On the Mining of Substitution Rules for Statistically Dependent Items

Wei-Guang Teng, Ming-Jyh Hsieh and Ming-Syan Chen
Department of Electrical Engineering
National Taiwan University
Taipei, Taiwan, ROC
E-mail: {eev, mintz}@arbor.ee.ntu.edu.tw, mschen@cc.ee.ntu.edu.tw

Abstract

In this paper, a new mining capability, called mining of substitution rules, is explored. A substitution refers to the choice made by a customer to replace the purchase of some items with that of others. The process of mining substitution rules can be decomposed into two procedures. The first procedure is to identify concrete itemsets among a large number of frequent itemsets, where a concrete itemset is a frequent itemset whose items are statistically dependent. The second procedure is then on the substitution rule generation. Two concrete itemsets X and Y form a substitution rule, denoted by X$\triangleright$Y to mean that X is a substitute for Y, if and only if (1) X and Y are negatively correlated and (2) the negative association rule X$\rightarrow \overline{Y}$ exists. In this paper, we derive theoretical properties for the model of substitution rule mining. Then, in light of these properties, algorithm SRM (standing for substitution rule mining) is designed and implemented to discover the substitution rules efficiently while attaining good statistical significance. Empirical studies are performed to evaluate the performance of algorithm SRM proposed. It is shown that algorithm SRM produces substitution rules of very high quality.

1. Introduction

Various data mining capabilities have been explored in the literature [5, 7, 14]. Among them, the one receiving a significant amount of research attention is on mining association rules [2]. Given a database of sales transactions, the goal of mining an association rule is to discover the relationship that the presence of some items in a transaction will imply the presence of other items in the same transaction. Note that in addition to the association rules, the data in a transaction database also possesses some other consumer purchase behaviors. Specifically, it is important to understand the choice made by consumers, which, corresponding to the purchase of some items instead of that of others, is termed *substitution* in this paper. For example, in a grocery store, the purchase of apples may be substituted for that of pears. Intuitively, the substitutes are of analogous properties and therefore are often possible choices for customers. However, in some cases, the substitutes could be formed due to purchasing purposes. For example, the purchase of roses may be substituted for that of a Teddy bear and a box of chocolates. The mining of substitution rules in a transaction database, same as that of association rules, will lead to very valuable knowledge in various aspects, including market prediction, user behavior analysis and decision support, to name a few. Despite of its importance, the mining of substitution rules, unlike that of association rules, has been little explored in the literature.

Mining negative association rules of the form $X \rightarrow \overline{Y}$, where $\overline{Y}$ means the absence of itemset Y, is useful for the mining of substitutive itemsets, since in a negative association rule, the presence of the antecedent itemset implies the absence of the positive counterpart of the consequent itemset, meaning that X could be a substitute for Y. It is noted that some efforts were elaborated upon the mining of negative association rules. In [15], the taxonomy of items is introduced and a heuristic of using similarity among items in the same category is utilized to facilitate the mining of negative association rules. On the other hand, a constraint-based approach is adopted in [3]. Notice, however, that in the negative association rule mining, the dependency of items in an itemset is not considered since the itemset frequency is the only measurement when generating frequent itemsets. In contrast, to discover substitution rules, one should first determine possible itemsets which could be choices for customers. The purchasing frequency, i.e., support of an itemset, is not adequate to identify these possible substitutes. The dependency of items has to be examined to identify concrete sets of items which are really purchased together by customers. Specifically, a frequent itemset whose items are statistically dependent is called a *concrete* itemset in this paper. Note that if a frequent itemset is not concrete, that itemset is likely to consist of frequent items which, though appearing together frequently due to their high individual occurrence counts, do not possess adequate depen-

0-7695-1754-4/02 $17.00 © 2002 IEEE

dency among themselves and are thus of little practical implication to be used as a whole in *either* the antecedent *or* the consequent of a substitution rule. In addition, the negative correlation of two itemsets should be verified if these two itemsets are considered to be substitutes for each other. Without considering these aspects, the mining of negative association rules is not applicable to the mining of substitution rules.

