GARM: Generalized Association Rule Mining

T. Hamrouni[1,2], S. Ben Yahia[1] and E. Mephu Nguifo[2]

[1] Department of Computer Science, Faculty of Sciences of Tunis, Tunis, Tunisia.
{tarek.hamrouni, sadok.benyahia}@fst.rnu.tn
[2] CRIL-CNRS, IUT de Lens, Lens, France.
{hamrouni, mephu}@cril.univ-artois.fr

Abstract. A thorough scrutiny of the literature dedicated to association rule mining highlights that a determined effort focused so far on mining the co-occurrence relations between items, *i.e.*, conjunctive patterns. In this respect, disjunctive patterns presenting knowledge about complementary occurring items were neglected in the literature. Nevertheless, recently a growing number of works is shedding light on their importance for the sake of providing a richer knowledge for users. For this purpose, we propose in this paper a new tool, called GARM, aiming at building a partially ordered structure amongst some particular disjunctive patterns, namely the disjunctive closed ones. Starting from this structure, deriving generalized association rules, *i.e.*, those offering conjunctive, disjunctive and negative connectors between items, becomes straightforward. Our experimental study put the focus on the mining performances as well as the quantitative aspect and proved the utility of the proposed approach.

Keywords: Data mining, disjunctive closed pattern, frequent essential pattern, disjunctive support, equivalence class, partially ordered structure, generalized association rules.

1 Introduction and Motivations

Association rule mining is a fundamental topic in Data mining [1]. It has been extensively investigated since its inception. Its key idea consists in looking for causal 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 is the market basket analysis, where the discovered rules can lead to important marketing and management strategic decisions. Recently, mining association rules was extended to various pattern classes like sequential patterns, graphs, etc. Nevertheless, the main moan that can be addressed to the contributions related to association rules is their focus on co-occurrences between items [2], probably as a heritage of the market basket analysis framework. Indeed, almost all related works neglect the other kinds of relations, like mutually exclusive occurrences [3], that can also bring information of worth interest for users.

In this paper, we propose a new tool, called GARM [1], covering the whole process allowing the extraction of generalized association rules. These latter generalize classical rules – positive rules – to offer disjunctive and negative connectors between items,

[1] GARM is the acronym of $\underline{g}$eneralized $\underline{a}$ssociation $\underline{r}$ule $\underline{m}$iner.

© Radim Belohlavek, Sergei O. Kuznetsov (Eds.): CLA 2008, pp. 145–156,
ISBN 978–80–244–2111–7, Palacký University, Olomouc, 2008.

in addition to the conjunctive one [4]. Our tool includes a first component making it possible extracting a concise representation of frequent patterns based on disjunctive patterns. Thanks to a second component, these latter will be partially structured *w.r.t.* set inclusion. Once the partially ordered structure obtained, generalized association rules can be easily derived thanks to the last component of our tool.

Noteworthily, extracting an exact concise representation of frequent patterns in the first component of the process makes it possible to exactly derive the different supports of each frequent pattern. This will make us able to compute the exact values of quality measures. Indeed, it was shown in [5] 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, using disjunctive patterns – in particular closed and essential patterns [6] – will provide 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 [2] – were explored in-depth in the literature through association rules having conjunction of literals, called *literalsets*, in premise and conclusion. This leads to what is commonly known as positive and negative association rules. While disjunctive association rules only have recently begin to grasp the interest of researchers.

In general, generalized association rules are useful in many applications. In particular, disjunctive association rules – having disjunction of items either in premise or in conclusion – were considered for two main purposes: On the one hand, they were used as an intermediate step for defining some concise representations for frequent patterns [1]. On the other hand, they were exploited to provide users with new forms of association rules [7, 8]. For example, the added-value of such association rules has been recently highlighted in [2]. It is however important to note that generalized association rules can be considered as particular GUHA rules [9].

Note that we restrict ourselves in this work to disjunctive closed patterns whose smallest seeds, *i.e.* essential patterns, are frequent with respect to a minimum conjunctive support threshold. This is argued by the fact that we aim at retaining the spirit of association rule mining where this threshold, as well as the confidence-based one, is used to dramatically limit the number of extracted association rules. In addition, the use of a partially ordered structure will make it possible to select representative subsets of rules to be extracted. This nucleus of rules will be of paramount help for avoiding to overwhelm users by highly-sized rule lists.

The remainder of the paper is organized as follows. The next section discusses the related work. Section 3 recalls the key notions used throughout this paper. The structural properties of the disjunctive search space are explored in Section 4, followed by a detailed description of the GARM tool having for purpose to offer a complete process for the extraction of generalized association rules in Section 5. Experimental results focusing on the mining time as well as the quantitative aspect are reported and discussed in Section 6. Section 7 concludes the paper and points out future works.

