.

  In the worst case, the complexity of the COVER_TEST step is hence bounded by $O(n \times \binom{n}{\lceil \frac{n}{2} \rceil} \times \mathbf{2}^n)$.

$\diamondsuit$

It is important to mention that the complexity in the worst case of the COVER_TEST step is not reachable in practice. Indeed, there is not a context that maximizes at the same time the size of $\mathcal{EDCI}$ and that of $\mathcal{BDCI}$ to reach the bounds we used in our computation. In addition, there is not a context that simultaneously gives the respective worst case theoretical complexities of the DCPR_MINER steps. Hence, the worst case complexity of DCPR_MINER is roughly bounded by the sum of those of its two steps.

6.6 Experimental Results

In this section, our objective is to show, through extensive experiments, that our concise representation provides interesting compactness rates compared respectively to the representations based on frequent closed itemsets, frequent (closed) non-derivable itemsets and frequent essential itemsets. Note that from a performance point of view, the disjunctive closed itemsets were shown in [Denden *et al.*, 2008] to be efficiently extracted using the DCPR_MINER algorithm.

The experiments were carried out on benchmark contexts (*cf.* Appendix A for a detailed description of these contexts). All experiments were carried out on a PC equipped with a 3GHz Pentium (R)

and 1.75GB of main memory, running the GNU/Linux distribution Fedora Core 7 (with 2GB of swap memory). In order to extract the aforementioned concise representations, we used the source codes of the following algorithms:

1. DCPR_MINER is used in order to extract the representation based on disjunctive closed itemsets.
2. LCM [Uno *et al.*, 2004] was applied to extract frequent (closed) itemsets. [5]
3. NDI [Calders and Goethals, 2007] allows the extraction of frequent non-derivable itemsets. [6]
4. FIRM is used in order to extract frequent closed non-derivable itemsets. [7]
5. MEP [Casali *et al.*, 2005a] extracts the representation based on frequent essential itemsets. [8]

Obtained results are presented as follows. Table 6.6 and Table 6.7 compare the size of $\mathcal{EDCI}$ (*resp.* $\mathcal{ADCI}$) with that of $\mathcal{FEI}$ (*resp.* $\mathcal{B}d^{+}(\mathcal{FI})$) on dense and sparse contexts, respectively. Both tables also compare the size of $\mathcal{ADCI}$ (*resp.* $\mathcal{B}d^{+}(\mathcal{FI})$) to that of $\mathcal{EDCI}$ (*resp.* $\mathcal{FEI}$). The purpose of these comparisons is to highlight the effect of the added part (*i.e.*, $\mathcal{ADCI}$ and $\mathcal{B}d^{+}(\mathcal{FI})$) *vs.* that of the core part (*i.e.*, $\mathcal{EDCI}$ and $\mathcal{FEI}$) on the size of the associated representation. Note that the symbol "/" indicates that a ratio cannot be computed, since the size of $\mathcal{ADCI}$ is equal to **0**. Figure 6.2 and Figure 6.4 graphically sketch the obtained results for dense and sparse contexts, respectively.

Table 6.8 and Table 6.10 compare the size of our concise representation DCIs_rep to those of the literature as well as to that of the set of frequent itemsets, respectively on dense and sparse contexts. In this respect, Table 6.9 and Table 6.11 present the compactness rates offered by our representation in comparison to the whole set of frequent itemsets and to the other representations. Obtained results are also graphically sketched by Figure 6.3 and Figure 6.5 for dense and sparse contexts, respectively. For these tables, the abbreviation "$\mathcal{FI}$" (*resp.* "FCIs_rep", "NDIs_rep", "CNDIs_rep", and "FEIs_rep") is used to stand for the set of frequent itemsets (*resp.* frequent closed, frequent non-derivable, frequent closed non-derivable and frequent essential itemset-based representation). We also use the symbol "-" to designate a case where an execution error occurred. For example, to show the cardinality of CNDIs_rep, the authors of [Muhonen and Toivonen, 2006] have chosen a specific interval of *minsupp* values for some contexts also used in our tests. Nevertheless, beyond these intervals, we noticed that their program comes to an end with an execution error.

In the literature dedicated to concise representations of frequent itemsets (*e.g.*, [Boulicaut *et al.*, 2003, Calders *et al.*, 2005]), it was shown that dense contexts present the most interesting cases. Indeed, within such contexts, the compactness ratio between the size of the set of frequent itemsets and those of concise representations is high. On the contrary, equivalence classes extracted from sparse contexts are often reduced to the associated generators and cannot be further compacted. The number of extracted frequent itemsets is hence small even for low *minsupp* values. This makes the size reduction rates brought by concise representations meaningless in such contexts. The next paragraphs give a thorough analysis of the obtained results.

[5] The source code of LCM is available at: *http://research.nii.ac.jp/~uno/code/lcm50.zip*.
[6] The source code of NDI is available at: *http://www.adrem.ua.ac.be/~goethals/software/files/ndi.tgz*.
[7] The source code of FIRM is available at: *http://www.cs.helsinki.fi/u/jomuhone/firm/firm-3-3-3.tar.gz*.
[8] The source code of MEP was kindly provided by its authors.

1. DCIs_rep *vs.* $\mathcal{FI}$:

1.1. Results on dense contexts: For the different *minsupp* values, the compactness rate offered by our concise representation DCIs_rep, *w.r.t.* the size of $\mathcal{FI}$, is considerably high (*cf.* Table 6.9). For example, it reaches almost **1, 116, 705** times for CONNECT with *minsupp* = **20%**. This clearly shows the necessity to set up concise representations for such type of contexts.

1.2. Results on sparse contexts: The size of DCIs_rep is lower than or equal to that of $\mathcal{FI}$ for the different contexts. However, the obtained results confirm that the compactness rates offered by the pioneer concise representations of the literature are often low on such datasets. Indeed, the size of DCIs_rep is almost equal to that of $\mathcal{FI}$ for different sparse datasets, such as KOSARAK and T10I4D100K. This makes the associated curves collapse (*cf.* Figure 6.5 (Left)). Interestingly, for the ACCIDENTS dataset, we note a reduction reaching **8.23** for *minsupp* = **20%** (*cf.* Table 6.11).

2. DCIs_rep *vs.* FEIs_rep:

2.1. Results on dense contexts: The size of DCIs_rep is always smaller than that of FEIs_rep (*cf.* Table 6.9). Considering Table 6.6, the cardinality of $\mathcal{EDCI}$ is always lower than that of $\mathcal{FEI}$. By comparing the respective cardinalities of $\mathcal{ADCI}$ and $\mathcal{B}d^{+}(\mathcal{FI})$, we note that the associated ratio reaches high values whenever the size of $\mathcal{ADCI}$ is smaller than that of $\mathcal{B}d^{+}(\mathcal{FI})$. This occurs for the CONNECT, CHESS and PUMSB contexts which explains why our representation is largely smaller than FEIs_rep on these contexts. For MUSHROOM, $\mathcal{ADCI}$ is smaller than $\mathcal{B}d^{+}(\mathcal{FI})$ for high *minsupp* values while it is the opposite for low ones, although the ratio values are too small. It is also the opposite for PUMSB* *w.r.t. minsupp* values while preserving the low ratio values.

When comparing the size of $\mathcal{ADCI}$ to that of $\mathcal{EDCI}$, we note that the former only constitutes at most **0.30** that of the latter. For the different contexts, the ratios also decrease proportionally to the decrease of *minsupp* values (*cf.* Table 6.6). This clearly shows that, in addition to ensuring the homogeneity of the representation, $\mathcal{ADCI}$ is very compact. While the size of $\mathcal{B}d^{+}(\mathcal{FI})$ often exceeds and reaches **20.48** times that of $\mathcal{FEI}$ (*cf.* Table 6.6). Thus, in addition to the heterogeneity caused by this border, its size for dense contexts makes the representation based on frequent essential itemsets, very large.

2.2. Results on sparse contexts: As for dense contexts, the size of DCIs_rep is always smaller than that of FEIs_rep (*cf.* Table 6.11). By comparing the respective size of the couple of sets constituting each representation, we notice that the size of $\mathcal{EDCI}$ is equal to that of $\mathcal{FEI}$ for the KOSARAK, RETAIL and T40I10D100K contexts. While its size is slightly reduced for T10I4D100K for very low *minsupp* values and ACCIDENTS for all *minsupp* values. Consequently, for the KOSARAK and T10I4D100K contexts, the curves representing the size of the sets $\mathcal{EDCI}$ and $\mathcal{FEI}$ collapse (*cf.* Figure 6.4 (Left)). In fact, for sparse contexts, the main advantage of our representation is that it avoids the use of elements from the conjunctive search space contrary to FEIs_rep, which heavily relies on $\mathcal{B}d^{+}(\mathcal{FI})$. This border clearly increases the size of FEIs_rep. For example, the size of $\mathcal{B}d^{+}(\mathcal{FI})$ reaches **13, 933.50** times the size of $\mathcal{ADCI}$ (*cf.* Table 6.7). It is worth noting that this latter set is almost empty for

all contexts, except ACCIDENTS. Indeed, its size for the other four contexts does not exceed **10**. In the figures associated to the KOSARAK and T40I10D100K datasets, only the curve representing the size of $\mathcal{B}d^{+}(\mathcal{FI})$ appears (*cf.* Figure 6.4 (Right)) since the size of $\mathcal{ADCI}$ is always equal to 0.

In comparison to dense contexts, the size of $\mathcal{B}d^{+}(\mathcal{FI})$ is more reduced for sparse ones. However, even for these latter contexts, $\mathcal{ADCI}$ offers better compactness rates *w.r.t.* the associated representation. Indeed, the size of $\mathcal{ADCI}$ does not exceed **0.11** for the ACCIDENTS context while decreasing whenever *minsupp* values lowered (*cf.* Table 6.7). As mentioned above, for the other four contexts, its size is equal (or almost equal) to **0** *w.r.t.* that of $\mathcal{EDCI}$. On the other hand, the size of $\mathcal{B}d^{+}(\mathcal{FI})$ can be even equal to that of $\mathcal{FEI}$ while being omnipresent for almost all contexts and especially RETAIL, T10I4D100K and T40I10D100K (*cf.* Table 6.7).

3. DCI*s_rep* *vs.* FCI*s_rep*, NDI*s_rep* and CNDI*s_rep*:

3.1. Results on dense contexts: For the CHESS, CONNECT and PUMSB contexts, the cardinality of DCI*s_rep* is significantly reduced compared to those of the other representations. It is also the case for the PUMSB* context *w.r.t.* FCI*s_rep* and FEI*s_rep*. Nevertheless, for the MUSHROOM context, the size of DCI*s_rep* is quite greater than the size of FCI*s_rep* and NDI*s_rep* for *minsupp* values lower than **20%**.
For the different contexts, the program allowing the extraction of CNDI*s_rep* comes to an end with an execution error for low *minsupp* values. This can be explained by the very high memory space used by this program when computing the associated closures of non-derivable itemsets. Indeed, all these latter itemsets are maintained in memory throughout this step of the execution. It is also worth noting that DCI*s_rep* is, in most cases, less sensitive to the variation of *minsupp* values than the other concise representations (*cf.* Figure 6.3).

3.2. Results on sparse contexts: Our representation is the smallest one for the ACCIDENTS context. For the remaining contexts, its size is almost equal to that of FCI*s_rep* while it is greater than those of NDI*s_rep* and, consequently, CNDI*s_rep*. In this respect, two remarks are noteworthy: (*i*) The size of CNDI*s_rep* is almost equal to that of NDI*s_rep*. Hence, in such contexts, computing the closed itemsets associated to non-derivable ones to obtain CNDI*s_rep* is often useless, since each itemset is equal to its closure. (*ii*) To belong to NDI*s_rep*, an itemset I must have a support not *exactly* derivable using the deduction rules based on the conjunctive supports of *all* its subsets [Calders and Goethals, 2007]. The main advantage of NDI*s_rep* is then brought by the large neighborhood explorations to retain or not an itemset within the representation. While in our cases, DCI*s_rep* relies on taking closures of essential itemsets. These latter itemsets are based on a simple comparison of their support with those of their *immediate* subsets. An important question is then: what will be the effect of enlarging the neighborhood for essential itemsets on the compactness rates?

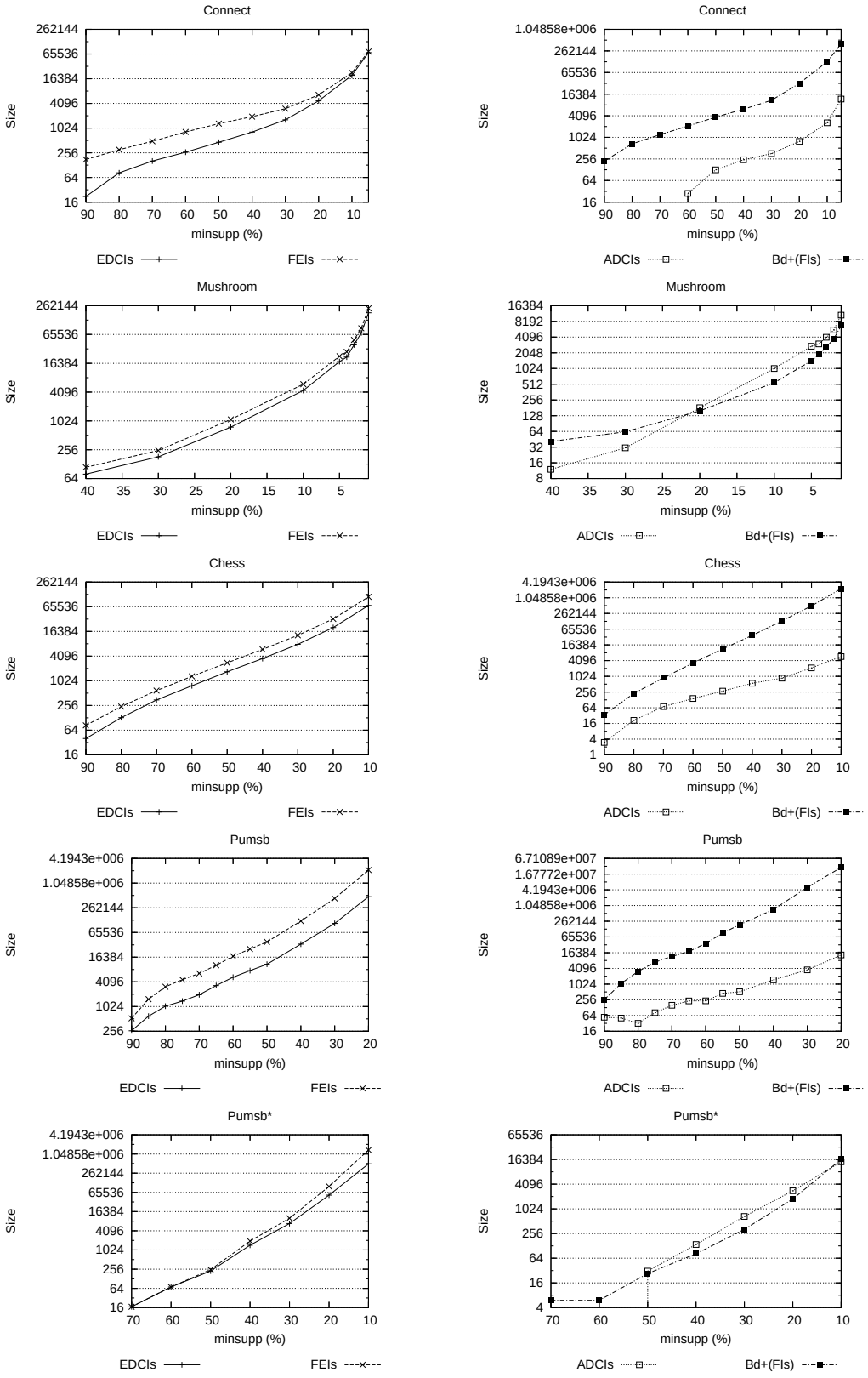


Figure 6.2: Size of $\mathcal{EDCI}$ *vs.* $\mathcal{FEI}$ (**Left**), and $\mathcal{ADCI}$ *vs.* $\mathcal{B}d^{+}(\mathcal{FI})$ (**Right**) for dense contexts.

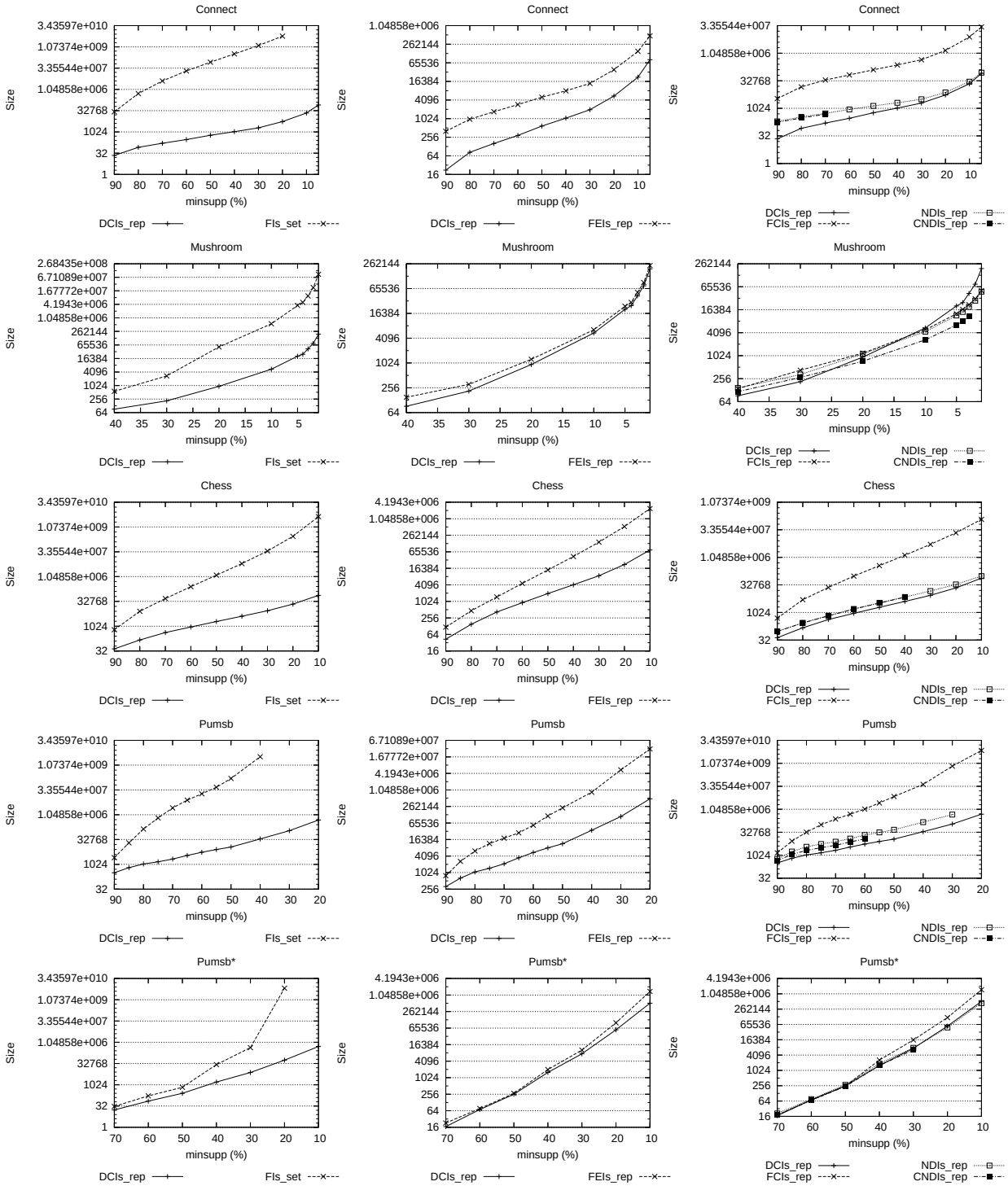


Figure 6.3: Size of DCI*s_rep vs.* the whole set of frequent itemsets (**Left**), FEI*s_rep* (**Middle**), and the remaining representations (**Right**) for dense contexts.

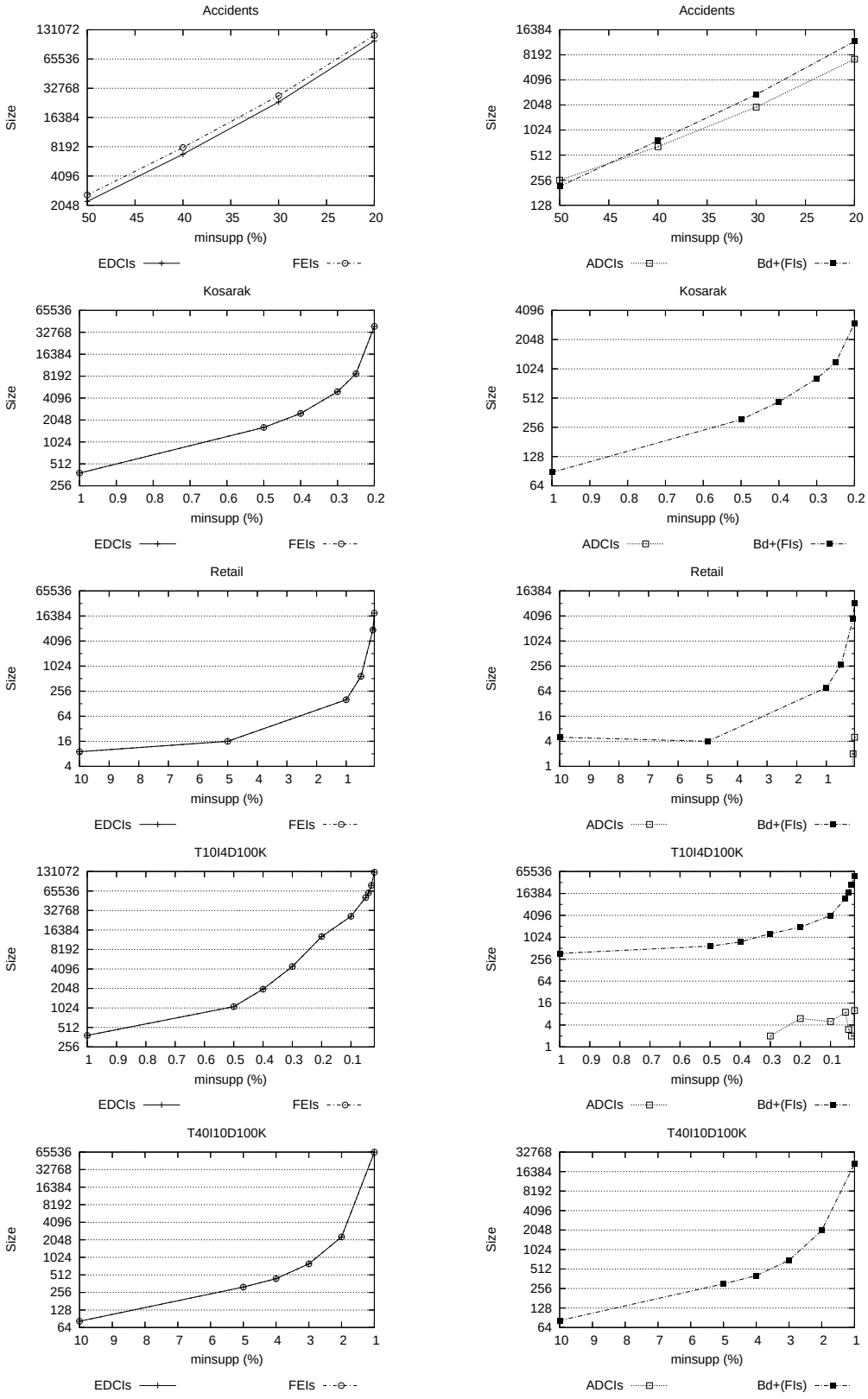


Figure 6.4: Size of $\mathcal{EDCI}$ *vs.* $\mathcal{FEI}$ (**Left**), and $\mathcal{ADCI}$ *vs.* $\mathcal{B}d^{+}(\mathcal{FI})$ (**Right**) for sparse contexts.

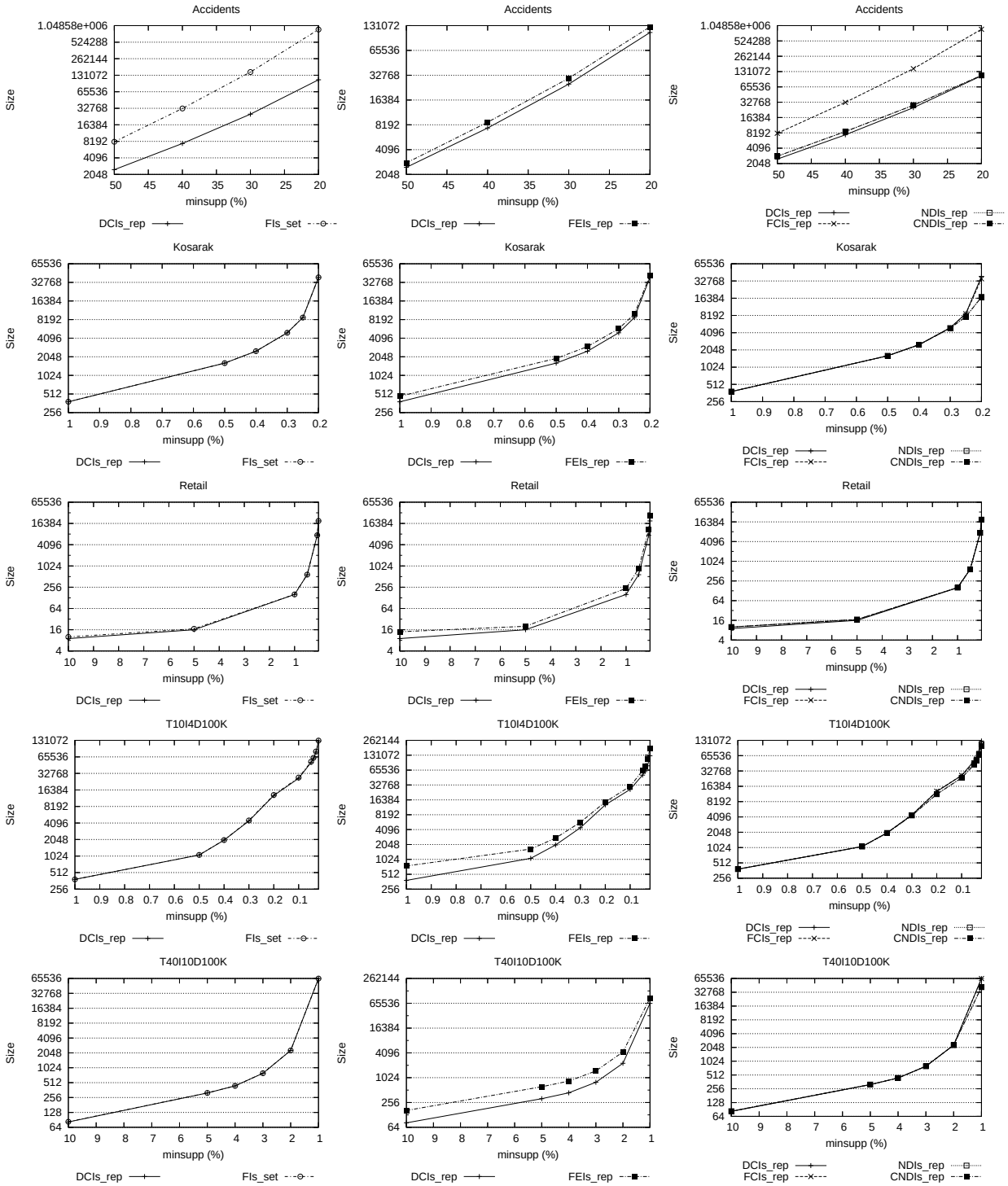


Figure 6.5: Size of DCI*s_rep vs.* the whole set of frequent itemsets (**Left**), FEI*s_rep* (**Middle**), and the remaining representations (**Right**) for sparse contexts.

minsupp	|**DCI**s_rep|		|**FEI**s_rep|		**Ratios**			
(%)	$\lvert\mathcal{EDCI}\rvert$	$\lvert\mathcal{ADCI}\rvert$	$\lvert\mathcal{FEI}\rvert$	$\lvert\mathcal{Bd}^+(\mathcal{FI})\rvert$	$\frac{\lvert\mathcal{FEI}\rvert}{\lvert\mathcal{EDCI}\rvert}$	$\frac{\lvert\mathcal{Bd}^+(\mathcal{FI})\rvert}{\lvert\mathcal{ADCI}\rvert}$	$\frac{\lvert\mathcal{ADCI}\rvert}{\lvert\mathcal{EDCI}\rvert}$	$\frac{\lvert\mathcal{Bd}^+(\mathcal{FI})\rvert}{\lvert\mathcal{FEI}\rvert}$
CONNECT								
90	23	0	177	222	**8.00**	/	**0.00**	**1.26**
80	84	0	305	673	**3.66**	/	**0.00**	**2.21**
70	162	0	491	1, 220	**3.04**	/	**0.00**	**2.49**
60	266	28	823	2, 103	**3.10**	**75.11**	**0.11**	**2.56**
50	463	127	1, 316	3, 748	**2.85**	**29.51**	**0.27**	**2.85**
40	820	243	1, 949	6, 213	**2.38**	**25.57**	**0.30**	**3.19**
30	1, 626	361	3, 045	11, 039	**1.87**	**30.58**	**0.22**	**3.63**
20	4, 726	789	6, 621	32, 583	**1.40**	**41.30**	**0.17**	**4.92**
10	19, 798	2, 603	22, 943	130, 986	**1.16**	**50.32**	**0.13**	**5.71**
5	70, 799	11, 940	75, 346	413, 053	**1.06**	**34.59**	**0.17**	**5.48**
MUSHROOM								
40	80	12	111	41	**1.39**	**3.42**	**0.15**	**0.37**
30	183	31	248	63	**1.36**	**2.03**	**0.17**	**0.26**
20	760	182	1, 101	158	**1.45**	**0.87**	**0.24**	**0.14**
10	4, 433	1, 025	5, 984	547	**1.35**	**0.53**	**0.23**	**0.09**
5	17, 815	2, 740	22, 966	1, 442	**1.29**	**0.53**	**0.15**	**0.06**
4	22, 128	3, 033	28, 317	1, 918	**1.28**	**0.63**	**0.14**	**0.07**
3	39, 723	4, 069	50, 553	2, 628	**1.27**	**0.65**	**0.10**	**0.05**
2	70, 845	5, 591	88, 507	3, 761	**1.25**	**0.67**	**0.08**	**0.04**
1	186, 274	10, 782	230, 475	6, 768	**1.24**	**0.63**	**0.06**	**0.03**
CHESS								
90	41	3	85	34	**2.10**	**11.33**	**0.08**	**0.40**
80	130	21	242	226	**1.87**	**10.76**	**0.16**	**0.94**
70	350	71	592	891	**1.69**	**12.55**	**0.20**	**1.51**
60	774	144	1, 315	3, 323	**1.70**	**23.08**	**0.19**	**2.53**
50	1, 696	276	2, 810	11, 463	**1.66**	**41.53**	**0.16**	**4.08**
40	3, 564	555	5, 978	38, 050	**1.68**	**68.56**	**0.16**	**6.37**
30	7, 958	867	13, 154	134, 624	**1.65**	**155.28**	**0.11**	**10.24**
20	20, 369	2, 149	33, 186	509, 355	**1.63**	**237.02**	**0.11**	**15.35**
10	70, 355	5, 844	114, 220	2, 339, 525	**1.62**	**400.33**	**0.08**	**20.48**
PUMSB								
90	264	55	530	259	**2.01**	**4.71**	**0.21**	**0.49**
85	597	51	1, 546	1, 083	**2.60**	**21.24**	**0.09**	**0.70**
80	1, 048	32	3, 107	3, 145	**2.97**	**98.28**	**0.03**	**1.01**
75	1, 388	82	4, 632	7, 076	**3.34**	**86.30**	**0.06**	**1.53**
70	1, 986	158	6, 582	11, 737	**3.32**	**74.28**	**0.08**	**1.78**
65	3, 319	233	10, 413	18, 179	**3.14**	**78.02**	**0.07**	**1.75**
60	5, 317	234	17, 257	37, 388	**3.25**	**159.78**	**0.04**	**2.17**
55	7, 694	451	25, 901	92, 221	**3.37**	**203.58**	**0.06**	**3.56**
50	11, 029	523	38, 643	193, 939	**3.50**	**370.82**	**0.05**	**5.02**
40	34, 086	1, 490	124, 719	741, 009	**3.66**	**497.32**	**0.04**	**5.94**
30	108, 851	3, 579	442, 793	5, 198, 357	**4.07**	**1, 452.46**	**0.03**	**11.74**
20	487, 187	13, 319	2, 182, 184	30, 222, 301	**4.48**	**2, 269.11**	**0.03**	**13.85**
PUMSB*								
70	18	0	18	6	**1.00**	/	**0.00**	**0.35**
60	72	0	72	6	**1.00**	/	**0.00**	**0.08**
50	225	31	249	27	**1.11**	**0.87**	**0.14**	**0.11**
40	1, 446	138	1, 947	82	**1.35**	**0.59**	**0.10**	**0.04**
30	6, 889	667	10, 076	324	**1.46**	**0.49**	**0.10**	**0.03**
20	53, 762	2, 826	100, 499	1, 786	**1.87**	**0.63**	**0.05**	**0.02**
10	513, 640	14, 329	1, 397, 666	16, 437	**2.72**	**1.15**	**0.03**	**0.01**

