Ann Math Artif Intell (2010) 59:201–222
DOI 10.1007/s10472-010-9192-z

Generalization of association rules through disjunction

Tarek Hamrouni · Sadok Ben Yahia · Engelbert Mephu Nguifo

Published online: 18 June 2010

Abstract Several efforts were devoted to mining association rules having conjunction of items in premise and conclusion parts. Such rules convey information about the co-occurrence relations between items. However, other links amongst items—like complementary occurrence of items, absence of items, etc.—may occur and offer interesting knowledge to end-users. In this respect, looking for such relationship is a real challenge since not based on the conjunctive patterns. Indeed, catching such links requires obtaining semantically richer association rules, the generalized ones. These latter rules generalize classic ones to also offer disjunction and negation connectors between items, in addition to the conjunctive one. For this purpose, we propose in this paper a complete process for mining generalized association rules starting from an extraction context. Our experimental study stressing on the mining performances as well as the quantitative aspect proves the soundness of our proposal.

Keywords Data mining · Disjunctive closed pattern · Disjunctive support · Equivalence class · Frequent essential pattern · Generalized association rules · Partially ordered structure · Lattices

Mathematics Subject Classifications (2010) 68 · 68Txx · 68T99 · 68Pxx · 68P30 · 62-07 · 97R50

T. Hamrouni (✉) · S. Ben Yahia
Computer Science Department, Faculty of Sciences of Tunis,
University Campus, 1060 Tunis, Tunisia
e-mail: tarek.hamrouni@fst.rnu.tn, hamrouni.tarek@gmail.com

S. Ben Yahia
e-mail: sadok.benyahia@fst.rnu.tn

T. Hamrouni
CRIL-CNRS, Lille Nord University, Artois, France
e-mail: hamrouni@cril.univ-artois.fr

E. Mephu Nguifo
LIMOS-CNRS, Blaise Pascal University, Campus des cézeaux, 63173 Aubière cedex, France
e-mail: engelbert.mephu_nguifo@univ-bpclermont.fr

1 Introduction and motivations

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 [1]. Indeed, almost all related work neglect the other kinds of relations, like mutually exclusive or complementary occurrences [2], which can also bring information of worth interest for the end-users. Such kind of knowledge can naturally be conveyed through disjunctive patterns, which have been shown to be closely related to different important pattern classes (cf. [3, 4] for a detailed description). 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 [1, 5]. For example, these rules were shown to be useful for software change impact analysis [6], feature model mining [7], and medical data analysis [8]. On the other hand, such kind of rules offers advantages compared to the hierarchy/taxonomy-based generalization [9]. 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. Note that these rules generalized through the use of the conjunction, disjunction, and negation connectors within items can be related to the rules defined in the general GUHA approach [10, 11].

In this paper, 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 [12]. 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 [5]. Moreover, the use of the disjunctive operator in association rules allows, for example, to obtain rules linking frequently occurring patterns and rare ones [13]. 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 patterns based on disjunctive patterns. Such a representation allows the derivation of the exact conjunctive supports of frequent patterns while preserving the easy access to their respective disjunctive and negative supports. This will allow us to compute the values of quality measures. Indeed, it was shown in [14, 15] 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 patterns—in particular disjunctive closed pattern [3] and essential patterns [16]—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* [17]. While disjunctive association rules only have recently begun to grasp the interest of researchers.

It is important to mention that we restrict ourselves in this work to disjunctive closed patterns whose minimal seeds, i.e., essential patterns, are frequent with respect

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

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 [18, 19]. In addition, the use of a partially ordered structure will make it possible 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.

The remainder of the paper is organized as follows. The next section recalls the key notions used throughout this paper. The structural properties of the disjunctive search space are explored in Section 3. After that, Section 4 extends the framework of classic association rules through taking into account the various possible connectors as well as negative items. Section 5 proposes algorithms covering the different steps of the mining process. Experimental results focusing on the mining time as well as the quantitative aspect are reported and analyzed in Section 6. Section 7 discusses the related work, while Section 8 concludes the paper and describes our main future work.

2 Basic concepts

In this section, we briefly present 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 the six transactions: (1, AB), (2, ACD), (3, CDE), (4, DEF), (5, ABCDE), and (6, ABC).[2]

In this work, we mainly concentrate on itemsets as a class of patterns. The following definition presents the supports that characterize a pattern.

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:

$$Supp(\wedge\ I) = |\ \{o \in \mathcal{O} \mid (\forall\ i \in I, (o, i) \in \mathcal{R})\}\ |$$

$$Supp(\vee\ I) = |\ \{o \in \mathcal{O} \mid (\exists\ i \in I, (o, i) \in \mathcal{R})\}\ |$$

$$Supp(\overline{I}) = |\ \{o \in \mathcal{O} \mid (\forall\ i \in I, (o, i) \notin \mathcal{R})\}\ |$$

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

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

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

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 in $(\mathbf{2}^{\mathcal{I}}, \subseteq)$[3] [20], the set of items $\mathcal{I}$ will be considered as only containing frequent items. Having the disjunctive supports of patterns subsets, we can derive their conjunctive supports using an inclusion–exclusion identity [21]. While their negative supports can be derived thanks to the De Morgan's law. This is stated by Lemma 1.

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

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

$$Supp(\overline{I}) = \mid \mathcal{O} \mid - Supp(\vee I) \tag{2}$$

3 Structural properties of the disjunctive search space

In this section, we will characterize disjunctive patterns through the associated equivalence classes induced by the disjunctive closure operator [3].

