Structures of Association Rule Set

Conference Paper · March 2012

DOI: 10.1007/978-3-642-28490-8_38

3 authors:

Anh Tran
University of Dalat

Tin C Truong
University of Dalat

Bac Le
Ho Chi Minh City University of Science

Structures of Association Rule Set

Anh Tran[1], Tin Truong[1] and Bac Le[2]

[1] University of Dalat, Dalat, Vietnam
anhtn@dlu.edu.vn, tintc@dlu.edu.vn

[2] University of Natural Science Ho Chi Minh, Ho Chi Minh, Vietnam
lhbac@fit.hcmuns.edu.vn

Abstract. This paper shows a mathematical foundation for almost important features in the problem of discovering knowledge by association rules. The class of frequent itemsets and the association rule set are partitioned into disjoint classes by two equivalence relations based on closures. Thanks to these partitions, efficient parallel algorithms for mining frequent itemsets and association rules can be obtained. Practically, one can mine frequent itemsets as well as association rules just in the classes that users take care of. Then, we obtain structures of each rule class using corresponding order relations. For a given relation, each rule class splits into two subsets of basic and consequence. The basic one contains minimal rules and the consequence one includes in the rules that can be deducted from those minimal rules. In the rest, we consider association rule mining based on order relation min. The explicit form of minimal rules according to that relation is shown. Due to unique representations of frequent itemsets through their generators and corresponding eliminable itemsets, operators for deducting all remaining rules are also suggested. Experimental results show that mining association rules based on relation min is better than the ones based on relations of minmin and minMax (considered recently in [7, 9] and [11, 10]) in terms of reduction in mining times as well as number of basic rules (thus users easily manage them).

Keywords: association rule; basic rule; consequence rule; generator; eliminable itemset; equivalence relation; order relation.

1 Introduction

Firstly introduced and researched by Agrawal et al. [1], association rule mining is one of the important problems in data mining. Traditional approach solves this problem in two phases of (1) to extract frequent itemsets whose the occurrences exceed minimum support in the data, and (2) to generate association rules from them with the given minimum confidence. The cardinalities of frequent itemset class $\mathcal{FS}$ and association rule set $\mathcal{ARS}$ can also grow unwieldy. The traditional algorithms (such as AIS [1], Apriori [2], Ap-genrules [2]) for finding those sets generate many candidates and check many sufficient conditions. Then, the running time and the memory capacity are usually enormous. Moreover, it is too complicated for users to understand and manage the results whose sizes are so big. Recently, some researchers have proposed a new framework for mining association rules. Firstly, frequent closed itemsets are mined by the efficient algorithms such as Charm-L [12], Closet [8]. The number of those

J.-S Pan, S.-M. Chen, N.T. Nguyen (Eds.): ACIIDS 2012, Part II, LNAI 7197, pp.361-370, 2012.

itemsets is usually less than the one of all frequent itemsets. Based on them, a small set of useful association rules (basic rules) is obtained. Zaki [11] considered the most-general rules and indicated how to find them. However, his method generates many candidates. Furthermore, there are no algorithms for finding the remaining rules (see [10]). With similar ideas, using the bases of exact [4] and approximate [6], Pasquier et al. [7] proposed the algorithms for mining the minimal rules and deriving the other ones from them. However, those algorithms miss some rules and waste much time to generate rules as well as to delete the repeated ones (see [9]).

In order to overcome these disadvantages, understanding structures of frequent itemsets and association rule set is essential. Based on frequent closed itemset lattice, we use two appropriate equivalence relations on $\mathcal{FS}$ and $\mathcal{ARS}$ to partition them into disjoint classes $\mathcal{FS}$(L) and $\mathcal{AR}$(L, S), where (L, S) is a pair of two nested non-empty frequent closed itemsets. All itemsets in each itemset class have the same closure so the same support. All rules in each rule class have the same support and confidence. Without loss of the generality, we only need to investigate each class independently. For each itemset class $\mathcal{FS}$(L), we show that the generators [3] are even minimal itemsets according to an appropriate order relation. In order to generate all remaining itemsets of the same class, we only need to add eliminable itemsets [9], [10] (this concept relates to sets of probability zero) to generators of L. To avoid the duplications, a simple sufficient condition related to generators and eliminable itemsets is checked. For each pair (L, S), based on structure of $\mathcal{FS}$(L) and different order relations over each class $\mathcal{AR}$(L, S), we show different structures of association rule set. According to given order relation, this set splits into subsets of basic and consequence. All rules in basic one are minimal. The consequence one contains non-minimal rules. Those rules are sufficiently deduced by adding, deleting or moving appropriate eliminable itemsets in both sides of the basic ones. A sufficient condition is verified to avoid the duplications. Experiments will figure out that mining association rules based on order relation min is better than the ones based on order relations of minMax and minmin in terms of reductions in number of basic rules as well as in mining times.

The paper is organized as follows. Section 2 reminds some elementary concepts of concept lattice, frequent itemsets, association rules and results of partitioning itemset class, generators and eliminable itemsets. It also shows unique representations of itemsets with the same closure based on equivalence relation on itemsets, their generators and eliminable itemsets. Section 3 partitions association rule set by a different equivalence relation and figures out its structures using different order relations over each equivalence rule class. Sections 4 and 5 show the experimental results and conclusion. Since the size of the paper is limited, we do not show some proofs, algorithms and similar results.