Table 6.6: Size of $\mathcal{EDCI}$ *vs.* $\mathcal{FEI}$ and $\mathcal{ADCI}$ *vs.* $\mathcal{Bd}^+(\mathcal{FI})$ for dense contexts.

minsupp	$\|\mathbf{DCI}s\_rep\|$		$\|\mathbf{FEI}s\_rep\|$		**Ratios**			
(%)	$\|\mathcal{EDCI}\|$	$\|\mathcal{ADCI}\|$	$\|\mathcal{FEI}\|$	$\|\mathcal{B}d^+(\mathcal{FI})\|$	$\frac{\|\mathcal{FEI}\|}{\|\mathcal{EDCI}\|}$	$\frac{\|\mathcal{B}d^+(\mathcal{FI})\|}{\|\mathcal{ADCI}\|}$	$\frac{\|\mathcal{ADCI}\|}{\|\mathcal{EDCI}\|}$	$\frac{\|\mathcal{B}d^+(\mathcal{FI})\|}{\|\mathcal{FEI}\|}$
ACCIDENTS								
50	2, 242	256	2, 613	216	**1.17**	**0.84**	**0.11**	**0.08**
40	6, 856	646	8, 044	762	**1.17**	**1.18**	**0.09**	**0.09**
30	23, 657	1, 932	27, 437	2, 729	**1.16**	**1.41**	**0.08**	**0.10**
20	100, 857	7, 267	114, 650	11, 896	**1.14**	**1.64**	**0.07**	**0.10**
KOSARAK								
1.00	384	0	384	88	**1.00**	/	**0.00**	**0.23**
0.50	1, 619	0	1, 619	307	**1.00**	/	**0.00**	**0.19**
0.40	2, 523	0	2, 523	467	**1.00**	/	**0.00**	**0.19**
0.30	5, 012	0	5, 012	814	**1.00**	/	**0.00**	**0.16**
0.25	8, 833	0	8, 833	1, 187	**1.00**	/	**0.00**	**0.13**
0.20	39, 465	0	39, 465	3, 022	**1.00**	/	**0.00**	**0.08**
RETAIL								
10.00	10	0	10	5	**1.00**	/	**0.00**	**0.56**
5.00	17	0	17	4	**1.00**	/	**0.00**	**0.25**
1.00	160	0	160	78	**1.00**	/	**0.00**	**0.49**
0.50	581	0	581	284	**1.00**	/	**0.00**	**0.49**
0.10	7, 587	2	7, 587	3, 452	**1.00**	**1, 726.00**	**0.00**	**0.46**
0.05	19, 234	5	19, 234	8, 143	**1.00**	**1, 628.60**	**0.00**	**0.42**
T10I4D100K								
1.00	386	0	386	370	**1.00**	/	**0.00**	**0.96**
0.50	1, 074	0	1, 074	585	**1.00**	/	**0.00**	**0.55**
0.40	2, 001	0	2, 001	761	**1.00**	/	**0.00**	**0.38**
0.30	4, 475	2	4, 475	1, 293	**1.00**	**646.50**	**0.00**	**0.29**
0.20	12, 945	6	12, 950	1, 938	**1.00**	**323.00**	**0.00**	**0.15**
0.10	26, 678	5	26, 688	4, 054	**1.00**	**810.80**	**0.00**	**0.15**
0.05	51, 907	9	51, 917	12, 062	**1.00**	**1, 340.22**	**0.00**	**0.23**
0.04	61, 006	3	61, 016	17, 725	**1.00**	**5, 908.33**	**0.00**	**0.29**
0.03	79, 971	2	79, 982	27, 867	**1.00**	**13, 933.50**	**0.00**	**0.35**
0.02	127, 519	10	127, 531	50, 258	**1.00**	**5, 025.80**	**0.00**	**0.39**
T40I10D100K								
10	83	0	83	82	**1.00**	/	**0.00**	**1.00**
5	317	0	317	302	**1.00**	/	**0.00**	**0.96**
4	441	0	441	405	**1.00**	/	**0.00**	**0.92**
3	794	0	794	700	**1.00**	/	**0.00**	**0.88**
2	2, 294	0	2, 294	2, 015	**1.00**	/	**0.00**	**0.88**
1	65, 237	0	65, 237	21, 692	**1.00**	/	**0.00**	**0.33**

Table 6.7: Size of $\mathcal{EDCI}$ *vs.* $\mathcal{FEI}$ and $\mathcal{ADCI}$ *vs.* $\mathcal{B}d^+(\mathcal{FI})$ for sparse contexts.

minsupp (%)	$\|\mathcal{FI}\|$	$\|$ **FCI**s_rep $\|$	$\|$ **NDI**s_rep $\|$	$\|$ **CNDI**s_rep $\|$	$\|$ **FEI**s_rep $\|$	$\|$ **DCI**s_rep $\|$
CONNECT						
90	27, 128	3, 487	199	177	399	**23**
80	533, 976	15, 108	348	305	978	**84**
70	4, 129, 840	35, 876	545	491	1, 711	**162**
60	21, 250, 672	68, 344	894	-	2, 926	**294**
50	88, 173, 344	130, 112	1, 397	-	5, 064	**590**
40	339, 915, 256	239, 373	2, 066	-	8, 162	**1, 063**
30	1, 331, 673, 368	460, 357	3, 221	-	14, 084	**1, 987**
20	6, 157, 510, 380	1, 483, 199	7, 574	-	39, 204	**5, 515**
10	-	8, 035, 412	29, 167	-	153, 929	**22, 401**
5	-	28, 384, 574	91, 050	-	488, 399	**82, 739**
MUSHROOM						
40	566	140	146	117	152	**92**
30	2, 736	427	329	275	311	**214**
20	53, 584	1, 197	1, 143	731	1, 259	**942**
10	574, 432	4, 885	4, 347	2, 655	6, 531	**5, 458**
5	3, 755, 512	12, 843	11, 569	6, 546	24, 408	**20, 555**
4	5, 131, 853	16, 733	14, 382	8, 240	30, 235	**25, 161**
3	9, 987, 059	22, 231	19, 426	10, 824	53, 181	**43, 792**
2	23, 596, 651	31, 768	28, 253	-	92, 268	**76, 436**
1	90, 751, 402	51, 640	48, 719	-	237, 243	**197, 056**
CHESS						
90	623	499	95	93	119	**44**
80	8, 228	5, 084	281	276	468	**151**
70	48, 732	23, 893	684	669	1, 483	**421**
60	254, 945	98, 393	1, 596	1, 567	4, 638	**918**
50	1, 272, 933	369, 451	3, 425	3, 341	14, 273	**1, 972**
40	6, 439, 703	1, 361, 158	7, 185	7, 015	44, 028	**4, 119**
30	37, 282, 963	5, 316, 468	15, 147	-	147, 778	**8, 825**
20	289, 154, 814	22, 808, 625	34, 761	-	542, 541	**22, 518**
10	4, 553, 779, 005	123, 243, 073	98, 664	-	2, 453, 745	**76, 199**
PUMSB						
90	2, 608	1, 467	586	460	789	**319**
85	20, 535	8, 514	1, 792	1, 147	2, 629	**648**
80	142, 157	33, 308	3, 642	2, 136	6, 252	**1, 080**
75	672, 630	101, 083	5, 549	3, 171	11, 708	**1, 470**
70	2, 698, 265	241, 259	7, 875	4, 564	18, 319	**2, 144**
65	8, 099, 128	496, 199	12, 609	7, 575	28, 592	**3, 552**
60	19, 529, 992	1, 074, 628	21, 323	12, 081	54, 645	**5, 551**
55	48, 790, 118	2, 729, 796	32, 121	-	118, 122	**8, 145**
50	165, 903, 541	7, 121, 265	47, 764	-	232, 582	**11, 552**
40	3, 474, 538, 312	44, 434, 213	149, 211	-	865, 728	**35, 576**
30	-	698, 928, 543	470, 828	-	5, 641, 150	**112, 430**
20	-	7, 453, 502, 677	-	-	32, 404, 485	**500, 506**
PUMSB*						
70	30	18	21	18	24	**18**
60	168	69	76	69	78	**72**
50	680	249	277	238	276	**256**
40	27, 355	2, 611	1, 884	1, 595	2, 029	**1, 594**
30	432, 699	16, 155	7, 926	6, 596	10, 401	**7, 556**
20	7, 122, 280, 454	122, 202	49, 642	-	102, 275	**56, 588**
10	-	1, 512, 866	450, 855	-	1, 414, 103	**527, 969**

Table 6.8: Size of the different concise representations for dense contexts.

minsupp (%)	$\lvert \mathbf{DCI}s\_rep \rvert$	$\frac{\lvert \mathcal{FI} \rvert}{\lvert \mathbf{DCI}s\_rep \rvert}$	$\frac{\lvert \mathbf{FCI}s\_rep \rvert}{\lvert \mathbf{DCI}s\_rep \rvert}$	$\frac{\lvert \mathbf{NDI}s\_rep \rvert}{\lvert \mathbf{DCI}s\_rep \rvert}$	$\frac{\lvert \mathbf{CNDI}s\_rep \rvert}{\lvert \mathbf{DCI}s\_rep \rvert}$	$\frac{\lvert \mathbf{FEI}s\_rep \rvert}{\lvert \mathbf{DCI}s\_rep \rvert}$
			CONNECT			
90	**23**	1, 233.09	158.50	9.05	8.05	18.09
80	**84**	6, 433.45	182.02	4.19	3.67	11.77
70	**162**	25, 651.18	222.83	3.39	3.05	10.62
60	**294**	72, 527.89	233.26	3.05	-	9.98
50	**590**	149, 700.07	220.90	2.37	-	8.60
40	**1, 063**	320, 070.86	225.40	1.95	-	7.68
30	**1, 987**	670, 530.40	231.80	1.62	-	7.09
20	**5, 515**	1, 116, 704.82	268.99	1.37	-	7.11
10	**22, 401**	-	358.72	1.30	-	6.87
5	**82, 739**	-	343.07	1.10	-	5.90
			MUSHROOM			
40	**92**	6.22	1.54	1.60	1.29	1.66
30	**214**	12.85	2.00	1.54	1.29	1.46
20	**942**	56.94	1.27	1.21	0.78	1.34
10	**5, 458**	105.27	0.90	0.80	0.49	1.20
5	**20, 555**	182.71	0.62	0.56	0.32	1.19
4	**25, 161**	203.97	0.67	0.57	0.33	1.20
3	**43, 792**	228.06	0.51	0.44	0.25	1.21
2	**76, 436**	308.72	0.42	0.37	-	1.21
1	**197, 056**	460.54	0.26	0.25	-	1.20
			CHESS			
90	**44**	14.49	11.60	2.21	2.16	2.74
80	**151**	54.85	33.89	1.87	1.84	3.11
70	**421**	116.03	56.89	1.63	1.59	3.53
60	**918**	278.02	107.30	1.74	1.71	5.06
50	**1, 972**	645.83	187.44	1.74	1.70	7.24
40	**4, 119**	1, 563.79	330.54	1.74	1.70	10.69
30	**8, 825**	4, 225.18	602.50	1.72	-	16.75
20	**22, 518**	12, 841.62	1, 012.95	1.54	-	24.09
10	**76, 199**	59, 762.45	1, 617.41	1.29	-	32.20
			PUMSB			
90	**319**	8.20	4.61	1.84	1.45	2.48
85	**648**	31.74	13.16	2.77	1.77	4.06
80	**1, 080**	131.75	30.87	3.38	1.98	5.79
75	**1, 470**	457.88	68.81	3.78	2.16	7.97
70	**2, 144**	1, 259.11	112.58	3.67	2.13	8.55
65	**3, 552**	2, 280.80	139.74	3.55	2.13	8.05
60	**5, 551**	3, 518.92	193.63	3.84	2.18	9.85
55	**8, 145**	5, 990.93	335.19	3.94	-	14.50
50	**11, 552**	14, 362.70	616.51	4.14	-	20.14
40	**35, 576**	97, 667.98	1, 249.03	4.19	-	24.34
30	**112, 430**	-	6, 216.62	4.19	-	50.18
20	**500, 506**	-	14, 891.96	-	-	64.74
			PUMSB*			
70	**18**	1.76	1.06	1.24	1.06	1.35
60	**72**	2.37	0.97	1.07	0.97	1.08
50	**256**	2.67	0.98	1.09	0.93	1.08
40	**1, 594**	17.17	1.64	1.18	1.00	1.27
30	**7, 556**	57.27	2.14	1.05	0.87	1.38
20	**56, 588**	125, 864.25	2.16	0.88	-	1.81
10	**527, 969**	-	2.87	0.85	-	2.68

Table 6.9: The compactness rates offered by the representation based on disjunctive closed itemsets for dense contexts.

minsupp (%)	$\|\mathcal{FI}\|$	$\|$ **FCI**s_rep $\|$	$\|$ **NDI**s_rep $\|$	$\|$ **CNDI**s_rep $\|$	$\|$ **FEI**s_rep $\|$	$\|$ **DCI**s_rep $\|$
ACCIDENTS						
50	8, 058	8, 058	2, 850	2, 850	2, 829	**2, 498**
40	32, 529	32, 529	8, 704	8, 704	8, 806	**7, 502**
30	149, 546	149, 530	28, 558	28, 558	30, 166	**25, 589**
20	889, 884	887, 389	110, 370	110, 367	126, 546	**108, 124**
KOSARAK						
1.00	384	384	384	384	472	**384**
0.50	1, 619	1, 619	1, 616	1, 616	1, 926	**1, 619**
0.40	2, 523	2, 522	2, 514	2, 513	2, 990	**2, 523**
0.30	5, 012	4, 983	4, 915	4, 895	5, 826	**5, 012**
0.25	8, 833	8, 771	7, 830	7, 786	10, 021	**8, 833**
0.20	39, 465	35, 865	17, 297	17, 117	42, 427	**39, 465**
RETAIL						
10.00	10	10	10	10	15	**10**
5.00	17	17	17	17	21	**17**
1.00	160	160	160	160	238	**160**
0.50	581	581	581	581	865	**581**
0.10	7, 590	7, 573	7, 580	7, 568	11, 039	**7, 589**
0.05	19, 243	19, 115	19, 178	19, 096	27, 377	**19, 239**
T10I4D100K						
1.00	386	386	386	386	756	**386**
0.50	1, 074	1, 074	1, 074	1, 074	1, 659	**1, 074**
0.40	2, 002	1, 993	1, 987	1, 979	2, 762	**2, 001**
0.30	4, 553	4, 510	4, 377	4, 369	5, 768	**4, 477**
0.20	13, 256	13, 108	11, 475	11, 434	14, 888	**12, 951**
0.10	27, 533	26, 807	24, 120	23, 901	30, 742	**26, 683**
0.05	53, 386	46, 994	44, 365	42, 800	63, 979	**51, 916**
0.04	62, 865	55, 844	53, 113	51, 266	78, 741	**61, 009**
0.03	82, 164	71, 266	69, 536	66, 462	107, 849	**79, 973**
0.02	129, 876	107, 823	109, 486	102, 869	177, 789	**127, 529**
T40I10D100K						
10	83	83	83	83	165	**83**
5	317	317	317	317	619	**317**
4	441	441	441	441	846	**441**
3	794	794	794	794	1, 494	**794**
2	2, 294	2, 294	2, 294	2, 294	4, 309	**2, 294**
1	65, 237	65, 237	42, 312	42, 312	86, 929	**65, 237**

Table 6.10: Size of the different concise representations for sparse contexts.

minsupp (%)	$\lvert \mathbf{DCI}s\_rep \rvert$	$\frac{\lvert \mathcal{FI} \rvert}{\lvert \mathbf{DCI}s\_rep \rvert}$	$\frac{\lvert \mathbf{FCI}s\_rep \rvert}{\lvert \mathbf{DCI}s\_rep \rvert}$	$\frac{\lvert \mathbf{NDI}s\_rep \rvert}{\lvert \mathbf{DCI}s\_rep \rvert}$	$\frac{\lvert \mathbf{CNDI}s\_rep \rvert}{\lvert \mathbf{DCI}s\_rep \rvert}$	$\frac{\lvert \mathbf{FEI}s\_rep \rvert}{\lvert \mathbf{DCI}s\_rep \rvert}$
ACCIDENTS						
50	**2, 498**	3.23	3.23	1.14	1.14	1.13
40	**7, 502**	4.34	4.34	1.16	1.16	1.17
30	**25, 589**	5.84	5.84	1.12	1.12	1.18
20	**108, 124**	8.23	8.21	1.02	1.02	1.17
KOSARAK						
1.00	**384**	1.00	1.00	1.00	1.00	1.23
0.50	**1, 619**	1.00	1.00	1.00	1.00	1.19
0.40	**2, 523**	1.00	1.00	1.00	1.00	1.19
0.30	**5, 012**	1.00	1.00	0.99	0.98	1.16
0.25	**8, 833**	1.00	0.99	0.89	0.88	1.13
0.20	**39, 465**	1.00	0.91	0.44	0.43	1.08
RETAIL						
10.00	**10**	1.00	1.00	1.00	1.00	1.56
5.00	**17**	1.00	1.00	1.00	1.00	1.25
1.00	**160**	1.00	1.00	1.00	1.00	1.49
0.50	**581**	1.00	1.00	1.00	1.00	1.49
0.10	**7, 589**	1.00	1.00	1.00	1.00	1.45
0.05	**19, 239**	1.00	0.99	1.00	0.99	1.42
T10I4D100K						
1.00	**386**	1.00	1.00	1.00	1.00	1.96
0.50	**1, 074**	1.00	1.00	1.00	1.00	1.55
0.40	**2, 001**	1.00	1.00	0.99	0.99	1.38
0.30	**4, 477**	1.02	1.01	0.98	0.98	1.29
0.20	**12, 951**	1.02	1.01	0.89	0.88	1.15
0.10	**26, 683**	1.03	1.00	0.90	0.90	1.15
0.05	**51, 916**	1.03	0.91	0.85	0.82	1.23
0.04	**61, 009**	1.03	0.92	0.87	0.84	1.29
0.03	**79, 973**	1.03	0.89	0.87	0.83	1.35
0.02	**127, 529**	1.02	0.85	0.86	0.81	1.39
T40I10D100K						
10	**83**	1.00	1.00	1.00	1.00	2.00
5	**317**	1.00	1.00	1.00	1.00	1.96
4	**441**	1.00	1.00	1.00	1.00	1.92
3	**794**	1.00	1.00	1.00	1.00	1.88
2	**2, 294**	1.00	1.00	1.00	1.00	1.88
1	**65, 237**	1.00	1.00	0.65	0.65	1.33

Table 6.11: The compactness rates offered by the representation based on disjunctive closed itemsets for sparse contexts.

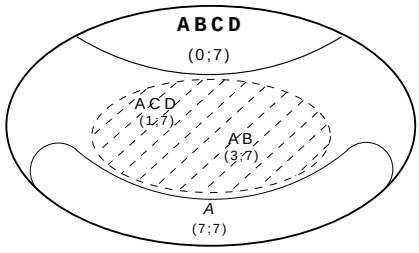


Figure 6.6: A disjunctive equivalence class: for each itemset, the associated couple of values gives its conjunctive support (on the left) and disjunctive support (on the right).

6.7 Related Work and Discussion

In this section, we discuss the main related work to the contributions proposed in this chapter.

First of all, let us make an alignment between the disjunctive and the conjunctive search spaces. We will hence find that disjunctive equivalence classes correspond to conjunctive equivalence classes – gathering itemsets having the same Galois closure [Ganter and Wille, 1999] – within the conjunctive search space. An essential itemset is then the mapping of the concept of *minimal generator* [Bastide *et al.*, 2000b]. While a disjunctive closed itemset is the mapping of the concept of *conjunctive closed itemset* [Pasquier *et al.*, 1999b]. It is also worth noting that a disjunctive equivalence class can contain itemsets having distinct conjunctive supports and, hence, belonging to distinct conjunctive equivalence classes. This is illustrated by the following example.

Example 47 *Consider the context shown in Table 6.1 (cf. page 110). The largest disjunctive equivalence class, i.e., whose support is equal to $|\mathcal{O}|$, namely* **7***, is depicted by Figure 6.6. It is composed by the maximal disjunctive closed itemset* `ABCD`*, its associated essential itemset* `A`*, and, some itemsets encompassed between* `A` *and* `ABCD` *(gathered within the dashed ellipse). Each itemset is associated to a couple of numerical values. The first one corresponds to its conjunctive support while the second value is equal to its disjunctive one. By examining the itemsets of this class, we note that their respective disjunctive supports are equal while their conjunctive supports can differ (e.g., Supp(*`ABCD`*) =* **0** *while Supp(*`A`*) =* **7***). Thus, the itemsets belonging to distinct conjunctive equivalence classes can share the same disjunctive closure and, hence, be gathered within the same disjunctive equivalence class.*

From the semantic aspect, contrary to frequent closed itemsets, disjunctive closed ones offer the possibility to take into account complementary information, *i.e.*, items that are for example mutually exclusive. For a given itemset I, its disjunctive closure gathers items whose appearances depend on that of a *nonempty* subset of I. This is not possible using frequent closed itemsets since this requires that *all* items of I simultaneously appear. Moreover, for an arbitrary itemset I, its associated closed itemset only gives an idea about the set of items S that closely depend on *all* the items of I. However, an item $i \in S$ can appear in a transaction that does not contain I, but only a proper subset. While the disjunctive closure of I gathers items that closely depend on the set of items contained in I. Indeed, the membership of an item to the disjunctive closure of I requires that a subset of I appears in the associated transactions. This can for example be useful for analyzing gene-expression data through localizing groups of genes of which the appearance depends on other groups. In addition, disjunctive closed itemsets offer an interesting starting

point for the extraction of generalized association rules [Nanavati *et al.*, 2001, Toivonen, 1996a] which can be useful in some real-life applications. Indeed, such rules offer the possibility to present conjunction, disjunction and negation of items in both premise and conclusion parts. In this respect, in addition to the conjunctive support, our representation offers direct access to the disjunctive support of frequent itemsets, and hence to their negative support through De Morgan's law.

The concepts of essential and disjunctive closed itemsets are also closely related to many important pattern classes as detailed in the following. They can be considered as particular cases of *composite items* [Ye and Keane, 1997] where the disjunction of (infrequent) items is used to compose new items, the *composite items*. By introducing composite items, Ye and Keane highlighted the usefulness of infrequent items in some applications. For example, consider the context of Table 6.1 and let $minsupp = \mathbf{4}$, B and C are hence infrequent items since their support is equal to **3**. Nevertheless, the support of B $\vee$ C is equal to **5** and, hence, $Supp(\texttt{B} \vee \texttt{C}) \geq minsupp$. The disjunct B $\vee$ C will be considered as a new item (a composite one) even if, actually, it is composed of two items. It will be used during the mining process since it is frequent which makes B and C useful. The work of Shima *et al.* [Shima *et al.*, 2004] can be considered as an extension of composite items, since it takes into account particular disjunctive normal forms (DNFs) where disjuncts may contain a conjunction of items – frequent closed itemsets – and not only a single item. Indeed, the authors proposed to extract minimal and closed DNFs. A minimal (*resp.* closed) DNF does not have a subset (*resp.* superset) with the same support. The disjuncts associated to such DNFs are thus constituted by frequent closed itemsets.

It is important to establish the link between essential itemsets and minimal transversals of a hypergraph [Eiter and Gottlob, 1995]. For this purpose, consider essential itemsets of the disjunctive equivalence class whose disjunctive support is equal to the cardinality of the whole set of objects $\mathcal{O}$. These itemsets are hence the minimal ones that intersect all objects of an arbitrary context. If we consider objects as hyperedges and items as vertices, the aforementioned itemsets are thus the minimal transversals of the corresponding hypergraph. Note that each set of essential itemsets corresponding to a given disjunctive equivalence class $\mathcal{C}$ can be considered as the minimal transversals of a hypergraph. This latter structure is then represented by the hyperedges corresponding to the objects verified by the itemsets belonging to $\mathcal{C}$. While the set of vertices corresponds to the set of items contained in the corresponding disjunctive closed itemset.

Essential and disjunctive closed itemsets can also be considered as specific cases of *error-tolerant itemsets* [Yang *et al.*, 2001]. Indeed, an itemset X is an error-tolerant itemset having an *error tolerance* ϵ and a support k *w.r.t.* a context $\mathcal{K}$ if there are k objects of $\mathcal{K}$ in which at least a fraction **1** - ϵ of the items from X are present [Yang *et al.*, 2001]. In our case, an essential or disjunctive closed itemset X is an error-tolerant itemset for $\epsilon = \frac{|X|-1}{|X|}$ and $\texttt{k} = Supp(\vee X)$. Indeed, the presence of one item of X in an object is sufficient to satisfy it. Note that if $\epsilon = \mathbf{0}$, *i.e.*, no error is allowed, $\texttt{k} = Supp(X)$. The conjunctive and disjunctive supports hence constitute the lower and upper bounds of the support of X, respectively.

We will establish the link between our work and that recently proposed in [Soulet and Crémilleux, 2008]. In this respect, our disjunctive closure operator h ensures obtaining a preserving function *w.r.t.* the disjunctive support according to [Soulet and Crémilleux, 2008]. Indeed, for an arbitrary itemset, our operator ensures that once $i \in h(X)$, we have: on the one hand, $Supp(\vee X) = Supp(\vee(X \cup \{i\}))$, and on

the other hand, $Supp(\vee Y) = Supp(\vee(Y \cup \{i\}))$, $\forall\ X \subseteq Y$. This latter equality results from the isotony property of any closure operator: if $i \in h(X)$, then $i \in h(Y)$, $\forall\ X \subseteq Y$.
In comparison to our work, that of [Soulet and Crémilleux, 2008] does not propose any concise representation for frequent itemsets using the preserving function associated to the disjunctive support. In addition, although they give a definition of a closure operator adequate to a condensable function, [9] the authors did not pay attention to the corresponding link between the power-set of items and that of objects as we did in this paper. Indeed, their definition mainly relies on an incremental augmentation of a given itemset X by those items that do not modify the value of the condensable function for X. The authors proposed to extract the closed itemsets adequate to a condensable function under an anti-monotone constraint applied to the minimal seeds giving these closures. However, they did not study the effect of such a pruning on the obtained set of closed itemsets *w.r.t.* the point of view of concise representations. Indeed, consider for example the closure operator associated to the disjunctive support, *i.e.*, our disjunctive closure operator h. The minimal elements within the associated equivalence classes are the essential itemsets. Let us also consider the anti-monotone constraint offered by the frequency constraint through setting a minimum support threshold *minsupp*. The obtained set once h is applied on essential itemsets, pruned *w.r.t.* *minsupp*, is $\mathcal{EDCI}$. However, this latter set is not an exact concise representation of the itemsets adequate to the disjunctive support, *i.e.*, disjunctive itemsets. Indeed, consider the context we used in this chapter depicted by Table 6.1 (*cf.* page 110). For *minsupp* = **1**, $\mathcal{EDCI}$ = {(B, 3), (C, 3), (D, 3), (BC, 5), (BD, 5), (CD, 5), (ABCD, 7)} (*cf.* Example 46, page 125 for details on the extraction of this set). If X = BCD, we will get ABCD as the closure of BCD. Indeed, the process used in [Soulet and Crémilleux, 2008] for getting the value of the function on an itemset X is the same as ours. Hence, it consists in looking for the smallest closure in the representation containing X. Consequently, the disjunctive support of BCD will be considered to be equal to **7**. This is obviously wrong since the closure of BCD is itself, and was already pruned since BCD is infrequent *w.r.t.* *minsupp* = **1** (its actual disjunctive support is **6**). Noteworthily, this constitutes one of the main contributions proposed in this paper. Indeed, the pruning of BCD also made $\mathcal{EDCI}$ not an exact concise representation for frequent itemsets (*cf.* Example 43, page 118). This motivated us to lead an in-depth exploration for ensuring the exactness of the proposed representation based on $\mathcal{EDCI}$ (*cf.* Subsection 6.4.3, page 118). Unfortunately, the authors in [Soulet and Crémilleux, 2008], neither stated explicitly that, once the anti-monotone constraint applied, the obtained set of closed itemsets – adequate to a condensable function f (*e.g.* $\mathcal{EDCI}$ for the disjunctive support and the *minsupp* constraint) – may not be an exact representation of itemsets adequate to f, nor highlighted an error bound or the need for adding other elements, as we did here, ensuring the exactness of the regeneration process.

Now, we will make the link between our work and that of Zhao *et al.* [Zhao *et al.*, 2006, Zhao, 2006]. Actually, the authors proposed connection operators to link $\mathcal{P}(\mathcal{I})$ and $\mathcal{P}(\mathcal{O})$ for the case of disjunctive Boolean expressions, called *OR-clauses*. Nevertheless, their definition of the operator Ψ linking $\mathcal{P}(\mathcal{O})$ to $\mathcal{P}(\mathcal{I})$ (performed by the operator f in our case) depends on the operator Φ ensuring the dual direction (performed by the operator g in our case) and was not independently given. The following definition presents the operators proposed in [Zhao *et al.*, 2006]:

Definition 67 (Connection operators)

[9] A condensable function is either a preserving function or a combination of more than one preserving function.