Definition 3 Let $\mathcal{K} = (\mathcal{O}, \mathcal{I}, \mathcal{R})$ be an extraction context. The disjunctive closure operator h is defined as follows:
$h : \mathcal{P}(\mathcal{I}) \rightarrow \mathcal{P}(\mathcal{I})$
$I \mapsto h(I) = \{i \in \mathcal{I} \mid (\forall\, o \in \mathcal{O})\ ((o, i) \in \mathcal{R}) \Rightarrow (\exists\, i_1 \in I)((o, i_1) \in \mathcal{R})\}$.

Roughly speaking, 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. Given an arbitrary pattern, its disjunctive closure is then equal to the maximal pattern, w.r.t. set inclusion, containing it and having the same disjunctive support. The following definition introduces a second characterization of the disjunctive closure [3].

Definition 4 The disjunctive closure of a pattern I is: $h(I) = \max_{\subseteq}\{I_1 \subseteq \mathcal{I} \mid (I \subseteq I_1) \wedge (Supp(\vee I) = Supp(\vee I_1))\} = I \cup \{i \in \mathcal{I} \backslash I \mid Supp(\vee\, I) = Supp(\vee\, (I \cup \{i\}))\}$.

Example 3 Considering our running dataset, we have h(D) = DEF. Indeed, DEF is the maximal pattern containing D and having a disjunctive support equal to that of D.

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* [16],

[3] Let a subset S of $\mathbf{2}^{\mathcal{I}}$ be an order ideal in $(\mathbf{2}^{\mathcal{I}}, \subseteq)$. Given $X \subseteq \mathcal{I}$, if $X \in S$, $\forall\, Y \subseteq X$, $Y \in S$. In addition, if $X \notin S$, $\forall\, Z \supseteq X$, $Z \notin S$.

while the *disjunctive closed pattern* [3] is the largest one. These particular patterns are defined as follows.

Definition 5

- A pattern $I \subseteq \mathcal{I}$ is a ***disjunctive closed pattern*** if $I = h(I)$ or, equivalently, $Supp(\vee I) < min\{Supp(\vee I_1) \mid I \subset I_1\}$ [3].
- A pattern $I \subseteq \mathcal{I}$ is an ***essential pattern*** if $\forall\ I_1 \subset I$, $I \nsubseteq h(I_1)$ or, equivalently, $Supp(\vee I) > max\{Supp(\vee I_1) \mid I_1 \subset I\}$ [16].

Example 4 Consider our running context. The pattern CDEF is disjunctively closed, whereas 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$)$.

Since the empty set does not contain any item, we cannot define disjunctive support on this pattern. However, to ensure that the set of essential patterns is an order ideal in $(\mathbf{2}^{\mathcal{I}}, \subseteq)$, we will implicitly consider the empty set as an essential pattern. The same process has been recently highlighted in [22]. In the remainder, $\mathcal{FEP}$ denotes the set of frequent essential patterns associated to a given context $\mathcal{K}$ and a fixed *minsupp* value. The associated set of disjunctive closures, denoted $\mathcal{EDCP}$, is then equal to $\{h(I) \mid I \in \mathcal{FEP}\}$.

4 Overview of generalized association rules

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.

4.1 Generalized association rule framework

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 metrics, namely the support and the confidence [19]. 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 rules extend classic ones by 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. They also offer links between items through the disjunction connector, in addition to the conjunction one. The definition of a generalized association rule requires that of a Boolean expression (*aka* Boolean attribute in [23]) which is as follows:

Definition 6 (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 5 Let $\mathtt{A}$, $\mathtt{B}$ and $\mathtt{C}$ be three items, then $(\mathtt{A} \wedge \mathtt{B}) \vee \overline{\mathtt{C}}$ is a Boolean expression.

Definition 7 (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 6 Let $\mathcal{I} = \{\mathtt{A}, \mathtt{B}, \mathtt{C}, \mathtt{D}, \mathtt{E}, \mathtt{F}\}$ be a set of items. The rules $\mathtt{A} \wedge \mathtt{B} \Rightarrow \mathtt{C} \wedge \overline{\mathtt{D}}$ and $\mathtt{A} \vee \mathtt{E} \Rightarrow \mathtt{F}$ are two examples of generalized association rules.

We now present the support and the confidence of a generalized association rule.

Definition 8 (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 2 *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 $Supp(\varrho(x_1, x_2, \ldots, x_n)) \neq 0$, then $Conf(R) \in [0, 1]$.*

Example 7 Consider our running context and the generalized association rule R: $\mathtt{A} \vee \mathtt{E} \Rightarrow \mathtt{D}$. $Supp(R) = Supp((\mathtt{A} \vee \mathtt{E}) \wedge \mathtt{D})$. Since the premise and the conclusion are simultaneously satisfied by the transactions **2**, **3**, **4** and **5**, then $Supp(R) = \mathbf{4}$. While $Conf(R) = \dfrac{Supp(R)}{Supp(\mathtt{A} \vee \mathtt{E})}$. Since the disjunctive pattern $\mathtt{A} \vee \mathtt{E}$ is also fulfilled by both transactions **1** and **6** (which do not contain $\mathtt{D}$), then $Supp(\mathtt{A} \vee \mathtt{E}) = \mathbf{6}$. Consequently, $Conf(R) = \dfrac{4}{6} = \mathbf{0.66}$.