Consequently, we develop in this paper a new model of mining substitution rules. The process of mining substitution rules can be decomposed into two procedures. The first procedure is to identify concrete itemsets among a large number of frequent itemsets. The second procedure is on the substitution rule generation. Two concrete itemsets X and Y form a substitution rule, denoted by $X \triangleright Y$ to mean that X is a substitute for Y, if and only if (1) X and Y are negatively correlated and (2) the negative association rule $X \rightarrow \overline{Y}$ exists. Without loss of generality, the chi-square test [8] is employed to identify concrete itemsets by statistically evaluating the dependency among items in individual itemsets. Moreover, the Pearson product moment correlation coefficient [8, 11] is utilized to measure the correlation between two itemsets. Explicitly, we derive theoretical properties for the model of mining substitution rules. Then, in light of these properties, algorithm SRM (standing for substitution rule mining) is designed and implemented to discover the substitution rules efficiently while attaining good statistical significance. For comparison purposes, a companion method which is extended from algorithm Apriori, called algorithm Apriori-Dual, is also implemented.

Extensive experimental studies have been conducted to provide many insights into algorithm SRM proposed. The quality of substitution rules in terms of statistical measurements is also evaluated. It is shown by experiments that algorithm SRM significantly outperforms algorithm Apriori-Dual. It is noted that algorithm SRM produces substitution rules of very high quality as measured by the correlation and the violation ratio [1]. The advantage of SRM is even more prominent when the transaction database is sparser.

The rest of the paper is organized as follows. The framework of mining negative association rules is explored and the model of substitution rule mining is presented in Section 2. Algorithm SRM and an illustrative example are described in details in Section 3. Several experiments are conducted in Section 4. This paper concludes with Section 5.

2. Model of Substitution Rule Mining

To facilitate our discussion, we shall first review the framework of negative association rules mining in Section 2.1. The model of substitution rule mining is then presented in Section 2.2.

2.1. Mining of Negative Association Rules

Same as in most prior work on mining association rules [2, 5], an itemset is a set containing one or many items. The support of an itemset X, denoted by S_X, is the fraction of transactions containing X in the whole dataset. The itemsets which meet the minimum support constraint are called frequent itemsets or large itemsets [5]. An association rule is an implication of the form $X \rightarrow Y$ with $X \cap Y = \varnothing$, where X and Y are both frequent itemsets. The support of the rule $X \rightarrow Y$, i.e., $Sup(X \rightarrow Y)$, is $S_{X \cup Y}$, and the confidence of the rule $X \rightarrow Y$, i.e., $Conf(X \rightarrow Y)$, is $\frac{S_{X \cup Y}}{S_X}$. Given a large database of transactions, the goal of mining association rules is to generate all rules that satisfy the user-specified constraints of minimum support and the minimum confidence, i.e., $Sup(X \rightarrow Y) \geq MinSup$ and $Conf(X \rightarrow Y) \geq MinConf$.

Definition 1: An itemset X is *positive* if and only if it contains no complement items, i.e., $X=\{i_1, i_2, ..., i_k\}$ where i_j is an item for $1 \leq j \leq k$. On the other hand, the *negative* itemset is an itemset containing one or more complement items. If a negative itemset is composed by complement items only, i.e., $\{\overline{i_1}, \overline{i_2}, ..., \overline{i_k}\}$, then this itemset is *pure negative* and can be denoted by $\overline{X}$.

A negative association rule refers to an association rule of which either the antecedent itemset, the consequent itemset, or both are negative. An example of mining negative itemsets through a naive approach is given below for illustrative purposes.

Example 1: Consider the transaction database in Table 1(a). We first append the complement items to each transaction as shown in Table 1(b). For example, the transaction with TID=1, i.e., {a, c, d} in Table 1(a), becomes {a, $\overline{b}$, c, d, $\overline{e}$, $\overline{f}$} in Table 1(b). The resulting database in Table 1(b) is the input to the itemset generation algorithm.