[2] A literal is an item or the negation of an item.

2 Related Work

Contributions related to association rule mining mainly concentrated on the classical 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 [10]. Recently, some works focused on introducing negative items. Nevertheless, the majority of items are not present in each transaction leading to explosive amounts of association rules with negation. Thus, existing approaches have tried to address this problem through the use of additional background information about the data, incorporating attribute correlations, and additional rule interestingness measures, etc. Here we will mainly detail the reduced number of related works on association rules relying on the disjunctive connector within items.

Some works [7, 8] were interested in using the disjunction connector within the association rule mining issue to define what is called *generalized association rules*. These rules grasped the interest of many researchers since they offer wealthier types of knowledge in many applications. In addition to the inclusive disjunction operator, *i.e.*, the operator $\vee$, Nanavati *et al.* in [8] were also interested in the exclusive disjunction operator, denoted $\oplus$. The authors hence proposed two kinds of rules which are 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 [7], 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, what is expressed in [8] using the operator $\oplus$. Other forms of generalized association rules were also described in [11]. In [12], 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 patterns. Elble *et al.* used disjunctive rules to handle numerical attributes by considering disjunctions between intervals [13]. This latter work extends other ones taking also into account categorical attributes (see [13] for references). Finally, it is worth noting that the disjunction connector has also been used to define some concise representations of frequent patterns through the so-called *disjunctive rule* (see for example [1] for references).

3 Basic Concepts

In this section, we briefly sketch the key notions that will be of use throughout the paper.

Definition 1. *An extraction context is a triplet $\mathcal{K} = (\mathcal{O}, \mathcal{I}, \mathcal{R})$ where $\mathcal{O}$ and $\mathcal{I}$ are, respectively, a finite set of objects (or transactions) and items (or attributes), and $\mathcal{R} \subseteq \mathcal{O} \times \mathcal{I}$ is a binary relation between the objects and items. A couple $(o, i) \in \mathcal{R}$ denotes that the object $o \in \mathcal{O}$ contains the item $i \in \mathcal{I}$.*

Example 1. *We will consider in the remainder a context that consists of transactions* (1, *AB*)*,* (2, *ACD*)*,* (3, *CDE*)*,* (4, *DEF*)*,* (5, *ABCDE*)*, and* (6, *ABC*) [3].

Definition 2. (SUPPORTS OF A PATTERN) *Let* $\mathcal{K} = (\mathcal{O}, \mathcal{I}, \mathcal{R})$ *be a context and* I *be a pattern. We mainly distinguish three kinds of supports related to* I*:*

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

Roughly speaking, the semantics of the aforementioned supports is as follows:
- *Supp*$(\wedge\ I)$ is the number of objects containing all items of I.
- *Supp*$(\vee\ I)$ is the number of objects containing at least one item of I.
- *Supp*$(\overline{I})$ is the number of objects that do not contain any item of I.

Note also that *Supp*$(\vee\ I)$ and *Supp*$(\overline{I})$ are two complementary quantities *w.r.t.* $|\mathcal{O}|$ in the sense that: *Supp*$(\vee\ I)$ + *Supp*$(\overline{I})$ = $|\mathcal{O}|$.

Example 2. *Consider our running context. We have Supp*$(\wedge$ *CDE*$)$ = $\mid \{3,5\} \mid$ = 2*, Supp*$(\vee$ *CDE*$)$ = $\mid \{2,3,4,5,6\} \mid$ = 5 *and Supp*$(\overline{CDE})$ = $\mid \{1\} \mid$ = 1.

Hereafter, *Supp*$(\wedge\ I)$ will simply be denoted *Supp*(I). In addition, if there is no risk of confusion, the *conjunctive support* will simply be called *support*. A pattern I is said to be *frequent* if *Supp*(I) is greater than or equal to a minimum support threshold, denoted *minsupp*. Since the set of frequent patterns is an order ideal, the set of items $\mathcal{I}$ will be considered as only containing frequent items. Lemma 1 states that conjunctive supports can be derived starting from disjunctive ones.

Lemma 1. *[14] Let* $I \subseteq \mathcal{I}$*. The following equalities hold:*

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

4 Structural Properties of the Disjunctive Search Space

In this section, we will characterize disjunctive patterns through the associated equivalence classes induced by the following closure operator:

Definition 3. *Let* $\mathcal{K}$ = ($\mathcal{O}$, $\mathcal{I}$, $\mathcal{R}$) *be an extraction context. The disjunctive closure operator* h *is defined as follows [6]:*

$h : \mathcal{P}(\mathcal{I}) \rightarrow \mathcal{P}(\mathcal{I})$

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

The disjunctive closure $h(I)$ of a pattern I is equal to the maximal set of items which *only* appear in the transactions that contain at least an item of I. The closure operator h induces an equivalence relation on the power-set of $\mathcal{I}$, which partitions it into so-called *disjunctive equivalence classes*. In each class, all the elements have the same disjunctive support. The smallest incomparable elements, *w.r.t.* set inclusion, of a disjunctive equivalence class are called essential patterns, while the disjunctive closed pattern is the largest one [6]. These particular patterns are defined as follows.