Let $\mathcal{K} = (\mathcal{O}, \mathcal{I}, \mathcal{M})$ be an extraction context, $\mathcal{P}(\mathcal{I})$ the set of all possible OR-clauses over $\mathcal{I}$, and $\mathcal{P}(\mathcal{O})$ the power-set of objects. Let $I \in \mathcal{P}(\mathcal{I})$ and $O \in \mathcal{P}(\mathcal{O})$. Given two partially ordered sets $(\mathcal{P}(\mathcal{I}), \subseteq)$ and $(\mathcal{P}(\mathcal{O}), \subseteq)$, the following operators form a connection over $\mathcal{P}(\mathcal{I})$ and $\mathcal{P}(\mathcal{O})$:

$\Phi : \mathcal{P}(\mathcal{I}) \rightarrow \mathcal{P}(\mathcal{O})$

$\quad I \quad \mapsto \Phi(I) = \{o \in \mathcal{O} |\ \exists\ i \in I \text{ s.t. } (o, i) \in \mathcal{M}\}$

$\Psi : \mathcal{P}(\mathcal{O}) \rightarrow \mathcal{P}(\mathcal{I})$

$\quad O \quad \mapsto \Psi(O) = \{i \in \mathcal{I} |\ \Phi(i) \subseteq O\}$

However, the authors neither gave the expression of the resulting closure operator nor carried out a thorough analysis of the inherent theoretical properties. We can clearly notice that these operators do not allow the direct computation of the disjunctive closure of an itemset. Indeed, the inter-dependency between the connection operators Φ and Ψ makes necessary to maintain the list of objects identifiers to which each item belongs (*aka* tidset [Zaki and Hsiao, 2002]) before starting the mining process. A main feature of our closure operator is that it does not present such an inter-dependency. Zhao *et al.* proposed an algorithm, called BLOSOM-CO, for mining closed OR-clauses (the equivalent of disjunctive closed itemsets in our case) whose *disjunctive* support is encompassed between two user-defined thresholds, and of size (*i.e.*, number of items) lower than a given threshold. However, they did not study the effect of setting the conjunctive frequency constraint during the mining process. They did not also explore the framework of concise representations for frequent itemsets. Due to the inter-dependency between the operators they proposed, the designed algorithm is based on a combination of a depth-first traversal of the search space and the use of tidsets. However, this leads to larger memory consumption for storing tidsets which constitutes a real hamper, especially for low *minsupp* values. It is also worth noting that the authors did not make the connection between minimal OR-clauses and essential itemsets.

From an algorithmic point of view, the DCPR_MINER algorithm can easily be adapted to efficiently extract a new exact concise representation associated to frequent *correlated* itemsets *w.r.t.* the *bond* measure [Omiecinski, 2003]. Even not mentioned in [Omiecinski, 2003], this measure is based on the disjunctive support. Indeed, the bond of an arbitrary itemset X is equal to the ratio between its conjunctive support and the cardinality of the set of objects that contain any item of X. This latter cardinality is obviously equal to its disjunctive support.

6.8 Conclusion

In this chapter, we introduced a new disjunctive closure operator and we thoroughly studied its theoretical properties. Based on this operator, we structurally characterize the disjunctive search space. Then, we introduced a new concise representation of frequent itemsets based on the disjunctive closed itemsets having at least a frequent essential itemset as a seed. In addition to interesting compactness rates, this representation allows a straightforward computation of the disjunctive and negative supports. Moreover, it is only composed of disjunctive closed itemsets which ensure its homogeneity. An algorithm, called DCPR_MINER, was proposed for its extraction. In nearly all experiments we performed, the obtained results showed that our representation is significantly smaller than the pioneering ones of the literature. Therefore we have proposed a concise representation (model) of frequent itemsets that distills the meaningful information with respect to the minimum description length principle, especially in the case of

dense contexts.

The next chapter proposes a complete approach allowing the extraction of (subset of) generalized association rules. The introduced approach relies on essential and disjunctive closed itemsets as a starting point.

Chapter 7

Generalization of Association Rules through Disjunction

7.1 Introduction

The main moan that can be addressed to the contributions related to association rules is their focus on the simultaneous occurrence (or co-occurrence) between items [Steinbach and Kumar, 2007]. Indeed, almost all related work neglect the other kinds of relations, like mutually exclusive or complementary occurrences [Tzanis and Berberidis, 2007], which can also bring information of worth interest for the end-users. Such kind of knowledge can naturally be conveyed through disjunctive patterns. In this regard, the added-value of association rules having disjunctions of literals [1] in the premise or conclusion part has been highlighted in some contributions [Nanavati *et al.*, 2001, Steinbach and Kumar, 2007]. For example, these rules were shown to be useful for software change impact analysis [Hattori *et al.*, 2008], and feature model mining [She, 2008]. In fact, such kind of rules offers advantages compared to the hierarchy/taxonomy-based generalization [Srikant and Agrawal, 1995]. Indeed, they do not depend upon a pre-defined taxonomy. They also do not suffer from the problem of overgeneralization since the taxonomy approach mainly considers fixed disjuncts.

In this chapter, we propose a new approach covering the whole process allowing the extraction of generalized association rules. These latter rules generalize positive ones by also allowing the disjunction and negation connectors between items [Toivonen, 1996a]. Indeed, in some situations, the information conveyed by a generalized association rule – and in particular disjunctive ones – may not be obtained even by a collection of conjunctive association rules [Nanavati *et al.*, 2001]. Moreover, the use of the disjunctive operator in association rules allows, for example, to obtain rules linking frequently occurring itemsets and rare ones. Such relationships are difficult to mine using conjunctive association rules unless the value of the minimum support threshold set too low, which leads to an overwhelming rule set.

As a starting point, the introduced approach relies on a concise representation of frequent itemsets based on disjunctive itemsets. Such a representation allows the derivation of the exact conjunctive supports of frequent itemsets while preserving the easy access to their respective disjunctive and negative supports.

[1] A literal is an item or the negation of an item.

This makes it possible to compute the values of quality measures. Indeed, it was shown in [Hébert and Crémilleux, 2007] that almost all interestingness measures for association rules are expressed depending on the support of the rule and those of its associated premise and conclusion. In addition, the use of disjunctive itemsets – in particular closed and essential itemsets – provides an interesting starting point towards mining association rules conveying complementary occurrences between items, rather than co-occurrences. Indeed, these latter relationships – co-occurrences within literals – were explored in-depth in the literature through association rules having conjunction of literals, called *literalsets*, in premise and conclusion parts. This leads to what is commonly known as *positive and negative association rules*. While disjunctive association rules only have recently begun to grasp the interest of researchers. In this respect, we give an overview of the possible mined forms of generalized association rules. After that, we select subsets of generalized rules to be extracted. This required the construction of a partially ordered structure, obtained *w.r.t.* set inclusion between disjunctive closed itemsets.

We restrict ourselves in this work to disjunctive closed itemsets whose minimal seeds, *i.e.*, essential itemsets, are frequent with respect to a minimum conjunctive support threshold. This is argued by the fact that, within the association rule framework, this threshold as well as the confidence-based one have a key role in the reduction of the number of extracted association rules [Ceglar and Roddick, 2006, Kryszkiewicz, 2002]. In addition, the use of a partially ordered structure will allow to select representative subsets of association rules. This nucleus of rules will be of paramount help for avoiding to overwhelm end-users by highly-sized rule lists. Moreover, once this structure built, extracting generalized rules becomes a straightforward task.

The remainder of the chapter is organized as follows. The next section starts by extending the framework of classic association rules through taking into account the various possible connectors as well as negative items. It then presents an overview of the possible mined forms of generalized association rules, and shows how are calculated the associated supports in the general case. Section 7.3 details the selection process of generalized rules to be extracted and their quality measures estimation using the adopted concise representation of frequent itemsets. Section 7.4 proposes algorithms covering the different steps of the extraction process. Experimental results focusing on the mining time as well as the quantitative aspect are reported and analyzed in Section 7.5. Section 7.6 discusses the related work.

7.2 Overview of Generalized Association Rule Forms

In this section, we are interested in going beyond classic association rules only conveying conjunction of items in the premise and/or conclusion parts. This is carried out through defining the framework of generalized association rules in the general case. Then, we describe some main rule forms, and show how their associated supports are computed.

7.2.1 Generalized Association Rule Framework

An association rule R: $X \Rightarrow Y$ based on an itemset Z, denoted *Z-based rule*, is such that $X = \{x_1, x_2, \ldots, x_n\} \subseteq \mathcal{I}$, and $Y = \{y_1, y_2, \ldots, y_m\} \subseteq \mathcal{I}$ be two itemsets, $X \cap Y = \emptyset$, and $X \cup Y = Z$. An association rule is usually considered as interesting *w.r.t.* two statistical metrics, namely the support and the confidence [Kryszkiewicz, 2002]. The formulae of these measures for an arbitrary rule are as

follows:

$$Supp(X \Rightarrow Y) = Supp(X \wedge Y); \text{ and, } Conf(X \Rightarrow Y) = \frac{Supp(X \wedge Y)}{Supp(X)} = \frac{Supp(X \Rightarrow Y)}{Supp(X)}$$

Let us recall that a rule is said to be *exact* whenever its confidence value is equal to **1**. Otherwise, it is said to be *approximate*. In addition, it is said to be *interesting* or *valid* if its support and confidence values are greater than or equal to their respective minimum thresholds *minsupp* and *minconf*. It is clear that whenever we have the ability to assess $Supp(X \Rightarrow Y)$, the derivation of the confidence value is straightforward, since we only have to divide the support of the rule by that of the premise part.

Generalized association rule forms extend the framework of classic association rules by:

1. Allowing the use of negative items, in addition to positive ones, within the same rule. The negative item $\overline{i}$ *w.r.t.* a positive item i conveys the information about the absence of i in transactions, rather than its presence.
2. Allowing the links between items using the disjunction connector, in addition to the conjunction one.

The definition of a generalized association rule requires that of a Boolean expression which is as follows:

Definition 68 (Boolean expression)
A Boolean expression is the logical connection of a set of items using the conjunction, disjunction and negation connectors.

Note that for a Boolean expression, parentheses are, whenever necessary, used to demarcate clauses and priority within operators. A clause is then composed by a set of literals linked using either the logical conjunction or the disjunction connector.

Example 48 *Let* A, B *and* C *be three items, then* $(A \wedge B) \vee \overline{C}$ *is a Boolean expression.*

Definition 69 (Generalized association rule)
Let $\mathcal{I}$ *be a set of items and* $x_i, y_j \in \mathcal{I}$. *A generalized association rule is of the form:*

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

where $\varrho(x_1, x_2, \ldots, x_n)$ *and* $\upsilon(y_1, y_2, \ldots, y_n)$ *are two Boolean expressions which do not have any item in common.*

Example 49 *Let* $\mathcal{I} = \{A, B, C, D, E\}$ *be a set of items. The rules* $A \wedge B \Rightarrow C \wedge \overline{D}$ *and* $A \vee E \Rightarrow D$ *are two examples of generalized association rules.*

We now present the support and the confidence of a generalized association rule.

Definition 70 (Support, Confidence of a Generalized association rule)
Let R *be a generalized association rule* $\varrho(x_1, x_2, \ldots, x_n) \Rightarrow \upsilon(y_1, y_2, \ldots, y_n)$,

- *The support of* R, $Supp(R)$, *is equal to the number of transactions that* ***simultaneously*** *satisfy both Boolean expressions* $\varrho(x_1, x_2, \ldots, x_n)$ *and* $\upsilon(y_1, y_2, \ldots, y_n)$. *Hence,*

$$Supp(R) = Supp(\varrho(x_1, x_2, \ldots, x_n) \wedge \upsilon(y_1, y_2, \ldots, y_n)).$$

- *The confidence of R, $Conf(R)$, is the ratio between its support and the support of the Boolean expression representing the premise part. Hence,*

$$Conf(R) = \frac{Supp(\varrho(x_1, x_2, \ldots, x_n) \wedge \upsilon(y_1, y_2, \ldots, y_n))}{Supp(\varrho(x_1, x_2, \ldots, x_n))}.$$

The next lemma states the interval in which varies the confidence of a generalized rule.

Lemma 9 *Let R: $\varrho(x_1, x_2, \ldots, x_n) \Rightarrow \upsilon(y_1, y_2, \ldots, y_n)$ be a generalized association rule. If $Supp(\varrho(x_1, x_2, \ldots, x_n)) \neq \mathbf{0}$, then $Conf(R) \in [\mathbf{0}, \mathbf{1}]$.*

Proof. The support of $\varrho(x_1, x_2, \ldots, x_n) \wedge \upsilon(y_1, y_2, \ldots, y_n)$ is lower than or equal to that of $\varrho(x_1, x_2, \ldots, x_n)$. Indeed, each transaction that satisfies the former also verifies the latter. Hence, $Conf(R) \leq \mathbf{1}$. Both supports also have positive values. Hence, $Conf(R) \geq \mathbf{0}$. Thus, $Conf(R) \in [\mathbf{0}, \mathbf{1}]$. $\diamondsuit$

Example 50 *Consider the context given by Table 2.1 (cf. page 12) and the generalized association rule R: $A \vee E \Rightarrow D$. $Supp(R) = Supp((A \vee E) \wedge D)$. Since the premise and the conclusion are simultaneously satisfied by the transactions* **1**, **2** *and* **4**, *then $Supp(R) = \mathbf{3}$. While $Conf(R) = \frac{Supp(R)}{Supp(A \vee E)}$. Since the disjunctive itemset $A \vee E$ is also fulfilled by the transaction* **3** *(which does not contain D), then $Supp(A \vee E) = \mathbf{4}$. Consequently, $Conf(R) = \frac{3}{4} = \mathbf{0.75}$.*

7.2.2 Support Retrieval of Generalized Association Rule Forms

Let $X = \{x_1, x_2, \ldots, x_n\} \subseteq \mathcal{I}$, and $Y = \{y_1, y_2, \ldots, y_m\} \subseteq \mathcal{I}$ be two itemsets *s.t.* $X \cap Y = \emptyset$. The generalized association rule forms using a conjunction, disjunction or negation of items in the premise and conclusion parts are as follows:

1. $\mathcal{R}_1$: $x_1 \wedge x_2 \wedge \ldots \wedge x_n \Rightarrow y_1 \wedge y_2 \wedge \ldots \wedge y_m$.
2. $\mathcal{R}_2$: $x_1 \wedge x_2 \wedge \ldots \wedge x_n \Rightarrow y_1 \vee\ y_2 \vee\ \ldots \vee\ y_m$.
3. $\mathcal{R}_3$: $x_1 \wedge x_2 \wedge \ldots \wedge x_n \Rightarrow \overline{y_1} \wedge \overline{y_2} \wedge \ldots \wedge \overline{y_m}$.
4. $\mathcal{R}_4$: $x_1 \vee\ x_2 \vee\ \ldots \vee\ x_n \Rightarrow y_1 \wedge y_2 \wedge \ldots \wedge y_m$.
5. $\mathcal{R}_5$: $x_1 \vee\ x_2 \vee\ \ldots \vee\ x_n \Rightarrow y_1 \vee\ y_2 \vee\ \ldots \vee\ y_m$.
6. $\mathcal{R}_6$: $x_1 \vee\ x_2 \vee\ \ldots \vee\ x_n \Rightarrow \overline{y_1} \wedge \overline{y_2} \wedge \ldots \wedge \overline{y_m}$.
7. $\mathcal{R}_7$: $\overline{x_1} \wedge \overline{x_2} \wedge \ldots \wedge \overline{x_n} \Rightarrow y_1 \wedge y_2 \wedge \ldots \wedge y_m$.
8. $\mathcal{R}_8$: $\overline{x_1} \wedge \overline{x_2} \wedge \ldots \wedge \overline{x_n} \Rightarrow y_1 \vee\ y_2 \vee\ \ldots \vee\ y_m$.
9. $\mathcal{R}_9$: $\overline{x_1} \wedge \overline{x_2} \wedge \ldots \wedge \overline{x_n} \Rightarrow \overline{y_1} \wedge \overline{y_2} \wedge \ldots \wedge \overline{y_m}$.

These association rules bring richer information to the end-user than those presented in the literature, since they involve various Boolean connectors in both the premise and the conclusion parts, and not only the conjunction one. We present now an overview of the process by which we are able to retrieve the

- $Supp((x_1 \wedge x_2 \wedge \ldots \wedge x_n) \wedge (y_1 \vee \ y_2 \vee \ \ldots \vee \ y_m)) = Supp(x_1 \wedge x_2 \wedge \ldots \wedge x_n) - Supp(x_1 \wedge x_2 \wedge \ldots \wedge x_n \wedge \overline{y_1} \wedge \overline{y_2} \wedge \ldots \wedge \overline{y_m})$ [Galambos and Simonelli, 2000] (3)

- $Supp(x_1 \wedge x_2 \wedge \ldots \wedge x_n \wedge \overline{y_1} \wedge \overline{y_2} \wedge \ldots \wedge \overline{y_m}) = \sum_{S \subseteq \{y_1,\ldots,y_m\}} (\text{-}1)^{|S|} Supp(x_1 \wedge x_2 \wedge \ldots \wedge x_n \wedge S)$ [Toivonen, 1996a] (4)

- $Supp(A \wedge B) = Supp(A) + Supp(B)$ - $Supp(A \vee B)$, where A and B are two Boolean expressions. (5)

- $Supp(A \wedge \overline{B}) = Supp(A \vee B)$ - $Supp(A)$, where A and B are two Boolean expressions. (6)

Table 7.1: Formulae used for support computation.

supports of the association rules presented above. For this purpose, the two formulae of Lemma 1 (*cf.* page 13) will be useful, in addition to the ones shown in Table 7.1.

The respective supports of the different proposed forms of association rules are then computed as follows:

1. $Supp(\mathcal{R}_1) = Supp(x_1 \wedge x_2 \wedge \ldots \wedge x_n \wedge y_1 \wedge y_2 \wedge \ldots \wedge y_m)$.

2. $Supp(\mathcal{R}_2) = Supp((x_1 \wedge x_2 \wedge \ldots \wedge x_n) \wedge (y_1 \vee y_2 \vee \ldots \vee y_m)) = Supp(x_1 \wedge x_2 \wedge \ldots \wedge x_n) - Supp(x_1 \wedge x_2 \wedge \ldots \wedge x_n \wedge \overline{y_1} \wedge \overline{y_2} \wedge \ldots \wedge \overline{y_m}) = Supp(x_1 \wedge x_2 \wedge \ldots \wedge x_n) - \sum_{S \subseteq \{y_1,\ldots,y_m\}} (\text{-}1)^{|S|} Supp(x_1 \wedge x_2 \wedge \ldots \wedge x_n \wedge S)$.

3. $Supp(\mathcal{R}_3) = Supp(x_1 \wedge x_2 \wedge \ldots \wedge x_n \wedge \overline{y_1} \wedge \overline{y_2} \wedge \ldots \wedge \overline{y_m}) = \sum_{S \subseteq \{y_1,\ldots,y_m\}} (\text{-}1)^{|S|} Supp(x_1 \wedge x_2 \wedge \ldots \wedge x_n \wedge S)$.

4. $Supp(\mathcal{R}_4) = Supp((x_1 \vee x_2 \vee \ldots \vee x_n) \wedge y_1 \wedge y_2 \wedge \ldots \wedge y_m) = Supp(y_1 \wedge y_2 \wedge \ldots \wedge y_m) - Supp(\overline{x_1} \wedge \overline{x_2} \wedge \ldots \wedge \overline{x_n} \wedge y_1 \wedge y_2 \wedge \ldots \wedge y_m) = Supp(y_1 \wedge y_2 \wedge \ldots \wedge y_m) - \sum_{S \subseteq \{x_1,\ldots,x_n\}} (\text{-}1)^{|S|} Supp(S \wedge y_1 \wedge \ldots \wedge y_m)$.

5. $Supp(\mathcal{R}_5) = Supp((x_1 \vee \ x_2 \vee \ \ldots \vee \ x_n) \wedge (y_1 \vee \ y_2 \vee \ \ldots \vee \ y_m)) = Supp(x_1 \vee \ x_2 \vee \ \ldots \vee \ x_n) + Supp(y_1 \vee \ y_2 \vee \ \ldots \vee \ y_m) - Supp(x_1 \vee \ x_2 \vee \ \ldots \vee \ x_n \vee \ y_1 \vee \ y_2 \vee \ \ldots \vee \ y_m)$.

6. $Supp(\mathcal{R}_6) = Supp((x_1 \vee x_2 \vee \ldots \vee x_n) \wedge \overline{y_1} \wedge \overline{y_2} \wedge \ldots \wedge \overline{y_m}) = Supp(\overline{y_1} \wedge \overline{y_2} \wedge \ldots \wedge \overline{y_m}) - Supp(\overline{x_1} \wedge \overline{x_2} \wedge \ldots \wedge \overline{x_n} \wedge \overline{y_1} \wedge \overline{y_2} \wedge \ldots \wedge \overline{y_m}) = \sum_{S \subseteq \{y_1,\ldots,y_m\}} (\text{-}1)^{|S|} Supp(S) - \sum_{Z' \subseteq \{x_1,\ldots,x_n,y_1,\ldots,y_m\}} (\text{-}1)^{|Z'|} Supp(Z')$.

7. $Supp(\mathcal{R}_7) = Supp(\overline{x_1} \wedge \overline{x_2} \wedge \ldots \wedge \overline{x_n} \wedge y_1 \wedge y_2 \wedge \ldots \wedge y_m) = \sum_{S \subseteq \{x_1,\ldots,x_n\}} (\text{-}1)^{|S|} Supp(S \wedge y_1 \wedge \ldots \wedge y_m)$.

8. $Supp(\mathcal{R}_8) = Supp(\overline{x_1} \wedge \overline{x_2} \wedge \ldots \wedge \overline{x_n} \wedge (y_1 \vee y_2 \vee \ldots \vee y_m)) = Supp(\overline{x_1} \wedge \overline{x_2} \wedge \ldots \wedge \overline{x_n}) - Supp(\overline{x_1} \wedge \overline{x_2} \wedge \ldots \wedge \overline{x_n} \wedge \overline{y_1} \wedge \overline{y_2} \wedge \ldots \wedge \overline{y_m}) = \sum_{S \subseteq \{x_1,\ldots,x_n\}} (\text{-}1)^{|S|} Supp(S) - \sum_{Z' \subseteq \{x_1,\ldots,x_n,y_1,\ldots,y_m\}} (\text{-}1)^{|Z'|} Supp(Z')$.

9. $Supp(\mathcal{R}_9) = Supp(\overline{x_1} \wedge \overline{x_2} \wedge \ldots \wedge \overline{x_n} \wedge \overline{y_1} \wedge \overline{y_2} \wedge \ldots \wedge \overline{y_m}) = |\mathcal{O}| - Supp(x_1 \vee \ x_2 \vee \ \ldots \vee \ x_n \vee \ y_1 \vee \ y_2 \vee \ \ldots \vee \ y_m)$.

In this respect, it is worth noting that disjunctive rules as defined in [Bykowski and Rigotti, 2001, Bykowski and Rigotti, 2003] and generalized disjunctive rules as defined in [Kryszkiewicz, 2002] are special cases of $\mathcal{R}_2$. The difference between them lies in the number of items used in the conclusion part. Moreover, an interesting case pointed out by $\mathcal{R}_5$-like rules occurs when $\{y_1, y_2, \ldots, y_m\}$ is an essential itemset and $h(\{y_1, y_2, \ldots, y_m\}) = \{x_1, x_2, \ldots, x_n\} \cup \{y_1, y_2, \ldots, y_m\} = \{x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m\}$ (and, hence, $Supp(\vee \{y_1, y_2, \ldots, y_m\}) = Supp(\vee \{x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m\})$, *cf.* Proposition 39, page 114). Indeed, in such a case, this rule is an exact one since its confidence value is equal to **1**:

$$Conf(\mathcal{R}_5) = \frac{Supp(\vee X) + Supp(\vee Y) - Supp(\vee (X \cup Y))}{Supp(\vee X)} =$$

$$\frac{Supp(\vee \{x_1, x_2, \ldots, x_n\}) + Supp(\vee \{y_1, y_2, \ldots, y_m\}) - Supp(\vee h(\{y_1, y_2, \ldots, y_m\}))}{Supp(\vee \{x_1, x_2, \ldots, x_n\})} = \mathbf{1}.$$

Moreover, $\mathcal{R}_5$ will have a *maximal* premise part and a *minimal* conclusion part, *w.r.t.* the number of items. This is at the opposite of minimal non-redundant rules where premise and conclusion parts are required to be *minimal* and *maximal*, respectively (*cf.* page 26). In addition, it is important to mention that, thanks to the properties of the closure operator h, the rules $\vee X \Rightarrow \vee Y$ and $h(X) \Rightarrow h(Y)$ have the same values of support and confidence.

Once a rule of the form $\mathcal{R}_5$ is extracted, it is straightforward to derive the corresponding one of the form $\mathcal{R}_9$ thanks to De Morgan's law (*cf.* Formula (2) in Lemma 1). Indeed, the support and the confidence of $\mathcal{R}_9$ are expressed as follows *w.r.t.* those of $\mathcal{R}_5$:

$$Supp(\mathcal{R}_9) = |\mathcal{O}| - Supp(\mathcal{R}_5), \text{ and,}$$

$$Conf(\mathcal{R}_9) = \frac{|\mathcal{O}| - Supp(\mathcal{R}_5)}{|\mathcal{O}| - Supp(x_1 \vee x_2 \vee \ldots \vee x_n)}$$

Note also that, thanks to Formula (3), we have $Supp(\mathcal{R}_3) = Supp(x_1 \wedge x_2 \wedge \ldots \wedge x_n)$ - $Supp(\mathcal{R}_2)$.

7.3 Selection of Subsets of Generalized Association Rules

7.3.1 Description of the Selected Subsets

To be able to derive the required information necessary for computing the associated quality measures of an association rule, a concise representation of frequent itemsets is adopted. In our case, we consider as a starting point the concise representation $\mathcal{DSSR}$ based on frequent essential itemsets and their associated disjunctive closed itemsets (*cf.* page 118). This choice is motivated by the fact that essential and disjunctive closed itemsets structurally characterize the associated disjunctive equivalence classes. Hence, they can be successfully used towards extracting representative subsets of generalized association rules. Interestingly, they are respectively the mapping of minimal generators and conjunctive closed itemsets. These latter itemsets were at the roots of the main generic bases of association rules proposed in the literature [Ceglar and Roddick, 2006, Kryszkiewicz, 2002]. This encourages the use of their correspondences within the disjunctive search space.

Example 51 *Consider the context given by Table 2.1 (cf. page 12). Table 7.2 presents the $\mathcal{DSSR}$ representation for minsupp = 1. In this respect, each disjunctive closed itemset is associated to its*

frequent essential itemsets and disjunctive support. Note that even not used during the rule mining step, the empty set is mentioned here for the sake of completeness of $\mathcal{DSSR}$.

Disjunctive closed itemset	Frequent essential itemsets	Disjunctive support
$\emptyset$	$\emptyset$	5
E	E	3
F	F	3
AB	A, B	3
CD	C, D	4
EF	EF	4
ABE	AE, BE	4
ABF	AF, BF	4
ABCDEF	AC, AD, BC, BD, CE, CF, DE, DF, AEF, BEF	5

Table 7.2: The $\mathcal{DSSR}$ representation for $minsupp = \mathbf{1}$.

Starting from $\mathcal{DSSR}$, the disjunctive support of each frequent itemset is at hand since the representation is composed of disjunctive itemsets. Its negative support simply follows using De Morgan's law. In addition, its conjunctive support can be deduced using Lemma 1 (*cf.* page 13). Now, we will present an overview of the process by which we retrieve subsets of generalized association rules and evaluate their associated supports using $\mathcal{DSSR}$. Rules can be classified according to the number of nodes (one or two) required for their extraction. We then distinguish two cases:

- **An intra-node rule:** it is extracted using itemsets standing within the same disjunctive equivalence class. Such a rule highlights relationships between a frequent essential itemset and its disjunctive closure f (here we have an f-based rule).
- **An inter-node rule:** it is extracted using two itemsets belonging to two comparable disjunctive equivalence classes. In this respect, let N_1 and N_2 be the respective nodes representing these classes within a partially ordered structure *w.r.t.* set inclusion. The associated disjunctive closed itemset of N_1, denoted f_1, is one of the immediate predecessors of that of N_2, denoted f_2. Let e_1 be a frequent essential itemset of f_1. An inter-node rule describes relationships between either f_1 and f_2 or e_1 and f_2 (here we have an f_2-based rule).

Both kinds of rules – intra-node and inter-node – can either be exact or approximate. [2] To reduce the number of mined rules, we mainly consider four rule forms under some constraints on the content of the premise and the conclusion parts. This is detailed in the following paragraphs.

Let X and Y be two itemsets such that either X or Y is a frequent essential itemset or a disjunctive closed one, and $Z = X \cup Y$ is a disjunctive closed itemset. The considered forms under the constraint

[2] It is worth noting that, in the classic association rule framework, an intra-node rule mined from a conjunctive equivalence class is always found to be an exact one.

on the premise X and the conclusion Y are as follows as well as the way of computation of the associated support:

- **Form 1:** `disjunction of items in premise and conclusion` $\vee X \Rightarrow \vee Y$: $Supp(\vee X \Rightarrow \vee Y) = Supp((\vee X) \wedge (\vee Y)) = Supp(\vee X) + Supp(\vee Y) - Supp((\vee X) \vee (\vee Y)) = Supp(\vee X) + Supp(\vee Y) - Supp(\vee Z)$,

- **Form 2:** `negation of items in premise and conclusion` $\overline{X} \Rightarrow \overline{Y}$: $Supp(\overline{X} \Rightarrow \overline{Y}) = Supp(\overline{X} \wedge \overline{Y}) = Supp(\overline{((\vee X) \vee (\vee Y))}) = Supp(\overline{Z}) = |\mathcal{O}| - Supp(\vee Z)$,

- **Form 3:** `disjunction of items in premise and negation of items in conclusion` $\vee X \Rightarrow \overline{Y}$: $Supp(\vee X \Rightarrow \overline{Y}) = Supp((\vee X) \wedge \overline{Y}) = Supp((\vee X) \vee (\vee Y)) - Supp(\vee Y) = Supp(\vee Z) - Supp(\vee Y)$, and,

- **Form 4:** `negation of items in premise and disjunction of items in conclusion` $\overline{X} \Rightarrow \vee Y$: $Supp(\overline{X} \Rightarrow \vee Y) = Supp(\overline{X} \wedge (\vee Y)) = Supp((\vee X) \vee (\vee Y)) - Supp(\vee X) = Supp(\vee Z) - Supp(\vee X)$.

Form 1 (*resp.* **2**, **3** and **4**) we select corresponds to an instantiation of the form $\mathcal{R}_5$ (*resp.* $\mathcal{R}_9$, $\mathcal{R}_6$ and $\mathcal{R}_8$) described in the previous section. Indeed, here we require the premise or the conclusion to be a frequent essential itemset and the rule to be based on a disjunctive closed itemset. Consequently, for each rule, the support of Z is known since it belongs to $\mathcal{DSSR}$. It is the same for either X or Y since one of them is assumed to be a frequent essential itemset or a disjunctive closed itemset. Once the respective supports of X, Y and Z are obtained, the derivation of the associated rules consists in simple arithmetic operations for computing the associated support and confidence values.

We are mainly interested in the aforementioned rule forms since there is a lack in the literature of algorithms designed for their extraction, especially those relying on the disjunction connector. The selected rules convey relationship between specific itemsets, namely disjunctive closed itemsets and essential ones. This restriction makes it possible to avoid the extraction of an overwhelming number of valid rules in the case where any itemset is allowed to be used in the premise or conclusion parts. Noteworthily, the used concise representation for frequent itemsets, namely $\mathcal{DSSR}$, is suitable for such an extraction. Indeed, it is composed of disjunctive closed itemsets and essential ones and, hence, does not require deriving their respective disjunctive supports. This allows, for example, the efficient derivation of association rules involving disjunction/negation of items in their respective premises and conclusions. On the contrary, many algorithms for extracting all valid positive association rules exist in the literature, mainly based on the pioneer Apriori algorithm [Agrawal and Srikant, 1994]. Moreover, the described process is complementary to that covered by both the Prince [Hamrouni *et al.*, 2005b] and the IMG_Extractor [Hamrouni *et al.*, 2006] algorithms we proposed for extracting lossless subsets of positive association rules as well as the adaptation of the Prince algorithm for mining generic association rules based on literalsets [Gasmi *et al.*, 2007]. Note however that the proposed process here also allows to compute the support and confidence values of each rule involving a conjunction of items, like in the form $\mathcal{R}_1$ (*cf.* previous section). Furthermore, Formula (4) makes it possible to consider a rule based on a literalset, whose positive variation is a frequent itemset. [3] Indeed, its support will be computed using some subsets of the

[3] The positive variation of $\{x_1, x_2, \ldots, x_n, \overline{y_1}, \overline{y_2}, \ldots, \overline{y_m}\}$ is equal to $\{x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m\}$.

positive variation, which is frequent. These subsets are then also frequent and, hence, their supports are exactly derivable from our representation.