Table 1. (a) The original transaction database; (b) After complement items are added

TID	Items
1	a, c, d
2	b, c
3	c
4	a, b, f
5	a, c, d
6	e
7	b, f
8	b, c, f
9	a, b, e
10	a, d

(a)

$\Longrightarrow$

TID	Items
1	a, $\overline{b}$, c, d, $\overline{e}$, $\overline{f}$
2	$\overline{a}$, b, c, $\overline{d}$, $\overline{e}$, $\overline{f}$
3	$\overline{a}$, $\overline{b}$, c, $\overline{d}$, $\overline{e}$, $\overline{f}$
4	a, $\overline{b}$, $\overline{c}$, $\overline{d}$, $\overline{e}$, f
5	a, $\overline{b}$, c, d, $\overline{e}$, $\overline{f}$
6	$\overline{a}$, $\overline{b}$, $\overline{c}$, $\overline{d}$, e, $\overline{f}$
7	$\overline{a}$, b, $\overline{c}$, $\overline{d}$, $\overline{e}$, f
8	$\overline{a}$, b, c, $\overline{d}$, $\overline{e}$, f
9	a, b, $\overline{c}$, $\overline{d}$, e, $\overline{f}$
10	a, $\overline{b}$, $\overline{c}$, d, $\overline{e}$, $\overline{f}$

(b)

Given that MinSup=0.3, all the frequent itemsets can then be discovered from Table 1(b) as summarized in Table 2. Note that we are only interested in those complement items whose positive counterparts are frequent for market basket analysis. As a result, the complement item $\overline{e}$ is not shown in Table 2, since the item e is not frequent.

Table 2. Frequent itemsets generated from the database in Table 1(b) (MinSup=0.3)

I_1	S_I
a	0.5
b	0.5
c	0.5
d	0.3
f	0.3
$\bar{a}$	0.5
$\bar{b}$	0.5
$\bar{c}$	0.5
$\bar{d}$	0.7
$\bar{f}$	0.7

I_2	S_I
a, d	0.3
a, $\bar{b}$	0.3
a, $\bar{c}$	0.3
a, $\bar{f}$	0.4
b, f	0.3
b, $\bar{a}$	0.3
b, $\bar{c}$	0.3
b, $\bar{d}$	0.5
c, $\bar{a}$	0.3
c, $\bar{b}$	0.3
c, $\bar{d}$	0.3
c, $\bar{f}$	0.4
d, $\bar{b}$	0.3

I_2	S_I
d, $\bar{f}$	0.3
f, $\bar{d}$	0.3
$\bar{a}$, $\bar{d}$	0.5
$\bar{a}$, $\bar{f}$	0.3
$\bar{b}$, $\bar{f}$	0.5
$\bar{c}$, $\bar{d}$	0.4
$\bar{c}$, f	0.3
$\bar{d}$, $\bar{f}$	0.4

I_3	S_I
a, d, $\bar{b}$	0.3
a, d, $\bar{f}$	0.3
a, $\bar{b}$, $\bar{f}$	0.3
b, $\bar{f}$, $\bar{d}$	0.3
b, $\bar{a}$, $\bar{d}$	0.3
b, $\bar{c}$, $\bar{d}$	0.3
c, $\bar{a}$, $\bar{d}$	0.3
c, $\bar{b}$, $\bar{f}$	0.3
d, $\bar{b}$, $\bar{f}$	0.3
$\bar{a}$, $\bar{d}$, $\bar{f}$	0.3

I_4	S_I
a, $\bar{d}$, $\bar{b}$, $\bar{f}$	0.3

Clearly, with this straightforward addition of complement items into the database, the mining of negative association rules can be performed by directly using methods devised for mining conventional association rules. However, this benefit may not be able to justify several drawbacks of this naive approach in practice. First, excessive storage space is required to store complement items and also the additional itemsets resulted. Next, many of the frequent itemsets generated are composed of complement items only. These itemsets are usually of little use in real applications. Finally, extra database scans are needed for the mining process. In real applications, this naive approach will suffer a prolonged execution time and make mining of negative association rules an infeasible task.

Once the negative itemsets are generated, one can discover all negative association rules in a straightforward manner. For two itemsets X and Y where $Y \subset X$, the rule $Y \rightarrow (X-Y)$ is output if the required MinConf is satisfied. However, for our purpose of discovering substitution rules, two positive itemsets are required to form a substitute pair. Thus, the algorithm Apriori-Dual, i.e., a companion method extended from algorithm Apriori, is proposed to generate only rules whose antecedent is positive and consequent is pure negative, i.e., $X \rightarrow \overline{Y}$ where X and Y are positive itemsets.