2 Preliminaries

2.1 Concept Lattice, Frequent Itemset and Association Rule

Given non-empty sets $\mathcal{O}$ containing objects (or transactions), $\mathcal{A}$ containing attributes (or items) related to objects o $\in$ $\mathcal{O}$. Let $\mathcal{R}$ be a binary relation on $\mathcal{O}$x$\mathcal{A}$. Consider two set functions: λ: $2^{\mathcal{O}} \rightarrow 2^{\mathcal{A}}$, ρ: $2^{\mathcal{A}} \rightarrow 2^{\mathcal{O}}$ defined as follows: $\forall A \subseteq \mathcal{A}$, $O \subseteq \mathcal{O}$: $\lambda(O)=\{a \in \mathcal{A} \mid (o, a) \in \mathcal{R}, \forall o \in O\}$, $\rho(A)=\{o \in \mathcal{O} \mid (o, a) \in R, \forall a \in A\}$, where $2^{\mathcal{O}}$ and $2^{\mathcal{A}}$ are the classes of all subsets of $\mathcal{O}$ and $\mathcal{A}$. Assign that, $\lambda(\varnothing)=\mathcal{A}$, $\rho(\varnothing)=\mathcal{O}$, $h=\lambda_o\rho$, $h'=\rho_o\lambda$. Sets of $h'(O)$ and $h(A)$ are in turn called the closures of O and A. Itemsets A and O are closed iff $h(A)=A$ [3] and $h'(O)=O$. If $A=\lambda(O)$ and $O=\rho(A)$, the pair $C=(O, A) \in \mathcal{O}x\mathcal{A}$ is called a concept. In the class of concepts $\mathcal{C} = \{C \in \mathcal{O}x\mathcal{A}\}$, if defining order relation $\prec$ as relation $\supseteq$ between subsets of $\mathcal{O}$, then $\mathcal{L} \equiv (\mathcal{C}, \prec)$ is a concept lattice [3]. On $\mathcal{O}$, consider σ-field [5] $\mathcal{F}_{max}=2^{\mathcal{O}}$ including all subsets of $\mathcal{O}$. Let $\mathcal{P}$ be a countable probability measure: $\mathcal{P}(O)=|O|/|\mathcal{O}|$, $\forall O \subseteq \mathcal{O}$. We have probability space $(\mathcal{O}, \mathcal{F}_{max}, \mathcal{P})$.

Let s_0 and c_0 be minimum support and minimum confidence. For any itemset S, the probability $\mathcal{P}(\rho(S)) = |\rho(S)| / |\mathcal{O}|$ is called the support of S, denoted by supp(S). An itemset S is frequent iff $supp(S) \geq s_0$ [1]. Let $\mathcal{CS}$, $\mathcal{FS}$ and $\mathcal{FCS} = \mathcal{CS} \cap \mathcal{FS}$ be respectively the classes of all closed itemsets, all frequent itemsets and all frequent closed itemsets. For every non-empty, strict subset L from S ($\varnothing \neq L \subset S$), $S \in \mathcal{FS}$ and $R=S \backslash L$, denote r: $L \rightarrow R$ as the rule created by L, R (or L, S). The conditional probability of $\rho(R)$ given $\rho(L)$: $c(r) \equiv \mathcal{P}[\rho(S)|\rho(L)] = \mathcal{P}[\rho(L) \cap \rho(R)]/\mathcal{P}(\rho(L)) = |\rho(S)|/|\rho(L)|$ is called the confidence of r. The rule r is called an association rule iff $c(r) \geq c_0$ [1]. Let $\mathcal{ARS}$ be the set of all association rules corresponding with s_0 and c_0. For two non-empty itemsets G, A: $\varnothing \neq G \subseteq A \subseteq \mathcal{A}$, G is called a generator [7] of A iff $h(G)=h(A)$ and ($\forall G'$: $\varnothing \neq G' \subset G \Rightarrow h(G') \subset h(G)$). Let $\mathcal{G}(A)$ be the class of all generators of A numbered by 1, 2, …: $\mathcal{G}(A) = \{A_i, i \in I=\{1, 2, \ldots, n_A\}, n_A \leq |A|\}$.

2.2 Partition of the Class of All Itemsets, Generators and Eliminable Itemsets

An equivalence relation based on the closures of itemsets partitions $\mathcal{FS}$ into disjoint equivalence classes. Using an appropriate order relation, we figure out that generators are minimal elements (in term of subset relation) over each class.

Definition 1 [9]. Closed mapping h: $2^{\mathcal{A}} \rightarrow 2^{\mathcal{A}}$ generates a binary relation $\sim_h$ in class $2^{\mathcal{A}}$: $\forall$ A, B $\subseteq \mathcal{A}$: $A \sim_h B$ *iff* $h(A) = h(B)$.

Theorem 1 [9] (*Partition of itemset class*). $\sim_{\mathcal{A}}$ is an equivalence relation. It partitions $2^{\mathcal{A}}$ into disjoint equivalence classes. All itemsets in each class have the same closure so the same support. We have:

$$2^{\mathcal{A}} = \sum_{A \in \mathcal{CS}} [A] = \sum_{A \in \mathcal{CS}} \{\varnothing \neq X \subseteq A \mid h(X) = A\} \;\; or \;\; \mathcal{FS} = \sum_{A \in \mathcal{FCS}} [A]$$

where: [A] denotes the equivalence class containing A and "+" denotes the union operator of two disjoint sets.

Definition 2. Consider order relation $\prec_{\mathcal{A}}$ (over the set $2^{\mathcal{A}} \backslash \{\varnothing\}$) defined as follows: $\forall$A, B: $\varnothing \neq$ A, B $\subseteq \mathcal{A}$: $A \prec_{\mathcal{A}} B$ *iff* ($A \subseteq B$ *and* $h(A) = h(B)$).