4.2 Support retrieval of generalized association rule forms

Generalized 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. However, computing their associated quality measures relies on a more complex process than that for positive rules. In this respect, in addition to both formulae shown in Lemma 1 (cf. page 3), the following ones are required for retrieving the supports of generalized rules, where x_i, $y_j \in \mathcal{I}$, and two Boolean expressions X and Y:

- $Supp((x_1 \wedge x_2 \wedge \ldots \wedge x_n) \wedge (y_1 \vee \ y_2 \vee \ \ldots \vee \ y_m)) = Supp(x_1 \wedge x_2 \wedge \ldots \wedge x_n) - Supp(x_1 \wedge x_2 \wedge \ldots \wedge x_n \wedge \overline{y_1} \wedge \overline{y_2} \wedge \ldots \wedge \overline{y_m})$ [21],
- $Supp(x_1 \wedge x_2 \wedge \ldots \wedge x_n \wedge \overline{y_1} \wedge \overline{y_2} \wedge \ldots \wedge \overline{y_m}) = \sum_{S \subseteq \{y_1, \ldots, y_m\}} (-1)^{|S|} Supp(x_1 \wedge x_2 \wedge \ldots \wedge x_n \wedge S)$ [12],
- $Supp(X \wedge Y) = Supp(X) + Supp(Y) - Supp(X \vee Y)$, and,
- $Supp(X \wedge \overline{Y}) = Supp(X \vee Y) - Supp(Y)$.

5 Extraction of generalized association rules

The process of mining generalized association rules is composed of three complementary steps which are as follows: (*i*) extracting an exact concise representation of frequent patterns based on disjunctive patterns; (*ii*) building a partially ordered structure w.r.t. set inclusion within disjunctive closed patterns; and, (*iii*) deriving generalized association rules from the built structure. The next paragraphs offer a detailed description of these steps.

5.1 Extracting a new concise representation based on disjunctive patterns

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

Theorem 1 *The set $\mathcal{EDCP} \cup \mathcal{FEP}$ of disjunctive patterns, associated to their disjunctive supports, is an exact representation of the set of frequent patterns $\mathcal{FP}$.*

Proof Let I be an arbitrary pattern. If there is a pattern I_1 *s.t.* $I_1 \in \mathcal{FEP}$ 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, and then compared to *minsupp* to retrieve its frequency status. If there is not such a pattern

$\mathcal{EDCP}$	Disj. Supp.	$\mathcal{FEP}$
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

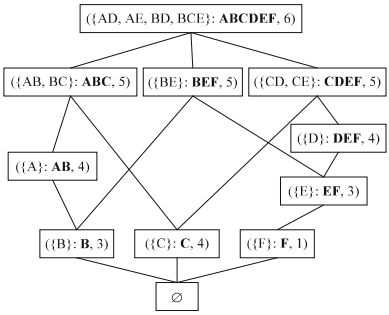

Fig. 1 (*Left*) The set $\mathcal{EDCP}$ and the associated disjunctive support and frequent essential patterns for $minsupp = \mathbf{1}$. (*Right*) The disjunctive equivalence classes partially ordered w.r.t. set inclusion

I_1, then I is necessarily encompassed between an *infrequent* essential pattern and its closure. Consequently, I is infrequent since the set of frequent patterns is an order ideal. □

Example 8 Figure 1 (Left) depicts 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. Figure 1 (Right) presents the associated Hasse diagram where for each disjunctive equivalence class, the associated disjunctive closed pattern f is accompanied by the set of its essential patterns FEP_f and its disjunctive support, under the form (FEP_f: f, $Supp(\vee f)$).

In the sequel, the representation based on the sets $\mathcal{FEP}$ and $\mathcal{EDCP}$ will be denoted $\mathcal{DSSR}$.[4] Starting from $\mathcal{DSSR}$, the conjunctive and negative supports of frequent patterns can 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. Please note that the associated mining algorithm, called DSSRM, is omitted here, due to space limitations (cf. [24] for details).

5.2 Building the partially ordered structure

In this subsection, we will propose a new algorithm, called POSB,[5] for partially sorting disjunctive closed patterns w.r.t. set inclusion. The POSB algorithm hence takes as input the representation $\mathcal{DSSR}$ *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 literature [25, 26], namely the sorting of disjunctive

[4]Stands for Disjunctive Search Space-based Representation.

[5]POSB is the acronym of Partially Ordered Structure Builder.

closed patterns, and the use of a border. Indeed, the set of disjunctive closed patterns $\mathcal{EDCP}$ is sorted w.r.t. the increasing pattern size. Since closures of equal size are not comparable, this sorting avoids unnecessary comparisons. In addition, it makes possible that the closure f under treatment 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 structure. This lower cover is composed by those closures which are *immediately covered* by f. On the other hand, the border $\mathcal{B}$ is found to be an anti-chain w.r.t. set inclusion containing maximal closures among those already treated.

In fact, both proposed algorithms in [25, 26] 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. Nevertheless, they are not directly adapted to our situation. Indeed, although the intersection of two disjunctive closed patterns is obviously a disjunctive closed pattern, this latter does not necessarily belong to $\mathcal{EDCP}$. This is due to the fact that it could have all its essential patterns infrequent and, hence, has been already pruned. On their side, the proposed algorithms in the literature 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. This is illustrated thanks to the following example.

Example 9 Consider a context containing the following transactions: A, B, ABC, ABD, and ABCD. Let $minsupp = \mathbf{2}$. In this situation, the set $\mathcal{FEP}$ of frequent essential patterns is equal to {A, B, C, D, AB}. The associated set $\mathcal{EDCP}$ of disjunctive closed patterns is then {C, D, ACD, BCD, ABCD}. By intersecting the disjunctive closures ACD and BCD, the result is CD which is not present in $\mathcal{EDCI}$ since the associated essential pattern, namely itself, is infrequent. Indeed, $Supp(\mathtt{CD}) = \mathbf{1} < \mathbf{2}$.