Algorithm Apriori-Dual

// Input: MinSup and MinConf
// Procedure of generating all frequent itemsets, including
// the negative ones
1. append the complement items whose positive counterpart is not original present to each transaction;
2. generate the set of frequent (positive and negative) items, i.e., L_1;
3. remove the negative items whose positive counterpart is not frequent from L_1;
4. **for** $k \geq 2$ **do{**
5. generate the candidate set of k-itemsets from L_{k-1}, i.e., $C_k = L_{k-1} \bowtie L_{k-1}$;
6. **if** (C_k is empty) **then break**;
7. scan the transactions to calculate supports of all candidate k-itemsets;
8. $L_k = \{ c \in C_k \mid S_c \geq \text{MinSup} \}$;
9. **}**

// Procedure of negative association rule generation
10. **foreach** negative itemset X in L_ks **do{**
11. let $\overline{Y}$ be the largest pure negative itemset that $\overline{Y} \subset X$;
12. **if** $(X-\overline{Y})$ is not an empty set // $(X-\overline{Y})$ is positive
13. **if** $(\text{Conf}((X-\overline{Y}) \rightarrow \overline{Y}) \geq \text{MinConf})$
14. output the rule $(X-\overline{Y}) \rightarrow \overline{Y}$;
15. **}**

As pointed out before, the problem formulation of the negative association rule mining is different from that of the substitution rule mining. In addition, since complement items are appended to the original transaction database, the computation cost of algorithm Apriori-Dual is, as conformed by our experimental results, very high. These drawbacks reduce the practicability of using algorithm Apriori-Dual for identifying substitute itemsets. Consequently, a new algorithm for mining substitution rules is proposed and will be described in later sections.

2.2. Mining of Substitution Rules

As mentioned before, the process of mining substitution rules can be decomposed into two procedures. The first one is to identify concrete itemsets among large amounts of itemsets. The second one is on the substitution rule generation. The chi-square test [8] is employed to identify concrete itemsets by statistically evaluating the dependency among items in individual itemsets. Also, the Pearson product moment correlation coefficient [8, 11] is utilized to measure the correlation between two itemsets.

2.2.1. Identification of Concrete Itemsets. Concrete itemsets are those possible itemsets which could be choices for customers with some purchasing purposes. To qualify an itemset as a concrete one, not only the purchasing frequency, i.e., support of an itemset, but also the dependency of items has to be examined to declare that these items are *purposely* purchased together by customers. One common approach is to evaluate the dependence among items in an itemset by the chi-square test [4, 10, 13]. Specifically, the chi-square value for an itemset can be derived in terms of supports and expected supports of its corresponding itemsets, as stated in Theorem 1 below.

Theorem 1: Let $X=\{x_1, x_2, ..., x_k\}$ be a positive k-itemset, the chi-square value for X is computed as

$$Chi(X) = n \times \left[\left(\sum_{I \in \{Y \mid Y^+ = X\}} \frac{S_I^2}{E_I} \right) - 1 \right],$$

where n is the number of total transactions, Y^+ denotes the positive itemset where all complement items in itemset Y are replaced by their positive counterparts, e.g., $\{a,\bar{b},\bar{c}\}^+=\{a,b,c\}$ where a, b and c are positive items, and $E_I = \prod_{i \in I} S_i$ is the expected support of I.

Proof: Since the itemset X is of size k and the presence of an item in each transaction is 0-1 valued, a corresponding $2\times2\times\cdots\times2$ k-dimensional contingency table can be constructed. Each dimension of this contigency table corresponds to the presence of an item, i.e., $x_i \in X$, in each transaction. The values of these 2^k cells are exactly the supports of itemsets $\{\overline{x_1}, \overline{x_2}, ..., \overline{x_k}\}$, $\{\overline{x_1}, \overline{x_2}, ..., x_k\}$, ..., $\{x_1, x_2, ..., x_k\}$ and the summation of these values is n, i.e., the number of transactions. Also, the corresponding itemsets above can be formulated as $\{Y \mid Y^+=X\}$. The chi-square value is then computed by