For each of the four rule forms, Table 7.3 summarizes the different possible content of the premise and conclusion parts. In this table, f_1 and f_2 denote two disjunctive closed itemsets *s.t.* f_1 is an immediate predecessor of f_2, while e_1 denotes a frequent essential itemset of f_1.

Form 1				**Form 2**			
R_{11}	$\vee e_1 \Rightarrow \vee(f_1 \backslash e_1)$	R_{12}	$\vee(f_1 \backslash e_1) \Rightarrow \vee e_1$	R_{21}	$\overline{e_1} \Rightarrow \overline{(f_1 \backslash e_1)}$	R_{22}	$\overline{(f_1 \backslash e_1)} \Rightarrow \overline{e_1}$
R_{13}	$\vee e_1 \Rightarrow \vee(f_2 \backslash e_1)$	R_{14}	$\vee(f_2 \backslash e_1) \Rightarrow \vee e_1$	R_{23}	$\overline{e_1} \Rightarrow \overline{(f_2 \backslash e_1)}$	R_{24}	$\overline{(f_2 \backslash e_1)} \Rightarrow \overline{e_1}$
R_{15}	$\vee f_1 \Rightarrow \vee(f_2 \backslash f_1)$	R_{16}	$\vee(f_2 \backslash f_1) \Rightarrow \vee f_1$	R_{25}	$\overline{f_1} \Rightarrow \overline{(f_2 \backslash f_1)}$	R_{26}	$\overline{(f_2 \backslash f_1)} \Rightarrow \overline{f_1}$
Form 3				**Form 4**			
R_{31}	$\vee e_1 \Rightarrow \overline{(f_1 \backslash e_1)}$	R_{32}	$\vee(f_1 \backslash e_1) \Rightarrow \overline{e_1}$	R_{41}	$\overline{e_1} \Rightarrow \vee(f_1 \backslash e_1)$	R_{42}	$\overline{(f_1 \backslash e_1)} \Rightarrow \vee e_1$
R_{33}	$\vee e_1 \Rightarrow \overline{(f_2 \backslash e_1)}$	R_{34}	$\vee(f_2 \backslash e_1) \Rightarrow \overline{e_1}$	R_{43}	$\overline{e_1} \Rightarrow \vee(f_2 \backslash e_1)$	R_{44}	$\overline{(f_2 \backslash e_1)} \Rightarrow \vee e_1$
R_{35}	$\vee f_1 \Rightarrow \overline{(f_2 \backslash f_1)}$	R_{36}	$\vee(f_2 \backslash f_1) \Rightarrow \overline{f_1}$	R_{45}	$\overline{f_1} \Rightarrow \vee(f_2 \backslash f_1)$	R_{46}	$\overline{(f_2 \backslash f_1)} \Rightarrow \vee f_1$

Table 7.3: The selected association rule forms.

The association rules of **Form 1** are shown in couples (R_{1i}, $R_{1(i+1)}$) with $i \in \{1, 2, 3\}$, *s.t.* the premise part of R_{1i} constitutes the conclusion part of $R_{1(i+1)}$, and vice versa. Such rules are ***reversed*** *w.r.t.* the content of the premise and conclusion parts. It is the same for rules of **Form 2**. Let us for example analyze the couple of rules (R_{11}, R_{12}) of **Form 1**, *i.e.*, R_{11}: $\vee e_1 \Rightarrow \vee(f_1 \backslash e_1)$ and R_{12}: $\vee(f_1 \backslash e_1) \Rightarrow \vee e_1$. Considering simultaneously both rules R_{11} and R_{12} aims at bringing information about the possible correlation between the disjunctive itemsets $\vee e_1$ and $\vee(f_1 \backslash e_1)$. In this respect, both rules have the same support value. While their respective confidence values depend on the support of their associated premise part. In the case where $Supp(\vee e_1) = Supp(\vee(f_1 \backslash e_1))$, then both rules R_{11} and R_{12} will have the same support and confidence values. Hence, their associated premise and conclusion parts depend one on the other by the same degree.

Each rule R_{3j} ($j \in \{1, 2, 3, 4, 5, 6\}$) of **Form 3** has its reverse in **Form 4**. For example, the reverse of R_{31} is R_{42}. In this respect, the rules R_{32}: $\vee(f_1 \backslash e_1) \Rightarrow \overline{e_1}$ (of **Form 3**) and R_{41}: $\overline{e_1} \Rightarrow \vee(f_1 \backslash e_1)$ (of **Form 4**) are given in Table 7.3 only for illustrative purpose. Indeed, both rules will always be discarded since having a support equal to **0**: $Supp(R_{32}) = Supp(R_{41}) = Supp((\vee(f_1 \backslash e_1)) \wedge \overline{e_1}) = Supp((\vee(f_1 \backslash e_1)) \vee (\vee e_1))$ - $Supp(\vee e_1) = Supp(\vee f_1)$ - $Supp(\vee e_1) = \mathbf{0}$ (e_1 is an essential itemset of f_1 and, hence, $Supp(\vee f_1) = Supp(\vee e_1)$).

Thus, in the general case, a rule R_1 has the same support as its reverse R_2. However, their confidences depend on the support of the premise and that of the conclusion of R_1, respectively. Thus, the validity of R_1 does not imply that R_2 is valid, unless the conclusion part of R_1 (being the premise part of R_2) has a support lower than or equal to that of the premise part of R_1. Experimental results will reveal that, in many cases, there are as many valid rules as valid reverse rules. Note also that the confidence measure of R_1 is equal to the *recall* measure [Geng and Hamilton, 2006] of R_2, and vice versa.

The next subsections explain two complementary tasks: the first focuses on how are respectively assessed the quality measures of the selected rules. The second subsection describes the elimination process of duplicated rules.

7.3.2 Assessing Quality Measures of Selected Rules

In this subsection, we will concentrate on the assessment of the respective support and confidence values of the selected generalized association rules. The same process as that we describe here applies for the remaining quality measures, using the rules support and those of their associated premises and conclusions [Geng and Hamilton, 2006, Hébert and Crémilleux, 2007]. These measures can be considered and their values used towards selecting the most appropriate ones for each association rule form.

In the remainder, for the sake of simplicity, we assume that X is a frequent essential itemset or a disjunctive closed itemset. Since $Y = Z \backslash X$, then Y does not necessarily belong to $\mathcal{DSSR}$ and, may even not be a frequent itemset. Nevertheless, its disjunctive support may be required to assess the interestingness measures of the associated rule (like in **Form 1**). To this end, we bound the support of Y using a lower bound, denoted *lb_Supp*, and an upper bound, denoted *ub_Supp*. These bounds are shown by Definition 72. This definition requires that we introduce specific subsets of the sets $\mathcal{FEI}$ and $\mathcal{EDCI}$ *w.r.t.* Y. This is done as follows:

Definition 71 (Minimal supersets and Maximal subsets)
Let $Y \subseteq \mathcal{I}$. The minimal supersets and maximal subsets of Y are as follows:

- *The set of minimal supersets of Y in $\mathcal{EDCI}$ is defined as follows:* $\text{Minimal\_Supersets}(Y) = min_{\subseteq}\{f \in \mathcal{EDCI} \mid Y \subseteq f$ *and* $\nexists\, f_1 \in \mathcal{EDCI}$ *s.t.* $Y \subset f_1 \subset f\}$.
- *The set of maximal subsets of Y in $\mathcal{FEI}$ is defined as follows:* $\text{Maximal\_Subsets}(Y) = max_{\subseteq}\{e \in \mathcal{FEI} \mid e \subseteq Y$ *and* $\nexists\, e_1 \in \mathcal{FEI}$ *s.t.* $e \subset e_1 \subset Y\}$.

The bounds are defined as follows:

Definition 72 (Upper and Lower bounds of disjunctive support)
Let $Y \subseteq \mathcal{I}$. The upper and lower bounds of the disjunctive support of Y are defined as follows:

- $ub\_Supp(\vee Y) = min\{Supp(\vee f) \mid f \in \text{Minimal\_Supersets}(Y)\}$,
- $lb\_Supp(\vee Y) = max\{Supp(\vee e) \mid e \in \text{Maximal\_Subsets}(Y)\}$.

Both sets $\text{Minimal\_Supersets}(Y)$ and $\text{Maximal\_Subsets}(Y)$ optimize the computation of the upper and lower bounds, respectively. Indeed, their introduction mainly relies on the fact that the disjunctive support proportionally decreases *w.r.t.* the reduction of itemsets size. Conversely, it augments whenever the itemsets size increases. Thus, to obtain the upper bound, it is sufficient to consider the minimal supersets among disjunctive closed itemsets covering Y. Whereas to get the lower bound, it is sufficient to consider maximal subsets among frequent essential itemsets contained in Y.

An interesting situation happens if Y belongs to $\mathcal{DSSR}$, or is encompassed between a frequent essential itemset and its disjunctive closure. In this case, $lb\_Supp(\vee Y) = ub\_Supp(\vee Y)$. Hence, the support and the confidence of each rule where Y is involved will be exactly computed. Otherwise, the value of support and that of confidence will be, respectively, bounded by a minimal and a maximal possible value using the bounds associated to the support of Y. This last case may lead to the appearance of a third type

of rules – in addition to exact and approximate – denoted *approximated rules*. Such rules are defined as follows:

Definition 73 (Approximated association rule)
An association rule is said to be approximated if it has either its support or its confidence not exactly determined.

Then, only approximated rules having minimum possible values of support and confidence greater than or equal to *minsupp* and *minconf*, respectively, will be retained. Note that an approximated rule is different from an approximate rule in the sense that the latter has its support and confidence exactly computed (with a confidence value lower than **1**), which is not the case of the former. Such approximated rules were shown to be of added value in the case of positive rules [Boulicaut *et al.*, 2003, Cheng *et al.*, 2008, Kanda *et al.*, 2001].

Noteworthily, the bounds $lb\_Supp(\vee Y)$ and $ub\_Supp(\vee Y)$ always exist. Indeed, since the set of items $\mathcal{I}$ is pruned *w.r.t. minsupp*, then Y will be composed of frequent items even if it is infrequent. These items are obviously frequent essential itemsets of size **1**, which ensures the existence of the lower bound $lb\_Supp(\vee Y)$. The itemset Y is also covered by at least a disjunctive closed itemset, namely Z, which ensures the existence of the upper bound $ub\_Supp(\vee Y)$.

Example 52 *Consider Table 7.2 depicting the $\mathcal{DSSR}$ representation associated to Table 2.1 (cf. page 12) and minsupp = **1**. Let minconf = **0.7**. Consider the intra-node rule R_1 of **Form 1** based on the disjunctive closed itemset ABCDEF and its frequent essential itemset AC: $\vee AC \Rightarrow \vee BDEF$. $Supp(R_1)$ = $Supp(\vee AC)$ + $Supp(\vee BDEF)$ - $Supp(\vee ABCDEF)$ = $Supp(\vee BDEF)$. Indeed, AC and ABCDEF belong to the same equivalence class. Since BDEF is neither a frequent essential itemset nor a disjunctive closed one, we need to evaluate its support using $\mathcal{DSSR}$. We have $BD \subseteq BDEF \subseteq h(BD) = ABCDEF$, then $lb\_Supp(\vee BDEF)$ = $ub\_Supp(\vee BDEF)$ = **5**. Hence, $Supp(R_1)$ = **5** and $Conf(R_1)$ = **1**. R_1 is hence a valid exact rule.*

*Consider now the inter-node rule R_2 of **Form 1** based on ABCDEF and its immediate predecessors ABF: $\vee ABF \Rightarrow \vee CDE$. $Supp(R_2)$ = $Supp(\vee ABF)$ + $Supp(\vee CDE)$ - $Supp(\vee ABCDEF)$. We will assess the support of CDE since not belonging to $\mathcal{DSSR}$. Since $CE \subseteq CDE \subseteq h(CE)$, then $lb\_Supp(\vee CDE)$ = $ub\_Supp(\vee CDE)$ = **5**. Hence, $Supp(R_2)$ = **4** + **5** - **5** = **4** and $Conf(R_1)$ = **1**. R_2 is also a valid exact rule although it relies on itemsets belonging to different equivalence classes, namely ABF and ABCDEF. Here, we took X = ABF. If we set Y = ABF in the sense that we consider ABF as a conclusion instead of a premise, then the obtained rule R_3: $\vee CDE \Rightarrow \vee ABF$ will have the same support as R_2 while being approximate. Indeed, its confidence value is equal to* $\frac{Supp(R_3)}{Supp(\vee CDE)} = \frac{4}{5} =$ **0.8**.

*Let us look to the inter-node rule R_3 of **Form 2** based on ABCDEF and its immediate predecessors ABF: $\overline{ABF} \Rightarrow \overline{CDE}$. In this case, $Supp(R_2)$ = $|\mathcal{O}|$ - $Supp(\vee ABCDEF)$ = **5** - **5** = **0**. Hence, this rule is not valid. It is the same for the rule $\overline{CDE} \Rightarrow \overline{ABF}$.*

As mentioned above, if Y is not encompassed between a frequent essential itemset and its disjunctive closure, then its support cannot be exactly determined and will simply be bounded. Now, let us discuss in the general case the effect of $lb\_Supp(\vee Y)$ and $ub\_Supp(\vee Y)$ on the minimum and maximum bounds that will have the support and confidence values of the associated rule. This is closely related to two main facts:

1. The sign "+" or "-" of the support of Y within the formula of the support of the rule, *i.e.*, the support of Y will be subtracted or added. For example, in **Form 1**, the sign is "+", while it is "-" in **Form 3**. In this respect, if the associated sign is "-", the maximal (*resp.* minimal) possible value of support of Y, *i.e.*, $ub\_Supp(\vee Y)$ (*resp.* $lb\_Supp(\vee Y)$), will lead to the minimal (*resp.* maximal) value of the support of the associated rule. The opposite reasoning applies for the sign "+".

2. The position of Y in the associated rule, *i.e.*, in the premise or conclusion part. Indeed, if Y is in the conclusion part, then its support will only be used in the computation of the support of the associated rules. While, if Y is in the premise part, its support will also contribute to the computation of the confidence of the rule. In this latter case, $ub\_Supp(\vee Y)$ (*resp.* $lb\_Supp(\vee Y)$) will lead to the minimal (*resp.* maximal) possible value of the confidence of the associated rule.

The different possible cases are summarized in Table 7.4. In this table, $ub\_Supp(R)$ (*resp.* $lb\_Supp(R)$) denotes the maximal (*resp.* minimal) possible value of the support of the association rule R. While, $ub\_Conf(R)$ (*resp.* $lb\_Conf(R)$) denotes the maximal (*resp.* minimal) possible value of the confidence of the association rule R. The symbol "□" indicates that the bound value of the support of Y does not affect neither the support nor the confidence bounds of R.

Associated bound	**Associated sign**		**Associated position**	
	"+"	"-"	Premise	Conclusion
$ub\_Supp(\vee Y)$	$ub\_Supp(R)$	$lb\_Supp(R)$	$lb\_Conf(R)$	□
$lb\_Supp(\vee Y)$	$lb\_Supp(R)$	$ub\_Supp(R)$	$ub\_Conf(R)$	□

Table 7.4: Summary of the approximations.

Example 53 *No approximated rule can be extracted starting from the DSSR representation depicted by Table 7.2. Indeed, the support of Y is always exactly derived. Let us consider then the extraction context of Chapter 6 (cf. Table 6.1, page 110). This context offers an interesting situation to illustrate the content of this subsection. Let minsupp = 1 and minconf = 0.5. As we saw in the aforementioned chapter, the disjunctive closure of the frequent essential itemset A is equal to ABCD. Consider now the following couple of rules of* ***Form 1****: R_1: $\vee A \Rightarrow \vee BCD$, and R_2: $\vee BCD \Rightarrow \vee A$. In this case, $Y = BCD$. However, BCD is an infrequent itemset whose disjunctive closure is equal to itself. We hence need to assess the disjunctive support of BCD. Since* MINIMAL_SUPERSETS(*BCD*) = {*ABCD*}, *and* MAXIMAL_SUBSETS(*BCD*) = {*BC, BD, CD*}, *we then deduce that:*

• $ub\_Supp(\vee BCD) = min\{Supp(\vee f) \mid f \in$ MINIMAL_SUPERSETS$(BCD)\} = min\{Supp(\vee BCD)\} = min\{\mathbf{7}\} = \mathbf{7}$.

• $lb\_Supp(\vee BCD) = max\{Supp(\vee e) \mid e \in$ MAXIMAL_SUBSETS$(BCD)\} = max\{Supp(\vee BC), Supp(\vee BD), Supp(\vee CD)\} = max\{\mathbf{5}, \mathbf{5}, \mathbf{5}\} = \mathbf{5}$.

The sign associated to BCD in the support formula of the rules R_1 and R_2 is "+". Indeed, $Supp(R_1) = Supp(R_2) = Supp(\vee A) + Supp(\vee BCD)$ - $Supp(\vee ABCD) = Supp(\vee BCD)$, since $Supp(\vee A) = Supp(\vee ABCD)$. Consequently, R_1 and R_2 will share the same bounds:

- $ub\_Supp(R_1) = ub\_Supp(R_2) = ub\_Supp(\vee BCD) = \mathbf{7}$.
- $lb\_Supp(R_1) = lb\_Supp(R_2) = lb\_Supp(\vee BCD) = \mathbf{5}$.

However, with respect to the confidence measure, we will not have the same scenario. Indeed, the position of BCD *– as a premise or a conclusion of the associated rule – will play a key role. In this respect,*

- $Conf(R_1) = \frac{Supp(R_1)}{Supp(A)} = \frac{Supp(\vee BCD)}{Supp(A)}$. *Hence,* $ub\_Conf(R_1) = \frac{7}{7}$, *while* $lb\_Conf(R_1) = \frac{5}{7}$.
- $Conf(R_2) = \frac{Supp(R_2)}{Supp(\vee BCD)} = \frac{Supp(\vee BCD)}{Supp(\vee BCD)} = \mathbf{1}$. *Hence,* $ub\_Conf(R_2) = lb\_Conf(R_2) = \mathbf{1}$.

The case of R_2 *is interesting, since although its support is only bounded, the value of its confidence is exactly computed.*

7.3.3 Eliminating Duplicated Rules

In this section, we focus on the selected rule forms (*cf.* Table 7.3, page 157) in order to avoid extracting the same intra- or inter-node rule more than once. Our aim is to locate the scenarios involving such situations. The different cases are discussed in the following paragraphs. We will use the $\mathcal{DSSR}$ representation associated to our running context (*cf.* Table 7.2, page 155) as a basis for illustrative examples.

- **Scenario 1:** If a frequent essential itemset e_1 is equal to its disjunctive closure f_1, then we did not use e_1 during the rule derivation step. This avoids extracting duplicated rules involving either e_1 or f_1, being equal. The use of f_1 only is indeed sufficient.

Example 54 *Let* f_2 *be an immediate successor of* f_1 *and consider for example* f_1 (*resp.* e_1 *and* f_2) *as equal to* F (*resp.* F *and* EF). *Since* $f_2 \backslash f_1 = f_2 \backslash e_1 = E$, *only rules involving* f_1 *and* f_2 *need to be mined. Indeed, let us look at the following couple of rules of* ***Form 1****:* R_1*:* $\vee(f_2 \backslash f_1) \Rightarrow \vee f_1$, *and* R_2*:* $\vee(f_2 \backslash e_1) \Rightarrow \vee e_1$. *Since* $e_1 = f_1 = F$, *then* $R_1 \equiv R_2$.

- **Scenario 2:** Suppose that a disjunctive closed itemset f has exactly two distinct associated essential itemsets e_1 and e_2 such that $e_1 \cup e_2 = f$ and $e_1 \cap e_2 = \emptyset$. In this case, it is sufficient to extract `intra-node` rules associated to only one essential itemset (either e_1 or e_2), since $f \backslash e_1 = e_2$ and, dually, $f \backslash e_2 = e_1$.

Example 55 *Let* f (*resp.* e_1 *and* e_2) *be equal to* AB (*resp.* A *and* B). *Since* $AB \backslash A = B$, *we only use either* A *or* B *for extracting intra-node rules. Indeed, the rules involving* A *are the same as those relying on* B. *This avoids duplicating generation of the same rules.*

- **Scenario 3:** Following the same spirit as the previous case, suppose that a disjunctive closed itemset f has exactly two immediate predecessors f_1 and f_2 *s.t.* $f_1 \cup f_2 = f$ and $f_1 \cap f_2 = \emptyset$. Then, it is sufficient to extract `inter-node` rules using either f_1 or f_2.

Example 56 *Let* f (*resp.* f_1 *and* f_2) *be equal to* ABE (*resp.* AB *and* E). *Since* $ABE \backslash AB = E$, *each rule involving* ABE *and* AB *has its duplicate within rules invoking both* ABE *and* E. *Thus, only using* AB *is sufficient.*

It is important to mention that the frequent essential itemsets associated to f_1 and f_2 do not involve any duplication and, hence, are used without any restriction.

Remark 8 *Suppose that a disjunctive closed itemset f has an essential itemset e and an immediate predecessor or one of its essential itemsets, say q, s.t. $q \cup e = f$ and $q \cap e = \emptyset$. Although this case also leads to duplicated rules (the same reasoning applies as the previous two cases), we will tolerate such a case since the associated duplicated rules do not involve the same disjunctive equivalence classes. Indeed, rules extracted using f and e are intra-node ones, since belonging to the same equivalence class. While those based on f and q are inter-node ones, since belonging to two comparable classes. This distinction may help interpreting such rules.*
Let us for example consider f (resp. e and q) as equal to `ABE` *(resp.* `AE` *and* `B`*).* `B` *is a frequent essential itemset associated to the closure* `AB` *which is an immediate predecessor of* `ABE`*. Each rule involving* `ABE` *and* `AE` *will be an intra-node rule. It will have its duplicate when using* `ABE` *and* `B`*. Nevertheless, the obtained rule in this latter case will be an inter-node one since* `ABE` *and* `B` *do not belong to the same equivalence class, contrary to* `ABE` *and* `AE`*.*

7.4 Extraction of Generalized Association Rules

In this section, we describe the process by which the selected generalized association rules will be extracted. In this respect, in order to derive inter-node rules (*cf.* previous section), disjunctive closed itemsets need to be sorted *w.r.t.* set inclusion. Then, a complementary algorithm will be used for deriving generalized rules. This is detailed in the following paragraphs.

7.4.1 Building the Partially Ordered Structure

Here, we describe an algorithm for building a partially ordered structure amongst disjunctive closed itemsets. This structure is formally defined as follows:

Definition 74 (Partially ordered structure)
The set of disjunctive closed itemsets $\mathcal{EDCI}$ forms a partially ordered structure $\mathcal{L} = (\mathcal{EDCI}, \subseteq)$ when $\mathcal{EDCI}$ is sorted with set inclusion between disjunctive closed itemsets. In this structure, each element f in $\mathcal{EDCI}$ is connected to the set of its immediate predecessors forming its lower cover:

- $Cov_l(f) = \{f_1 \mid f_1 \in \mathcal{EDCI}$ *and* $f_1 \subset f$ *and* $\nexists\, f_2 \in \mathcal{EDCI}$ *s.t.* $f_1 \subset f_2 \subset f\}$.

It is also connected to the set of its immediate successors forming its upper cover:

- $Cov^u(f) = \{f_1 \mid f_1 \in \mathcal{EDCI}$ *and* $f \subset f_1$ *and* $\nexists\, f_2 \in \mathcal{EDCI}$ *s.t.* $f \subset f_2 \subset f_1\}$.

The construction of this structure is carried out using a new algorithm, called POSB. [4] The POSB algorithm takes as input the $\mathcal{DSSR}$ representation *s.t.* to each disjunctive closed itemset is associated its set of frequent essential itemsets and disjunctive support. A node in the partially ordered structure will be associated to each disjunctive closed itemset.

The pseudo-code of POSB is shown by Algorithm 10, while its associated notations are summarized in Table 7.5. Our algorithm inherits two main optimizations used in the literature by algorithms dedicated

[4] POSB is the acronym of Partially Ordered Structure Builder.

to the Hasse diagram building (like [Baixeries *et al.*, 2009, Valtchev *et al.*, 2000] [5]). These optimizations are the sorting of disjunctive closed itemsets, and the use of a border. Indeed, the set of disjunctive closed itemsets $\mathcal{EDCI}$ is sorted *w.r.t.* the increasing itemset size. Since closures of equal size are not comparable, this sorting avoids unnecessary comparisons. In addition, it makes possible that the closure under treatment to be of the largest size in comparison to the already handled closures. Thus, it suffices to find its lower cover among the nodes inserted in the partially ordered structure. On the other hand, the border is found to be an anti-chain *w.r.t.* set inclusion containing maximal closures among those already treated.

However, the algorithms proposed in the literature do not directly fit to our situation. Indeed, although the intersection of two disjunctive closed itemsets is obviously a disjunctive closed itemset, this latter does not necessarily belong to $\mathcal{EDCI}$. This is due to the fact that it could have all its essential itemsets infrequent and, hence, has been already pruned. On their side, the algorithms for building the Hasse diagram mainly rely on the fact that the intersection of two concepts was already treated and it suffices to locate the corresponding node within the already built part of the Hasse diagram. Recall that for obtaining the set $\mathcal{EDCI}$, two constraints of different natures were combined: a monotone constraint through the disjunctive support and an anti-monotone one through setting *minsupp* (*cf.* page 13). Consequently, some disjunctive closed itemsets resulting from the intersection of others may be pruned since they have all their essential itemsets infrequent. This is illustrated thanks to the following example.

Example 57 *Consider a context containing the following transactions:* A, B, ABC, ABD, *and* $ABCD$. *Let* $minsupp = \mathbf{2}$. *In this situation, the set of frequent essential itemsets* $\mathcal{FEI}$ *is equal to* $\{A, B, C, D, AB\}$. *The associated set of disjunctive closed itemsets* $\mathcal{EDCI}$ *is then* $\{C, D, ACD, BCD, ABCD\}$. *By intersecting the closures* ACD *and* BCD, *the result is* CD *which is not present in* $\mathcal{EDCI}$ *since the associated essential itemset, namely itself, is infrequent since* $Supp(CD) = \mathbf{1}$.

In the case of the Valtchev et al. algorithm, the elements to be sorted are associated to the Galois closure operator. More precisely, they correspond to the conjunctive closed itemsets. For $minsupp = \mathbf{2}$, *they form the set of frequent closed itemsets* $\mathcal{FCI}$ *equal to* $\{\emptyset, A, B, AB, ABC, ABD, ACD, ABCD\}$. *In this case, the intersection of each couple of elements from* $\mathcal{FCI}$ *also belongs to* $\mathcal{FCI}$.

The POSB algorithm incrementally inserts disjunctive closed itemsets one at a time to a structure which is only partially finished to obtain at the end the entire one (*cf.* Algorithm 10). Let f be the current disjunctive closed itemset to be inserted in the partially ordered structure. f will be compared to the elements of the border $\mathcal{B}$ (*cf.* lines 5-11). If an element $b \in \mathcal{B}$ is included in f (*cf.* lines 7-9), then it is an element of its lower cover. A link between the node representing b and that representing f will be constructed thanks to the Lower_Cover_Insertion procedure (*cf.* Algorithm 11). The element b will then be deleted from the border. If b is not included in f but its intersection with f is not empty (*cf.* lines 10-11), then the Lower_Cover_Management procedure will identify the common immediate predecessors of both b and f (*cf.* Algorithm 12). Finally, f will be added to the border. It is important to note that in the Lower_Cover_Management procedure, a prohibited list is associated to each disjunctive closed itemset to be inserted in the partially ordered structure. Indeed, when updating the

[5] Both described algorithms in [Baixeries *et al.*, 2009, Valtchev *et al.*, 2000] construct the Hasse diagram representing the subset-superset relationship among concepts in the Galois lattice. They begin at the bottom of the lattice and then recursively identify the lower neighbors of each concept.

f	:	A disjunctive closed itemset.
$\mathcal{B}$	:	The set containing the elements of the border.
b	:	An element of the border $\mathcal{B}$.
$Prohibited\_List$	:	The list of the treated nodes in the partially ordered structure.

Table 7.5: Notations used by the POSB algorithm.

precedence link between disjunctive closed itemsets, a node can be visited more than once since it can be an immediate predecessor of many other nodes. This list will avoid such useless treatments by only allowing the visit of nodes that do not belong to it.

Algorithm 10: POSB

Input: - The set $\mathcal{EDCI}$ of disjunctive closed itemsets.

Output: - The set $\mathcal{EDCI}$ ordered by set inclusion.

```
1  Begin
2    B := ∅ ;
3    ForEach (f ∈ EDCI) Do
4      Prohibited_List = ∅;
5      ForEach (b ∈ B) Do
6        inter := b ∩ f;
7        If (inter = b) Then
8          Lower_Cover_Insertion(f, b);
9          B := B \ b;
10       Else If (inter ≠ ∅) Then
11         Lower_Cover_Management(f, b);
12     B := B ∪ f ;
13 End
```

Example 58 *Consider the set of disjunctive closed itemsets sorted by increasing size in Table 7.2. The first two disjunctive closed itemsets* `E` *and* `F` *share the same size. They are hence immediately inserted in the border, since no precedence relation can link them. After that, the disjunctive closed itemset* `AB` *is compared to the element of the border, i.e.,* `E` *and* `F`*. Since none of them is included in* `AB`*, this closure is simply added to the border. It is the same for the disjunctive closed itemset* `CD`*.*

At this step, the border is composed by `E`*,* `F`*,* `AB` *and* `CD`*. The closure under treatment now is* EF*. Since both* `E` *and* `F` *are included in* `EF`*, they will be removed from the border and inserted as an immediate predecessor of* `EF`*. This is done thanks to a call to the* Lower_Cover_Insertion *procedure. Then,* `ABE` *will be compared to the element of the border. Since* `AB` *is included in* `ABE`*, it will also be removed from the border and set as an immediate predecessor of* `ABE`*. The intersection of this latter with* `EF` *is neither equal to*

Algorithm 11: LOWER_COVER_INSERTION

Input: - A disjunctive closed itemset f, and an element *pred* to be inserted in its lower cover.

Output: - The updated lower cover of f.

```
1 Begin
2   ForEach (l ∈ Cov_l(f)) Do
3     inter := l ∩ pred;
4     If (inter = pred) Then
5       return;
6     Else If (inter = l) Then
7       Cov_l(f) := Cov_l(f) \ l;
8   Cov_l(f) := Cov_l(f) ∪ pred;
9 End
```

EF nor to the empty set. Consequently, the POSB *algorithm calls the* LOWER_COVER_MANAGEMENT *procedure. This latter procedure will search for the immediate predecessors of* ABE *among the predecessors of* EF*. It also stores in a prohibited list the visited nodes which makes it possible to avoid performing the same treatment more than once. The precedence relation between* ABE *and its immediate predecessor* E *is then established.*

The border now contains EF*,* CD*, and* ABE *to which will be compared the closure to be inserted in the partially ordered structure, i.e.,* ABF*. The intersection of this latter with* EF *is equal to* F*, which will be inserted as an immediate predecessor of* ABF*. It is the same for* AB *w.r.t. the element of border* ABE*.* ABF *will then be added to the border without removing any element since it covers none of them. The last closure to be treated is* ABCDEF*. Since it subsumes all elements of the border, a precedence link will be established between each element and* ABCDEF*.*

At the end of the POSB *algorithm execution, we obtain the partially ordered structure illustrated by Figure 7.1. In this figure, the disjunctive closed itemsets listed in Table 7.2 are partially ordered w.r.t. set inclusion. Each one of them represents an equivalence class, to which is associated the corresponding frequent essential itemsets and disjunctive support.*

Thus, the construction of the precedence links in our situation requires more attention than in the case of manipulating conjunctive closed itemsets. Noteworthily, the POSB algorithm can be used in the general case for partially ordering itemsets pruned *w.r.t.* a conjunction of constraints of different natures (like the monotone and anti-monotone ones in our case).

The next theorem states the soundness and the correctness of the POSB algorithm.