Proposition 1. $\prec_{\mathcal{A}}$ *is a partial order relation. A minimal element G of* equivalence class [A] *is a generator of* A and $\mathcal{G}(A) \neq \varnothing$.

Definition 3 (*Eliminable itemsets*) [9]. In $2^{\mathcal{A}}$, a subset *R is called eliminable itemset in S iff* $R \subset S$ *and* $\rho(S) = \rho(S \backslash R)$. Denote the class of all eliminable itemsets in S by $\mathcal{N}(S)$ and assign that $\mathcal{N}^*(S) := \mathcal{N}(S) \backslash \{\varnothing\}$.

Proposition 2 (*Recognizing an eliminable itemset*). $\forall R \subset S$, we have:

a) $R \in \mathcal{N}(S) \Leftrightarrow \rho(S \backslash R) \subseteq \rho(R) \Leftrightarrow h(S) = h(S \backslash R) \Leftrightarrow supp(S) = supp(S \backslash R) \Leftrightarrow \mathcal{P}(\rho(S \backslash R) \backslash \rho(S)) = 0$. Obviously, the set of transactions containing S is a subset of the set of transactions containing S\R, i.e., $\rho(S) \subseteq \rho(S \backslash R)$. When R is an eliminable itemset in S, the equality occurs. Then, the probability of $\rho(S \backslash R) \backslash \rho(S)$ is equal to zero. Thus, deleting R in S does not change the support of S.

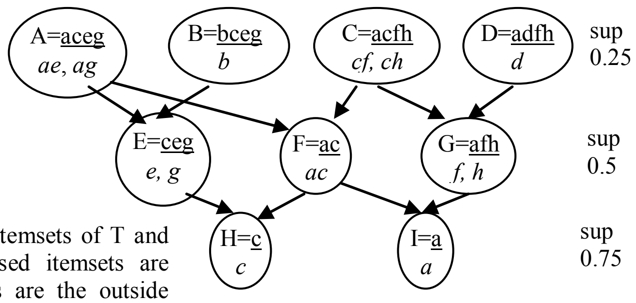
b) $\mathcal{N}(S) = \{A: A \subseteq S \backslash G_0,\ G_0 \in \mathcal{G}(S)\}$ [9].

c) $\mathcal{N}(S) = \bigcup_{G_0 \in \mathcal{G}(S)} \mathcal{N}(S, G_0)$, *where* $\mathcal{N}(S, G_0) = \{A: A \subseteq S \backslash G_0\}$.

Example 1. Figure 1 contains database T, the lattice of closed itemsets on T and the corresponding generators. This figure is used in examples in the rest of the paper. $\mathcal{A}$ = {a, b, c, d, e, f, g, h} is partitioned into nine disjointed equivalence classes: [A], [B] ... [I]. With X=aceg (i.e. {a, c, e, g}) we have: $\mathcal{G}(X)$ = {ae, ag}, [X] = {ae, ag, ace, acg, aeg, aceg} (the supports of all itemsets in [X] are equal to 0.25) and $\mathcal{N}^*(X, ae)$ = {cg, c, g}, $\mathcal{N}^*(X, ag)$={ce, c, e}, $\mathcal{N}^*(X) = \mathcal{N}^*(X, ae) \cup \mathcal{N}^*(X, ag)$ = {cg, c, g, e, ce}.

Trans	Items
1	a, c, e, g
2	a, c, f, h
3	a, d, f, h
4	b, c, e, g

(a) Database T.



(b) The lattice of closed itemsets of T and their generators (the closed itemsets are underlined, their supports are the outside numbers and their generators are italicized).

Fig. 1. Database T, the lattice of closed itemsets of T and their generators.

2.3 Unique Representation of Itemsets with the Same Closure

We show unique representations of itemsets of the same closure with and without constraints that play an important role in deriving non-repeatedly consequence rules.

Proposition 3. For every itemset X: $\varnothing \neq X \subseteq \mathcal{A}$, denote *equivalence class restricted on X* as follows: $\lfloor X \rfloor = \{X' \subseteq X \mid X' \neq \varnothing, h(X') = h(X)\}$ or $\lfloor X \rfloor = \{X' \subseteq X \mid X' \in [X] \backslash \{\varnothing\}\}$. The following statements hold:

a) $\lfloor X \rfloor \subseteq [X]$; $\forall X' \in \lfloor X \rfloor$, $\mathcal{G}(X') \subseteq \mathcal{G}(X)$; and $X \in \mathcal{CS} \Leftrightarrow \lfloor X \rfloor = [X]$.

b) *Closure-preserved property of eliminable itemset*:

$\forall X' = X_0 + Y \in \lfloor X \rfloor$, where $X_0 \in \mathcal{G}(X)$, $Y \subseteq X \backslash X_0$, then: $\forall X'' \subseteq X' \backslash X_0$, $X' \backslash X'' \in \lfloor X \rfloor$

$\forall X' \in \lfloor X \rfloor, X'' \subseteq X \backslash X' \Rightarrow X'+X'' \in \lfloor X \rfloor$

c) *Representation of itemsets*: $X' \in \lfloor X \rfloor \Leftrightarrow \exists X_i \in \mathcal{G}(X), X'' \in \mathcal{N}(X, X_i): X' = X_i+X''$.