[3] We use a separator-free form for the sets, *e.g.*, *ABC* stands for the set of items {A, B, C}.

Definition 4.
• *A pattern $I \subseteq \mathcal{I}$ is a* ***disjunctive closed pattern*** *if $I = h(I)$ or, equivalently, Supp$(\vee I)$ $< min\{$Supp$(\vee I') \mid I' \subseteq \mathcal{I}$ s.t. $I \subset I'\}$.*
• *A pattern $I \subseteq \mathcal{I}$ is an* ***essential pattern*** *if $\forall\ I' \subset I$, $I \nsubseteq h(I')$ or, equivalently, Supp$(\vee$ $I) > max\{$Supp$(\vee I') \mid I' \subseteq \mathcal{I}$ s.t. $I' \subset I\}$.*

Example 3. *Consider our running context. The pattern CDEF is disjunctively closed, while BE is not, since Supp$(\vee$ BE$)$ = Supp$(\vee$ BEF$)$. On the other hand, the pattern AC is essential, while DE is not, since Supp$(\vee$ DE$)$ = Supp$(\vee$ D$)$.*

In the remainder, $\mathcal{FEP}_{\mathcal{K}}$ [4] denotes the set of frequent essential patterns associated to a given context $\mathcal{K}$ and a fixed *minsupp* value. The associated set of disjunctive closure will further be denoted $\mathcal{EDCP}_{\mathcal{K}}$ [5]. This latter set is hence equal to $\{h(I) \mid I \in \mathcal{FEP}_{\mathcal{K}}\}$.

To establish the link with conjunctive equivalence class – gathering patterns having the same Galois closure [15] – we notice that essential patterns (*resp.* disjunctive closed patterns) are equivalent to minimal generators *aka* free-sets (*resp.* closed patterns) (see [1] for references). These latter patterns were at the basis of the main concise representations of association rules that were proposed in the literature [10]. This clearly motivates the use of their correspondences within the disjunctive search space.

5 Detailed Description of the GARM Tool

As mentioned in the first section, the GARM tool is composed of three complementary components which are as follows: (i) Extracting an exact concise representation of frequent patterns based on disjunctive closed patterns and frequent essential ones. (ii) Building a partially ordered structure *w.r.t.* set inclusion within disjunctive closed patterns. Each one of these latter will be accompanied by its set of frequent essential patterns. (iii) Deriving generalized association rules from the built structure.

5.1 Extracting a New Concise Representation based on Disjunctive Patterns

Our representation is based on the sets $\mathcal{FEP}_{\mathcal{K}}$ and $\mathcal{EDCP}_{\mathcal{K}}$, as stated by Theorem 1.

Theorem 1. *The set $\mathcal{EDCP}_{\mathcal{K}} \cup \mathcal{FEP}_{\mathcal{K}}$ is an exact concise representation of the set of frequent patterns $\mathcal{FP}_{\mathcal{K}}$ [16].*

Example 4. *Figure 1 (Left) lists the set of disjunctive closed patterns associated to the running context. For each closed pattern, its associated disjunctive support and frequent essential patterns, for minsupp = $\mathbf{1}$, are also given.*

This representation will be denoted $\mathcal{DSSR}_{\mathcal{K}}$ [6]. It is extracted thanks to an adaptation of our DCPR_MINER [7] algorithm [17], what constitutes the first component of the

[4] Stands for frequent essential patterns.

[5] Stands for essential disjunctive closed patterns.

[6] Stands for disjunctive search space-based representation.

[7] DCPR_MINER is the acronym of disjunctive closed pattern-based representation miner.

$\mathcal{EDCP}_{\mathcal{K}}$	**Disj. Supp.**	$\mathcal{FEP}_{\mathcal{K}}$
B	3	B
C	4	C
F	1	F
AB	4	A
EF	3	E
ABC	5	AC, BC
BEF	5	BE
DEF	4	D
CDEF	5	CD, CE
ABCDEF	6	AD, AE, BD, BCE

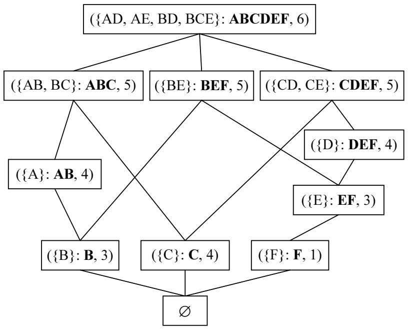

Fig. 1. (**Left**) The set $\mathcal{EDCP}_{\mathcal{K}}$ and the associated disjunctive support and frequent essential patterns for *minsupp* = 1. (**Right**) The equivalence classes partially ordered *w.r.t.* set inclusion.