Theorem 17 *The* POSB *algorithm is sound and correct. It exactly determines the lower cover of each closure belonging to* $\mathcal{EDCI}$*.*

Algorithm 12: LOWER_COVER_MANAGEMENT

Input: - A disjunctive closed itemset f, and an element b of the border $\mathcal{B}$.

Output: - The updated lower cover of f.

```
1  Begin
2    ForEach (pred_b ∈ Cov_l(b)) Do
3      If (pred_b ∉ Prohibited_List) Then
4        inter := pred_b ∩ f;
5        If (inter = pred_b) Then
6          LOWER_COVER_INSERTION(f, pred_b);
7        Else If (inter ≠ ∅) Then
8          LOWER_COVER_MANAGEMENT(f, pred_b);
9        Prohibited_List := Prohibited_List ∪ pred_b;
10 End
```

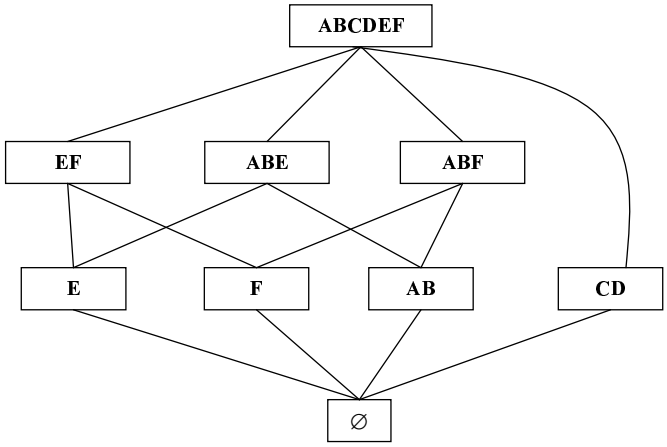


Figure 7.1: The partially ordered structure associated to the disjunctive closed itemsets given by Table 7.2.

Proof. The POSB algorithm sorts the closure of $\mathcal{EDCI}$ by increasing size. Thus, during its processing, a disjunctive closed itemsets $f \in \mathcal{EDCI}$ is necessarily of size higher than or equal to those already inserted in the partially ordered structure. Hence, all elements belonging to its lower cover were already treated. In addition, f will be correctly linked to its lower cover. Indeed, the comparison, *w.r.t.* set inclusion, between f and already treated closures is performed starting from the largest ones, *i.e.*, the elements of the border $\mathcal{B}$, until reaching the largest closures included in f. These latter closures correspond to the elements of the lower cover of f. $\diamondsuit$

FEI_f	:	The set of frequent essential itemsets associated to a disjunctive closed itemset f.
Set_Prem_Cl_f	:	The set containing frequent essential and disjunctive closed itemsets that will play the role of premise or conclusion *w.r.t.* a rule based on a disjunctive closed itemset f.
$\mathcal{EGAR}$	:	The set of valid exact generalized association rules.
$\mathcal{AGAR}$	:	The set of valid approximate generalized association rules.
$\mathcal{ApGAR}$	:	The set of valid approximated generalized association rules.

Table 7.6: Notations used by the GARS algorithm.

Let us now analyze the worst case complexity of the POSB algorithm. In the worst case, each essential itemset is frequent and equal to its disjunctive closed itemset. Hence, the set $\mathcal{EDCI}$ contains all disjunctive closed itemsets since there is no pruned disjunctive closure. The partially ordered structure constitutes thus a complete lattice. In this case, the complexity of the POSB algorithm is of the same order of magnitude than that of the algorithm proposed in [Valtchev *et al.*, 2000].

7.4.2 Deriving Generalized Association Rules

In this subsection, we describe the GARS algorithm [6] allowing the extraction of the selected generalized association rules. Its pseudo-code is given by Algorithm 13. The associated notations are listed in Table 7.6. To this algorithm, we brought some modifications allowing to avoid the derivation of useless rules (*cf.* previous section).

For each disjunctive closed itemset $f \in \mathcal{EDCI}$, the first step in the GARS algorithm consists in searching for the subsets that will play the role of premise and, then, conclusion of each rule based on f. These itemsets are its frequent essential itemsets, its immediate predecessors contained in $Cov_l(f)$, and their respective frequent essential itemsets (*cf.* line 3).

For each element X of Set_Prem_Cl_f (*cf.* lines 4-10), the algorithm determines *the difference*, denoted Y, between f and X (*i.e.*, $Y = f \setminus X$). Then, the Compute_Bounds procedure computes the upper and lower bounds of the support of Y (*cf.* line 6). After that, two cases have to be distinguished:

1. If the upper and lower bounds of the support of Y are equal (*cf.* lines 7-8), then $Supp(\vee Y)$ is exactly known. The Generate_Rules_Exact_Bounds procedure is hence called. Indeed, in this case, each rule using X (in premise or conclusion) and Y (conversely, in conclusion or premise) will be determined with its exact value of support and confidence (*cf.* Subsection 7.3.1 for the different forms of selected generalized rules and their associated formulae). The *minsupp* and *minconf* thresholds are then used to only retain valid rules. Then, for each valid rule, its value of confidence allows distinguishing its membership to the set $\mathcal{EGAR}$ of exact generalized association rules or to the set $\mathcal{AGAR}$ of approximate ones.

2. If the upper bound of the disjunctive support of Y is different from the lower one (*cf.* lines 9-10),

[6] GARS is the acronym of Generalized Association Rules Selector.

Algorithm 13: GARS

Input: - The partially ordered structure, *minsupp* and *minconf*.

Output: - The sets $\mathcal{EGAR}$, $\mathcal{AGAR}$ and $\mathcal{ApGAR}$.

```
1  Begin
2    ForEach (f ∈ EDCI) Do
3      Set_Prem_Cl_f := FEI_f ∪ Cov_l(f) ∪ {e | e ∈ FEI_f1 s.t. f1 ∈ Cov_l(f) };
4      ForEach (X ∈ Set_Prem_Cl_f) Do
5        Y := f \ X;
6        Compute_Bounds(up_Supp(∨Y), lp_Supp(∨Y));
7        If (up_Supp(∨Y) = lp_Supp(∨Y)) Then
8          Generate_Rules_Exact_Bounds(f, X, Y, Supp(∨Y), minsupp,
           minconf);
9        Else
10         Generate_Rules_Approximated_Bounds(f, X, Y, up_Supp(∨Y),
           lp_Supp(∨Y), minsupp, minconf);
11 End
```

then the Generate_Rules_Approximated_Bounds procedure is called. In this situation, the support and/or the confidence of rules using Y may not be exactly determined. Consequently, their associated lower and upper bounds are computed (*cf.* Subsection 7.3.2). If the support of a rule, under this case, is exactly determined then it is simply compared to *minsupp*. Otherwise, the lower bound of support must be higher than or equal to *minsupp*. The same reasoning applies for the confidence computation. Indeed, if the confidence value is exactly computed then it is simply compared to *minconf*. Otherwise, the lower bound of the confidence value must be greater than or equal to *minconf*. A rule which fulfills the validity conditions *w.r.t. minsupp* and *minconf* is qualified to be valid. In this situation, if either its support or its confidence is approximately determined, the associated valid rule will be inserted in the set $\mathcal{ApGAR}$. Otherwise, it is added according to its confidence value to $\mathcal{EGAR}$ or $\mathcal{AGAR}$.

The next theorem shows the soundness and the correctness of the GARS algorithm.

Theorem 18 *The* GARS *algorithm is sound and correct. It exactly determines all valid selected generalized association rules.*

Proof. The GARS algorithm iterates on the elements of $\mathcal{EDCI}$. For each closure $f \in \mathcal{EDCI}$, it determines its subsets – essential and disjunctive closed itemsets – that will be used in the associated generalized rules. The computation of the support and confidence of mined rules is performed according to the associated formulae and compared to the minimum thresholds to only retain valid rules. Thus, all valid selected rules will be mined. $\diamondsuit$

7.5 Experimental Results

In this section, we will describe the experimental results we obtained. Through the carried out experiments, we focused on the mining time as well as the number of extracted valid rules. The considered benchmark contexts are described in Appendix A. All experiments were carried out on a PC equipped with a 3GHz Pentium (R) and 1.75GB of main memory, running the GNU/Linux distribution Fedora Core 7 (with 2GB of swap memory).

The whole process for extracting the generalized association rules was implemented in C++ into a tool, called GARM. [7] To the best of our knowledge, our tool is the unique one allowing the extraction of generalized association rules through a dedicated exploration of the disjunctive search space. Moreover, no previous approach has considered essential and disjunctive closed itemsets as a basis for mining generalized association rules. In addition, there is no publicly available tool for mining disjunctive rules. The purpose of our experiments is twofold. On the one hand, we focus on a comparison of the mining time of the different components covering the process of generalized association rule mining. Recall that the GARM tool gathers the following components:

1. The first one extracts the $\mathcal{DSSR}$ representation thanks to two complementary steps. These latter are carried out thanks to a slight modification of our DCPR_Miner algorithm (*cf.* page 123). The first step allows the extraction of the sets $\mathcal{EDCI}$ and $\mathcal{FEI}$. The second makes it possible to gather, for a given disjunctive closed itemset, its associated frequent essential itemsets.

2. The second component constructs the partially ordered structure *w.r.t.* set inclusion between disjunctive closed itemsets using the POSB algorithm.

3. The third one derives the valid generalized association rules which are under the selected rule forms (*cf.* Table 7.3, page 157). In addition, it simultaneously eliminates duplicate ones. This is carried out thanks to the GARS algorithm.

On the other hand, we concentrate on the quantitative aspect through a comparison of the number of mined valid rules *w.r.t.* their associated type, *i.e.*, exact, approximate or approximated. The different experiments are carried out by varying either *minsupp* or *minconf* values.

Before going in detail in the interpretation of the obtained results, it is worth recalling that, once the support of the difference evaluated using the $\mathcal{DSSR}$ representation (*cf.* previous section), the derivation of the valid association rules having one of the considered forms becomes an easy task. Indeed, this consists in simple arithmetic operations using the respective supports of the premise, the conclusion and the rule. In addition, the extraction of approximated rules can be made optional in our tool. Indeed, we can easily restrict the extraction to valid rules whose respective supports and confidences are exactly determined. Note also that the restriction can be carried out *w.r.t.* association rule forms by allowing only some of them to be mined. Here, we preferred to omit such restrictions, which can however be useful according to the application to only retain rules of interest for the end-users.

We will begin by describing the effect of the *minsupp* variation on: (i) the runtime of the different components of GARM, (ii) the number of mined rules through a global point of view only distinguishing

[7] GARM is the acronym of Generalized Association Rule Miner.

rules *w.r.t.* their type and, then, thanks to a local point of view taking into account also the rule forms. Then, we study the effect of the *minconf* variation on both runtime and mined rule number. In the different tables sketching the obtained results, "Comp." (*resp.* "Approx" and "Apted") stands for "Component" (*resp.* "Approximate" and "Approximated").

7.5.1 Effect of the *minsupp* Variation

In these experiments, the value of *minsupp* varies and that of *minconf* is set to the associated relative minimum support threshold, *i.e.*, $\frac{minsupp}{|\mathcal{O}|}$.

Table 7.7 and Table 7.8 present representative results on the mining time (in seconds) of the three components of GARM for dense and sparse contexts, respectively. While Figure 7.2 and Figure 7.3 graphically sketch the obtained results for dense and sparse contexts, respectively. Our results show the efficiency of our tool towards extracting generalized association rules. In this respect, the time consumed by each component, *w.r.t.* the total time, closely depends on the context characteristics. Nevertheless, the second and third components are in general faster than the first one. Interestingly, once the partially ordered structure built thanks to the second component, the derivation of generalized association rules performed by the third one is in almost all cases the fastest step (*cf.* the last three columns in Table 7.7 and Table 7.8). This highlights the added value of such a structure not only for reducing the number of mined rules but also as a basis for efficient computations of the required supports. With respect to the variation of *minsupp* values, we note that as far as the value of *minsupp* decreases, the number of frequent essential itemsets and, hence, disjunctive closed itemsets increases. This augmentation leads to the increase of the mining time as well as the number of extracted generalized association rules.

For dense and sparse contexts respectively, Table 7.9 and Table 7.10 show our main results on the total number of valid generalized association rules distinguished *w.r.t.* their type (*i.e.*, exact, approximate and approximated). These results are also respectively shown in Figure 7.4 and Figure 7.5. Obtained results highlight that the number of mined generalized association rules closely depends on the context density. Indeed, the higher the value of this latter, the larger the associated equivalence classes are. This increases the number of essential itemsets per class. Consequently, the number of rules involving essential itemsets and disjunctive closed itemsets will greatly augment. This fact augments the number of rules even for high *minsupp* values for the dense contexts such as CONNECT and PUMSB. In this respect, it is always worth recalling that generalized association rules – disjunctive ones in particular – reach minimum support threshold much easier than conjunctive association rules. This fact highlights the added-value, *w.r.t.* the rule number reduction, of only considering frequent essential itemsets and their closure, and not any itemset.

For the KOSARAK, RETAIL, and T40I10D100K contexts, we only obtained approximate generalized association rules. Indeed, the number of exact rules is equal to **0** for the tested *minsupp* values. This is due to the fact that, for these contexts, each frequent essential itemset is equal to its disjunctive closure, which is not the case for contexts such as MUSHROOM and PUMSB. Moreover, the number of approximated rules is also equal to **0**. This is explained as follows. Let us recall that we search for the support of the difference between the disjunctive closed itemset, on which is based the rule, and the premise (or conclusion) containing either a disjunctive closed itemset or a frequent essential rules. In the case of RETAIL, KOSARAK, and T40I10D100K contexts, the support of the difference is always exactly

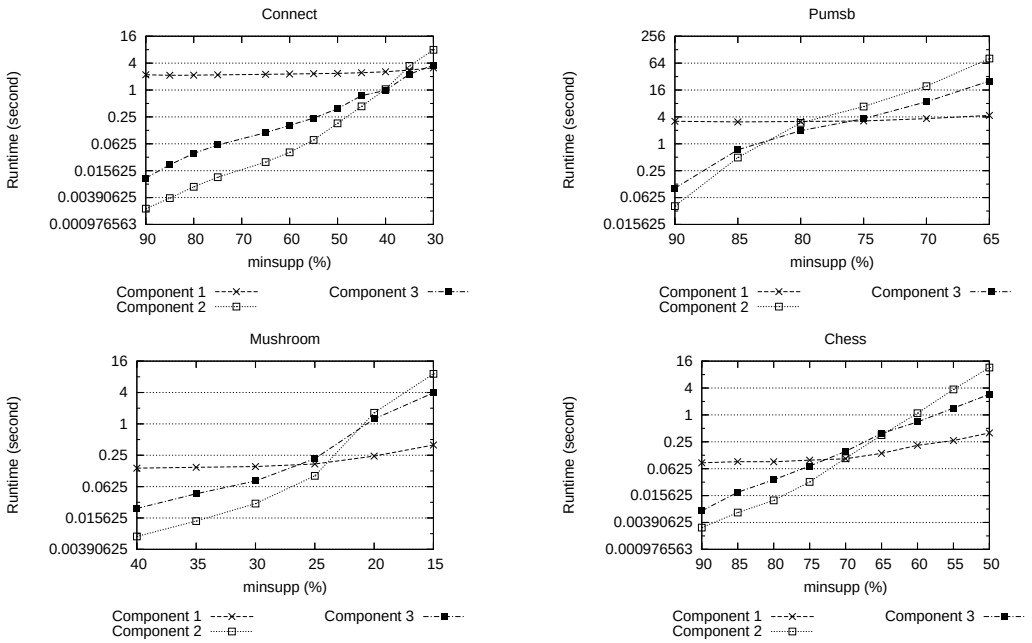


Figure 7.2: Mining time of generalized association rules from dense contexts.

determined, which leads to the absence of approximated association rules. Indeed, this difference is always encompassed between a frequent essential itemset and its disjunctive closure. Since each equivalence class is limited to a unique element, the difference is an essential itemset (equal to its closure) which explains why its support is always determined.

For the T10I4D100K, some exact and approximated generalized association rules are mined for low *minsupp* values. This can be explained by the fact that although almost all disjunctive equivalence classes contain a unique essential itemset, some of the classes have frequent essential itemsets that are different from their disjunctive closure. This makes possible to extract rules that are either exact or approximated. Note however that the number of approximated rules closely depends on the *minsupp* value. Indeed, the appearance of such rules is connected to the possibility (or not) to exactly derive the support of the difference. Such a derivation relies on the content of disjunctive equivalence classes, and more precisely, frequent essential itemsets and their associated closures. The appearance of these latter itemsets in the $\mathcal{DSSR}$ representation depends on the *minsupp* value. Note that the same scenario also occurred for the CONNECT dataset when the *minsupp* value was lowered from **75%** to **65%**.

For the smallest value of *minsupp* per dataset used in our tests, Table 7.11 also details for each benchmark dataset the number of generalized association rules per type (exact, approximate or approximated) and per rule form (*cf.* Table 7.3, page 157). In this table and for lack of space, "T" (*resp.* "E", "A", "Ap", and "Tot") refers to "Type" (*resp.* "Exact", "Approximate", "Approximated", "Total"). Note also that some dataset names are abbreviated. For the KOSARAK, RETAIL and T40I10D100K datasets, we omit the repartition of exact and approximated rules since their total number is equal to **0**.
Table 7.11 mainly highlights that **Form 1** – involving disjunction of items in premise and in conclusion – has the higher number of valid rules. This can be explained by the fact that the disjunction of items

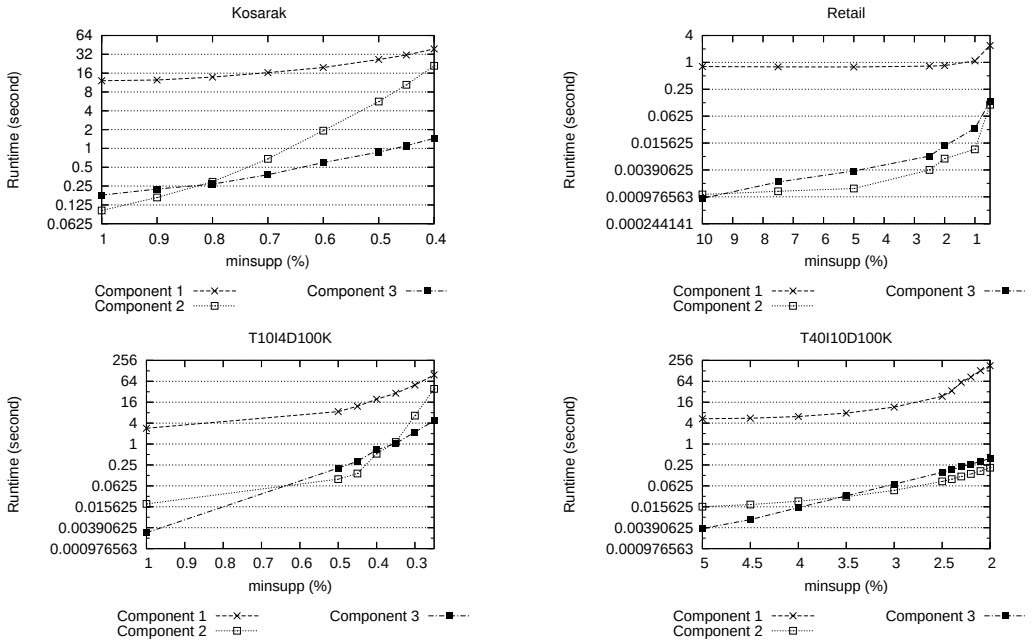


Figure 7.3: Mining time of generalized association rules from sparse contexts.

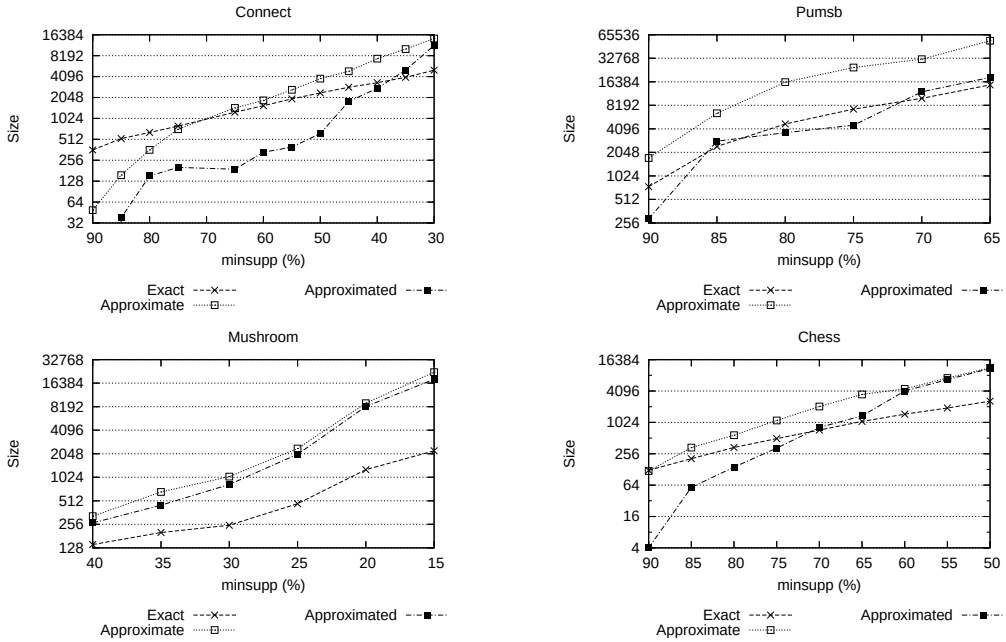


Figure 7.4: Number of mined generalized association rules from dense contexts.

Context	*minsupp* (%)	Comp. 1 Step 1	Comp. 1 Step 2	Comp. 2	Comp. 3	Total time	Comp. 1 / Total time (%)	Comp. 2 / Total time (%)	Comp. 3 / Total time (%)
CONNECT	*90*	2.1984	0.0020	0.0022	0.0107	**2.2133**	*99.42*	*0.10*	*0.48*
	85	2.1365	0.0028	0.0038	0.0212	**2.1643**	*98.84*	*0.18*	*0.98*
	80	2.1496	0.0034	0.0068	0.0380	**2.1978**	*97.96*	*0.31*	*1.73*
	75	2.1802	0.0044	0.0112	0.0588	**2.2546**	*96.90*	*0.50*	*2.60*
	65	2.2408	0.0074	0.0244	0.1108	**2.3834**	*94.33*	*1.02*	*4.65*
	60	2.2709	0.0098	0.0402	0.1618	**2.4827**	*91.86*	*1.62*	*6.52*
	55	2.3069	0.0127	0.0760	0.2352	**2.6308**	*88.17*	*2.89*	*8.94*
	50	2.3427	0.0160	0.1798	0.3827	**2.9212**	*80.74*	*6.16*	*13.10*
	45	2.4174	0.0213	0.4302	0.7472	**3.6161**	*67.44*	*11.90*	*20.66*
	40	2.5328	0.0243	1.0443	0.9813	**4.5827**	*55.80*	*22.79*	*21.41*
	35	2.7689	0.0299	3.4640	2.2542	**8.5170**	*32.86*	*40.67*	*26.47*
	30	3.0990	0.0400	7.9639	3.6207	**14.7236**	*21.32*	*54.09*	*24.59*
PUMSB	*90*	3.1823	0.0052	0.0403	0.1015	**3.3293**	*95.74*	*1.21*	*3.05*
	85	3.0641	0.0162	0.4919	0.7354	**4.3076**	*71.51*	*11.42*	*17.07*
	80	3.1239	0.0342	2.9364	1.9693	**8.0638**	*39.17*	*36.41*	*24.42*
	75	3.1990	0.0505	6.8037	3.6467	**13.6999**	*23.72*	*49.66*	*26.62*
	70	3.5852	0.0778	19.5460	8.7276	**31.9366**	*11.47*	*61.20*	*27.33*
	65	4.1989	0.1229	81.0964	25.3739	**110.7921**	*3.90*	*73.20*	*22.90*
MUSH-ROOM	*40*	0.1400	0.0013	0.0069	0.0238	**0.1720**	*82.15*	*4.01*	*13.84*
	35	0.1446	0.0018	0.0136	0.0455	**0.2055**	*71.24*	*6.62*	*22.14*
	30	0.1490	0.0029	0.0296	0.0803	**0.2618**	*58.02*	*11.31*	*30.67*
	25	0.1655	0.0043	0.1008	0.2153	**0.4859**	*34.95*	*20.75*	*44.30*
	20	0.2274	0.0129	1.6332	1.2396	**3.1131**	*7.72*	*52.46*	*39.82*
	15	0.3701	0.0232	9.1475	3.9346	**13.4754**	*2.92*	*67.88*	*29.20*
CHESS	*90*	0.0837	0.0009	0.0030	0.0071	**0.0947**	*89.33*	*3.17*	*7.50*
	85	0.0884	0.0014	0.0065	0.0185	**0.1148**	*78.22*	*5.66*	*16.12*
	80	0.0875	0.0021	0.0122	0.0352	**0.1370**	*65.40*	*8.91*	*25.69*
	75	0.0936	0.0033	0.0316	0.0718	**0.2003**	*48.38*	*15.78*	*35.84*
	70	0.0995	0.0050	0.1070	0.1521	**0.3636**	*28.74*	*29.43*	*41.83*
	65	0.1303	0.0079	0.3538	0.3858	**0.8778**	*15.74*	*40.31*	*43.95*
	60	0.1966	0.0122	1.0902	0.6955	**1.9945**	*10.47*	*54.66*	*34.87*
	55	0.2468	0.0200	3.7233	1.4192	**5.4093**	*4.93*	*68.83*	*26.24*
	50	0.3678	0.0269	11.4860	2.8852	**14.7659**	*2.67*	*77.79*	*19.54*

Table 7.7: Mining time (in second) of generalized association rules from dense contexts.

Context	*minsupp* (%)	Comp. 1 Step 1	Comp. 1 Step 2	Comp. 2	Comp. 3	Total time	Comp. 1 / Total time (%)	Comp. 2 / Total time (%)	Comp. 3 / Total time (%)
KOSARAK	*1.00*	12.0809	0.0032	0.1019	0.1792	**12.3652**	*97.73*	*0.82*	*1.45*
	0.90	12.4514	0.0037	0.1645	0.2239	**12.8435**	*96.98*	*1.28*	*1.74*
	0.80	13.8207	0.0045	0.2924	0.2685	**14.3861**	*96.10*	*2.03*	*1.87*
	0.70	16.2875	0.0061	0.6825	0.3794	**17.3555**	*93.88*	*3.93*	*2.19*
	0.60	19.6858	0.0088	1.9262	0.5942	**22.2150**	*88.65*	*8.67*	*2.68*
	0.50	26.4366	0.0125	5.6164	0.8738	**32.9393**	*80.30*	*17.05*	*2.65*
	0.45	31.2231	0.0157	10.4114	1.1056	**42.7558**	*73.06*	*24.35*	*2.59*
	0.40	39.0474	0.0197	21.0048	1.4477	**61.5196**	*63.51*	*34.14*	*2.35*
RETAIL	*10.00*	0.8074	0.0004	0.0011	0.0009	**0.8098**	*99.75*	*0.14*	*0.11*
	7.50	0.7914	0.0005	0.0013	0.0021	**0.7953**	*99.57*	*0.17*	*0.26*
	5.00	0.7909	0.0004	0.0015	0.0037	**0.7965**	*99.35*	*0.19*	*0.46*
	2.50	0.8252	0.0006	0.0039	0.0080	**0.8377**	*98.58*	*0.47*	*0.95*
	2.00	0.8463	0.0008	0.0070	0.0135	**0.8676**	*97.64*	*0.80*	*1.56*
	1.00	1.0789	0.0014	0.0113	0.0334	**1.1250**	*96.03*	*1.00*	*2.97*
	0.50	2.3869	0.0040	0.1127	0.1331	**2.6367**	*90.68*	*4.27*	*5.05*
T10I4-D100K	*1.00*	2.8381	0.0024	0.0189	0.0028	**2.8622**	*99.24*	*0.66*	*0.10*
	0.50	8.5903	0.0069	0.0975	0.2035	**8.8982**	*96.62*	*1.10*	*2.28*
	0.45	12.2011	0.0088	0.1424	0.3180	**12.6703**	*96.37*	*1.12*	*2.51*
	0.40	19.4357	0.0145	0.5272	0.6773	**20.6547**	*94.17*	*2.55*	*3.28*
	0.35	29.0230	0.0192	1.1450	1.0416	**31.2288**	*93.00*	*3.66*	*3.34*
	0.30	49.9765	0.0457	6.5793	2.2136	**58.8151**	*85.05*	*11.19*	*3.76*
	0.25	97.5089	0.0597	38.8410	4.6734	**141.0830**	*69.16*	*27.53*	*3.31*
T40I10-D100K	*5.00*	5.3281	0.0021	0.0157	0.0037	**5.3496**	*99.64*	*0.29*	*0.07*
	4.50	5.5353	0.0023	0.0181	0.0067	**5.5624**	*99.55*	*0.33*	*0.12*
	4.00	6.1850	0.0028	0.0226	0.0148	**6.2252**	*99.40*	*0.36*	*0.24*
	3.50	7.7510	0.0036	0.0304	0.0322	**7.8172**	*99.20*	*0.39*	*0.41*
	3.00	11.5457	0.0050	0.0465	0.0698	**11.6670**	*99.00*	*0.40*	*0.60*
	2.50	23.6067	0.0087	0.0842	0.1538	**23.8534**	*99.01*	*0.35*	*0.64*
	2.40	34.0378	0.0087	0.0976	0.1852	**34.3293**	*99.18*	*0.28*	*0.54*
	2.30	59.5699	0.0098	0.1161	0.2232	**59.9190**	*99.43*	*0.20*	*0.37*
	2.20	84.9241	0.0112	0.1381	0.2637	**85.3371**	*99.53*	*0.16*	*0.31*
	2.10	129.7740	0.0129	0.1675	0.3167	**130.2711**	*99.63*	*0.13*	*0.24*
	2.00	180.3630	0.0149	0.2081	0.3864	**180.9724**	*99.67*	*0.12*	*0.21*

Table 7.8: Mining time (in second) of generalized association rules from sparse contexts.