$$Chi(X) = \sum_{c\in cells} \frac{(O_c - E_c)^2}{E_c},$$

where O_c is the observed value and E_c is the expected value of cell c in the contingency table. For any itemset I and its corresponding cell c that $I^+=X$, we have

$$O_c = n \times S_I \text{ and } E_c = n \times \prod_{i\in I} S_i.$$

With some algebraic manipulations, we have

$$\begin{aligned} Chi(X) &= \sum_{c\in cells} \frac{O_c^2 - 2O_cE_c + E_c^2}{E_c} \\ &= \sum_{c\in cells} \frac{O_c^2}{E_c} - 2\sum_{c\in cells} O_c + \sum_{c\in cells} E_c \\ &= \left(\sum_{c\in cells} \frac{O_c^2}{E_c}\right) - n \quad \left(\because \sum_{c\in cells} O_c = \sum_{c\in cells} E_c = n\right) \\ &= \left(\sum_{I\in\{Y\mid Y^+=X\}} \frac{n^2\times {S_I}^2}{n\times\prod_{i\in I} S_i}\right) - n \\ &= n\times\left[\left(\sum_{I\in\{Y\mid Y^+=X\}} \frac{{S_I}^2}{E_I}\right) - 1\right]. \end{aligned}$$ **Q.E.D.**

To utilize the chi-square test to verify whether the occurrences of given items are dependent, two contradictive hypotheses are made

$\begin{cases} H_0: \text{The occurrences of all items}(x_1 \sim x_k) \text{ are independent,} \\ H_1: H_0 \text{ is rejected.} \end{cases}$

With Theorem 1, to declare the dependency among items in an itemset X, or to support hypothesis H_1, the chi-square value for X is required to be no less than a threshold, i.e., $Chi(X) \geq \chi^2_{df(X),\alpha}$.

In addition, it follows from advanced statistics and information theory [9] that corresponding degree of freedom for this test can be denoted by

$$df(X) = \prod_i c(v_i) - \sum_i [c(v_i) - 1] - 1 = 2^k - k - 1$$

where $c(v_i)$ is the number of categories in dimension i, i.e., $c(v_i)=2$ for all dimensions since the presence of an item in each transaction is 0-1 valued.

We comment that the results derived in Theorem 1 are essential for our mining of substitution rules and are not subsumed by the work in [4]. In [4], it was stated that "if S is correlated with significance level α, any superset of S is also correlated with significance level α." From Theorem 1 in [4], one may mistakenly assume that the chi-square test for itemsets at a given significance level is upward closed (as stated in Theorem 1 in [4].) However, as also noted in [6], this upward closure property is not fully correct. Explicitly, the first statement of the proof of Theorem 1 in [4] "The key observation in proving this is that not matter what k is, the chi-squared statistic has only one degree of freedom" which its subsequent proof is based upon is not true, thereby leading to incorrect conclusions. A counterexample of Theorem 1 in [4] is given in the Appendix for interested readers. Specifically, as opposed to what Theorem 1 in [4] suggests, all correlated itemsets, rather than only *minimally* correlated ones, should be discovered. This in turn justifies the necessity of our development of the process to identify concrete itemsets in this paper.

Without loss of generality, a *concrete* itemset is thus defined to be a frequent itemset which is positively correlated given a significance level α (usually $\alpha = 0.05$), if it contains more than one item. Note that the significance level of a concrete itemset is expected to be at least no less than that of its subsets. For example, if the itemset {flashlight, battery} has a quite high chi-quare value, then its superset, e.g., {flashlight, battery, pencil), could still have a high chi-square value ($>\chi^2_{df(X),\alpha}$) even though pencil is not so correlated with the other items.

Definition 2: A positive frequent itemset $X=\{x_1, x_2, ..., x_k\}$ is called a *concrete* itemset, if and only if
(1) k=1, or
(2) k≥2,

$$S_X > \prod_{x_i\in X} S_{x_i} \text{ and } Chi(X) \geq \chi^2_{df(X),\alpha},$$

where $\prod_{x_i\in X} S_{x_i}$ corresponds to the *expected* support for itemset X, and $\chi^2_{df(X),\alpha}$ is the value of chi-square distribution with degree of freedom $df(X)$ at probability α. Note that $S_X > \prod_{x_i\in X} S_{x_i}$ is required to ensure that all $x_i \in X$ are positively correlated.