An itemset X could have different generators, so itemsets generated in $\lfloor X \rfloor$ by proposition 3.c could be repeated. The theorem 2 shows unique representation of each of them.

Theorem 2 (*Unique representation of itemsets by their generators and eliminable itemsets*). $\forall \emptyset \neq X', X \subseteq \mathcal{A}$, let $X_U = \bigcup_{X_i \in \mathcal{G}(X)} X_i$, $X_{U,i} = X_U \backslash X_i$, $X\_ = X \backslash X_U$, $\mathcal{IS}(X) = \{X'=X_i+X'_i+X^{\sim} \mid X_i \in \mathcal{G}(X), X^{\sim} \subseteq X\_, X'_i \subseteq X_{U,i}, i=1 \text{ or } (i>1: X_k \not\subset X_i+X'_i, \forall k: 1 \leq k<i)\}$ and denote $\mathcal{FS}(X) = \mathcal{IS}(X)$, *if* $supp(X) \geq s_0$:

a) *All itemsets of* $\mathcal{IS}(X)$ *are non-repeatedly generated.* **b)** $\lfloor X \rfloor = \mathcal{IS}(X)$.

Proof: (a) Assume that there exist i, k such that $i>k\geq 1$ and $X_i+X'_i+X^{\sim}_i \equiv X_k+X'_k+X^{\sim}_k$, where $X_i, X_k \in \mathcal{G}(X)$, $X^{\sim}_i, X^{\sim}_k \subseteq X\_$, $X'_i \subseteq X_{U,i}$, $X'_k \subseteq X_{U,k}$. Since $X_k \cap X^{\sim}_i = \emptyset$, $X_k \subset X_i+X'_i$ (the equality does not occur because $X_i, X_k \in \mathcal{G}(X)$). It contradicts to the selection of the index i! Thus, all itemsets of $\mathcal{IS}(X)$ are non-repeatedly generated.
(b) **"$\subseteq$":** If $X' \in \lfloor X \rfloor$, by proposition 3.c, assume that i is the minimum index such that $X_i \in \mathcal{G}(X)$, $X''_i \subseteq X \backslash X_i$ and $X' = X_i+X''_i$. Let $X'_i = X''_i \cap X_U$, $X^{\sim} = X''_i \backslash X_U = X' \backslash X_U$, then $X'_i \subseteq X_{U,i}$, $X^{\sim} \subseteq X\_$ and $X' = X_i+X'_i+X^{\sim}$. Assume that there exists the index k such that $1 \leq k<i$, $X_k \in \mathcal{G}(X)$, $X_k \subset X_i+X'_i$. Then $X'=X_k+X''_k$, where $X''_k=X'_k+X^{\sim}$ and $X'_k=(X_i+X'_i) \backslash X_k \subseteq X \backslash X_k$, $X^{\sim} \subseteq X \backslash X_k$. Therefore, $X''_k \subseteq X \backslash X_k$: It is absurd!
"$\supseteq$": It is easy to prove. □

Example 2. Consider class [X], where X=aceg, $\mathcal{G}(X) = \{X_1=ae, X_2=ag\}$. Then, X_U=aeg, $X_{U,1}$=g, $X_{U,2}$=e and $X\_$=c. By theorem 2, $X'=aceg \in \mathcal{IS}(X)$ and $X''=cg \in \mathcal{N}^*(X)$ are uniquely generated: $X'=X_1+X'_1+X^{\sim}$ and $X''=X'_1+X^{\sim}$, where $X'_1=g \subseteq X_{U,1}$, $X^{\sim}=c \subseteq X\_$. By proposition 3.c, X' has two duplicate representations: X' = ae+cg = ag+ce. If the condition ($i>1: X_k \not\subset X_i+X'_i, \forall k: 1 \leq k<i$) is absent, duplicate X' is generated once again: $X'=X_2+X'_2+X^{\sim}$, where $X'_2=e \subseteq X_{U,2}$. Hence, all itemsets in $\lfloor X \rfloor$=[X]=$\mathcal{IS}$(X)={ae, aeg, aegc, aec, ag, agc} are non-repeatedly generated.

For two itemsets X, Y: $\emptyset \neq X, Y \subseteq \mathcal{A}$ and $X \cap Y = \emptyset$, we denote the set of itemsets (with constraint X) Y' contained in Y that the closure of Y'+X is the same with the one of Y+X as follows: $\lfloor Y \rfloor_X := \{Y' \subseteq Y \mid h(X+Y') = h(X+Y)\}$. Let $Y_{min,X}$ = Minimal$\{Y_k \equiv Z_k \backslash X \mid Z_k \in \mathcal{G}(X+Y)\}$ be the class containing all minimal sets (in term of relation "$\subseteq$") of $\{Z_k \backslash X \mid Z_k \in \mathcal{G}(X+Y)\}$, $Y_{X,U} = \bigcup_{Y_k \in Y_{min,X}} Y_k$, $Y_{X,U,k} = Y_{X,U} \backslash Y_k$, $Y_{X\_} = Y \backslash Y_{X,U}$, $\mathcal{IS}(Y)_X = \{Y'=Y_k+Y'_k+Y^{\sim} \mid Y_k \in Y_{min,X}, Y^{\sim} \subseteq Y_{X\_}, Y'_k \subseteq Y_{X,U,k}, k=1 \text{ or } (k>1 \text{ and } Y_j \not\subset Y_k+Y'_k, \forall j: 1 \leq j<k)\}$ and denote $\mathcal{FS}(Y)_X = \mathcal{IS}(Y)_X$, *if* $supp(X+Y) \geq s_0$. Theorem 3 shows unique representation of itemsets with constraint X.