In the case of, for example, 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 patterns. For $minsupp = \mathbf{2}$, they form the set $\mathcal{FCP}$ of frequent closed patterns equal to {∅, A, B, AB, ABC, ABD, ACD, ABCD}. In this case, the intersection of each couple of elements from $\mathcal{FCP}$ also belongs to $\mathcal{FCP}$.

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 (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 2). 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 procedure will identify the common immediate predecessors of both b and f (cf. Algorithm 3). Finally, f will be added to 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.

Algorithm 1 POSB

Input: The set $\mathcal{EDCP}$ of disjunctive closed patterns.
Output: The disjunctive closed patterns ordered by set inclusion.

```
1  Begin
2    B := ∅ ;
3    Foreach (f ∈ EDCP) 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
```

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.

```
1  Begin
2    Foreach (l ∈ Lower_Cover(f)) do
3      inter := Lower_Cover(f))
4    inter := l ∩ pred;
5    If (inter = pred) then
6      return;
7    Else If (inter = l) then
8      Lower_Cover(f) := Lower_Cover(f) \ l;
9    Lower_Cover(f) := Lower_Cover(f) ∪ pred;
10 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.

```
1  Begin
2    Foreach (pred_b ∈ Lower_Cover(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
```

5.3 Deriving generalized association rules

As shown in Subsection 5.1, the $\mathcal{DSSR}$ representation allows computing the disjunctive, conjunctive and negative supports of each set of positive and negative

items whose positive variation[6] is based on a frequent pattern. Moreover, once the partially ordered structure built, selecting subsets of generalized association rules can be easily carried out. Thus, in the following paragraphs, we present an overview of the process by which we retrieve selected generalized association rules and evaluate their associated supports through traversing the partially ordered structure.

5.3.1 Description of the selected subsets

Rules can be classified according to the number of nodes (one or two) two) 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 $Z = f$, cf. Subsection 4.1).
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 $Z = f_2$, cf. Subsection 4.1).

Both kinds of rules—intra-node and inter-nodes—can either be exact or approximate.[7] 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 patterns such that either X or Y is a frequent essential pattern or a disjunctive closed one, and $Z = X \cup Y$ is a disjunctive closed pattern. The considered forms under the constraint 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* $\overline{X} \Rightarrow \vee Y$: $Supp(\overline{X} \Rightarrow \vee Y) = Supp(\overline{X} \wedge (\vee Y)) = Supp((\vee X) \vee (\vee Y)) - Supp(\vee X) = Supp(\vee Z) - Supp(\vee X)$.

5.3.2 Assessing quality measures of selected rules

The different forms we selected require the premise or the conclusion to be a frequent essential pattern (or its negation) and the rule to be based on a disjunctive

[6]The positive variation of $\{x_1, x_2, \ldots, x_n, \overline{y_1}, \overline{y_2}, \ldots, \overline{y_m}\}$ is equal to $\{x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m\}$.

[7]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.

closed pattern. 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 pattern or a disjunctive closed pattern. 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.

In the remainder, for the sake of simplicity, we assume that X is a frequent essential pattern or a disjunctive closed pattern. Since $Y = Z \setminus X$, then Y does not necessarily belong to $\mathcal{DSSR}$ and, may even not be a frequent pattern. 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 10. This definition requires that we introduce specific subsets of the sets $\mathcal{FEP}$ and $\mathcal{EDCP}$ w.r.t. Y. This is done as follows:

Definition 9 (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 $\mathcal{EDCP}$ is $\textsc{Minimal\_Supersets}(Y) = \min_{\subseteq}\{f \in \mathcal{EDCP} \mid Y \subseteq f \text{ and } \nexists\, f_1 \in \mathcal{EDCP} \text{ s.t. } Y \subset f_1 \subset f\}$.
- The set of maximal subsets of Y in $\mathcal{FEP}$ is $\textsc{Maximal\_Subsets}(Y) = \max_{\subseteq}\{e \in \mathcal{FEP} \mid e \subseteq Y \text{ and } \nexists\, e_1 \in \mathcal{FEP} \text{ s.t. } e \subset e_1 \subset Y\}$.

The bounds are then defined as follows:

Definition 10 (Upper and Lower bounds of disjunctive support) Let $Y \subseteq \mathcal{I}$. The upper and lower bounds of the disjunctive support of Y are as follows:

- $ub\_Supp(\vee Y) = min\{Supp(\vee f) \mid f \in \textsc{Minimal\_Supersets}(Y)\}$,
- $lb\_Supp(\vee Y) = max\{Supp(\vee e) \mid e \in \textsc{Maximal\_Subsets}(Y)\}$.

Both sets $\textsc{Minimal\_Supersets}(Y)$ and $\textsc{Maximal\_Subsets}(Y)$ 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 patterns size. Conversely, it augments whenever the patterns size increases. Thus, to obtain the upper bound, it is sufficient to consider the minimal supersets among disjunctive closed patterns covering Y. Whereas to get the lower bound, it is sufficient to consider maximal subsets among frequent essential patterns contained in Y.