GARM tool. Starting from $\mathcal{DSSR}_{\mathcal{K}}$, the conjunctive and negative supports of frequent patterns can thus be deduced using disjunctive supports. This representation also allows the derivation of the support of each literalset whose positive variation is based on a frequent pattern. This is carried out using the following formula [4]: *Supp*$(x_1 \wedge x_2 \wedge \ldots \wedge x_n \wedge \overline{y_1} \wedge \overline{y_2} \wedge \ldots \wedge \overline{y_m})$ = $\sum\limits_{S \subseteq \{y_1, \ldots, y_m\}} (-1)^{|S|}$*Supp*$(x_1 \wedge x_2 \wedge \ldots \wedge x_n \wedge S)$, such that its positive variation, namely $\{x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m\}$, belongs to $\mathcal{FP}_{\mathcal{K}}$.

5.2 Building the Partially Ordered Structure

In this section, we will propose a new algorithm, called POSB [8], for partially sorting disjunctive closed patterns *w.r.t.* set inclusion. The POSB algorithm hence takes as input the representation $\mathcal{DSSR}_{\mathcal{K}}$ *s.t.* to each disjunctive closed pattern is associated its set of frequent essential patterns and disjunctive support. A node in the partially ordered structure will be associated to each disjunctive closed pattern. The pseudo-code of POSB is shown by Algorithm 1. Our algorithm inherits two main optimizations used in the algorithm proposed by Valtchev *et al.* [18], namely the sorting of disjunctive closed patterns, and the use of a border. Indeed, the set of disjunctive closed patterns $\mathcal{EDCP}_{\mathcal{K}}$ is sorted *w.r.t.* the increasing pattern size. Since closures of equal size cannot be comparable, this sorting avoids unnecessary comparisons. In addition, it makes possible that the closure f under treatment be of the largest size *w.r.t.* already treated ones. Thus, it suffices to find its lower cover among the nodes inserted in the structure. This lower cover is composed by those closures which are *immediately covered* by f.

On the other hand, the border $\mathcal{B}$ is an anti-chain *w.r.t.* set inclusion containing maximal closures among those already treated. In fact, the Valtchev *et al.* algorithm constructs the Hasse diagram representing the subset-superset relationship among concepts in the Galois lattice. It begins at the top of the lattice and then recursively identifies the lower neighbors of each concept. Nevertheless, it is not directly adapted to our situation. Indeed, although the intersection of two disjunctive closed patterns is obviously

[8] POSB is the acronym of partially ordered structure builder.

Algorithm 1: POSB

```
Input: The set $\mathcal{EDCP}_{\mathcal{K}}$ of disjunctive closed patterns.
Output: The disjunctive closed patterns ordered by set inclusion.
Begin
    $\mathcal{B} := \emptyset$ ;
    Foreach ($f \in \mathcal{EDCP}_{\mathcal{K}}$) do
        $Prohibited\_List = \emptyset$;
        Foreach ($b \in \mathcal{B}$) do
            $inter := b \cap f$;
            If ($inter = b$) then
                LOWER_COVER_INSERTION($f$, $b$);
                $\mathcal{B} := \mathcal{B} \setminus b$;
            Else If ($inter \neq \emptyset$) then
                LOWER_COVER_MANAGEMENT($f$, $b$);
        $\mathcal{B} := \mathcal{B} \cup f$ ;
End
```

a disjunctive closed pattern, this latter does not necessarily belong to $\mathcal{EDCP}_{\mathcal{K}}$. This is due to the fact that it could have all its essential patterns infrequent and, hence, has been already pruned. On its side, the proposed algorithm in [18] relies on the fact that the intersection of two concepts was already treated and it suffices to locate the corresponding node within the Hasse diagram.

In Algorithm 1, disjunctive closed patterns are inserted one at a time to a structure which is only partially finished to obtain at the end the entire one. Let f be the current disjunctive closed pattern to be inserted in the partially ordered structure. f will be compared to the elements of the border $\mathcal{B}$. If an element $b \in \mathcal{B}$ is included in f, 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 2). The element b will then be deleted from the border. If b is not included in f but their intersection is not empty, then the LOWER_COVER_MANAGEMENT procedure will identify the common immediate predecessors of b and f (*cf.* Algorithm 3). 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 pattern to be inserted in the partially ordered structure. Indeed, when updating the precedence link between disjunctive closed patterns, 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.