Theorem 3 (*Unique representation of itemsets with constraint*). $\forall \emptyset \neq X, Y \subseteq \mathcal{A}$ and $X \cap Y = \emptyset$,

a) $Y' \in \lfloor Y \rfloor_X \Leftrightarrow \exists Z_0 \in \mathcal{G}(X+Y), Y'' \in \mathcal{N}(X+Y): Y'=(Z_0+Y'')\backslash X$.

b) $\lfloor Y \rfloor_X = \mathcal{IS}(Y)_X$. **c)** *All itemsets of* $\mathcal{IS}(Y)_X$ *are non-repeatedly derived.*

3 Structures of Association Rule Set

This section partitions rule set into disjoint rule classes using an equivalence relation based on the closure L of left-handed side and the one S of two-sided union of rules. Due to this relation, we only independently consider each class $\mathcal{AR}$(L, S) where (L, S) denotes a pair of two nested non-empty frequent closed itemsets (this notion is used in the rest of the paper). The rules in $\mathcal{AR}$(L, S) are all in the form: r:L'→S'\L', where ∅≠L'⊂S' and L'∈[L], S'∈[S]. Thus, they also have the same confidence.

The size of $\mathcal{AR}$(L, S) can be still big. Then, we can just mine the set of basic rules. When it is necessary, remaining rules can be generated from it. Pasquier et al. [7] and Zaki [11] considered mining basic rules in forms of minimal and most general. In [9], [10], they are also shown in the forms of minMax, minmin. By the viewpoint based on order relation, we see that there are different forms of basic rules such as MaxMax, Maxmin and min. This section proposes five order relations in order to obtain five pairs of sets of basic and consequence. For a given relation, the basic one contains minimal rules and the consequence one includes in the rules that can be deducted from them. In the rest, we show the explicit form of minimal rules according to relation min (min basic rules). Thanks to unique representations of itemsets, we propose how to derive non-repeatedly all remaining rules of $\mathcal{AR}$(L, S).

3.1 Partition of Association Rule Set by Equivalence Relation

Definition 4 (*Equivalence relation on association rule set*) [9]. Let $\sim_r$ be a binary relation on $\mathcal{ARS}$ defined as follows: ∀L',S',Ls,Ss⊆$\mathcal{A}$, ∅≠L'⊂S', ∅≠Ls⊂Ss, r:L'→S'\L', s:Ls→Ss\Ls: $s \sim_r r$ *iff* (*Ls* ∈ *[L'] and Ss* ∈ *[S']*).

Theorem 4 (*Partition of the association rule set*) [9]. Relation $\sim_r$ is an equivalence relation. It *partitions* $\mathcal{ARS}$ *into disjoint equivalence rule classes* $\mathcal{AR}$(L, S): $\mathcal{ARS} = \sum_{(L,S)} \mathcal{AR}(L,S)$. All rules in each class have the same support and confidence.

Then, we need only to independently investigate structure of each equivalence rule class $\mathcal{AR}$(L, S) = {r:L'→R' | L'∈[L], L'+R'∈[S]}. For example, $\mathcal{AR}$(ceg, aceg) = {e→acg, e→a, e→ac, e→ag, ec→ag, eg→ac, egc→a, ec→a, eg→a, g→ace, g→a, g→ac, g→ae, gc→ae, gc→a} and $\mathcal{AR}$(ceg, ceg) = {e→cg, e→c, e→g, g→ce, g→c, g→e, eg→c, ec→g, gc→e}. Obviously, $\mathcal{AR}$(L, L)=∅, ∀L∈$\mathcal{FCS}$: L∈$\mathcal{G}$(L). Then, when considering $\mathcal{AR}$(L, L), we always suppose that L∉$\mathcal{G}$(L).

3.2 Basic and Consequence Sets According to Different Order Relations

Definition 5 (*Order relations over each rule class*). Consider binary relations over $\mathcal{AR}$(L, S) defined by: $\forall r_j$: $L_j \rightarrow R_j \in \mathcal{AR}$(L, S), $S_j=L_j+R_j$, j=1,2:

a) $r_1 \prec_{minMax} r_2$ *iff* $(L_1 \subseteq L_2$ *and* $R_1 \supseteq R_2)$. **b)** $r_1 \prec_{minmin} r_2$ *iff* $(L_1 \subseteq L_2$ *and* $R_1 \subseteq R_2)$.

c) $r_1 \prec_{MaxMax} r_2$ *iff* $(L_1 \supseteq L_2$ *and* $S_1 \supseteq S_2)$. **d)** $r_1 \prec_{Maxmin} r_2$ *iff* $(L_1 \supseteq L_2$ *and* $R_1 \subseteq R_2)$.

e) $r_1 \prec_{min} r_2$ *iff* $(L_1 \supseteq L_2$ *and* $R_1 \supseteq R_2$, *if* $L \subset S)$; $(L_1 \subseteq L_2$ *and* $R_1 \supseteq R_2$, *if* $L = S)$.

It is easy to prove that the above binary relations are partial order relations over $\mathcal{AR}(L, S)$. Basic rule set $\mathcal{B}_{name}(L, S)$ contains minimal elements with respect to order relation $\prec_{name}$ (name $\in$ {minmin, minMax, Maxmin, MaxMax, min}). Corresponding consequence rule set is denoted as $\mathcal{C}_{name}(L, S) := \mathcal{AR}(L, S) \setminus \mathcal{B}_{name}(L, S)$. It contains all non-minimal rules, i.e., for every consequence rule r_c, there exists a basic rule r_b such that: $r_b \prec_{name} r_c$. We need to figure out: (1) *how to determine basic rules* r_b, and (2) *how to derive non-repeatedly all consequence rules* r_c *of them*?