An interesting situation happens if Y belongs to $\mathcal{DSSR}$, or is encompassed between a frequent essential pattern and its disjunctive closure. Indeed, $lb\_Supp(\vee Y) = ub\_Supp(\vee Y)$. 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 *approached rules*.[8] Such rules are defined as follows:

Definition 11 An association rule is said to be *approached* if it has either its support or its confidence not exactly determined.

Then, only approached rules having minimum possible values of support and confidence greater than or equal to *minsupp* and *minconf*, respectively, will be retained. Note that an approached 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 approached rules were shown to be of added value in the case of positive rules [27].

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 patterns of size **1**, which ensures the existence of the lower bound $lb\_Supp(\vee Y)$. The pattern Y is also covered by at least a disjunctive closed pattern, namely Z, which ensures the existence of the upper bound $ub\_Supp(\vee Y)$.

Example 10 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. $Supp(R_1) = Supp(\vee \text{BCE}) + Supp(\vee \text{ADF}) - Supp(\vee \text{ABCDEF}) = Supp(\vee \text{ADF})$ (since $h(\text{BCE}) = \text{ABCDEF}$). Since ADF $\notin \mathcal{DSSR}$, we need to evaluate its support. Since AD $\subseteq$ ADF $\subseteq$ h(AD) = ABCDEF (cf. Fig. 1 (Left)), then $lb\_Supp(\vee \text{ADF}) = \mathbf{6}$. 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. Fig. 1 (Right)): (Right)): $\vee$ ABC $\Rightarrow \in \mathcal{EDCP}$. Hence, $Supp(R_2) = Supp(\vee \text{ABC}) + Supp(\vee \text{DEF}) - Supp(\vee \text{ABCDEF}) = \mathbf{5} + \mathbf{4} - \mathbf{6} = \mathbf{3}$, 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 as R_2. Nevertheless, its confidence is equal to $\mathbf{0.75}$. Hence, R_3 is a valid rule while R_2 is not.

5.3.3 Associated mining algorithm

Now, we describe the GARS[9] algorithm allowing the extraction of the selected generalized association rules. Its pseudo-code is given by Algorithm 4. For each disjunctive closed pattern $f \in \mathcal{EDCP}$, the first step in GARS consists in searching for the set SET_PREM_CL$_f$ gathering the subsets that will play the role of premise and, then, conclusion of each rule based on f. These patterns are composed by the

[8] We use in this paper "approached rules" instead of the commonly used "approximated rules" in order to avoid confusion with "approximate rules".

[9] GARS is the acronym of Generalized Association Rules Selector.

Algorithm 4 GARS

Input: - The partially ordered structure, *minsupp* and *minconf*.
Output: - The sets $\mathcal{EGAR}$, $\mathcal{AGAR}$ and $\mathcal{ApGAR}$.
1 **Begin**
2 **Foreach** ($f \in \mathcal{EDCP}$) **do**
3 SET_PREM_CL$_f$:= FEP $_f \cup$ *Cov*$_l(f) \cup \{e \mid e \in$ FEP $_{f_1}$ *s.t.* $f_1 \in$ *Cov*$_l(f)\}$;
4 **Foreach** ($X \in$ SET_PREM_CL$_f$) **do**
5 $Y := f \setminus X$;
6 COMPUTE_BOUNDS (*up_Supp*($\vee Y$), *lp_Supp*($\vee Y$));
7 **If** (*up_Supp*($\vee Y$) = *lp_Supp*($\vee Y$)) **then**
8 GENERATE_RULES_EXACT_BOUNDS(f, X, Y, *Supp*($\vee Y$), *minsupp*, *minconf*);
9 **Else**
10 GENERATE_RULES_APPROXIMATED_BOUNDS(f, X, Y, *up_Supp*($\vee Y$), *lp_Supp*($\vee Y$), *minsupp*, *minconf*);
11 **End**

set of its frequent essential patterns, denoted FEP$_f$, and the set of its immediate predecessors equal to *Cov*$_l(f)$ as well as their respective frequent essential patterns (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 invoked. 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. 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), then the GENERATE_RULES_APPROXIMATED_BOUNDS procedure is invoked. 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. 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*. On the other hand, 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 $\mathcal{ApGAR}$ of valid approached generalized association rules. Otherwise, it is added according to its confidence value to $\mathcal{EGAR}$ or $\mathcal{AGAR}$.

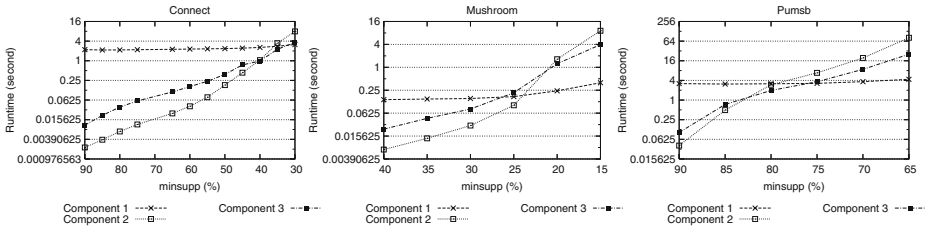

Fig. 2 Mining time of generalized association rules from dense contexts

6 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. 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.[10] 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 patterns as a basis for mining generalized association rules.

Here we scrutinize obtained representative results on six benchmark datasets, namely Connect, Mushroom, and Pumsb which are commonly considered as dense, and on the other hand Kosarak, Retail and T40I10D100K commonly considered as sparse.[11] In these experiments, the value of *minsupp* varies and that of *minconf* is set to the associated relative minimum support threshold, i.e., $\frac{minsupp}{|\mathcal{O}|}$. 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 three components: (*i*) extraction of the $\mathcal{DSSR}$ representation; (*ii*) building of the partially ordered structure; and, (*iii*) deriving the valid generalized association rules which are under the selected rule forms. On the other hand, we put the focus 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 approached.

Figures 2 and 3 graphically show representative results on the mining time (in seconds) of the three components of GARM for dense and sparse contexts, respectively. The obtained 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