The value of $\chi^2_{df(X),\alpha}$ can be obtained by the table look-up. As mentioned earlier, the usual value $\alpha = 0.05$ is used in this study for statistical significance. Considering itemset {a, d} in Table 2 for example, $S_{ad} = 0.3 > S_a \times S_d = 0.6 \times 0.3$. Also, the chi-square value for {a, d} is

$$\begin{aligned} Chi(\{a,d\}) &= n\times\left[\left(\frac{{S_{\bar a\bar d}}^2}{E_{\bar a\bar d}} + \frac{{S_{\bar a d}}^2}{E_{\bar a d}} + \frac{{S_{a\bar d}}^2}{E_{a\bar d}} + \frac{{S_{ad}}^2}{E_{ad}}\right) - 1\right] \\ &= 10\times\left(\frac{0.5^2}{0.5\times0.7} + 0 + \frac{0.2^2}{0.5\times0.7} + \frac{0.3^2}{0.5\times0.3} - 1\right) \\ &= 4.29 > \chi^2_{1,0.05} = 3.84. \end{aligned}$$

Thus, {a, d} is a concrete itemset.

2.2.2. Testing of Negative Correlation. To evaluate the correlation between two *concrete* itemsets, we adopt the measurement of Pearson product moment correlation coefficient [8]. Theorem 2 states that the correlation coefficient of two itemsets can be determined by their supports.

Theorem 2: Let X and Y be two itemsets with $X \cap Y = \emptyset$. The correlation coefficient of X and Y can be formulated in terms of their supports. Explicitly,

$$\rho(X,Y) = \frac{Cov(X,Y)}{\sqrt{Var(X) \cdot Var(Y)}} = \frac{S_{XY} - S_X \cdot S_Y}{\sqrt{S_X(1-S_X)S_Y(1-S_Y)}}.$$

Proof: Since variables corresponding to occurrence of items in a transaction database are all 0-1 valued, it follows that

$$EX = EX^2 = S_X,\ EY = EY^2 = S_Y \text{ and } E(XY) = S_{XY}$$

where E stands for the expected value. According to the definition of correlation coefficient, we have

$$\begin{aligned}\rho(X,Y) &= \frac{Cov(X,Y)}{\sqrt{Var(X) \cdot Var(Y)}} \\ &= \frac{E[(X-EX)(Y-EY)]}{\sqrt{E[(X-EX)^2] \cdot E[(Y-EY)^2]}} \\ &= \frac{E(XY) - (EX)(EY)}{\sqrt{[EX^2 - (EX)^2] \cdot [EY^2 - (EY)^2]}} \\ &= \frac{S_{XY} - S_X \cdot S_Y}{\sqrt{[S_X - (S_X)^2][S_Y - (S_Y)^2]}} \\ &= \frac{S_{XY} - S_X \cdot S_Y}{\sqrt{S_X(1-S_X)S_Y(1-S_Y)}}\end{aligned}$$ **Q.E.D.**

Note that when both variables to be correlated are binary as in this case, we may use the phi coefficient of correlation as stated in [11, 12] instead of $\rho(X,Y)$ in Theorem 2. However, the phi coefficient of correlation and the Pearson product moment correlation coefficient are in fact algebraically equivalent and give identical numerical results. Therefore, for notational simplicity, we employ $\rho(X,Y)$ to express the results of Theorem 2.

Consequently, a substitution rule can be defined as below.

Definition 3: Given two itemsets X and Y and $X \cap Y = \emptyset$, X is a substitute for Y, denoted by $X \triangleright Y$, if and only if
(1) both X and Y are concrete,
(2) X and Y are negatively correlated, i.e., $\rho(X,Y) < -\rho_{\min} \leq 0$ (usually $\rho_{\min}=0$ for simplicity), and
(3) the negative association rule $X \rightarrow \overline{Y}$ is valid, i.e., $Sup(X \rightarrow \overline{Y}) \geq MinSup$ and $Conf(X \rightarrow \overline{Y}) \geq MinConf$.