The works related to relations of minmin and minMax have been considered by Pasquier et al. [7] and Zaki [11]. Overcoming the weaknesses in their results, in [10], [11] we indicated the better mining algorithms based on those relations. In the rest of the paper we consider structure of association rule set based on relation min (for relations of Maxmin and MaxMax, one can get the similar results). We will show that mining rules based on this relation is better than the ones based on minmin relation and minMax one in terms of reductions in number of basic rules and mining times.

3.3 Generating Min Basic Rules and Deriving Non-repeatedly Consequence Ones

Theorem 5 shows explicit form of min basic rules (basic rules determined by $\prec_{min}$).

Theorem 5 (*Explicit form of min basic rules*)**.** Let $L \subseteq S$. Then

a) $\mathcal{B}_{min}(L, S) = \{r_b: L \to S\backslash L\}$, *if* $L \subset S$.

b) $\mathcal{B}_{min}(L, L) = \{r_b: L_i \to L\backslash L_i \mid L_i \in \mathcal{G}(L)\}$, *if* $L \notin \mathcal{G}(L)$.

For example, $\mathcal{B}_{min}$(ceg, aceg) = {ceg→a} and $\mathcal{B}_{min}$(ceg, ceg) = {e→cg, g→ce}. Using closure-preserved property of eliminable itemset, from each rule r_0, for generating consequence rules that belong to the same class, we need only delete or move eliminable itemsets in both sides of r_0. These operators are used (by users) for generating preserved-confidence and non-repeated consequence rules from each basic rule. For different rules, however, corresponding consequence sets could include duplicate rules. To avoid the duplications, we replace generating consequence rule set from each rule with the one from each rule set containing rules of the same left-handed or right-handed side.

Definition 6 (*Operators for deriving non-repeatedly all consequence rules*).

a) L⊂S: $\mathcal{B}_{min\text{-}L\text{-}R}(L, S) = \{r_c: L' \to R' \mid R' \in \mathcal{FS}(S\backslash L)_L,\ L' \in \mathcal{FS}(L)$ *and* $(L' \subset L$ *or* $R' \subset S\backslash L)\}$,

$\mathcal{B}_{min\text{-}L+R}(L, S) = \{r_c: L'' \to R' + (L'\backslash L'') \mid L' \in \mathcal{FS}(L),\ R' \in \mathcal{FS}(S\backslash L)_L,\ L'' \in \mathcal{FS}(L')\backslash\{L'\}\}$.

b) L=S: $\mathcal{B}_{min\text{-}R+L}(L, L) = \{r_c: L_i + R'' \to R'\backslash R'' \mid L_i \in \mathcal{G}(L),\ \varnothing \neq R' \subseteq L\backslash L_i,$

$\varnothing \neq R'' \subset R',\ i=1$ *or* $(i>1: L_k \not\subset L_i + R'',\ \forall k: 1 \le k < i)\}$,

$\mathcal{B}_{min\text{-}R}(L, L) = \{r_c: L_i \to R' \mid \varnothing \neq R' \subset L\backslash L_i,\ L_i \in \mathcal{G}(L)\}$.

To illustrate the definition, considering (L, S) = (ceg, aceg), we have: $\mathcal{B}_{min\text{-}L\text{-}R}$(L, S) = {e→a, ce→a, eg→a, g→a, cg→a} and $\mathcal{B}_{min\text{-}L+R}$(L, S) = {e→ac, e→ag, g→ae, e→acg, ce→ag, eg→ac, g→ace, cg→ae, g→ac}. Theorem 6 shows structure of the set of all min consequence rules that are non-repeatedly derived from min basic ones.

Theorem 6 (*Deriving non-repeatedly all min consequence rules*).

a) L ⊂ S: i) All *rules in* $\mathcal{B}_{min\text{-}L\text{-}R}(L, S)$, $\mathcal{B}_{min\text{-}L+R}(L, S)$ *are non-repeatedly generated.*

ii) All *rules in* $\mathcal{B}_{min\text{-}L\text{-}R}(L, S)$, $\mathcal{B}_{min\text{-}L+R}(L, S)$, $\mathcal{B}_{min}(L, S)$ *are totally different.*

iii) $\mathcal{C}_{min}(L, S) = \mathcal{B}_{min\text{-}L\text{-}R}(L, S) + \mathcal{B}_{min\text{-}L+R}(L, S)$.

b) L = S: i) All *rules in* $\mathcal{B}_{min\text{-}R}(L, L)$, $\mathcal{B}_{min\text{-}R+L}(L, L)$ *are non-repeatedly generated.*

ii) All *rules in* $\mathcal{B}_{min\text{-}R}(L, L)$, $\mathcal{B}_{min\text{-}R+L}(L, L)$, $\mathcal{B}_{min}(L, S)$ *are totally different.*

iii) $\mathcal{C}_{min}(L, L) = \mathcal{B}_{min\text{-}R}(L, L) + \mathcal{B}_{min\text{-}R+L}(L, L)$.