[10] GARM is the acronym of Generalized Association Rule Miner. The software GARM is available at: http://fc.isima.fr/~mephu/FILES/GARM_software.zip.

[11] These datasets are available at: http://fimi.cs.helsinki.fi/data.

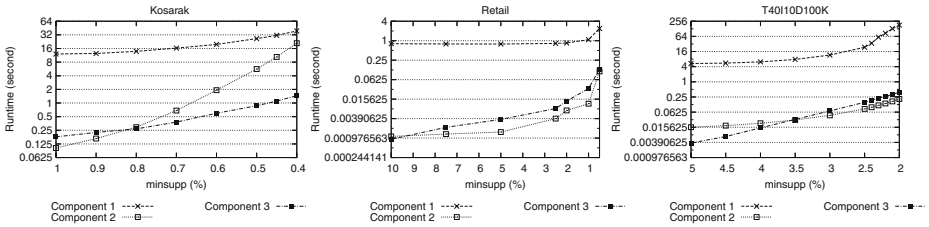

Fig. 3 Mining time of generalized association rules from sparse contexts

generalized association rules performed by the third one is in almost all cases the fastest step. 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 patterns and, hence, disjunctive closed patterns 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, Figs. 4 and 5 show our main results on the total number of valid generalized association rules distinguished w.r.t. their type. 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 patterns per class. Consequently, the number of rules involving essential patterns and disjunctive closed patterns 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 patterns and their closure, and not any pattern. On the other hand, 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 pattern is equal to its disjunctive closure, which is not the case for contexts such as Mushroom and Pumsb. Moreover, the number of approached 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

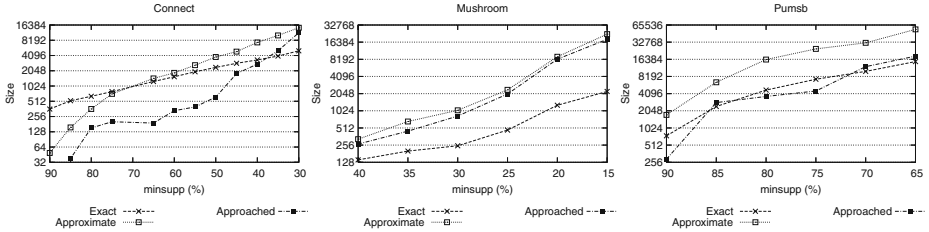

Fig. 4 Number of mined generalized association rules from dense contexts

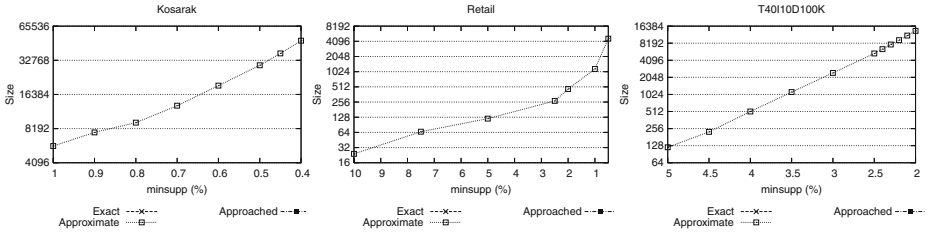

Fig. 5 Number of mined generalized association rules from sparse contexts

closed pattern, on which is based the rule, and the premise (or conclusion) containing either a disjunctive closed pattern or a frequent essential rules. In the case of RETAIL, KOSARAK, and T40I10D100K contexts, the support of the difference is always exactly determined, which leads to the absence of approached association rules. Indeed, this difference is always encompassed between a frequent essential pattern and its disjunctive closure. Since each equivalence class is limited to a unique element, the difference is an essential pattern (equal to its closure) which explains why its support is always determined.

Let us now 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 slightly affects that of the third component since *minconf* is only used in this component. On the other hand, the number of exact generalized association rules does not change when the value of *minconf* is modified. Indeed, these rules have always a confidence value equal to **1**. Only the number of approximate and approached 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 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* (cf. [4] for more details).

7 Related work

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 [18, 19].

Some work 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 is often extracted. Thus, existing approaches have tried to address this problem through the use of additional background information about the data [28], incorporating item correlations [29], and additional rule interestingness measures [30], etc.

In the remainder of this section, we describe related work on association rules relying on the disjunction connector within items. Our description is divided into two parts: the first concentrates on the GUHA approach which constitutes a main related work to ours. The second part is dedicated to the remaining related work.

7.1 The GUHA approach

A main related approach to our work consists in the GUHA approach, developed since the mid-sixties. GUHA stands for General Unary Hypotheses Automaton. Many works in the literature either describe the original GUHA methods (like [10, 11, 31–33]), extend them [8, 23] or apply them in real-life application [8, 34]. The GUHA methods are realized by GUHA procedures such as 4FT procedure that we will describe here since it is the most related one to our work [23]. Note however that GUHA is not in principle restricted to mining association rules, the most used GUHA procedures mine for generalized association rules, as defined in [23]. The 4FT procedure mines for rules under the form $\varphi \approx \psi$ where φ and ψ are two Boolean attributes (or equivalently, Boolean expression), that may be deduced starting from categorial attributes. On the other hand, $\approx$ is a 4ft-quantifier which expresses a kind of dependency between φ and ψ [35]. φ then represents the premise part of a generalized association rules in our case, while ψ represents the conclusion part. The relation between φ and ψ is thus evaluated on the basis of a 4ft table [23] as shown in Table 1.

A 4ft table is constituted by a quadruplet of natural numbers a, b, c, and d over a data matrix (the extraction context $\mathcal{K}$ in our case) $\mathcal{M}$ so that :

- a is the number of objects of $\mathcal{M}$ satisfying φ and ψ.
- b is the number of objects of $\mathcal{M}$ satisfying φ and not satisfying ψ.
- c is the number of objects of $\mathcal{M}$ not satisfying φ but satisfying ψ.
- d is the number of objects of $\mathcal{M}$ satisfying neither φ nor ψ.

Note that the sum of these four numbers corresponds to the number of rows of the matrix, what corresponds to the size of the objects set, $|\mathcal{O}|$, in our case.

A 4ft-quantifier is then a condition over the 4ft table. Many quantifiers are considered under the GUHA methods. The main related one to our work is the founded implication quantifier introduced in [10]. This quantifier is defined through the following condition:

$$a \geq Base \wedge \frac{a}{a+b} \geq p, \quad \text{such that} \quad Base > 0 \ \text{ and } \ 0 < p \leq 1.$$

Base and p are two threshold parameters of the procedure. The *Base* parameter represents the minimum absolute number of objects that must satisfy both φ and ψ. This parameter corresponds to the minimum support threshold *minsupp* we used in our work. On the other hand, the p parameter indicates that at least $100p$ per cent of objects satisfying φ satisfy also ψ. This parameter corresponds to the minimum confidence threshold *minconf* we used in our work.

The GUHA methods thus offer a general framework for mining different kinds of association rules. The rules we concentrate on in this work can be considered as particular generalized association rules through a structural characterization of the disjunctive search space. This allows us to define specific rule forms, rather than

Table 1 The 4ft table

$\mathcal{M}$	ψ	$\neg\psi$
φ	a	b
$\neg\varphi$	c	d

mining the whole set of valid generalized association rules, while taking into account particular elements within this search space, namely disjunctive closed patterns and essential patterns.

7.2 Other related work

Some other works [5, 36] 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 $\oplus$ [5]. 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 [36], 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 [5] using the operator $\oplus$. Other forms of generalized association rules were also described in [37]. In [38], 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 [39]. On the other hand, Elble et al. used disjunctive rules to handle numerical attributes by considering disjunctions between intervals [40]. In classification association rule mining, a disjunctive rule having a premise (*resp.* conclusion) composed by a conjunction (*resp.* disjunction) of items is called *a multiple target rule* [41]. Finally, it is worth noting that such a rule form has also been used as an intermediate step for defining concise representations for frequent patterns (e.g., those based on disjunction-free sets [42] and (generalized) disjunction-free generators [19]).

8 Conclusion and further research

In this paper, 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. 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 patterns. 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 pattern. On the other hand, the fact that this representation is based on disjunctive patterns, namely essential and disjunctive closed patterns, 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 patterns. 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 of 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. On the other hand, the number of exact, approximate and approached rules closely depends on dataset characteristics.

Other avenues for future work mainly address the following points: first, a detailed comparison of our approach to the already proposed tools under the general GUHA approach is under investigation. 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. This issue is highly correlated with that aiming at going beyond the support-confidence framework. We then plan to lead a study aiming at selecting the right quality measures according to each rule form [14, 15]. This allows us to further reduce the number of mined rules while retaining those which are interesting for end-users w.r.t. the couple (*rule form*, *metric*). 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. Even not mentioned in [43], this measure is based on the disjunctive support. Indeed, the bond of an arbitrary pattern 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. Finally, the application of the proposed approach on real-life data will be a key step for highlighting the interest of the generalized association rules.

Acknowledgements We would like to thank the anonymous reviewers for their helpful comments and suggestions.

References

1. Steinbach, M., Kumar, V.: Generalizing the notion of confidence. Knowl. Inf. Syst. **12**(3), 279–299 (2007)
2. Tzanis, G., Berberidis, C.: Mining for mutually exclusive items in transaction databases. Int. J. Data Warehousing Mining **3**(3), 45–59 (2007)
3. Hamrouni, T., Ben Yahia, S., Mephu Nguifo, E.: Sweeping the disjunctive search space towards mining new exact concise representations of frequent itemsets. Data Knowl. Eng. **68**(10), 1091–1111 (2009)
4. Hamrouni, T.: Mining concise representations of frequent patterns through conjunctive and disjunctive search spaces. Ph.D. thesis, University of Tunis El Manar (Tunisia) and University of Artois (France). Available at: http://tel.archives-ouvertes.fr/tel-00465733 (2009)
5. 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 (CIKM 2001), pp. 482–489. Atlanta, Georgia, USA (2001)
6. Hattori, L., dos Santos, G., Cardoso, F., Sampaio, M.: Mining software repositories for software change impact analysis: a case study. In: Proceedings of the 23rd Brasilian Symposium on Database (SBBD 2008). Campinas, Brazil (2008)

7. She, S.: Feature model mining. Master thesis, University of Waterloo, Waterloo, Ontario, Canada (2008)
8. Ralbovský, M., Kuchar, T.: Using disjunctions in association mining. In: Proceedings of the 7th Industrial Conference on Data Mining (ICDM 2007). LNCS, vol. 4597, pp. 339–351. Springer, Leipzig, Germany (2007)
9. Srikant, R., Agrawal, R.: Mining generalized association rules. In: Proceedings of the 21th International Conference on Very Large Data Bases (VLDB 1995), pp. 407–419. Zurich, Switzerland (1995)