Context	*minsupp* (%)	# Exact	# Appro	# Apted	Total number	$\frac{\#\text{ Exact}}{\text{Total number}}$ (%)	$\frac{\#\text{ Appro}}{\text{Total number}}$ (%)	$\frac{\#\text{ Apted}}{\text{Total number}}$ (%)
CONNECT	*90*	359	49	0	**408**	*87.99*	*12.01*	*0.00*
	85	524	156	38	**718**	*72.98*	*21.73*	*5.29*
	80	641	361	152	**1, 154**	*55.55*	*31.28*	*13.17*
	75	795	717	202	**1, 714**	*46.38*	*41.83*	*11.79*
	65	1, 259	1, 449	190	**2, 898**	*43.44*	*50.00*	*6.56*
	60	1, 561	1, 859	334	**3, 754**	*41.58*	*49.52*	*8.90*
	55	1, 954	2, 640	396	**4, 990**	*39.16*	*52.91*	*7.93*
	50	2, 388	3, 824	620	**6, 832**	*34.95*	*55.97*	*9.08*
	45	2, 879	4, 891	1, 794	**9, 564**	*30.10*	*51.14*	*18.76*
	40	3, 351	7, 413	2, 778	**13, 542**	*24.75*	*54.74*	*20.51*
	35	3, 994	10, 208	5, 086	**19, 288**	*20.71*	*52.92*	*26.37*
	30	5, 060	14, 498	11, 488	**31, 046**	*16.30*	*46.70*	*37.00*
PUMSB	*90*	746	1, 734	290	**2, 770**	*26.93*	*62.60*	*10.47*
	85	2, 437	6, 475	2, 854	**11, 766**	*20.71*	*55.03*	*24.26*
	80	4, 735	16, 211	3, 660	**24, 606**	*19.24*	*65.88*	*14.88*
	75	7, 317	24, 997	4, 554	**36, 868**	*19.85*	*67.80*	*12.35*
	70	10, 075	31, 901	12, 194	**54, 170**	*18.60*	*58.89*	*22.51*
	65	14, 965	54, 879	18, 616	**88, 460**	*16.92*	*62.04*	*21.04*
MUSH-ROOM	*40*	141	326	267	**734**	*19.21*	*44.41*	*36.38*
	35	201	665	450	**1, 316**	*15.27*	*50.53*	*34.20*
	30	249	1, 044	829	**2, 122**	*11.73*	*49.20*	*39.07*
	25	470	2, 390	2, 018	**4, 878**	*9.64*	*49.00*	*41.36*
	20	1, 284	9, 078	8, 242	**18, 604**	*6.90*	*48.80*	*44.30*
	15	2, 239	22, 691	18, 370	**43, 300**	*5.17*	*52.40*	*42.43*
CHESS	*90*	123	119	4	**246**	*50.00*	*48.37*	*1.63*
	85	206	336	58	**600**	*34.33*	*56.00*	*9.67*
	80	338	582	142	**1, 062**	*31.82*	*54.80*	*13.38*
	75	500	1, 116	328	**1, 944**	*25.72*	*57.41*	*16.87*
	70	729	2, 071	820	**3, 620**	*20.14*	*57.21*	*22.65*
	65	1, 066	3, 522	1, 364	**5, 952**	*17.91*	*59.17*	*22.92*
	60	1, 470	4, 506	4, 102	**10, 078**	*14.59*	*44.71*	*40.70*
	55	1, 934	7, 296	6, 780	**16, 010**	*12.08*	*45.57*	*42.35*
	50	2, 616	11, 472	11, 102	**25, 190**	*10.39*	*45.54*	*44.07*

Table 7.9: Number of mined generalized association rules from dense contexts.

Context	*minsupp* (%)	# Exact	# Appro	# Apted	Total number	$\frac{\#\text{ Exact}}{\text{Total number}}$ (%)	$\frac{\#\text{ Appro}}{\text{Total number}}$ (%)	$\frac{\#\text{ Apted}}{\text{Total number}}$ (%)
KOSARAK	*1.00*	0	5, 750	0	**5, 750**	*0*	*100*	*0*
	0.90	0	7, 286	0	**7, 286**	*0*	*100*	*0*
	0.80	0	9, 248	0	**9, 248**	*0*	*100*	*0*
	0.70	0	13, 046	0	**13, 046**	*0*	*100*	*0*
	0.60	0	19, 614	0	**19, 614**	*0*	*100*	*0*
	0.50	0	29, 648	0	**29, 648**	*0*	*100*	*0*
	0.45	0	37, 696	0	**37, 696**	*0*	*100*	*0*
	0.40	0	48, 760	0	**48, 760**	*0*	*100*	*0*
RETAIL	*10.00*	0	24	0	**24**	*0*	*100*	*0*
	7.50	0	66	0	**66**	*0*	*100*	*0*
	5.00	0	120	0	**120**	*0*	*100*	*0*
	2.50	0	270	0	**270**	*0*	*100*	*0*
	2.00	0	464	0	**464**	*0*	*100*	*0*
	1.00	0	1, 160	0	**1, 160**	*0*	*100*	*0*
	0.50	0	4, 622	0	**4, 622**	*0*	*100*	*0*
T10I4-D100K	*1.00*	0	88	0	**88**	*0.00*	*100.00*	*0.00*
	0.50	0	6, 842	0	**6, 842**	*0.00*	*100.00*	*0.00*
	0.45	0	10, 060	0	**10, 060**	*0.00*	*100.00*	*0.00*
	0.40	6	22, 299	7	**22, 312**	*0.03*	*99.94*	*0.03*
	0.35	30	34, 296	0	**34, 326**	*0.09*	*99.91*	*0.00*
	0.30	198	70, 964	8	**71, 170**	*0.28*	*99.71*	*0.01*
	0.25	426	149, 199	5	**149, 630**	*0.28*	*99.71*	*0.01*
T40I10-D100K	*5.00*	0	120	0	**120**	*0*	*100*	*0*
	4.50	0	224	0	**224**	*0*	*100*	*0*
	4.00	0	512	0	**512**	*0*	*100*	*0*
	3.50	0	1, 128	0	**1, 128**	*0*	*100*	*0*
	3.00	0	2, 454	0	**2, 454**	*0*	*100*	*0*
	2.50	0	5, 404	0	**5, 404**	*0*	*100*	*0*
	2.40	0	6, 438	0	**6, 438**	*0*	*100*	*0*
	2.30	0	7, 778	0	**7, 778**	*0*	*100*	*0*
	2.20	0	9, 282	0	**9, 282**	*0*	*100*	*0*
	2.10	0	11, 148	0	**11, 148**	*0*	*100*	*0*
	2.00	0	13, 546	0	**13, 546**	*0*	*100*	*0*

Table 7.10: Number of mined generalized association rules from sparse contexts.

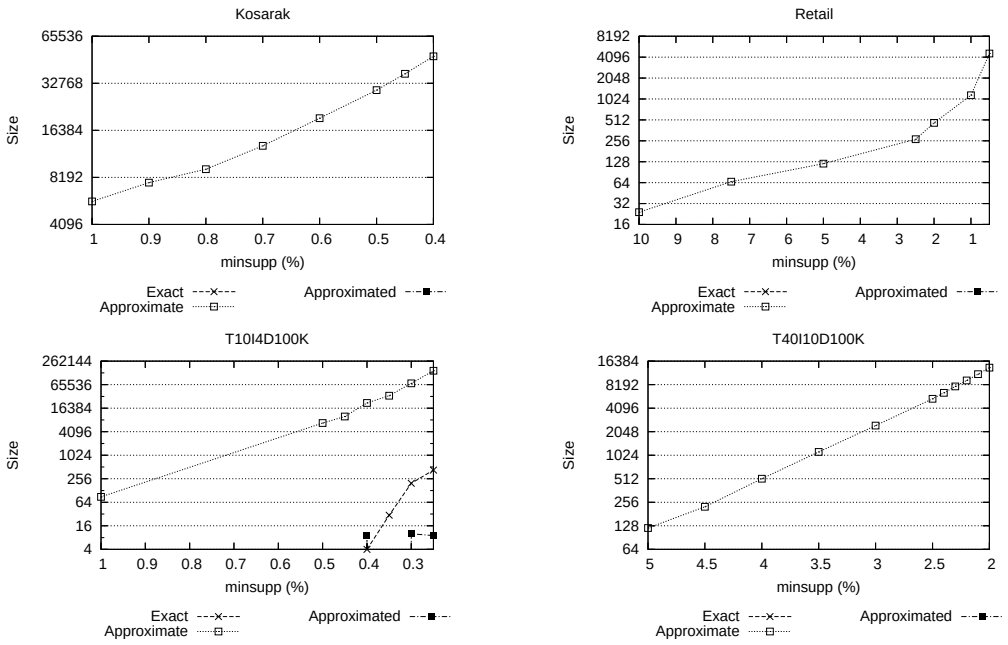


Figure 7.5: Number of mined generalized association rules from sparse contexts.

reaches minimum thresholds much easier than the other forms. In addition, we note that, in many cases, there are as many valid rules as valid reverse rules. For example, the number of rules following R_{35} is always the same as those of R_{46}.

7.5.2 Effect of the *minconf* Variation

Now, we concentrate on the variation of the mining time and the number of extracted rules when the value of *minsupp* is fixed and that of *minconf* varies. With respect to the mining time, such a variation only affects that of the third component since *minconf* is only used in this component. The number of approximate and approximated rules can also change, which is not the case of the number of exact generalized association rules. Indeed, exact rules have always a confidence value equal to **1**. These variations are depicted by Table 7.12 and Table 7.13 for dense and sparse contexts, respectively. They are also respectively sketched by Figure 7.6 and Figure 7.7. The selected *minsupp* value is given under the name of the context in the associated table.

According to the obtained results, we notice that, in general, the mining time of generalized association rules increases proportionally to the decrease of *minconf* values. Nevertheless, the augmentation is not very sensitive to the variation of *minconf*. Moreover the number of approximate and approximated rules decreases when the value of *minconf* increases. This can be explained by the fact that we only retain valid rules, *i.e.*, those the minimum confidence value of which is higher than or equal to *minconf*. Once this latter is set to a higher value, the validity constraint becomes harder to be verified by a rule, even if its support is greater than or equal to *minsupp*.

The experiments we carried out confirm that, once the partially ordered structure built, the derivation of generalized association rules becomes straightforward. Indeed, this last component does not influence

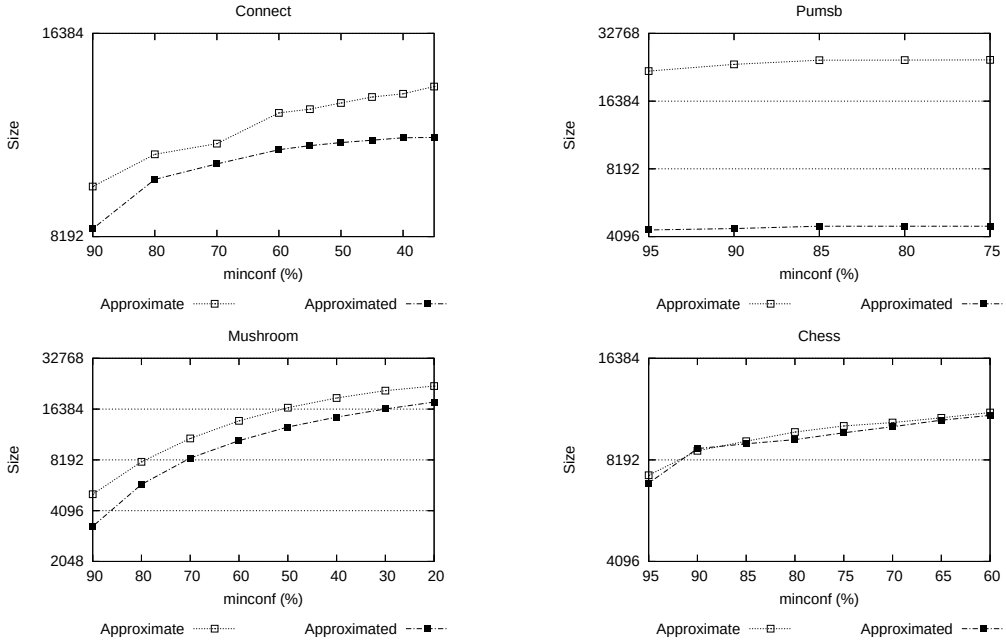


Figure 7.6: Variation of the number of mined generalized association rules *w.r.t. minconf* values for dense contexts.

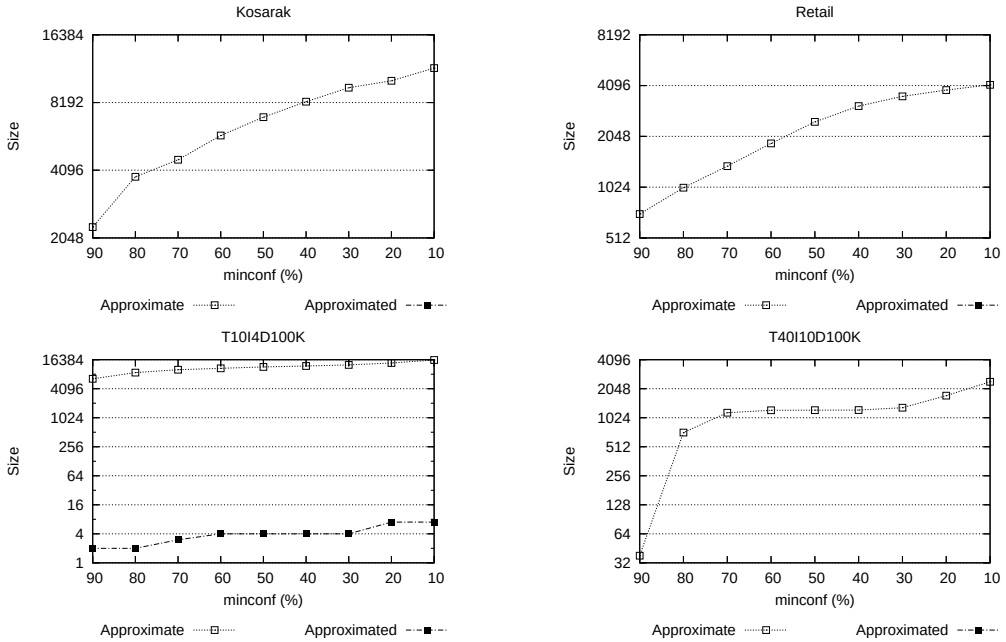


Figure 7.7: Variation of the number of mined generalized association rules *w.r.t. minconf* values for sparse contexts.

Dataset	T	Form 1							Form 2						
		R11	R12	R13	R14	R15	R16	Tot	R21	R22	R23	R24	R25	R26	Tot
Connect	E	1,420	2,220	524	0	890	0	**5,054**	3	0	0	0	0	3	**6**
(30%)	A	803	0	1,732	2,256	2,832	3,722	**11,345**	0	0	0	0	3	0	**3**
	Ap	807	807	2,974	2,974	1,962	1,962	**11,486**	0	0	0	0	0	0	**0**
Pumsb	E	3,197	7,802	3,541	0	425	0	**14,965**	0	0	0	0	0	0	**0**
(65%)	A	4,605	0	12,378	15,919	10,776	11,201	**54,879**	0	0	0	0	0	0	**0**
	Ap	2,089	2,089	5,910	5,910	1,309	1,309	**18,616**	0	0	0	0	0	0	**0**
Mush.	E	229	612	402	0	268	0	**1,511**	638	2	0	0	0	88	**728**
(15%)	A	628	0	1,121	1,523	2,716	2,984	**8,972**	0	0	1,556	0	1,875	1,695	**5,126**
	Ap	1,150	1,150	3,661	3,661	1,973	1,973	**13,568**	0	391	0	1,077	0	571	**2,039**
Chess	E	703	1,295	417	0	201	0	**2,616**	0	0	0	0	0	0	**0**
(50%)	A	592	0	958	1,375	4,173	4,374	**11,472**	0	0	0	0	0	0	**0**
	Ap	1,087	1,087	2,930	2,930	1,534	1,534	**11,102**	0	0	0	0	0	0	**0**
Ko. (0.4%)	A	0	0	0	0	6,552	6,552	**13,104**	0	0	0	0	6,552	6,552	**13,104**
Re. (0.5%)	A	0	0	0	0	624	624	**1,248**	0	0	0	0	624	624	**1,248**
T10	E	0	213	0	0	0	0	**213**	213	0	0	0	0	0	**213**
(0.25%)	A	426	0	420	420	18,589	18,589	**38,444**	0	0	420	0	18,590	19,009	**38,019**
	Ap	0	0	0	0	1	1	**2**	0	0	0	0	0	1	**1**
T40 (2%)	A	0	0	0	0	1,695	1,695	**3,390**	0	0	0	0	1,695	1,695	**3,390**

Dataset	T	Form 3							Form 4						
		R31	R32	R33	R34	R35	R36	Tot	R41	R42	R43	R44	R45	R46	Tot
Connect	E	0	0	0	0	0	0	**0**	0	0	0	0	0	0	**0**
(30%)	A	42	0	0	0	1,127	405	**1,574**	0	42	2	0	405	1,127	**1,576**
	Ap	0	0	0	2	0	0	**2**	0	0	0	0	0	0	**0**
Pumsb	E	0	0	0	0	0	0	**0**	0	0	0	0	0	0	**0**
(65%)	A	0	0	0	0	0	0	**0**	0	0	0	0	0	0	**0**
	Ap	0	0	0	0	0	0	**0**	0	0	0	0	0	0	**0**
Mush.	E	0	0	0	0	0	0	**0**	0	0	0	0	0	0	**0**
(15%)	A	273	0	100	266	2,145	928	**3,712**	0	273	889	100	1,474	2,145	**4,881**
	Ap	223	0	220	623	354	546	**1,966**	0	223	0	220	0	354	**797**
Chess	E	0	0	0	0	0	0	**0**	0	0	0	0	0	0	**0**
(50%)	A	0	0	0	0	0	0	**0**	0	0	0	0	0	0	**0**
	Ap	0	0	0	0	0	0	**0**	0	0	0	0	0	0	**0**
Ko. (0.4%)	A	0	0	0	0	6,373	4,903	**11,276**	0	0	0	0	4,903	6,373	**11,276**
Re. (0.5%)	A	0	0	0	0	554	509	**1,063**	0	0	0	0	509	554	**1,063**
T10	E	0	0	0	0	0	0	**0**	0	0	0	0	0	0	**0**
(0.25%)	A	213	0	0	0	18,822	17,333	**36,368**	0	213	0	0	17,333	18,822	**36,368**
	Ap	0	0	0	0	1	0	**1**	0	0	0	0	0	1	**1**
T40 (2%)	A	0	0	0	0	1,689	1,694	**3,383**	0	0	0	0	1,694	1,689	**3,383**

Table 7.11: Detailed number of mined generalized association rules per selected form.

the performances of the GARM tool.

7.6 Related Work and Discussion

Contributions related to association rule mining mainly concentrated on the classic rule form, namely that presenting conjunction of items in both premise and conclusion parts. In this respect, many concise representations for such rules were proposed in the literature [Ceglar and Roddick, 2006, Kryszkiewicz, 2002].

Some works focused on taking into account negative items within the mined association rules. Since the majority of items are not present in each object, a huge quantity of association rules with negation

Context	*minconf* (%)	Runtime Comp. 3	# Appro	# Apted	Total rule number	$\frac{\#\text{ Appro}}{\text{Total number}}$ (%)	$\frac{\#\text{ Apted}}{\text{Total number}}$ (%)
CONNECT	*35*	**3.6136**	13, 666	11, 487	**25, 153**	*54.33*	*45.67*
(30%)	*40*	**3.5639**	13, 331	11, 475	**24, 806**	*53.74*	*46.26*
	45	**3.5537**	13, 190	11, 380	**24, 570**	*53.68*	*46.32*
	50	**3.5499**	12, 924	11, 290	**24, 214**	*53.37*	*46.63*
	55	**3.6312**	12, 653	11, 172	**23, 825**	*53.10*	*46.90*
	60	**3.6007**	12, 494	11, 018	**23, 512**	*53.14*	*46.86*
	70	**3.5580**	11, 250	10, 498	**21, 748**	*51.73*	*48.27*
	80	**3.4814**	10, 847	9, 958	**20, 805**	*52.14*	*47.86*
	90	**3.4294**	9, 717	8, 412	**18, 129**	*53.60*	*46.40*
PUMSB	*75*	**3.6832**	24, 997	4, 554	**29, 551**	*84.59*	*15.41*
(75%)	*80*	**3.6261**	24, 943	4, 554	**29, 497**	*84.56*	*15.44*
	85	**3.6818**	24, 924	4, 553	**29, 477**	*84.55*	*15.45*
	90	**3.6390**	23, 880	4, 444	**28, 324**	*84.31*	*15.69*
	95	**3.7789**	22, 280	4, 384	**26, 664**	*83.56*	*16.44*
MUSHROOM	*20*	**3.7657**	22, 454	18, 031	**40, 485**	*55.46*	*44.54*
(15%)	*30*	**3.6490**	21, 065	16, 419	**37, 484**	*56.20*	*43.80*
	40	**3.5339**	19, 075	14, 690	**33, 765**	*56.49*	*43.51*
	50	**3.4037**	16, 700	12, 829	**29, 529**	*56.55*	*43.45*
	60	**3.2730**	13, 955	10, 648	**24, 603**	*56.72*	*43.28*
	70	**3.1144**	10, 979	8, 399	**19, 378**	*56.66*	*43.34*
	80	**3.0111**	7, 981	5, 858	**13, 839**	*57.67*	*42.33*
	90	**2.7817**	5, 125	3, 309	**8, 434**	*60.77*	*39.23*
CHESS	*60*	**2.9597**	11, 321	11, 095	**22, 416**	*50.50*	*49.50*
(50%)	*65*	**2.9615**	10, 910	10, 739	**21, 649**	*50.39*	*49.61*
	70	**2.8043**	10, 568	10, 275	**20, 843**	*50.70*	*49.30*
	75	**2.7948**	10, 327	9, 857	**20, 184**	*51.16*	*48.84*
	80	**2.7880**	9, 903	9, 408	**19, 311**	*51.28*	*48.72*
	85	**2.8126**	9, 305	9, 143	**18, 448**	*50.44*	*49.56*
	90	**2.7629**	8, 717	8, 848	**17, 565**	*49.63*	*50.37*
	95	**2.6172**	7, 377	6, 991	**14, 368**	*51.34*	*48.66*

Table 7.12: Variation of the mining time and the number of mined generalized association rules *w.r.t. minconf* values for dense contexts.

Context	*minconf* (%)	Runtime Comp. 3	# Appro	# Apted	Total rule number	$\frac{\#\text{ Appro}}{\text{Total number}}$ (%)	$\frac{\#\text{ Apted}}{\text{Total number}}$ (%)
KOSARAK	*10*	**0.3394**	11, 658	0	**11, 658**	*100*	*0*
(**0.70%**)	*20*	**0.2980**	10, 235	0	**10, 235**	*100*	*0*
	30	**0.2794**	9, 536	0	**9, 536**	*100*	*0*
	40	**0.2419**	8, 260	0	**8, 260**	*100*	*0*
	50	**0.2078**	7, 052	0	**7, 052**	*100*	*0*
	60	**0.1727**	5, 842	0	**5, 842**	*100*	*0*
	70	**0.1357**	4, 560	0	**4, 560**	*100*	*0*
	80	**0.1148**	3, 828	0	**3, 828**	*100*	*0*
	90	**0.0701**	2, 289	0	**2, 289**	*100*	*0*
RETAIL	*10*	**0.1194**	4, 139	0	**4, 139**	*100*	*0*
(**0.50%**)	*20*	**0.1116**	3, 853	0	**3, 853**	*100*	*0*
	30	**0.1029**	3, 537	0	**3, 537**	*100*	*0*
	40	**0.0900**	3, 099	0	**3, 099**	*100*	*0*
	50	**0.0728**	2, 496	0	**2, 496**	*100*	*0*
	60	**0.0549**	1, 861	0	**1, 861**	*100*	*0*
	70	**0.0408**	1, 364	0	**1, 364**	*100*	*0*
	80	**0.0308**	1, 017	0	**1, 017**	*100*	*0*
	90	**0.0219**	709	0	**709**	*100*	*0*
T10I4D100K	*10*	**0.4975**	16, 163	7	**16, 170**	*99.96*	*0.04*
(**0.40%**)	*20*	**0.4336**	13, 975	7	**13, 982**	*99.95*	*0.05*
	30	**0.3982**	12, 753	4	**12, 757**	*99.97*	*0.03*
	40	**0.3805**	12, 131	4	**12, 135**	*99.97*	*0.03*
	50	**0.3632**	11, 557	4	**11, 561**	*99.97*	*0.03*
	60	**0.3417**	10, 857	4	**10, 861**	*99.96*	*0.04*
	70	**0.3218**	10, 128	3	**10, 131**	*99.97*	*0.03*
	80	**0.2868**	8, 824	2	**8, 826**	*99.98*	*0.02*
	90	**0.2229**	6, 577	2	**6, 579**	*99.97*	*0.03*
T40I10D100K	*10*	**0.0682**	2, 417	0	**2, 417**	*100*	*0*
(**3.00%**)	*20*	**0.0496**	1, 735	0	**1, 735**	*100*	*0*
	30	**0.0377**	1, 302	0	**1, 302**	*100*	*0*
	40	**0.0350**	1, 231	0	**1, 231**	*100*	*0*
	50	**0.0351**	1, 228	0	**1, 228**	*100*	*0*
	60	**0.0348**	1, 225	0	**1, 225**	*100*	*0*
	70	**0.0329**	1, 154	0	**1, 154**	*100*	*0*
	80	**0.0213**	719	0	**719**	*100*	*0*
	90	**0.0017**	38	0	**38**	*100*	*0*

Table 7.13: Variation of the mining time and the number of mined generalized association rules *w.r.t. minconf* values for sparse contexts.

is often extracted. Thus, existing approaches have tried to address this problem through the use of additional background information about the data [Savasere *et al.*, 1998], incorporating item correlations [Antonie and Zaïane, 2004], and additional rule interestingness measures [Morzy, 2006, Wu *et al.*, 2004], etc.

In [Kim, 2003, Nanavati *et al.*, 2001], the authors were interested in using the disjunction connector within the association rule mining task. In addition to the inclusive disjunction connector, *i.e.*, the operator $\vee$, Nanavati *et al.* were also interested in the exclusive disjunction connector, denoted $\oplus$ [Nanavati *et al.*, 2001]. In this respect, two items A and B are said to be mutually exclusive, *i.e.*, $\mathtt{A} \oplus \mathtt{B}$, whenever the negative association rule $\mathtt{A} \Rightarrow \overline{\mathtt{B}}$ (or equivalently, $\mathtt{B} \Rightarrow \overline{\mathtt{A}}$) is an exact rule. The authors hence proposed two kinds of rules: the simple disjunctive rules and the generalized disjunctive ones. Simple disjunctive rules are those having either the premise or the conclusion (*i.e.*, not simultaneously both) composed by a disjunction of items. This disjunction can be inclusive (the simultaneous occurrence of items is possible) or exclusive (two distinct items cannot occur together). On the other hand, generalized disjunctive rules are disjunctive rules whose premises or conclusions contain a conjunction of disjunctions. These disjunctions can either be inclusive or exclusive. In [Kim, 2003], the author mainly focuses on getting out association rules having conclusions containing mutually exclusive items, *i.e.*, the presence of one of them leads to the absence of the others. This is expressed in [Nanavati *et al.*, 2001] using the operator $\oplus$. Other forms of generalized association rules were described in [Grün, 1998]. They are as follows: for all x_i, $y_j \in \mathcal{I}$,

1. Rules having their premise or conclusion part composed of negated items, *i.e.*, those of the form $x_1 \wedge x_2 \wedge \ldots \wedge x_n \Rightarrow \overline{y_1} \wedge \overline{y_2} \wedge \ldots \wedge \overline{y_m}$ or $\overline{x_1} \wedge \overline{x_2} \wedge \ldots \wedge \overline{x_n} \Rightarrow y_1 \wedge y_2 \wedge \ldots \wedge y_m$.
2. Rules having disjunctive premises or conclusions, *i.e.*, those of the form $x_1 \wedge x_2 \wedge \ldots \wedge x_n \Rightarrow y_1 \vee y_2 \vee \ldots \vee y_m$ or $x_1 \vee x_2 \vee \ldots \vee x_n \Rightarrow y_1 \wedge y_2 \wedge \ldots \wedge y_m$.

All these rule forms were included and enriched thanks to those taken into account in our work. In [Shima *et al.*, 2005], Shima *et al.* extract what they called *disjunctive closed rules*. In their work, a disjunctive closed rule simply stands for a clause under the disjunctive normal form (DNF) such that its disjuncts are constituted by frequent closed itemsets [Shima *et al.*, 2004]. Elble *et al.* used disjunctive rules to handle numerical attributes by considering disjunctions between intervals [Elble *et al.*, 2003]. In classification association rule mining, a disjunctive rule having a premise (*resp.* conclusion) composed by a conjunction (*resp.* disjunction) of items is called *multiple target rule* [Li and Jones, 2006]. Finally, it is worth recalling that such a rule form has also been used as an intermediate step for defining concise representations for frequent itemsets (*e.g.*, those based on disjunction-free sets [Bykowski and Rigotti, 2001, Bykowski and Rigotti, 2003] and (generalized) disjunction-free generators [Kryszkiewicz, 2002]).

7.7 Conclusion

In this chapter, we introduced a novel approach for extracting generalized association rules. We started by extending the framework of classic association rules through taking into account various connectors as well as negative items. An overview of the possible mined forms of generalized association rules was also presented, in addition to how are calculated the associated supports in the general case. To avoid that our approach be restrictive to some association rule forms regardless the others, we adopted

as a starting point an exact concise representation of frequent itemsets. On the one hand, having at hand such a representation allows the exact derivation of the support of each literalset whose positive variation is a frequent itemset. On the other hand, the fact that this representation is based on disjunctive itemsets, namely essential and disjunctive closed itemsets, makes easier the extraction of rules containing disjunction of items as well as negated ones.

As a next step, towards reducing the number of mined rules, a selection process of subsets of generalized association rules was then described. As a result, we mainly concentrated on four generalized association rule forms. We also distinguished both intra-node and inter-nodes rules. These latter rules required the construction of a partially ordered structure obtained *w.r.t.* set inclusion between disjunctive closed itemsets. An approximation process of the quality measures was also discussed. This approximation is useful once a given support required for their computation may not be exactly derived from the representation. We also led a study allowing the elimination of duplicated rules.

For mining generalized association rules, we designed new complementary algorithms covering the different steps of our approach. This results in a new tool, called GARM. The experimental tests consisted essentially in analyzing the behavior of our tool regarding the mining time of its components and the number of mined association rules per type and per rule form. Experimental results proved the effectiveness of the proposed approach, and that the number of exact, approximate and approximated rules closely depends on dataset characteristics.

Part IV

Conclusion

Chapter 8

Conclusion and Future Work

8.1 Conclusion

The increasing opportunity of quickly collecting and cheaply storing large volumes of data highlighted the need for extracting concise information to be efficiently manipulated and intuitively analyzed. In this situation, the use of data mining tools become of paramount importance in order to transform the stored data into possible useful knowledge through the extraction of patterns (*e.g.* itemsets, association rules, clusters, etc.). To help the end-users efficiently and effectively interpreting and analyzing the extracted patterns, the size of the pattern sets should be as concise as possible while preserving as much as possible their hidden interesting information.

In this thesis, we were mainly interested in two complementary pattern classes, namely frequent itemsets and association rules. These pattern classes are the most mined ones in data mining. When confronted to real-life applications, the number of frequent itemsets and association rules proves to be very large hampering their effective interpretations by the end-users. In this situation, a high number of approaches was devoted to the proposal of concise representations of frequent patterns aiming at only retaining subsets of the whole set, while being able to derive non-retained (or redundant) patterns without information loss. Such subsets are called *exact concise representations*.

A central concept in the design of almost all concise representations of frequent itemsets as well as association rules is the concept of *minimal generator*. In this respect, we carried out a critical survey of exact concise representations of frequent itemsets proposed in the literature. Its main result is that the different representations are either based on a generalization/extension of minimal generators, or can be obtained using these latter patterns as an efficient computation mean. Moreover, the most known concise representations of association rules (*aka generic bases*) convey rules with minimal premise parts, composed by minimal generators. Nevertheless, several minimal generators characterize the same set of objects and, consequently, convey redundant knowledge.

In this regard, we focused on the intrinsic properties of the minimal generator family that can be drawn from a context. Our review of the literature showed that a unique attempt was proposed for removing redundancy within minimal generators, through the succinct system of minimal generators [Dong *et al.*, 2005]. This system splits the whole set of minimal generators into two parts: the first contains useful (or

succinct) minimal generators, and the second part gathers those that can be derived starting from the succinct ones. Our study of the structural properties of this system revealed that it preserves the order ideal property. This important result was not established in the original contribution [Dong *et al.*, 2005]. In addition, after a thorough study of the aforementioned system *w.r.t.* the exactness of the regeneration process, we proved that it misses some redundant minimal generators in some cases, which makes it not an exact representation of the minimal generator set. In this situation, we introduced a new lossless system. The key idea of our proposal is that minimal generators belonging to the same equivalence class can be split into finer classes using a substitution process applied on their respective subsets. This process is based on the Armstrong axiom of pseudo-transitivity [Armstrong, 1974]. It makes possible retaining a representative minimal generator per substitution-based class. The obtained system constitutes hence a *perfect cover* of the minimal generator set. Since it is not an order ideal contrary to the original system, we proposed a hybrid family – the *directed substitution-free sets* – whose purpose is to simultaneously maintain the main feature of each system, *i.e.*, the order ideal structure while being lossless. This was done by simply adding some particular elements to the first system. The proposed systems were then extended to association rules by removing redundancy from generic bases. Carried out experiments confirm that our study makes it possible getting, in average, almost as many closed itemsets as *irreducible* minimal generators. Thus, it allows to eliminate, as much as possible, redundant generic association rules.