Example 5. *The associated structure to our running context is given by Figure 1* (*Right*).

5.3 Deriving Generalized Association Rules

Once the partially ordered structure built, deriving (subsets) generalized association rules can be easily done. An association rule R: $X \Rightarrow Y$ based on a pattern 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 patterns, $X \cap Y = \emptyset$, and $X \cup Y = Z$. An association rule is usually considered as interesting *w.r.t.* two statistical measures, namely the support and the confidence. The formulae of these measures for an arbitrary rule are as follows:

Algorithm 2: LOWER_COVER_INSERTION

```
Input: A disjunctive closure $f$, and an element $pred$ to be inserted in its lower cover.
Output: The updated lower cover of $f$.
Begin
    Foreach ($l \in Lower\_Cover(f)$) do
        $inter$ := $l \cap pred$;
        If ($inter$ = $pred$) then
            return;
        Else If ($inter$ = $l$) then
            $Lower\_Cover(f)$ := $Lower\_Cover(f) \setminus l$;
    $Lower\_Cover(f)$ := $Lower\_Cover(f) \cup pred$;
End
```

Algorithm 3: LOWER_COVER_MANAGEMENT

```
Input: A disjunctive closed pattern $f$, and an element $b$ of the border $\mathcal{B}$.
Output: The updated lower cover of $f$.
Begin
    Foreach ($pred\_b \in Lower\_Cover(b)$) do
        If ($pred\_b \notin Prohibited\_List$) then
            $inter$ := $pred\_b \cap f$;
            If ($inter$ = $pred\_b$) then
                LOWER_COVER_INSERTION($f$, $pred\_b$);
            Else If ($inter \neq \emptyset$) then
                LOWER_COVER_MANAGEMENT($f$, $pred\_b$);
            $Prohibited\_List$ := $Prohibited\_List \cup pred\_b$;
End
```

$$\textit{Supp}(X \Rightarrow Y) = \textit{Supp}(X \wedge Y), \text{ and, } \textit{Conf}(X \Rightarrow Y) = \frac{\textit{Supp}(X \wedge Y)}{\textit{Supp}(X)}$$

A rule is said to be *exact* if its confidence is equal to $\mathbf{1}$. 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 are able to evaluate *Supp*$(X \Rightarrow Y)$, the derivation of the confidence value will be straightforward.

Let us now adapt the association rule framework to our context. As shown in Subsection 5.1, the $\mathcal{DSSR}_{\mathcal{K}}$ representation allows deriving the disjunctive, conjunctive and negative supports of each set of positive and negative items whose positive variation is based on a frequent pattern. In the sequel, we present an overview of the process by which we retrieve generalized association rules and evaluate their associated supports through traversing the partially ordered structure. Rules can be classified according to the number of nodes required for their extraction. We then distinguish two cases:

1. **An intra-node rule**: it requires a unique node and highlight relationships between a frequent essential pattern and its disjunctive closure f (here Z = f).
2. **An inter-nodes rule**: it is extracted using two nodes N_1 and N_2 *s.t.* the associated disjunctive closure 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 pattern of f_1. An inter-nodes rule describes relationships between either f_1 and f_2 or e_1 and f_2 (here Z = f_2).

Both kinds of rules – intra-node and inter-nodes – can be either exact or approximate.

Different forms of generalized association rules can be extracted starting from our representation (*cf.* [16] for a detailed description). To limit the number of possible extracted rule forms, we mainly focus here on the following ones:

1. **Form 1**: *disjunction of items in premise and conclusion* $\vee X \Rightarrow \vee Y$: $\mathit{Supp}(\vee X \Rightarrow \vee Y) = \mathit{Supp}(\vee X \wedge \vee Y) = \mathit{Supp}(\vee X) + \mathit{Supp}(\vee Y)$ - $\mathit{Supp}((\vee X) \vee (\vee Y)) = \mathit{Supp}(\vee X) + \mathit{Supp}(\vee Y)$ - $\mathit{Supp}(\vee Z)$,
2. **Form 2**: *negation of items in premise and conclusion* $\overline{X} \Rightarrow \overline{Y}$: $\mathit{Supp}(\overline{X} \Rightarrow \overline{Y}) = \mathit{Supp}(\overline{X} \wedge \overline{Y}) = \mathit{Supp}(\overline{((\vee X) \vee (\vee Y))}) = \mathit{Supp}(\overline{Z}) = |\mathcal{O}|$ - $\mathit{Supp}(\vee Z)$,
3. **Form 3**: *disjunction of items in premise and negation of items in conclusion* $\vee X \Rightarrow \overline{Y}$: $\mathit{Supp}(\vee X \Rightarrow \overline{Y}) = \mathit{Supp}(\vee X \wedge \overline{Y}) = \mathit{Supp}((\vee X) \vee (\vee Y))$ - $\mathit{Supp}(\vee Y) = \mathit{Supp}(\vee Z)$ - $\mathit{Supp}(\vee Y)$, and,
4. **Form 4**: *negation of items in premise and disjunction of items in conclusion* $\overline{X} \Rightarrow \vee Y$: $\mathit{Supp}(\overline{X} \Rightarrow \vee Y) = \mathit{Supp}(\overline{X} \wedge \vee Y) = \mathit{Supp}((\vee X) \vee (\vee Y))$ - $\mathit{Supp}(\vee X) = \mathit{Supp}(\vee Z)$ - $\mathit{Supp}(\vee X)$,