Proof: (a) *"L ⊂ S":* (i), (ii): It is easy to prove them. (iii) *"⊇":* For every r_c:L'→R'∈$\mathcal{B}_{min\text{-}L\text{-}R}$(L, S), where R'∈$\mathcal{FS}$(S\L)$_L$, L'∈$\mathcal{FS}$(L) and (L'⊂L or R'⊂S\L), then R'∈$\mathcal{FS}$(S\L)$_{L'}$, h(L'+R')=S and r_c∈ $\mathcal{C}_{min}$(L, S). For every r_c:L''→R'+(L'\L'') ∈$\mathcal{B}_{min\text{-}L+R}$(L, S), where L'∈$\mathcal{FS}$(L), L''∈$\mathcal{FS}$(L')\{L'}, R'∈$\mathcal{FS}$(S\L)$_L$, we have R'∈$\mathcal{FS}$(S\L)$_{L'}$, h(L''+R'+(L'\L'')) = h(L'+R') = h(L+R') = S and r_c∈$\mathcal{C}_{min}$(L, S). *"⊆":* For every r_c:L''→R''∈$\mathcal{C}_{min}$(L, S): h(L''+R'')=S, h(L'')=L and (L''⊂L or R''≠S\L), we have R''=R'$_0$+R', where R'$_0$=R''∩L, R'=R''\L, L'=L''+R'$_0$⊇L'', L''+R''⊆ L∪R''=L+R'⊆S. Then R'$_0$=L'\L'', h(L'')=h(L')=L and h(L+R')=S, i.e., L'∈$\mathcal{FS}$(L), L''∈$\mathcal{FS}$(L')⊆$\mathcal{FS}$(L), R'∈$\mathcal{FS}$(S\L)$_L$ and r_c:L''→R'+(L'\L''). If R'$_0$=(L'\L'') = ∅, then (R'=R''≠S\L or L''⊂L). Thus r_c:L''→R'∈$\mathcal{B}_{min\text{-}L\text{-}R}$(L, S). If R'$_0$=(L'\L'') ≠ ∅, then L''∈$\mathcal{FS}$(L')\{L'}, r_c:L''→ R'+(L'\L'')∈$\mathcal{B}_{min\text{-}L+R}$(L, S).

(b) It is similarly proved. □

Based on theorems 5 and 6, the fast algorithms for mining (directly) min basic rules and generating (non-repeatedly) all consequence ones are obtained.

4 Experimental results

Four benchmark databases in [13] are used during these experiments: Pumsb contains 49046 transactions, 7117 items (P, 49046, 7117); Mushroom (M), 8124, 119; Connect (C), 67557, 129; Pumsb* (P*), 49046, 7117. Experiments will show the efficiency of mining rules using relation min with the ones using relations of minmin and minMax.

Consider Table 1 where: the number of all rules is shown in column #AR and the percent ratios of the cardinalities of basic sets of min, minmin and minMax to #AR are in turn shown in columns of B_m, B_{mm} and B_{mM}. It shows that *the cardinality of min basis is smaller than the ones of bases of mm and mM*. For the present experiments, the reduction in the number of basic rules ranges from a factor of 1.0 to 2.1 times. Table 2 figures out that *the times for mining sets of basic rule and all rules based on relation min* (in columns TB_m, T_m) *are less than the ones based on relations of mm* (in columns TB_{mm}, T_{mm}) *and mM* (in columns TB_{mM}, T_{mM}). The reduction in the time for mining *basic* rules ranges from a factor of 1.1 to 4.4 times. The one in the time for mining *all* rules ranges from 1.0 to 1.9 times. Figures of 2 and 3 show the effect of minimum confidence on the mining times. The time to mine *basic* rules using relation min is less much than the one using relation mm. In comparison with relation mM, it is noteless.

However, when mining *all* rules, the conversion comes. Hence, we can conclude that *mining rules using relation min is better in terms of reductions in number of basic rules as well as in running times.*

Table 1. The sizes of basic sets (MS = min sup, MC = min confidence: %).

DB (1)	MS=MC (2)	#AR	B_m (%)	B_{mm} (%)	B_{mm}/B_m	B_{mM} (%)	B_{mM}/B_m
C	90	3640704	8.77	8.92	*1.0*	8.77	*1.0*
	80	326527774	1.21	1.23	*1.0*	1.21	*1.0*
M	20	19191656	0.15	0.31	*2.1*	0.18	*1.2*
	15	34505370	0.18	0.33	*1.8*	0.21	*1.2*
P	85	1408950	31.55	51.64	*1.6*	41.21	*1.3*
	80	28267480	13.19	27.47	*2.1*	19.46	*1.5*
P*	40	5659536	2.99	4.68	*1.6*	3.49	*1.2*
	30	311729540	0.88	1.67	*1.9*	1.14	*1.3*

Table 2. The times for mining basic and all rules using relations of min, minmin and minMax.

(1)-(2)	TB_m	TB_{mm}	TB_{mm} / TB_m	TB_{mM}	TB_{mM} / TB_m	T_m	T_{mm}	T_{mm} / T_m	T_{mM}	T_{mM} / T_m
C-90	1.31	1.66	*1.3*	1.41	*1.1*	49	53	*1.1*	53	*1.1*
C-80	18.16	21.50	*1.2*	19.53	*1.1*	2674	2878	*1.1*	3612	*1.4*
M-20	0.09	0.41	*4.4*	0.14	*1.5*	88	91	*1.0*	164	*1.9*
M-15	0.25	0.72	*2.9*	0.31	*1.2*	158	163	*1.0*	265	*1.7*
P-85	1.70	4.42	*2.6*	2.22	*1.3*	40	54	*1.4*	44	*1.1*
P-80	15.36	56.13	*3.7*	22.45	*1.5*	504	738	*1.5*	646	*1.3*
P*-40	0.68	1.59	*2.3*	0.84	*1.2*	51	57	*1.1*	71	*1.4*
P*-30	12.45	36.70	*2.9*	16.38	*1.3*	2597	2511	*1.0*	3625	*1.4*