We also explored the disjunctive search space. This was motivated by the fact that, in some applications, the mined knowledge on the complementary occurrences of items – conveyed through disjunctive support – brings richer knowledge to the end-users. In this respect, we took as a starting point the unique concise representation of frequent itemsets whose some of the elements belong to the disjunctive search space. This representation is that based on frequent essential itemsets [Casali *et al.*, 2005a]. Since several essential itemsets can satisfy the same set of objects, they hence characterize the same class. In this situation, we introduced a new closure operator dedicated to the disjunctive search space. We also thoroughly studied the induced structural properties. Once applied, each disjunctive equivalence class is characterized by a unique disjunctive closed itemsets. This operator offered a new characterization of essential itemsets. Indeed, they constitute the minimal elements within the equivalence classes the proposed operator induces. Based on disjunctive closed itemsets, we proposed a new exact concise representation of frequent itemsets. This representation is the first one uniquely relying on disjunctive itemsets and, hence, obtained through only the traversal of the disjunctive search space. The obtained experimental results highlighted interesting compactness rates.

Our exploration of the disjunctive search space allows the direct derivation of the disjunctive and negative supports of itemsets. This constituted a motivating starting point for mining *generalized association rules*. We thus considered as a starting point an exact concise representation of frequent itemsets only containing specific elements from the disjunctive search space, namely frequent essential itemsets and their associated disjunctive closed itemsets. Although we mainly focused on rules containing disjunction of items as well as negated ones, having at hand such a concise representation is important. Indeed, it allows the derivation of the exact frequency of each literalset whose positive variation is a frequent itemset. This makes our proposal generic and, thus, not restricted to some association rule forms, regardless the others. In addition to the two commonly considered types of rules – exact and approximate – we also considered *approximated* rules, and we detailed how quality measures of these rules are ap-

proximated. Our experiments showed the usefulness of exploring the disjunctive search space towards extracting generalized association rules.

8.2 Short and Long Term Perspectives

The obtained results open some short-term and long-term perspectives. Short-term perspectives are as follows:

Post-processing the mined association rules: Although we concentrated in this thesis on subsets of (generalized) association rules, the number of mined rules remains large in some cases. This highlights the importance of post-processing the mined rules towards further pruning them *w.r.t.* user's preference, use of a combination of quality measures or summarization techniques, etc.

Reducing the disjunctive closed itemset-based concise representation: The concise representation DCIs_rep we proposed (*cf.* page 121) is homogeneous in the sense that it is only composed by disjunctive closed itemsets associated to their disjunctive supports. However, it results from the union of two disjoint sets, namely $\mathcal{EDCI}$ and $\mathcal{ADCI}$. Both sets convey different knowledge since the former has the set of frequent essential itemsets as a seed, while the latter – added to ensure the exact regeneration of frequent itemsets – has some particular infrequent essential itemsets as a generator. Note that these latter elements can further be pruned using a recent optimization highlighted in [Kryszkiewicz, 2009]. In addition, an issue which deserves further exploration is about the possible existence of a regeneration mechanism ensuring the exact regeneration of frequent itemsets starting from $\mathcal{EDCI}$, without using $\mathcal{ADCI}$.

Long-term perspectives are described in the following:

Generalization of the obtained results to other pattern classes and similar constructs: The implications of the redundancy removal we carried out for the minimal generator set will constitute an interesting issue. Indeed, this key concept – minimal generator – is at the roots of the design of different pattern classes. Minimal generators also have similar constructs in different important fields. It will hence be challenging to study which proposed results can be directly applied and those requiring to be adapted according to the pattern class under treatment. For example, through the conjunctive/disjunctive search space, the redundancy removal within each set of itemsets fulfilling the order ideal property, like the non-derivable itemsets, essential itemsets, etc., follows the same process as for minimal generators. Nevertheless, when dealing with minimal generators of sequential patterns, the fact that in an equivalence class may cohabit more than one closed sequential pattern [Lo *et al.*, 2008] should be taken into consideration.

Interestingly, the succinct and informative association rules, we obtained once the redundancy within minimal generators taken into consideration, can be extended to implication-closed sets. Indeed, the closure operator induced by an extraction context and the closure operator induced by the set of implications that are valid in this context coincide [Hermann and Sertkaya, 2008].

The proposed disjunctive closure operator can also be extended to the general case of Boolean expressions, instead of single items, as well as to other pattern classes. We also think being a promising issue the extension of the proposed process for generalized association rule mining to

take into account different tasks, like classification through decision trees where disjunctive rules are of paramount importance. Another interesting perspective is to extract new classification rules having as a premise part containing disjunction of items, instead of the ubiquitous conjunction of items.

Design, implementation and evaluation of new mining algorithms: From an algorithmic point of view, at its current stage, our study of the redundancy within minimal generators focuses on the efficient generation of the DSFS family. Thus, besides our own method, other algorithms from the literature working with MGs could be adapted for this task, both breadth-first search ones, *e.g.*, TITANIC [Stumme *et al.*, 2002], and depth-first ones, *e.g.*, the right-to-left search GR-GROWTH algorithm [Li *et al.*, 2006]. Of course, it will be interesting to compare performances of these algorithms on different datasets. The same strategy applies for the extraction of the disjunctive closed itemset-based representation of frequent itemsets. In this respect, we can benefit from the significant performance improvements of algorithms dedicated to frequent closed itemset mining [Bayardo *et al.*, 2004, Ben Yahia *et al.*, 2006]. The next step in this direction is the design of efficient expansion methods, *i.e.*, ones yielding to the entire sets from the representations proposed in this thesis.

Setting hybrid mining approaches: In this important issue, we intend to set up a platform that offers an adaptive selection of the "most adequate" mining tool of interesting patterns according to the dataset under treatment. The main goal is to guide, through meta-rules, for example the choice of the search space – conjunctive or disjunctive – to be explored. Our investigations show that this issue is highly correlated with that of determining the relation between their associated closure operators applied, respectively, on a given context and its dual. This latter context is obtained by replacing the presence of an item in the initial context by its absence and vice versa. This will make it possible setting up a hybrid approach aiming at exploring either the conjunctive or the disjunctive search space according to context characteristics. Indeed, thanks to such a relation, we can for example adapt the extraction algorithms of conjunctive/disjunctive closed itemsets proposed in the literature by choosing either the original context, or the dual context.

A more challenging problem is how to decide which one choosing to be mined: the initial context or its dual? The answer should take into consideration the sparseness/density of both contexts as well as the mining task to which is dedicated the algorithm (for example, frequent itemsets *vs.* frequent *closed* itemsets). In this respect, we started a study based on the succinct system of minimal generators (SSMG) having for purpose a formal characterization of the sparseness of a context [Hamrouni *et al.*, 2009a]. As a result, we introduced the first formal definition of this concept. From a practical point of view, to ensure the efficiency of the mining task, the detection of context sparseness should be performed on the fly. This can for example be done using a representative sample of the whole context [Toivonen, 1996b]. However, this raises the following questions:

(i) What will be the "good" size of this sample?

(ii) What will be the optimal number of portions the whole interval of sparseness values should be divided into to ensure an acceptable precision degree of the obtained contexts categorization?

The answer *a priori* closely depends on the difficulty level of the mining process. A theoretical study

and validating experiments have to be carried out to get bounds for sample sizes, in connection with the desired accuracy of the results. Another important theoretical issue consists in analyzing the relation between the sparseness measure of a context and that of its dual. This question is highly correlated with that of determining the relation between their respective lattices, obtained thanks to the Galois closure operator and the proposed disjunctive closure operator, respectively. An in-depth analysis of the common characteristics of both conjunctive and disjunctive search spaces as well as their differences *w.r.t.* the mined patterns is thus a thriving issue.

Further exploration of generalized association rules: In this thesis, we mainly concentrated on generalized rules under a generalization of the support-confidence framework. In order to reduce even more the number of extracted rules while retaining interesting ones for the end-users, the selection of the right quality measures [Geng and Hamilton, 2006, Hébert and Crémilleux, 2007] that suit each generalized association rule form is necessary. These measures should then be generalized to also take into consideration the disjunction and negation of items. This will allow guiding the mining process according to the couple (*rule form*, *measure*). In this respect, the proposed process can easily be adapted to efficiently extract generalized association rules based on correlated patterns *w.r.t.* the *bond* measure [Omiecinski, 2003]. In addition, searching for the relationships between the various rule forms deserves a thorough investigation. Its main motivation is to extend the concept of concise representation to generalized association rules. The purpose is thus to only retain a subset of valid rules – *w.r.t.* a given set of quality measures – while being able to derive the remaining redundant ones without information loss. Adequate axiomatic systems need thus to be set up taking into account the used connectors between items. Noteworthily, this exploration can exploit the results offered by the general GUHA approach [Hájek and Havránek, 1978, Hajek and Holena, 2003].

Extension of visualization tools: To make easier the manipulation and interpretation of generalized association rules by the end-users, visualization tools are also of paramount importance. Indeed, these latter make possible the end-users to concentrate on particular areas of interest, by zooming them for example. This motivates the extension of existing prototypes for association rule visualization to the generalized case. For example, our CbVAR prototype [Ben Yahia *et al.*, 2009a, Couturier *et al.*, 2007] was designed to visualize thousands of positive association rules thanks to a clustering-based technique. Interestingly, this prototype covers the mining step as well as the visualization step of association rules. In the generalized case, the mining component can be ensured using the proposed GARM tool. However, the visualization step should be extended *w.r.t.* the connectors involved within the visualized rules and the type of items (positive or negative). The integration of quality measures, in addition to the support and confidence ones, as well as user-specified constraints will also be helpful for further pattern pruning.

Note that the visualization component of CbVAR can be straightforwardly used once coupled with the proposed IMG_Extractor algorithm, *w.r.t.* succinct association rules. However, we think that a careful study of the effect of the total order relation choice, on the quality of the extracted rules according to the data under consideration, presents an interesting issue towards increasing the knowledge usefulness.

Extraction of rare patterns and minimal transversal of hypergraphs: In this thesis, we mainly

focused on *frequent* itemsets as well as *valid* association rules *w.r.t.* a minimum support threshold *minsupp*. It is however worth noting the increase interest within the data mining community in getting out rare patterns [Weiss, 2004], and especially rare itemsets and rare association rules [Liu *et al.*, 1999, Yun *et al.*, 2003]. Rare patterns convey information about rare situations/events useful for detecting, for example, the causes of rare diseases from medical data [Koh and Rountree, 2005], the suspicious transactions from financial data [Manning *et al.*, 2008], etc. Such patterns are often missed in a frequent pattern mining process since their frequencies do not reach the *minsupp* threshold. In this situation, the adaptation of the concise representations proposed here should take into account the difference in the structural properties of both sets of frequent itemsets and of rare ones. Indeed, the former is an order ideal, while the second is an order filter (a rare itemset has all its superset also rare).

We also aim at exploiting our exploration of the disjunctive search space towards the efficient detection of minimal transversals of a hypergraph [Berge, 1989, Fredman and Khachiyan, 1996]. This latter structure can be represented through an extraction context such that objects represent hyperedges, while items stand for the corresponding vertices. Within the aforementioned search space, essential itemsets belonging to the disjunctive equivalence class having for support the cardinality of the whole object set are the corresponding minimal transversals. Since this equivalence class is the top one within the disjunctive lattice (its disjunctive closure being equal to the whole set of items), levelwise search algorithms would examine many unnecessary candidates. In this situation, it will be interesting to benefit from the proposed disjunctive closure operator in order to efficiently detect the targeted class and, hence, its essential itemsets.

Application of the obtained results for real-life datasets: The application of the obtained results on real-life datasets is actually a challenging task. In this respect, it is highly interesting to extract the proposed concise representations to deal with bioinformatics data [Wang *et al.*, 2005] and, in particular, gene-expression data analysis. In gene-expression datasets, items represent gene expression properties, while objects stand for biological situations (or biological experiments). The frequent itemsets hence denote sets of genes that are frequently co-regulated and thus can be suspected to participate to a common function within the cells [Besson *et al.*, 2005]. These datasets offer the most interesting cases for the application of concise representation since they are dense and strongly correlated [Becquet *et al.*, 2002, Besson *et al.*, 2005, Gasmi *et al.*, 2005]. They hence produce equivalence classes containing a large number of patterns, sharing common characteristics.

In this situation, the redundancy removal within the minimal generator/essential itemset set can be useful towards reducing the number of generalized association rules that can be drawn from such datasets. In addition, taking into consideration the absence of genes (*i.e.*, negative items) can be a useful tool for analyzing the effect of the absence of one or more genes in the evolution of the other ones. A biologist can beforehand select the most interesting rule forms *w.r.t.* the research purposes. For example, the possible mined gene interactions can be as follows:

- Intra-biological situation correlations: in this first case, the interactions are between genes that simultaneously appear in the same biological experiments. The use of minimal generators and their associated closures can be helpful in this case.

- Inter-biological situations correlations: in this second case, the interactions are between complementary occurrence genes, *i.e.*, those that characterize different experiments. Here, the use of essential and disjunctive closed itemsets can be helpful to catch useful information.

Bibliography

[Agrawal *et al.*, 1993] Agrawal, R., Imielinski, T. and Swami, A. (1993). Mining association rules between sets of items in large databases. *In Proceedings of the ACM-SIGMOD International Conference on Management of Data (SIGMOD 1993), Washington D. C., USA*, pages 207–216.

[Agrawal *et al.*, 1996] Agrawal, R., Mannila, H., Srikant, R., Toivonen, H. and Verkamo, A. I. (1996). Fast discovery of association rules. *In Advances in Knowledge Discovery and Data Mining, AAAI Press, Menlo Park, CA, USA*, pages 307–328.

[Agrawal and Srikant, 1994] Agrawal, R. and Srikant, R. (1994). Fast algorithms for mining association rules. *In Proceedings of the 20th International Conference on Very Large Data Bases (VLDB 1994), Santiago, Chile*, pages 478–499.

[Antonie and Zaïane, 2004] Antonie, M. and Zaïane, O. R. (2004). Mining positive and negative association rules: An approach for confined rules. *In Proceedings of the 8th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD 2004), LNCS, volume 3202, Springer-Verlag, Pisa, Italy*, pages 27–38.

[Armstrong, 1974] Armstrong, W. W. (1974). Dependency structures of database relationships. *In Proceedings of IFIP Congress, Geneva, Switzerland*, pages 580–583.

[Ashrafi *et al.*, 2007] Ashrafi, M. Z., Taniar, D. and Smith, K. (2007). Redundant association rules reduction techniques. *International Journal Business Intelligence and Data Mining*, 2(1):29–63.

[Baixeries *et al.*, 2009] Baixeries, J., Szathmary, L., Valtchev, P. and Godin, R. (2009). Yet a faster algorithm for building the Hasse diagram of a concept lattice. *In Proceedings of the 7th International Conference on Formal Concept Analysis (ICFCA 2009), LNCS, volume 5548, Springer-Verlag, Darmstadt, Germany*, pages 162–177.

[Balcázar *et al.*, 2007] Balcázar, J. L., Bifet, A. and Lozano, A. (2007). Mining frequent closed unordered trees through natural representations. *In Proceedings of the 15th International Conference on Conceptual Structures (ICCS 2007), LNCS, volume 4604, Springer-Verlag, Sheffield, UK*, pages 347–359.

[Balcázar and Casas-Garriga, 2007] Balcázar, J. L. and Casas-Garriga, G. (2007). Horn axiomatizations for sequential data. *Theoretical Computer Science*, 371(3):247–264.

[Baralis and Chiusano, 2004] Baralis, E. and Chiusano, S. (2004). Essential classification rule sets. *ACM Transactions on Database Systems*, 29(4):635–674.

[Barbut and Monjardet, 1970] Barbut, M. and Monjardet, B. (1970). *Ordre et classification. Algèbre et Combinatoire.* Hachette, Tome II.

[Bastide *et al.*, 2000a] Bastide, Y., Pasquier, N., Taouil, R., Stumme, G. and Lakhal, L. (2000a). Mining minimal non-redundant association rules using frequent closed itemsets. *In Proceedings of the 1st International Conference on Computational Logic (DOOD 2000), LNAI, volume 1861, Springer-Verlag, London, UK*, pages 972–986.

[Bastide *et al.*, 2000b] Bastide, Y., Taouil, R., Pasquier, N., Stumme, G. and Lakhal, L. (2000b). Mining frequent patterns with counting inference. *ACM-SIGKDD Explorations*, 2(2):66–75.

[Bayardo, 1998] Bayardo, R. J. (1998). Efficiently mining long patterns from databases. *In Proceedings of the International Conference on Management of Data (SIGMOD 1998), Seattle, Washington, USA*, pages 85–93.

[Bayardo *et al.*, 2004] Bayardo, R. J., Goethals, B. and Zaki, M. J. (2004). Frequent itemset mining implementations repository. Available at http://fimi.cs.helsinki.fi/, accessed on July 24th, 2009.

[Becquet *et al.*, 2002] Becquet, C., Blachon, S., Jeudy, B., Boulicaut, J.-F. and Gandrillon, O. (2002). Strong-association-rule mining for large-scale gene-expression data analysis: a case study on human SAGE data. *Genome Biology*, 3.

[Ben Yahia *et al.*, 2009a] Ben Yahia, S., Couturier, O., Hamrouni, T. and Mephu Nguifo, E. (2009a). *Meta-knowledge based approach for an interactive visualization of large amounts of association rules*, pages 202–225. In the Book on Post-Mining of Association Rules: Techniques for Effective Knowledge Extraction, IGI Global Publisher.

[Ben Yahia *et al.*, 2009b] Ben Yahia, S., Gasmi, G. and Mephu Nguifo, E. (2009b). A new generic basis of factual and implicative association rules. *To appear in Intelligent Data Analysis (IDA) - An International Journal, IOS Press.*

[Ben Yahia *et al.*, 2006] Ben Yahia, S., Hamrouni, T. and Mephu Nguifo, E. (2006). Frequent closed itemset based algorithms: A thorough structural and analytical survey. *ACM-SIGKDD Explorations*, 8(1):93–104.

[Ben Yahia and Mephu Nguifo, 2004] Ben Yahia, S. and Mephu Nguifo, E. (2004). Revisiting generic bases of association rules. *In Proceedings of 6th International Conference on Data Warehousing and Knowledge Discovery (DaWaK 2004), LNCS, volume 3181, Springer-Verlag, Zaragoza, Spain*, pages 58–67.

[Berge, 1989] Berge, C. (1989). *Hypergraphs.* North Holland, Amsterdam.

[Berry and Linoff., 2004] Berry, M. J. A. and Linoff., G. S. (2004). *Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management, Second Edition.* Wiley Publishing.

[Besson *et al.*, 2005] Besson, J., Robardet, C., Boulicaut, J.-F. and Rome, S. (2005). Constraint-based concept mining and its application to microarray data analysis. *Intelligent Data Analysis*, 9(1):59–82.

[Bonchi and Lucchese, 2006] Bonchi, F. and Lucchese, C. (2006). On condensed representations of constrained frequent patterns. *Knowledge and Information Systems*, 9(2):180–201.

[Boulicaut *et al.*, 2003] BOULICAUT, J.-F., BYKOWSKI, A. and RIGOTTI, C. (2003). Free-sets: A condensed representation of Boolean data for the approximation of frequency queries. *Data Mining and Knowledge Discovery*, 7(1):5–22.

[Boulicaut and Jeudy, 2001] BOULICAUT, J.-F. and JEUDY, B. (2001). Mining free itemsets under constraints. *In Proceedings of the International Database Engineering and Application Symposium (IDEAS 2001), Grenoble, France*, pages 322–329.

[Brijs, 2003] BRIJS, T. (2003). Retail market basket data set. *In Proceedings of the IEEE ICDM Workshop on Frequent Itemset Mining Implementations (FIMI 2003)*, volume 90 de *CEUR Workshop Proceedings, Melbourne, Florida, USA*.

[Brijs *et al.*, 1999] BRIJS, T., SWINNEN, G., VANHOOF, K. and WETS, G. (1999). Using association rules for product assortment decisions: A case study. *In Proceedings of the 6th ACM-SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 1999), San Diego, CA, USA*, pages 254–260.

[Bykowski and Rigotti, 2001] BYKOWSKI, A. and RIGOTTI, C. (2001). A condensed representation to find frequent patterns. *In Proceedings of the 12th ACM Symposium on Principles Of Database Systems (PODS 2001), ACM Press, Santa Barbara, CA, USA*, pages 267–273.

[Bykowski and Rigotti, 2003] BYKOWSKI, A. and RIGOTTI, C. (2003). DBC: A condensed representation of frequent patterns for efficient mining. *Information Systems*, 28(8):949–977.

[Calders, 2004] CALDERS, T. (2004). Deducing bounds on the support of itemsets. *In Database Support for Data Mining Applications: Discovering Knowledge with Inductive Queries, LNCS, volume 2682, Springer-Verlag*, pages 214–233.

[Calders and Goethals, 2002] CALDERS, T. and GOETHALS, B. (2002). Mining all non-derivable frequent itemsets. *In Proceedings of the 6th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD 2002), LNAI, volume 2431, Springer-Verlag, Helsinki, Finland*, pages 74–85.

[Calders and Goethals, 2003] CALDERS, T. and GOETHALS, B. (2003). Minimal k-free representations of frequent sets. *In Proceedings of the 7th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD 2003), LNAI, volume 2838, Springer-Verlag, Cavtat-Dubrovnik, Croatia*, pages 71–82.

[Calders and Goethals, 2005] CALDERS, T. and GOETHALS, B. (2005). Depth-first non-derivable itemset mining. *In Proceedings of the 5th SIAM International Conference on Data Mining (SDM 2005), Newport Beach, CA, USA*, pages 250–261.

[Calders and Goethals, 2007] CALDERS, T. and GOETHALS, B. (2007). Non-derivable itemset mining. *Data Mining and Knowledge Discovery*, 14(1):171–206.

[Calders *et al.*, 2005] CALDERS, T., RIGOTTI, C. and BOULICAUT, J.-F. (2005). A survey on condensed representations for frequent sets. *In Constraint Based Mining and Inductive Databases, LNAI, volume 3848, Springer-Verlag*, pages 64–80.

[Casali *et al.*, 2003] CASALI, A., CICCHETTI, R. and LAKHAL, L. (2003). Mining concise representations of frequent multidimensional patterns. *In Proceedings of the 11th International Conference on Conceptual Structures (ICCS 2003), LNAI, volume 2746, Springer-Verlag, Dresden, Germany*, pages 351–361.

[Casali *et al.*, 2005a] CASALI, A., CICCHETTI, R. and LAKHAL, L. (2005a). Essential patterns: A perfect cover of frequent patterns. *In Proceedings of the 7th International Conference on Data Warehousing and Knowledge Discovery (DaWaK 2005), LNCS, volume 3589, Springer-Verlag, Copenhagen, Denmark*, pages 428–437.

[Casali *et al.*, 2005b] CASALI, A., CICCHETTI, R., LAKHAL, L. and LOPES, S. (2005b). Couvertures parfaites des motifs fréquents. *Revue d'Ingénierie des Systèmes d'Information (ISI), Hermès-Lavoisier*, 10(2):117–138.

[Casali *et al.*, 2009] CASALI, A., NEDJAR, S., CICCHETTI, R., LAKHAL, L. and NOVELLI, N. (2009). Lossless reduction of datacubes using partitions. *International Journal of Data Warehousing and Mining, IGI Global Publisher*, 5(1):18–35.

[Ceglar and Roddick, 2006] CEGLAR, A. and RODDICK, J. F. (2006). Association mining. *ACM Computing Surveys*, 38(2).

[Cheng *et al.*, 2006] CHENG, J., KE, Y. and NG, W. (2006). δ-tolerance closed frequent itemsets. *In Proceedings of the 6th IEEE International Conference on Data Mining (ICDM 2006), Hong Kong, China*, pages 139–148.

[Cheng *et al.*, 2008] CHENG, J., KE, Y. and NG, W. (2008). Effective elimination of redundant association rules. *Data Mining and Knowledge Discovery*, 16(2):221–249.

[Couturier *et al.*, 2007] COUTURIER, O., HAMROUNI, T., BEN YAHIA, S. and MEPHU NGUIFO, E. (2007). A scalable association rule visualization towards displaying large amount of knowledge. *In Proceedings of the 11nd International Conference on Information Visualization (IV 2007), IEEE Computer Society Press, Zurich, Switzerland*, pages 657–663.

[Davey and Priestley, 2002] DAVEY, B. and PRIESTLEY, H. (2002). *Introduction to Lattices and Order*. Cambridge University Press.

[Denden *et al.*, 2008] DENDEN, I., HAMROUNI, T. and BEN YAHIA, S. (2008). Efficient exploration of the disjunctive lattice towards extracting concise representations of frequent patterns (in French). *In Proceedings of the 9th African Conference on Research in Computer Science and Applied Mathematics (CARI 2008), Rabat, Morocco*, pages 443–450.

[Deogun and Jiang, 2005] DEOGUN, J. S. and JIANG, L. (2005). SARM - succinct association rule mining: An approach to enhance association mining. *In Proceedings of the 15th International Symposium on Methodologies for Intelligent Systems (ISMIS 2005), LNAI, volume 3488, Springer-Verlag, Saratoga Springs, New York, USA*, pages 121–130.

[Dong *et al.*, 2005] DONG, G., JIANG, C., PEI, J., LI, J. and WONG, L. (2005). Mining succinct systems of minimal generators of formal concepts. *In Proceedings of the 10th International Conference on Database Systems for Advanced Applications (DASFAA 2005), LNCS, volume 3453, Springer-Verlag, Beijing, China*, pages 175–187.

[Eiter and Gottlob, 1995] EITER, T. and GOTTLOB, G. (1995). Identifying the minimal transversals of a hypergraph and related problems. *SIAM Journal on Computing*, 24(6):1278–1304.

[Elble *et al.*, 2003] ELBLE, J., HEEREN, C. and PITT, L. (2003). Optimized disjunctive association rules via sampling. *In Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM 2003), Melbourne, Florida, USA*, pages 43–50.

[Fayyad *et al.*, 1996] FAYYAD, U. M., PIATETSKY-SHAPIRO, G. and SMYTH, P. (1996). From data mining to knowledge discovery in database. *Artificial Intelligence Magazine*, 17(3):37–54.

[Flouvat *et al.*, 2005] FLOUVAT, F., MARCHI, F. D. and PETIT, J.-M. (2005). A thorough experimental study of datasets for frequent itemsets. *In Proceedings of the 5th IEEE International Conference on Data Mining (ICDM 2005), IEEE Computer Society Press, New Orleans, USA*, pages 162–169.

[Frawley *et al.*, 1992] FRAWLEY, W. J., PIATETSKY-SHAPIRO, G. and MATHEUS, C. J. (1992). Knowledge discovery in databases - an overview. *Artificial Intelligence Magazine*, 13:57–70.

[Fredman and Khachiyan, 1996] FREDMAN, M. L. and KHACHIYAN, L. (1996). On the complexity of dualization of monotone disjunctive normal forms. *Journal of Algorithms*, 21:618–628.

[Galambos and Simonelli, 2000] GALAMBOS, J. and SIMONELLI, I. (2000). *Bonferroni-type inequalities with applications.* Springer.

[Ganter and Wille, 1999] GANTER, B. and WILLE, R. (1999). *Formal Concept Analysis.* Springer.

[Gasmi *et al.*, 2007] GASMI, G., BEN YAHIA, S., MEPHU NGUIFO, E. and BOUKER, S. (2007). Extraction of association rules based on literalsets. *In Proceedings of the 9th International Conference on Data Warehousing and Knowledge Discovery (DaWaK 2007), LNCS, volume 4654, Springer-Verlag, Regensburg, Germany*, pages 293–302.

[Gasmi *et al.*, 2005] GASMI, G., HAMROUNI, T., ABDELHAK, S., BEN YAHIA, S. and MEPHU NGUIFO, E. (2005). Extracting generic basis of association rules from SAGE data. *In The ECML/PKDD Discovery Challenge on gene expression data co-located with the 9th European Conference on Conference on Machine Learning and Knowledge Discovery in Databases (ECML/PKDD 2005), Porto, Portugal*, pages 84–89.

[Gély *et al.*, 2005] GÉLY, A., MEDINA, R., NOURINE, L. and RENAUD, Y. (2005). Uncovering and reducing hidden combinatorics in Guigues-Duquenne bases. *In Proceedings of the 3rd International Conference on Formal Concept Analysis (ICFCA 2005), LNAI, volume 3403, Springer-Verlag, Lens, France*, pages 235–248.

[Geng and Hamilton, 2006] GENG, L. and HAMILTON, H. J. (2006). Interestingness measures for data mining: A survey. *ACM Computing Surveys*, 38:1–31.

[Geurts, 2003] GEURTS, K. (2003). Traffic accidents data set. *In Proceedings of the IEEE ICDM Workshop on Frequent Itemset Mining Implementations (FIMI 2003)*, volume 90 de *CEUR Workshop Proceedings, Melbourne, Florida, USA.*

[Goethals *et al.*, 2005] GOETHALS, B., MUHONEN, J. and TOIVONEN, H. (2005). Mining non-derivable association rules. *In Proceedings of the 5th SIAM International Conference on Data Mining (SDM 2005), Newport Beach, CA, USA*, pages 239–249.

[Goethals and Zaki, 2003] GOETHALS, B. and ZAKI, M. J. (2003). FIMI'03: Workshop on frequent itemset mining implementations. *In Proceedings of the IEEE ICDM Workshop on Frequent Itemset Mining Implementations (FIMI 2003)*, CEUR Workshop Proceedings 90, Melbourne, Florida, USA.

[Grün, 1998] GRÜN, G. A. (1998). New forms of association rules. Rapport technique TR 1998-15, School of Computing Science, Simon Fraser University, Burnaby, BC, Canada.

[Grunwald, 2007] GRUNWALD, P. D. (2007). *The Minimum Description Length Principle.* MIT Press.

[Guigues and Duquenne, 1986] GUIGUES, J. L. and DUQUENNE, V. (1986). Familles minimales d'implications informatives résultant d'un tableau de données binaires. *Mathématiques et Sciences Humaines*, 24(95):5–18.

[Guillet and Hamilton, 2007] GUILLET, F. and HAMILTON, H. J. (2007). *Quality Measures in Data Mining.* Studies in Computational Intelligence, volume 43, Springer.

[Hájek and Havránek, 1978] HÁJEK, P. and HAVRÁNEK, T. (1978). *Mechanizing Hypothesis Formation: Mathematical Foundations for a General Theory.* Springer-Verlag.

[Hajek and Holena, 2003] HAJEK, P. and HOLENA, M. (2003). Formal logics of discovery and hypothesis formation by machine. *Theoretical Computer Science*, 292(2):345–357.

[Hamrouni *et al.*, 2006] HAMROUNI, T., BEN YAHIA, S. and MEPHU NGUIFO, E. (2006). Redundancy-free generic bases of association rules. *In Proceedings of the 8th French Conference on Machine Learning (CAp 2006), Presses Universitaires de Grenoble, Trégastel, France*, pages 363–378.

[Hamrouni *et al.*, 2008a] HAMROUNI, T., BEN YAHIA, S. and MEPHU NGUIFO, E. (2008a). Succinct minimal generators: Theoretical foundations and applications. *International Journal of Foundations of Computer Sciences, World Scientific Publishing Company, Guest Editors: R. Belohlávek*, 19(2):271–296.

[Hamrouni *et al.*, 2008b] HAMROUNI, T., BEN YAHIA, S. and MEPHU NGUIFO, E. (2008b). GARM: Generalized association rule mining. *In Proceedings of the 6th International Conference on Concept Lattices and their Applications (CLA 2008), Olomouc, Czech Republic*, pages 145–156.