where either X or Y is a frequent essential pattern or a disjunctive closed one, and $Z = X \cup Y$ is a disjunctive closed pattern (as described above). For each rule, the support of Z is known. It is the same for either X or Y since one of them is assumed to be a frequent essential pattern or a disjunctive closed pattern. For the sake of simplicity, we assume in the remainder that X is a frequent essential pattern or a disjunctive closed pattern. Since $Y = Z \backslash X$, then Y does not necessarily belong to $\mathcal{DSSR}_{\mathcal{K}}$ and, may even not be a frequent pattern. Nevertheless, its disjunctive support is required to evaluate that of the associated rule. To this end, we bound the support of Y using a lower bound, denoted *lb_Supp*, and an upper bound, denoted *ub_Supp*, computed as follows:

- ***lb_Supp***$(\vee Y) = max\{\mathit{Supp}(\vee e) \mid e \in \mathcal{FEP}_{\mathcal{K}}$ and $e \subseteq Y\}$,
- ***ub_Supp***$(\vee Y) = min\{\mathit{Supp}(\vee f) \mid f \in \mathcal{EDCP}_{\mathcal{K}}$ and $Y \subseteq f\}$.

In this respect, if Y is encompassed between a frequent essential pattern and its disjunctive closure, then *lb_Supp*$(\vee Y)$ = *ub_Supp*$(\vee Y)$. Hence, the support and confidence of the associated rule will be exactly computed. Otherwise, these latter measures will be bounded by a minimal and a maximal possible value using the bounds associated to Y. Such rules, further denoted *approximated* rules, are defined as follows:

Definition 5. *An association rule is said to be approximated if it has either its support or its confidence not exactly determined.*

Then, only valid 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 not equal to $\mathbf{1}$), what is not the case of the former. In this respect, approximated rules were shown to convey interesting knowledge in the case of positive rules (see for example [19]).

Noteworthily, the bounds *lb_Supp*$(\vee Y)$ and *ub_Supp*$(\vee Y)$ always exist. Indeed, on the one hand, 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 obviously belong to $\mathcal{FEP}_{\mathcal{K}}$, what ensures the existence of the lower bound. On the other hand, Y is covered by at least a disjunctive closed pattern, namely Z, what ensures the existence of the upper bound.

Example 6. *Let minsupp* = $\mathbf{1}$ *and let minconf* = $\mathbf{0.7}$. *Consider the intra-node rule* R_1 *of* ***Form 1*** *based on the disjunctive closed pattern ABCDEF and its frequent essential pattern BCE:* $\vee$ *BCE* $\Rightarrow$ $\vee$ *ADF. Supp*(R_1) = *Supp*($\vee$ *BCE*) + *Supp*($\vee$ *ADF*) - *Supp*($\vee$ *ABCDEF*) = *Supp*($\vee$ *ADF*) (*since h*(*BCE*) = *ABCDEF*). *Since ADF* $\notin \mathcal{DSSR}_{\mathcal{K}}$, *we need to evaluate its support. Since AD* $\subseteq$ *ADF* $\subseteq$ *h*(*AD*) = *ABCDEF* (*cf. Figure 1* (*Left*)), *then lb_Supp*($\vee$ *ADF*) = *ub_Supp*($\vee$ *ADF*) = $\mathbf{6}$. *Hence, Supp*(R_1) = $\mathbf{6}$ *and Conf*(R_1) = $\mathbf{1}$. R_1 *is hence a valid rule. Now, consider the inter-nodes rule* R_2 *of* ***Form 1*** *based on ABCDEF and one of its immediate predecessors, namely ABC* (*cf. Figure 1* (*Right*))*:* $\vee$ *ABC* $\Rightarrow$ $\vee$ *DEF. In this case, DEF* $\in \mathcal{EDCP}_{\mathcal{K}}$. *Hence, Supp*$(\mathrm{R}_2)$ = *Supp*($\vee$ *ABC*) + *Supp*($\vee$ *DEF*) - *Supp*($\vee$ *ABCDEF*) = $\mathbf{5}$ + $\mathbf{4}$ - $\mathbf{6}$ = $\mathbf{3}$, *and Conf*(R_2) = $\mathbf{0.6}$. *Here, we took* X = *ABC. If we set* Y = *ABC, then the associated rule* R_3 = $\vee$ *DEF* $\Rightarrow$ $\vee$ *ABC will have the same support than* R_2. *Nevertheless, its confidence is equal to* $\mathbf{0.75}$. *Hence,* R_3 *is a valid rule while* R_2 *is not.*