10. Hájek, P., Havel, I., Chytil, M.: The GUHA method of automatic hypotheses determination. Computing **1**, 293–308 (1966)
11. Hájek, P., Havránek, T.: Mechanizing Hypothesis Formation: Mathematical Foundations for a General Theory. Springer, New York (1978)
12. Toivonen, H.: Discovering of frequent patterns in large data collections. Ph.D. thesis, University of Helsinki, Helsinki, Finland (1996)
13. Weiss, G.M.: Mining with rarity: a unifying framework. ACM-SIGKDD Explor. **6**(1), 7–19 (2004)
14. 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 (MLDM 2007). LNCS, vol. 4571, pp. 533–547. Springer, Leipzig, Germany (2007)
15. Geng, L., Hamilton, H.J.: Interestingness measures for data mining: a survey. ACM Comput. Surv. **38**(3), 1–31 (2006)
16. Casali, A., Cicchetti, R., Lakhal, L.: Essential patterns: a perfect cover of frequent patterns. In: Proceedings of the 7th International Conference on Data Warehousing and Knowledge Discovery (DaWaK 2005). LNCS, vol. 3589, pp. 428–437. Springer, Copenhagen, Denmark (2005)
17. Wu, X., Zhang, C., Zhang, S.: Efficient mining of both positive and negative association rules. ACM Trans. Inf. Sys. **22**(3), 381–405 (2004)
18. Ceglar, A., Roddick, J.F.: Association mining. ACM Comput. Surv. **38**(2), 1–42 (2006)
19. Kryszkiewicz, M.: Concise representations of frequent patterns and association rules. Habilitation dissertation, Institute of Computer Science, Warsaw University of Technology, Warsaw, Poland (2002)
20. Ganter, B., Wille, R.: Formal Concept Analysis. Springer, New York (1999)
21. Galambos, J., Simonelli, I.: Bonferroni-type Inequalities with Applications. Springer, New York (2000)
22. Kryszkiewicz, M.: Closures of downward closed representations of frequent patterns. In: Proceedings of the 4th International Conference on Hybrid Artificial Intelligence Systems (HAIS 2009). LNCS, vol. 5572, pp. 104–112. Springer, Salamanca, Spain (2009)
23. Rauch, J.: Logic of association rules. Appl. Intell. **22**(1), 9–28 (2005)
24. Hamrouni, T., Ben Yahia, S., Mephu Nguifo, E.: Optimized mining of a concise representation for frequent patterns based on disjunctions rather than conjunctions. In: Proceedings of the 23rd International Florida Artificial Intelligence Research Society Conference (FLAIRS 2010), pp. 422–427. AAAI, Daytona Beach, Florida, USA (2010)
25. Baixeries, J., Szathmary, L., Valtchev, P., Godin, R.: 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, vol. 5548, pp. 162–177. Springer, Darmstadt, Germany (2009)
26. 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 (LaCIM 2000), pp. 293–306. Montréal, Canada (2000)
27. Cheng, J., Ke, Y., Ng, W.: Effective elimination of redundant association rules. Data Min. Knowl. Discov. **16**(2), 221–249 (2008)
28. Savasere, A., Omiecinski, E., Navathe, S.: Mining for strong negative associations in a large database of customer transactions. In: Proceedings of the 14th International Conference on Data Engineering (ICDE 1998), pp. 494–502. IEEE Computer Society, Orlando, Florida, USA (1998)
29. Antonie, M., Zaïane, O.R.: 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, vol. 3202, pp. 27–38. Springer, Pisa, Italy (2004)
30. Morzy, M.: Efficient mining of dissociation rules. In: Proceedings of the 8th International Conference on Data Warehousing and Knowledge Discovery (DaWaK 2006). LNCS, vol. 4081, pp. 228–237. Springer, Krakow, Poland (2006)
31. Hájek P. (guest ed.): Int. J. Man-Mach. Stud. **10** (1978, special issue on GUHA)

32. Hájek P. (guest ed.): Int. J. Man-Mach. Stud. **15** (1981, second special issue on GUHA)
33. Hájek, P., Holena, M., Rauch, J.: The GUHA method and its meaning for data mining. J. Comput. Syst. Sci. **76**(1), 34–48 (2010)
34. Rauch, J., Simunek, M.: Dealing with background knowledge in the SEWEBAR project. In: Berendt, B. et al. (eds.) Knowledge Discovery Enhanced with Semantic and Social Information, SCI 220, pp 89–106. Springer, Berlin (2009)
35. Rauch, J.: Classes of association rules: an overview. In: Lin, T., et al. (eds.) Data Mining: Foundations and Practice, SCI 118, pp. 315–337. Springer, Berlin (2008)
36. 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 (IKS 2003), pp. 155–157. Scottsdale, AZ, USA (2003)
37. Grün, G.A.: New forms of association rules. Technical Report TR 1998-15, School of Computing Science, Simon Fraser University, Burnaby, BC, Canada (1998)
38. 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 (CME 2005), pp. 321–325. Takamatsu, Japan (2005)
39. Shima, Y., Mitsuishi, S., Hirata, K., Harao, M.: Extracting minimal and closed monotone DNF formulas. In: Proceedings of the 7th International Conference Discovery Science (DS 2004). LNCS, vol. 3245, pp. 298–305. Springer, Padova, Italy (2004)
40. Elble, J., Heeren, C., Pitt, L.: Optimized disjunctive association rules via sampling. In: Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM 2003), pp. 43–50. Melbourne, Florida, USA (2003)
41. Li, J., Jones, J.: Using multiple and negative target rules to make classifiers more understandable. Knowl.-Based Syst. **19**(6), 438–444 (2006)
42. Bykowski, A., Rigotti, C.: DBC: a condensed representation of frequent patterns for efficient mining. Inf. Syst. **28**(8), 949–977 (2003)
43. Omiecinski, E.R.: Alternative interest measures for mining associations in databases. IEEE Trans. Knowl. Data Eng. **15**(1), 57–69 (2003)