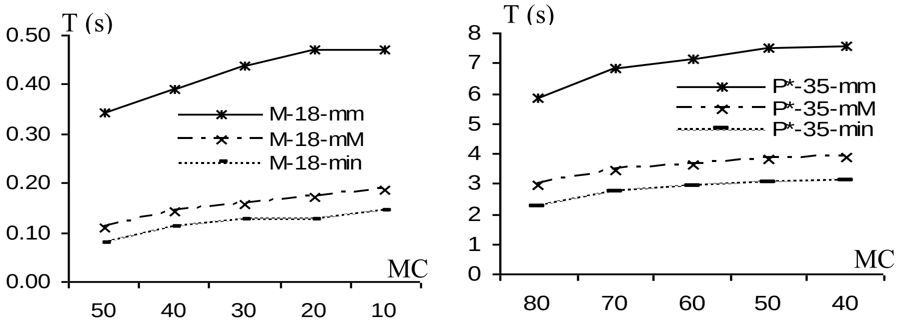


Fig. 2. The times for mining *basic* sets: M with MS = 18 and P* with MS = 35.

5 Conclusion

An equivalence relation partitions the class of itemsets into disjoint classes. Each class contains itemsets of the same closure so the same support. Their unique representations are figured out. The association rule set is also partitioned into disjoint rule classes by an appropriate equivalence relation. Each rule class contains all association rules having the same closures of left-handed sides and two-sided unions, so the same confidence. Each rule class splits into different sets of basic and consequence based on different order relations. According to relation min, the explicit form of basic rules is shown. Operators for deducing non-repeatedly all remaining rules are also obtained. The paper shows that mining association rules based on order relation min is better the ones based on relations of minmin and minMax.

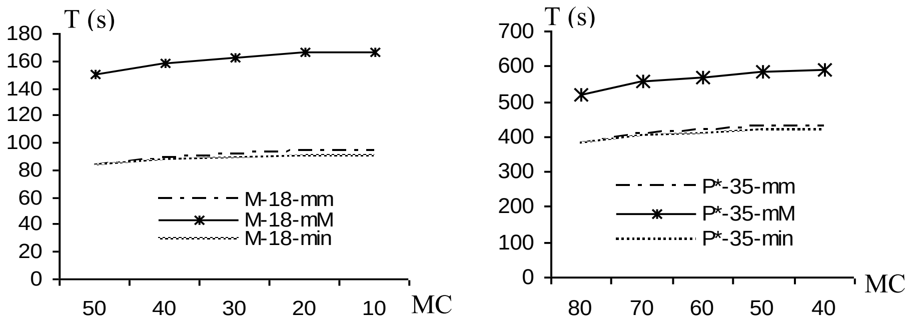


Fig. 3. The times for mining all rules based on relations of min, minmin and minMax: M with MS = 18 and P* with MS = 35.

References

1. Agrawal, R., Imielinski, T., Swami, N.: Mining association rules between sets of items in large databases. In: Proceedings of the ACM SIGMOID, pp. 207-216 (1993).
2. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules. In: Proceedings of the 20th International Conference on Very Large Data Bases, pp. 478-499 (1994).
3. Bao, H.T.: An approach to concept formation based on formal concept analysis. IEICE Trans. Information and systems, vol. E78-D, no. 5 (1995).
4. Duquenne, V., Guigues, J.-L.: Famille minimale d’implications informatives re’sultant d’un tableau de donne´es binaires. Math. et Sci. Hum., vol. 24, no. 95, pp. 5–18 (1986).
5. Feller, W.: An introduction to probability theory and its applications. New York. John Wiley & sons. Inc., vol. I, Chapman & Hall. Ltd. London (1950).
6. Luxenburger, M.: Implications partielles dans un contexte. Math., Inf. et Sci. Hum., vol. 29, no. 113, pp. 35-55 (1991).
7. Pasquier, N., Taouil, R., Bastide, Y., Stumme, G., Lakhal, L.: Generating a condensed representation for association rules. J. of Intelligent Information Systems, vol. 24, no. 1, pp. 29-60 (2005).
8. Pei, J., Han, J., Mao, R.: CLOSET: An Efficient Algorithm for Mining Frequent Closed Itemsets. In: Proceedings of the DMKDWorkshop on Research Issues in Data Mining and Knowledge Discovery, pp. 21–30 (2000).
9. Tin, T., Anh, T.: Structure of set of association rules based on concept lattice. SCI 283, Adv. in Intelligent. Information and Database System, pp. 217-227, Springer-Verlag (2010).
10. Tin, T., Anh, T., Thong, T.: Structure of association rule set based on min-min basic rules. In: Proceedings of the 2010 IEEE-RIVF International Conference on Computing and Communication Technologies, pp. 83-88 (2010).
11. Zaki, M.J.: Mining non-redundant association rules. Data mining and knowledge discovery, no 9, pp. 223-248 (2004).
12. Zaki, M.J., Hsiao, C-J.: Efficient algorithms for mining closed itemsets and their lattice structure. IEEE Trans. Knowledge and data engineering, vol. 17, no. 4, pp. 462-478 (2005).
13. Frequent Itemset Mining Dataset Repository, http://fimi.cs.helsinki.fi/data/ (2009).