3. SRM: Substitution Rule Mining

Given the definitions of concrete itemsets and substitution rules, a detailed description of algorithm SRM for mining substitution rules is given.

Algorithm SRM

// Input: MinSup, MinConf, and $\rho_{\min}$
// Procedure of identifying concrete itemsets
1. generate the set of all frequent (positive) items, i.e., L_1, and assign L_1 to the set of concrete itemsets;
2. **for** $k \geq 2$ **do{**
3. generate the candidate set of k-itemsets from L_{k-1}, i.e., $C_k = L_{k-1} \bowtie L_{k-1}$;
4. **if** (C_k is empty) **then break**;
5. scan the transactions to calculate supports of all candidate k-itemsets;
6. $L_k = \{ c \in C_k \mid S_c \geq MinSup \}$;
7. **foreach** frequent itemset X in L_k **do{**
8. **if** $(S_X > \prod_{x_i \in X} S_{x_i})$ **&&** $(Chi(X) \geq \chi^2_{df(X),\alpha})$
9. add X to the set of concrete itemsets;
10. }
11. **}**

// Procedure of substitution rule generation
12. **foreach** pair of concrete itemsets X, Y **do{**
13. **if** $(\rho(X,Y) < -\rho_{\min})$
14. **if** $(Sup(X \rightarrow \overline{Y}) \geq MinSup)$ **&&** $(Conf(X \rightarrow \overline{Y}) \geq MinConf)$ // $X \rightarrow \overline{Y}$ is valid
15. output the substitution rule $X \triangleright Y$;
16. **}**

The execution of algorithm SRM can be best understood by the example below.

Example 2: Consider the transaction database in Table 1(a). Algorithm SRM first performs the procedure of identifying concrete itemsets, i.e., operations from line 1 to line 11 in algorithm SRM. Given MinSup=0.2 and MinConf=0.7, the frequent itemsets can be first obtained as in Table 3.

The dependency among items in these frequent itemsets is then evaluated. By Definition 2, chi-square tests of concreteness are performed on each k-itemset for $k \geq 2$. The chi-square values of these frequent itemsets are also shown in Table 3 where only two frequent 2-itemsets are found concrete. Note that {a, c, d} fails to pass the test since $df(\{a, c, d\}) = 2^3\text{-}3\text{-}1 = 4$ and $Chi(\{a,c,d\}) = 6.38 < \chi^2_{4,0.05} = 9.49$.

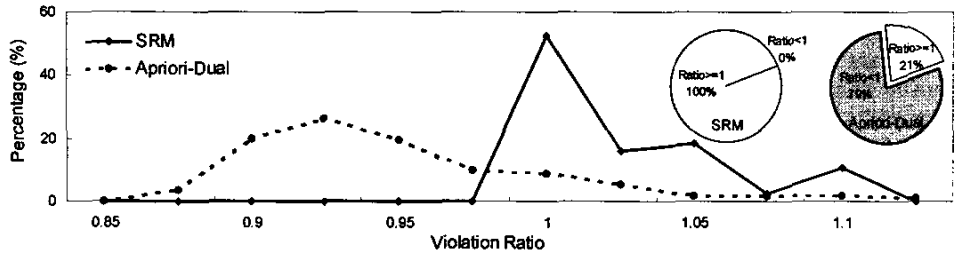
Table 3: Frequent (positive) itemsets, their supports and chi-square values generated from Table 1(a) (concrete itemsets are in italics)

I_1	S_I
a	*0.5*
b	*0.5*
c	*0.5*
d	*0.3*
e	*0.2*
f	*0.3*

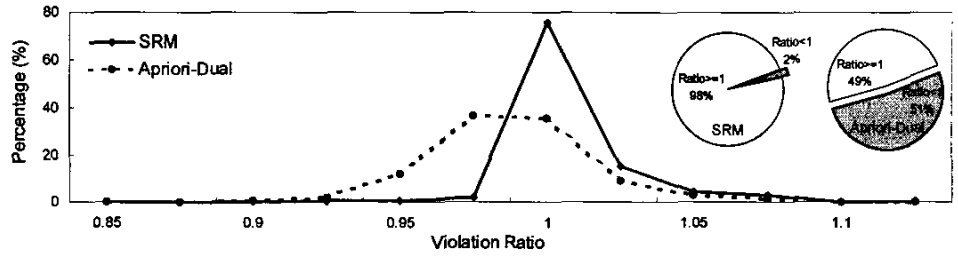
I_2	S_I	*Chi*(I)
a, b	0.2	0.4
a, c	0.2	0.4
a, d	*0.3*	*4.29*
b, c	0.2	0.4
b, f	*0.3*	*4.29*
c, d	0.2	0.48