6 Experimental Results

Our experiments [9] focused on the mining time as well as the number of extracted valid rules *w.r.t.* their associated type, *i.e.*, exact, approximate or approximated. They were carried out on a PC equipped with a Pentium (R) having 3GHz as clock frequency and 1.75GB of main memory, running the GNU/Linux distribution Fedora Core 7 (with 2GB of swap memory). The compiler gcc 4.1.2 is used to generate the executable code starting from our C++ implementation.

Table 1. Mining time of generalized association rules on benchmark contexts.

Context	*minsupp* (%)	Component 1	Component 2	Component 3	Total time
CONNECT	*80.00*	2.1530	0.0068	0.0380	**2.1978**
	60.00	2.2807	0.0402	0.1618	**2.4827**
	40.00	2.5571	1.0443	0.9813	**4.5827**
PUMSB	*90.00*	3.1875	0.0403	0.1015	**3.3293**
	80.00	3.1581	2.9364	1.9693	**8.0638**
	70.00	3.6630	19.5460	8.7276	**31.9366**
KOSARAK	*0.90*	12.4551	0.1645	0.2239	**12.8435**
	0.70	16.2936	0.6825	0.3794	**17.3555**
	0.50	26.4491	5.6164	0.8738	**32.9393**
RETAIL	*2.00*	0.8471	0.0039	0.0135	**0.8645**
	1.00	1.0803	0.0113	0.0334	**1.1250**
	0.50	2.3909	0.1127	0.1331	**2.6367**

In the proposed experiments, the *minconf* value is set to the relative minimum support value, *i.e.*, $\frac{\mathit{minsupp}}{|\mathcal{O}|}$. Table 1 presents the mining time in seconds of the three components of GARM. This table shows the efficiency of our tool towards extracting generalized associated rules. Indeed, even for low *minsupp* values, GARM remains very fast. In this respect, the time consumed by each component, *w.r.t.* the total time,

[9] Test contexts are available at: *http://fimi.cs.helsinki.fi/data.*

Table 2. Number of extracted generalized association rules on benchmark contexts.

Context	*minsupp* (%)	Exact	Approximate	Approximated	Total number
Connect	*80.00*	620	316	152	**1, 088**
	60.00	1, 533	1, 337	354	**3, 224**
	40.00	3, 319	5, 813	3, 130	**12, 262**
Pumsb	*90.00*	566	1, 322	730	**2, 618**
	80.00	4, 376	13, 426	5, 002	**22, 804**
	70.00	9, 409	26, 747	14, 870	**51, 026**
Kosarak	*0.90*	0	7, 586	0	**7, 586**
	0.70	0	13, 046	0	**13, 046**
	0.50	0	29, 648	0	**29, 648**
Retail	*2.00*	0	464	0	**464**
	1.00	0	1, 160	0	**1, 160**
	0.50	0	4, 622	0	**4, 622**

closely depends on the context characteristics. Nevertheless, the second and third components are in general faster than the first one. On the other hand, Table 2 highlights that the number of extracted rules closely depends on the context density. Indeed, the higher the value of this latter, the larger the associated equivalence classes are, and the greater the number of frequent essential patterns and closed ones is. This fact augments the number of rules even for high *minsupp* values for dense contexts. Interestingly enough, the number of exact and approximated rules for Retail and Kosarak is equal to $\mathbf{0}$ for the tested *minsupp* values. This is due to the fact that for both contexts, each essential pattern is equal to its disjunctive closure what is not the case for the Connect and Pumsb contexts. Please note that the mining time and the number of extracted rules when *minconf* varies is omitted here, due to space limitations.