[Hamrouni *et al.*, 2009a] HAMROUNI, T., BEN YAHIA, S. and MEPHU NGUIFO, E. (2009a). Assessing local and global sparseness measure for frequent itemset contexts. *To appear in the International Journal of Computing & Information Sciences, Guest Editors: J. Diatta.*

[Hamrouni *et al.*, 2009b] HAMROUNI, T., BEN YAHIA, S. and MEPHU NGUIFO, E. (2009b). Sweeping the disjunctive search space towards mining new exact concise representations of frequent itemsets. *Data & Knowledge Engineering*, 68(10):1091–1111.

[Hamrouni *et al.*, 2005a] HAMROUNI, T., BEN YAHIA, S. and SLIMANI, Y. (2005a). Avoiding the itemset closure computation "pitfall". *In Proceedings of the 3rd International Conference on Concept Lattices and their Applications (CLA 2005), Olomouc, Czech Republic*, pages 46–59.

[Hamrouni *et al.*, 2005b] HAMROUNI, T., BEN YAHIA, S. and SLIMANI, Y. (2005b). PRINCE: An algorithm for generating rule bases without closure computations. *In Proceedings of the 7th International Conference on Data Warehousing and Knowledge Discovery (DaWaK 2005), LNCS, volume 3589, Springer-Verlag, Copenhagen, Denmark*, pages 346–355.

[Hamrouni *et al.*, 2007a] Hamrouni, T., Denden, I., Ben Yahia, S. and Mephu Nguifo, E. (2007a). A new concise representation of frequent patterns through disjunctive search space. *In Proceedings of the 5th International Conference on Concept Lattices and their Applications (CLA 2007), Montpellier, France*, pages 50–61.

[Hamrouni *et al.*, 2007b] Hamrouni, T., Valtchev, P., Ben Yahia, S. and Mephu Nguifo, E. (2007b). About the lossless reduction of the minimal generator family of a context. *In Proceedings of the 5th International Conference on Formal Concept Analysis (ICFCA 2007), LNAI, Springer-Verlag, volume 4390, Clermont-Ferrand, France*, pages 130–150.

[Han *et al.*, 2007] Han, J., Cheng, H., Xin, D. and Yan, X. (2007). Frequent pattern mining: current status and future directions. *Data Mining and Knowledge Discovery*, 15(1):55–86.

[Han and Kamber, 2006] Han, J. and Kamber, M. (2006). *Data Mining: Concepts and Techniques. Second edition.* Morgan Kaufmann Publishers.

[Hattori *et al.*, 2008] Hattori, L., dos Santos, G., Cardoso, F. and Sampaio, M. (2008). Mining software repositories for software change impact analysis: A case study. *In Proceedings of the 23rd Brasilian Symposium on Database (SBBD 2008), Campinas, Brazil.*

[Hébert *et al.*, 2007] Hébert, C., Bretto, A. and Crémilleux, B. (2007). A data mining formalization to improve hypergraph minimal transversal computation. *Fundamenta Informaticae, IOS Press*, 80(4): 1–19.

[Hébert and Crémilleux, 2007] Hébert, C. and Crémilleux, B. (2007). A unified view of objective interestingness measures. *In Proceedings of the 5th International Conference Machine Learning and Data Mining in Pattern Recognition (MLDM 2007), LNCS, volume 4571, Springer-Verlag, Leipzig, Germany*, pages 533–547.

[Hermann and Sertkaya, 2008] Hermann, M. and Sertkaya, B. (2008). On the complexity of computing generators of closed sets. *In Proceedings of the 6th International Conference on Formal Concept Analysis (ICFCA 2008), LNAI, volume 4933, Springer-Verlag, Montréal, Canada*, pages 158–168.

[Kanda *et al.*, 2001] Kanda, K., Haraguchi, M. and Okubo, Y. (2001). Constructing approximate informative basis of association rules. *In Proceedings of the 4th International Conference Discovery Science (DS 2001), LNCS, volume 2226, Springer-Verlag, Washington, DC, USA*, pages 141–154.

[Kim, 2003] Kim, H. D. (2003). Complementary occurrence and disjunctive rules for market basket analysis in data mining. *In Proceedings of the 2nd IASTED International Conference Information and Knowledge Sharing (IKS 2003), Scottsdale, AZ, USA*, pages 155–157.

[Koh and Rountree, 2005] Koh, Y. S. and Rountree, N. (2005). Finding sporadic rules using apriori-inverse. *In Proceedings of the International 9th Pacific-Asia Conference on Knowledge Data Discovery (PAKDD 2005), LNAI, volume 3518, Springer-Verlag, Hanoi, Vietnam*, pages 97–106.

[Kryszkiewicz, 1998] Kryszkiewicz, M. (1998). Representative association rules and minimum condition maximum consequence association rules. *In Proceedings of the 2nd European Symposium on Principles of Data Mining and Knowledge Discovery (PKDD 1998), LNCS, volume 1510, Springer-Verlag, Nantes, France*, pages 361–369.

[Kryszkiewicz, 2001] Kryszkiewicz, M. (2001). Concise representation of frequent patterns based on disjunction-free generators. *In Proceedings of the 1st IEEE International Conference on Data Mining (ICDM 2001), San Jose, CA, USA*, pages 305–312.

[Kryszkiewicz, 2002] Kryszkiewicz, M. (2002). Concise representations of frequent patterns and association rules. Habilitation dissertation, Institute of Computer Science, Warsaw University of Technology, Warsaw, Poland.

[Kryszkiewicz, 2009] Kryszkiewicz, M. (2009). Closures of downward closed representations of frequent patterns. *In Proceedings of the 4th International Conference on Hybrid Artificial Intelligence Systems (HAIS 2009), LNCS, volume 5572, Springer-Verlag, Salamanca, Spain*, pages 104–112.

[Kuznetsov and Obiedkov, 2002] Kuznetsov, S. O. and Obiedkov, S. A. (2002). Comparing performance of algorithms for generating concept lattices. *Journal of Experimental and Theoretical Artificial Intelligence*, 14(2-3):189–216.

[Lee *et al.*, 2006] Lee, A. J. T., Lin, W. C. and Wang, C.-S. (2006). Mining association rules with multi-dimensional constraints. *The Journal of Systems and Software*, 79(1):79–92.

[Li *et al.*, 2005] Li, H., Li, J., Wong, L., Feng, M. and Tan, Y. (2005). Relative risk and odds ratio: a data mining perspective. *In Proceedings of the 24th ACM SIGMOD-SIGACT-SIGART symposium on Principles Of Database Systems (PODS 2005), Baltimore, Maryland, USA*, pages 368–377.

[Li, 2006] Li, J. (2006). On optimal rule discovery. *IEEE Transactions on Knowledge and Data Engineering*, 18(4):460–471.

[Li and Jones, 2006] Li, J. and Jones, J. (2006). Using multiple and negative target rules to make classifiers more understandable. *Knowledge Based Systems*, 19(6):438–444.

[Li *et al.*, 2006] Li, J., Li, H., Wong, L., Pei, J. and Dong, G. (2006). Minimum description length principle: Generators are preferable to closed patterns. *In Proceedings of the 21st National Conference on Artificial Intelligence (AAAI 2006), Boston, Massachusetts, USA*, pages 409–414.

[Li *et al.*, 2007] Li, J., Liu, G. and Wong, L. (2007). Mining statistically important equivalence classes and delta-discriminative emerging patterns. *In Proceedings of the 13th ACM International Conference on Knowledge Discovery and Data Mining (KDD 2007), ACM Press, San Jose, CA, USA*, pages 430–439.

[Liu *et al.*, 1999] Liu, B., Hsu, W. and Ma, Y. (1999). Mining association rules with multiple minimum supports. *In Proceedings of the 5th ACM International Conference on Knowledge Discovery and Data Mining (KDD 1999), ACM Press, San Diego, CA, USA*, pages 337–341.

[Liu *et al.*, 2007] Liu, G., Li, J. and Wong, L. (2007). A new concise representation of frequent itemsets using generators and a positive border. *Knowledge and Information Systems*, 15(1):55–86.

[Lo *et al.*, 2008] Lo, D., Khoo, S.-C. and Li, J. (2008). Mining and ranking generators of sequential patterns. *In Proceedings of the 8th SIAM International Conference on Data Mining (SDM 2008), Atlanta, Georgia, USA*, pages 553–564.

[Luxenburger, 1991] Luxenburger, M. (1991). Implications partielles dans un contexte. *Mathématiques, Informatique et Sciences Humaines*, 29(113):35–55.

[Maier, 1983] MAIER, D. (1983). *The theory of Relational Databases.* Computer Science Press.

[Mannila and Toinoven, 1996] MANNILA, H. and TOINOVEN, H. (1996). Multiple uses of frequent sets and condensed representations. *In Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD 1996), Portland, Oregon, USA*, pages 189–194.

[Mannila and Toivonen, 1997] MANNILA, H. and TOIVONEN, H. (1997). Levelwise search and borders of theories in knowledge discovery. *Data Mining and Knowledge Discovery*, 3(1):241–258.

[Manning *et al.*, 2008] MANNING, A. M., HAGLIN, D. J. and KEANE, J. A. (2008). A recursive search algorithm for statistical disclosure assessment. *Data Mining and Knowledge Discovery*, 16(2):165–196.

[Medina *et al.*, 2006] MEDINA, R., NOURINE, L. and RAYNAUD, O. (2006). Interactive association rules discovery. *In Proceedings of the 4th International Conference on Formal Concept Analysis (ICFCA 2006), LNAI, volume 3874, Springer-Verlag, Dresden, Germany*, pages 177–190.

[Mephu Nguifo, 1994] MEPHU NGUIFO, E. (1994). Galois lattice: A framework for concept learning, design, evaluation and refinement. *In Proceedings of the IEEE International Conference on Tools with Artificial Intelligence (ICTAI 1994), New-Orleans, USA*, pages 461–467.

[Mielikäinen *et al.*, 2006] MIELIKÄINEN, T., PANOV, P. and DZEROSKI, S. (2006). Itemset support queries using frequent itemsets and their condensed representations. *In Proceedings of the 9th International Conference Discovery Science (DS 2006), LNCS, volume 4265, Springer-Verlag, Barcelona, Spain*, pages 161–172.

[Morzy, 2006] MORZY, M. (2006). Efficient mining of dissociation rules. *In Proceedings of the 8th International Conference on Data Warehousing and Knowledge Discovery (DaWaK 2006), LNCS, volume 4081, Springer-Verlag, Krakow, Poland*, pages 228–237.

[Muhonen and Toivonen, 2006] MUHONEN, J. and TOIVONEN, H. (2006). Closed non-derivable itemsets. *In Proceedings of the 10th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD 2006), LNAI, volume 4213, Springer-Verlag, Berlin, Germany*, pages 601–608.

[Nanavati *et al.*, 2001] NANAVATI, A. A., CHITRAPURA, K. P., JOSHI, S. and KRISHNAPURAM, R. (2001). Mining generalised disjunctive association rules. *In Proceedings of the 10th International Conference on Information and Knowledge Management (CIKM 2001), Atlanta, Georgia, USA*, pages 482–489.

[Narushima, 1982] NARUSHIMA, H. (1982). Principle of inclusion-exclusion on partially order sets. *Discrete Mathematics*, 42:243–250.

[Omiecinski, 2003] OMIECINSKI, E. R. (2003). Alternative interest measures for mining associations in databases. *IEEE Transactions on Knowledge and Data Engineering*, 15(1):57–69.

[Pasquier, 2000] PASQUIER, N. (2000). Datamining : Algorithmes d'extraction et de réduction des règles d'association dans les bases de données. Thesis, École Doctorale Sciences pour l'Ingénieur de Clermont Ferrand, Université Clermont Ferrand II, France.

[Pasquier, 2009] PASQUIER, N. (2009). *Frequent closed itemset based condensed representations for association rules*, pages 248–273. In the Book on Post-Mining of Association Rules: Techniques for Effective Knowledge Extraction, IGI Global Publisher.

[Pasquier *et al.*, 1999a] Pasquier, N., Bastide, Y., Taouil, R. and Lakhal, L. (1999a). Discovering frequent closed itemsets for association rules. *In Proceedings of 7th International Conference on Database Theory (ICDT 1999), LNCS, volume 1540, Springer-Verlag, Jerusalem, Israel*, pages 398–416.

[Pasquier *et al.*, 1999b] Pasquier, N., Bastide, Y., Taouil, R. and Lakhal, L. (1999b). Efficient mining of association rules using closed itemset lattices. *Information Systems*, 24(1):25–46.

[Pei *et al.*, 2004] Pei, J., Dong, G., Zou, W. and Han, J. (2004). Mining condensed frequent-pattern bases. *Knowledge and Information Systems*, 6(5):570–594.

[Pei *et al.*, 2006] Pei, J., Yuan, Y., Lin, X., Jin, W., Ester, M., Liu, Q., Wang, W., Tao, Y., Yu, J. X. and Zhang, Q. (2006). Towards multidimensional subspace skyline analysis. *ACM Transactions on Database Systems*, 31(4):1335–1381.

[Pfaltz and Taylor, 2002] Pfaltz, J. L. and Taylor, C. M. (2002). Scientific knowledge discovery through iterative transformation of concept lattices. *In Proceedings of the Workshop on Discrete Applied Mathematics in conjunction with the 2nd SIAM International Conference on Data Mining, Arlington, Virginia, USA*, pages 65–74.

[Phan Luong, 2002] Phan Luong, V. (2002). The closed keys base of frequent itemsets. *In Proceedings of the 4th International Conference on Data Warehousing and Knowledge Discovery (DaWaK 2002), LNCS, volume 2454, Springer-Verlag, Aix-en-Provence, France*, pages 181–190.

[Raïssi *et al.*, 2008] Raïssi, C., Calders, T. and Poncelet, P. (2008). Mining conjunctive sequential patterns. *Data Mining and Knowledge Discovery*, 17(1):77–93.

[Ralbovský and Kuchar, 2007] Ralbovský, M. and Kuchar, T. (2007). Using disjunctions in association mining. *In Proceedings of the 7th Industrial Conference on Data Mining (ICDM 2007), LNCS, volume 4597, Springer-Verlag, Leipzig, Germany*, pages 339–351.

[Rissanen, 1978] Rissanen, J. (1978). Modeling by shortest data description. *Automatica*, 14:465–471.

[Sassi *et al.*, 2007] Sassi, M., Grissa-Touzi, A. and Ounelli, H. (2007). Clustering quality evaluation based on fuzzy FCA. *In Proceedings of 18th International Conference Database and Expert Systems Applications (DEXA 2007), LNCS, Springer-Verlag, volume 4653, Regensburg, Germany*, pages 639–649.

[Savasere *et al.*, 1998] Savasere, A., Omiecinski, E. and Navathe, S. (1998). Mining for strong negative associations in a large database of customer transactions. *In Proceedings of the 14th International Conference on Data Engineering (ICDE 1998), IEEE Computer Society Press, Orlando, Florida, USA*, pages 494–502.

[She, 2008] She, S. (2008). Feature model mining. Master Thesis, University of Waterloo, Waterloo, Ontario, Canada.

[Shima *et al.*, 2005] Shima, Y., Hirata, K., Harao, M., Yokoyama, S., Matsuoka, K. and Izumi, T. (2005). Extracting disjunctive closed rules from MRSA data. *In Proceedings of the 1st International Conference on Complex Medical Engineering (CME 2005), Takamatsu, Japan*, pages 321–325.

[Shima *et al.*, 2004] SHIMA, Y., MITSUISHI, S., HIRATA, K. and HARAO, M. (2004). Extracting minimal and closed monotone DNF formulas. *In Proceedings of the 7th International Conference Discovery Science (DS 2004), LNCS, volume 3245, Springer-Verlag, Padova, Italy*, pages 298–305.

[Soulet and Crémilleux, 2008] SOULET, A. and CRÉMILLEUX, B. (2008). Adequate condensed representations of patterns. *Data Mining and Knowledge Discovery*, 17(1):94–110.

[Srikant and Agrawal, 1995] SRIKANT, R. and AGRAWAL, R. (1995). Mining generalized association rules. *In Proceedings of the 21th International Conference on Very Large Data Bases (VLDB 1995), Zurich, Switzerland*, pages 407–419.

[Srikant *et al.*, 1997] SRIKANT, R., VU, Q. and AGRAWAL, R. (1997). Mining association rules with item constraints. *In Proceedings of the 3rd International Conference on Knowledge Discovery in Databases and Data Mining (KDD 1997), Newport Beach, CA, USA*, pages 67–73.

[Steinbach and Kumar, 2007] STEINBACH, M. and KUMAR, V. (2007). Generalizing the notion of confidence. *Knowledge and Information Systems*, 12(3):279–299.

[Stumme *et al.*, 2001] STUMME, G., TAOUIL, R., BASTIDE, Y., PASQUIER, N. and LAKHAL, L. (2001). Intelligent structuring and reducing of association rules with formal concept analysis. *In Proceedings of the Joint German/Austrian Conference on AI: Advances in Artificial Intelligence, LNCS, volume 2174, Springer-Verlag, Vienna, Austria*, pages 335–350.

[Stumme *et al.*, 2002] STUMME, G., TAOUIL, R., BASTIDE, Y., PASQUIER, N. and LAKHAL, L. (2002). Computing Iceberg concept lattices with TITANIC. *Data & Knowledge Engineering*, 42(2):189–222.

[Stumme *et al.*, 1998] STUMME, G., WILLE, R. and WILLE, U. (1998). Conceptual knowledge discovery in databases using formal concept analysis methods. *In Proceedings of the 2nd European Symposium on Principles of Data Mining and Knowledge Discovery (PKDD 1998), LNCS, volume 1510, Springer-Verlag, Nantes, France*, pages 450–458.

[Toivonen, 1996a] TOIVONEN, H. (1996a). Discovering of frequent patterns in large data collections. Thesis, University of Helsinki, Helsinki, Finland.

[Toivonen, 1996b] TOIVONEN, H. (1996b). Sampling large databases for association rules. *In Proceedings of the 22th International Conference on Very Large Data Bases (VLDB 1996), Bombay, India*, pages 134–145.

[Tzanis and Berberidis, 2007] TZANIS, G. and BERBERIDIS, C. (2007). Mining for mutually exclusive items in transaction databases. *International Journal of Data Warehousing and Mining, IGI Global Publisher*, 3(3):45–59.

[Uno *et al.*, 2004] UNO, T., ASAI, T., UCHIDA, Y. and ARIMURA, H. (2004). An efficient algorithm for enumerating closed patterns in transaction databases. *Proceedings of the 7th International Conference on Discovery Science (DS 2004), Padova, Italy*, pages 16–31.

[Valtchev *et al.*, 2004] VALTCHEV, P., MISSAOUI, R. and GODIN, R. (2004). Formal concept analysis for knowledge discovery and data mining: The new challenges. *In Proceedings of the 2nd International Conference on Formal Concept Analysis (ICFCA 2004), LNCS, Springer-Verlag, volume 2961, Sydney, Australia*, pages 352–371.

[Valtchev *et al.*, 2000] Valtchev, P., Missaoui, R. and Lebrun, P. (2000). A fast algorithm for building the Hasse diagram of a Galois lattice. *In Proceedings of the Conference on Combinatorics, Computer Science and Applications (LaCIM 2000), Montréal, Canada*, pages 293–306.

[Wang *et al.*, 2005] Wang, J., Zaki, M. J., Shasha, D. and Toivonen, H. (2005). *Data Mining in Bioinformatics.* Springer.

[Weiss, 2004] Weiss, G. M. (2004). Mining with rarity: A unifying framework. *ACM-SIGKDD Explorations*, 6(1):7–19.

[Wu *et al.*, 2004] Wu, X., Zhang, C. and Zhang, S. (2004). Efficient mining of both positive and negative association rules. *ACM Transactions on Information Systems*, 22(3):381–405.

[Xie *et al.*, 2006] Xie, Z., Chen, H. and Li, C. (2006). MFIS-mining frequent itemsets on data streams. *In Proceedings of the 2nd International Conference on Advanced Data Mining and Applications (ADMA 2006), LNCS, volume 4093, Springer-Verlag, Xi'an, China*, pages 1085–1093.

[Xie and Liu, 2005] Xie, Z. and Liu, Z. (2005). From intent reducts for attribute implications to approximate intent reducts for association rules. *In Proceedings of the 5th International Conference on Computer and Information Technology (CIT 2005), IEEE Computer Society Press, Shanghai, China*, pages 162–169.

[Xin *et al.*, 2007] Xin, D., Han, J., Yan, X. and Cheng, H. (2007). On compressing frequent patterns. *Data & Knowledge Engineering*, 60(1):5–29.

[Yan and Han, 2003] Yan, X. and Han, J. (2003). CloseGraph: Mining closed frequent graph patterns. *In Proceedings of the 9th ACM International Conference on Knowledge Discovery and Data Mining (KDD 2003), Washington, DC, USA*, pages 286–295.

[Yang *et al.*, 2001] Yang, C., Fayyad, U. M. and Bradley, P. S. (2001). Efficient discovery of error-tolerant frequent itemsets in high dimensions. *In Proceedings of the 7th ACM International Conference on Knowledge Discovery and Data Mining (KDD 2001), San Francisco, CA, USA*, pages 194–203.

[Ye and Keane, 1997] Ye, X. and Keane, J. A. (1997). Mining composite items in association rules. *In Proceedings of the 10th IEEE International Conference on Systems, Man, and Cybernetics (SMC 1997), Orlando, Florida, USA*, pages 1367–1372.

[Yun *et al.*, 2003] Yun, H., Ha, D., Hwang, B. and Ryu, K. (2003). Mining association rules on significant rare data using relative support. *The Journal of Systems and Software*, 67:181–191.

[Zaki, 2004] Zaki, M. J. (2004). Mining non-redundant association rules. *Data Mining and Knowledge Discovery*, 9(3):223–248.

[Zaki and Hsiao, 2002] Zaki, M. J. and Hsiao, C. J. (2002). CHARM: An efficient algorithm for closed itemset mining. *In Proceedings of the 2nd SIAM International Conference on Data Mining, Arlington, Virginia, USA*, pages 34–43.

[Zhao, 2006] Zhao, L. (2006). Mining subspace and boolean patterns from data. Thesis, Rensselaer Polytechnic Institute, Troy, New York, USA.

[Zhao *et al.*, 2006] ZHAO, L., ZAKI, M. J. and RAMAKRISHNAN, N. (2006). BLOSOM: A framework for mining arbitrary Boolean expressions. *In Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2006), Philadelphia, PA, USA*, pages 827–832.

[Zheng *et al.*, 2001] ZHENG, Z., KOHAVI, R. and MASON, L. (2001). Real world performance of association rule algorithms. *In Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM Press*, pages 401–406.

Part V

Appendix

Appendix A

Description of Benchmark Contexts

The benchmark contexts we used in our experiments are freely downloadable from the FIMI website at: *http://fimi.cs.helsinki.fi/data*. The first five contexts are commonly considered in the literature to be dense, *i.e.*, containing many long frequent itemsets at various levels of *minsupp* values [Bayardo, 1998]. While the last five are considered to be sparse, *i.e.*, containing a large number of items but only a few of them frequently occur in the context. Here, we describe the content of each context.

- **Chess:** This dataset is derived from the steps of Chess games. The format for the objects in this dataset is a sequence of **37** item values. Each object is a board-description for this chess endgame. The first **36** items describe the board. The last ($\mathbf{37}^{th}$) item is the classification: "win" or "no win". This context is available in the UC Irvine Machine Learning Database Repository (at *http://www.ics.uci.edu/~mlearn/MLRepository.html*).

- **Connect:** This dataset contains all legal **8**-ply positions in the game of connect-4 in which neither player has won yet, and in which the next move is not forced. This context is also available in the UC Irvine Machine Learning Database Repository.

- **Mushroom:** This dataset includes descriptions of hypothetical samples corresponding to **23** species of gilled mushrooms. Each species is identified as definitely edible, definitely poisonous, or of unknown edibility and not recommended. This latter class was combined with the poisonous one. This context is also available in the UC Irvine Machine Learning Database Repository.

- **Pumsb:** Pumsb contains census data from PUMS (Public Use Microdata Samples). Each object represents the answers to a census questionnaire, including the age, tax-filing status, marital status, income, sex, veteran status, and location of residence of the respondent.

Pumsb*: Pumsb* contains census data from PUMS (Public Use Microdata Samples). Each object represents the answers to a census questionnaire, including the age, tax-filing status, marital status,

income, sex, veteran status, and location of residence of the respondent. PUMSB* is obtained after deleting all frequent items for a minimum support threshold set to **80**% in the original PUMSB dataset (*i.e.*, items which appear in more than **80**% of the transactions contained in PUMSB).

Note that the versions of the five aforementioned datasets available on the FIMI website were prepared by R. J. Bayardo from the UCI datasets and PUMS.

- **ACCIDENTS:** This dataset of traffic accidents is obtained from the National Institute of Statistics (NIS) for the region of Flanders (Belgium) for the period **1991-2000** [Geurts, 2003]. The traffic accident data contains a rich source of information on the different circumstances in which the accidents have occurred: details about the accident (type of collision, injuries,...), traffic conditions (maximum speed, priority regulation, ...), environmental conditions (weather, light conditions, time of the accident, ...), road conditions (road surface, obstacles, ...), human conditions (alcohol, ...) and geographical conditions (location, physical characteristics,...).

- **KOSARAK:** This dataset was provided to the FIMI website by F. Bodon. It contains data corresponding to (anonymized) click-stream data of a Hungarian on-line news portal.

- **RETAIL:** The dataset contains information about the market basket of clients in a Belgian supermarket [Brijs, 2003, Brijs *et al.*, 1999]. Data were collected over three non-consecutive periods. This results in approximately **5** months of data. Each record in the dataset contains information about the date of purchase, the receipt number, the article number, the number of items purchased, the article price in Belgian Francs and the customer number. Although most of the products are identified by a unique bar code, some article numbers in the dataset represent a group of products rather than an individual product item. In total, **5, 133** customers have purchased at least one product in the supermarket during the data collection period.

- **T10I4D100K:** This dataset is a synthetic dataset generated using the generator from the IBM Almaden Quest research group, based on the algorithm introduced in [Agrawal and Srikant, 1994]. This generator is available at: *http://www.almaden.ibm.com/software/quest/Resources/index.shtml*. The goal of this generation is to create objects similar to those obtained in a supermarket environment. By applying certain distribution laws, the obtained datasets tend to mimic the real world compared to given characteristics. The data are generated in order to correspond, on average, to the input characteristics while respecting a certain distribution and the existence of exceptions.

  The different parameters for the generation are as follows:

  1. The average size of objects (T),
  2. The average size of maximal potentially frequent itemsets (I),
  3. The number of objects (D).

- **T40I10D100K:** Identically to T10I4D100K, this dataset is also generated by the generator from the IBM Almaden Quest research group. The differences between this dataset and T10I4D100K are the parameters given to the generator.

Table A.1 summarizes the characteristics of the considered contexts. Note that the respective numbers of items shown in Table A.1 for the PUMSB and PUMSB* datasets are somewhat different from the numbers reported in many contributions. Indeed, we only consider items that appear at least one time. For example, although the minimum item identifier in PUMSB is **0** and the maximum item identifier is **7, 116**, there are only **2, 113** distinct item identifers that appear in the dataset.

Context	**# of items**	**# of objects**	**Avg. size of objects**	**Max. size of objects**
CHESS	**75**	**3, 196**	**37.00**	**37**
CONNECT	**129**	**67, 557**	**43.00**	**43**
MUSHROOM	**119**	**8, 124**	**23.00**	**23**
PUMSB	**2, 113**	**49, 046**	**74.00**	**74**
PUMSB*	**2, 088**	**49, 046**	**50.48**	**63**
ACCIDENTS	**468**	**340, 183**	**33.81**	**52**
KOSARAK	**41, 270**	**990, 002**	**8.10**	**2, 498**
RETAIL	**16, 470**	**88, 162**	**10.31**	**77**
T10I4D100K	**870**	**100, 000**	**10.10**	**30**
T40I10D100K	**942**	**100, 000**	**39.61**	**78**

Table A.1: Characteristics of the considered benchmark contexts.

Appendix B

Selected Publication List

- **Exploration of the conjunctive search space:**
  - **Journals:**
    - T. Hamrouni, S. Ben Yahia, E. Mephu Nguifo: Succinct minimal generators: Theoretical foundations and applications. International Journal of Foundations of Computer Sciences (IJFCS), World Scientific Publishing Company, 19(2):271–296, April 2008. Guest Editor: R. Belohlávek.
    - S. Ben Yahia, T. Hamrouni, E. Mephu Nguifo: Frequent closed itemset based algorithms: A thorough structural and analytical survey. ACM-SIGKDD Explorations, 8(1):93–104, June 2006.
  - **Conferences:**
    - T. Hamrouni, P. Valtchev, S. Ben Yahia, E. Mephu Nguifo: About the lossless reduction of the minimal generator family of a context. In: Proceedings of the 5th International Conference on Formal Concept Analysis (ICFCA 2007), LNAI, volume 4390, Springer-Verlag, Clermont Ferrand, France. pages 130-150
    - T. Hamrouni, S. Ben Yahia, E. Mephu Nguifo: Redundancy-free generic bases of association rules. In: Proceedings of the 8th French Conference on Machine Learning (CAp 2006), Presses Universitaires de Grenoble, Trégastel, France. pages 363-378
    - T. Hamrouni, S. Ben Yahia, Y. Slimani: Prince: An algorithm for generating rule bases without closure computations. In: Proceedings of the 7th International Conference on Data Warehousing and Knowledge Discovery (DaWaK 2005), LNCS, volume 3589, Springer-Verlag, Copenhagen, Denmark. pages 346-355 [1]

- **Exploration of the disjunctive search space:**
  - **Journal:**

[1] This publication was carried out during my master thesis.

- T. Hamrouni, S. Ben Yahia, E. Mephu Nguifo: Sweeping the disjunctive search space towards mining new exact concise representations of frequent itemsets. Data & Knowledge Engineering, Elsevier, 68(10):1091–1111, October 2009.

∗ **Conferences:**

- T. Hamrouni, S. Ben Yahia, E. Mephu Nguifo: GARM: Generalized Association Rule Mining. In: Proceedings of the 6th International Conference on Concept Lattices and their Applications (CLA 2008), Olomouc, Czech Republic. pages 145–156
- T. Hamrouni, I. Denden, S. Ben Yahia, E. Mephu Nguifo: A new concise representation of frequent patterns through disjunctive search space. In: Proceedings of the 5th International Conference on Concept Lattices and their Applications (CLA 2007), Montpellier, France. pages 50–61
- T. Hamrouni, I. Denden, S. Ben Yahia, E. Mephu Nguifo, Y. Slimani: The closed essential itemsets: A new concise representation (in French). In: Proceedings of the 7th French Conference on Knowledge Extraction and Management (EGC 2007), Namur, Belgium. pages 241-252

• **Visualization of association rules:**

∗ **Book chapter:**

- S. Ben Yahia, O. Couturier, T. Hamrouni, E. Mephu Nguifo: Meta-knowledge based approach for an interactive visualization of large amounts of association rules. In the Book on Post-Mining of Association Rules: Techniques for Effective Knowledge Extraction, IGI Global Publisher. pages 202–225 (2009)

∗ **Conference:**

- O. Couturier, T. Hamrouni, S. Ben Yahia, E. Mephu Nguifo: A scalable association rule visualization towards displaying large amount of knowledge. In: Proceedings of the 11nd IEEE International Conference on Information Visualization (IV 2007), IEEE Computer Society Press, Zurich, Switzerland. pages 657-663

• **Study of context density:**

∗ **Journal:**

- T. Hamrouni, S. Ben Yahia, E. Mephu Nguifo: Assessing local and global sparseness measure for frequent itemset contexts. To appear in the International Journal of Computing & Information Sciences (IJCIS). Guest Editor: J. Diatta. (2009)