7 Conclusion and Perspectives

In this paper, we presented a complete tool, called GARM, allowing the extraction of generalized association rules. Our tool is composed of three components. The first consists in extracting a concise representation of frequent patterns based on disjunctive closed ones. The second component aimed at partially ordering these closure *w.r.t.* set inclusion. Once the structure built, extracting subsets of generalized association rules becomes a straightforward task thanks to the last component. Carried out experiments proved the effectiveness of the proposed tool. It is also important to mention that our GARM tool is easily adaptable to the case where the input is composed by conjunctive (closed) patterns instead of disjunctive ones.

Other avenues for future work mainly address the following points: First, a detailed comparison of our approach to the general GUHA approach [9] will be carried out. Second, the relationships between the various rule forms will be studied. The purpose is to only retain a lossless subset of rules while being able to derive the remaining redundant ones. Adequate axiomatic systems need thus to be set up.

Acknowledgments: We would like to thank anonymous reviewers for their helpful comments and suggestions. We are also grateful to Mrs. Nassima Ben Younes for fruitful discussions and help in the implementation of the tool. This work is supported by the French-Tunisian project CMCU-Utique 05G1412.

References

1. Ceglar, A., Roddick, J.F.: Association mining. ACM Computing Surveys, **volume 38(2)** (2006)
2. Steinbach, M., Kumar, V.: Generalizing the notion of confidence. Knowledge and Information Systems, **volume 12(3)** (2007) 279–299
3. Tzanis, G., Berberidis, C.: Mining for mutually exclusive items in transaction databases. International Journal of Data Warehousing and Mining, **volume 3(3)** (2007) 45–59
4. Toivonen, H.: Discovering of frequent patterns in large data collections. PhD thesis, University of Helsinki, Helsinki, Finland (1996)
5. Hébert, C., Crémilleux, B.: A unified view of objective interestingness measures. In: Proceedings of the 5th International Conference Machine Learning and Data Mining in Pattern Recognition, Springer-Verlag, LNCS, volume 4571. (2007) 533–547
6. Hamrouni, T., Denden, I., Ben Yahia, S., Mephu Nguifo, E.: A new concise representation of frequent patterns through disjunctive search space. In: Proceedings of the 5th International Conference on Concept Lattices and their Applications. (2007) 50–61
7. Kim, H.D.: Complementary occurrence and disjunctive rules for market basket analysis in data mining. In: Proceedings of the 2nd IASTED International Conference Information and Knowledge Sharing. (2003) 155–157
8. Nanavati, A.A., Chitrapura, K.P., Joshi, S., Krishnapuram, R.: Mining generalised disjunctive association rules. In: Proceedings of the 10th International Conference on Information and Knowledge Management. (2001) 482–489
9. Hájek, P., Havránek, T.: Mechanizing Hypothesis Formation: Mathematical Foundations for a General Theory. Springer-Verlag (1978)
10. Kryszkiewicz, M.: Concise representations of association rules. In: Proceedings of the ESF Exploratory Workshop on Pattern Detection and Discovery in Data Mining, Springer-Verlag, LNCS, volume 2447. (2002) 92–109
11. Grün, G.A.: New forms of association rules. Technical Report TR 1998-15, School of Computing Science, Simon Fraser University, Burnaby, BC, Canada (1998)
12. Shima, Y., Hirata, K., Harao, M., Yokoyama, S., Matsuoka, K., Izumi, T.: Extracting disjunctive closed rules from MRSA data. In: Proceedings of the 1st International Conference on Complex Medical Engineering. (2005) 321–325
13. Elble, J., Heeren, C., Pitt, L.: Optimized disjunctive association rules via sampling. In: Proceedings of the 3rd IEEE International Conference on Data Mining. (2003) 43–50
14. Galambos, J., Simonelli, I.: Bonferroni-type inequalities with applications. Springer (2000)
15. Ganter, B., Wille, R.: Formal Concept Analysis. Springer (1999)
16. Hamrouni, T., Denden, I., Ben Yahia, S., Mephu Nguifo, E.: Exploring the disjunctive search space towards discovering new exact concise representations for frequent patterns. Technical report, CRIL-CNRS of Lens, Lens, France (2007)
17. Denden, I., Hamrouni, T., Ben Yahia, S.: Efficient exploration of the disjunctive lattice towards extracting concise representations of frequent patterns. To appear in the Proceedings of the 9th African Conference on Research in Computer Science and Applied Mathematics (in French). (2008)
18. Valtchev, P., Missaoui, R., Lebrun, P.: A fast algorithm for building the Hasse diagram of a Galois lattice. In: Proceedings of the Conference on Combinatorics, Computer Science and Applications. (2000) 293–306
19. Boulicaut, J.F., Bykowski, A., Rigotti, C.: Free-sets: A condensed representation of Boolean data for the approximation of frequency queries. Data Mining and Knowledge Discovery **volume 7(1)** (2003) 5–22