I_3	S_I	*Chi*(I)
a, c, d	0.2	6.38

Next, in the procedure of substitution rule generation, i.e., operations from line 12 to line 16 in algorithm SRM, the candidate substitution pairs can then be generated by



(a) Violation ratio distribution curve for results of D_{dense}



(b) Violation ratio distribution curve for results of D_{sparse}

Figure 1. Violation ratio distribution curves

joining on these concrete itemsets. By examining the support, confidence and correlation of these candidate pairs, substitution rules can be generated as in Table 4.

Table 4: Substitution rules discovered with MinSup=0.2, MinConf=0.7, and $\rho_{\min}$=-0.5

Rule(X⊳Y)	Sup	Conf	Correlation(X, Y)
{b}⊳{d}	0.5	1	-0.65
{d}⊳{b}	0.3	1	-0.65
{a, d}⊳{b}	0.3	1	-0.65

4. Experimental Results

The simulation model of our experimental studies is described in Section 4.1. The quality of substitution rules generated is evaluated in Section 4.2.

4.1. Simulation Model

As mentioned in Section 2.1, mining negative association rules by appending complement items to the original transaction database incur both an excessive storage space and a huge computational cost. Without the process of generating rules with the required form as adopted by Apriori-Dual, the computation time of the naive approach for generating negative association rules is shown by our experiments to be longer, in several orders, than those of both the algorithms Apriori-Dual and SRM. Therefore, only the algorithms Apriori-Dual and SRM are being compared in following experiments.

We use two synthetic datasets, i.e., D_{dense} and D_{sparse}, generated by a randomized transaction generation algorithm in [16]. The values of parameters used to generate the datasets are summarized in Table 5, where both the dense and the sparse dataset distributions are considered.

Table 5. Parameter settings of the synthetic datasets

	D_{dense}	D_{sparse}	Meaning
T	10	5	Average size of transactions
I	50	100	Number of items
D	10,000	10,000	Number of transactions

4.2. Evaluation of Rule Quality

To evaluate the quality of a substitution rule, we may count the number of transactions which contain only one of the substitutive itemsets in the rule, i.e., the antecedent or the consequent. Hence, the violation ratio proposed in [1] is adopted. Specifically, a pair of substitutive itemsets is said to be in violation if exactly only one of them is present in a transaction. The violation ratio is defined as the ratio of the number of real violations to the expected number of violations. Thus, the larger the value of the violation ratio of a rule, the more likely its antecedent and consequent itemsets are substitutes for each other. Note that the violation ratio of an interesting substitution rule should be lager than one.

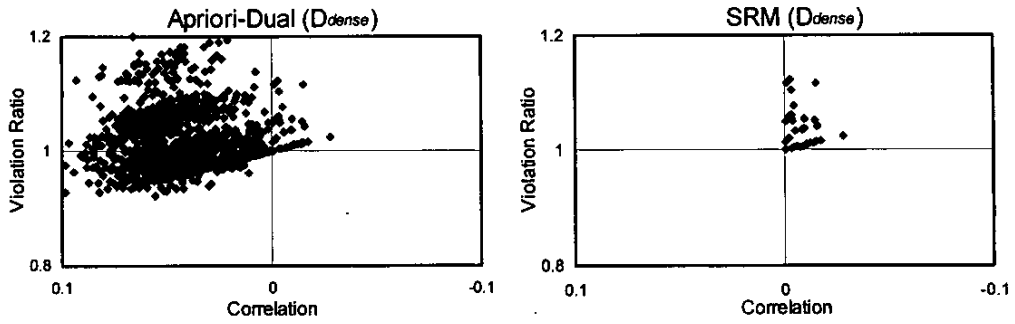


Figure 2. Quality matrix in the dense dataset

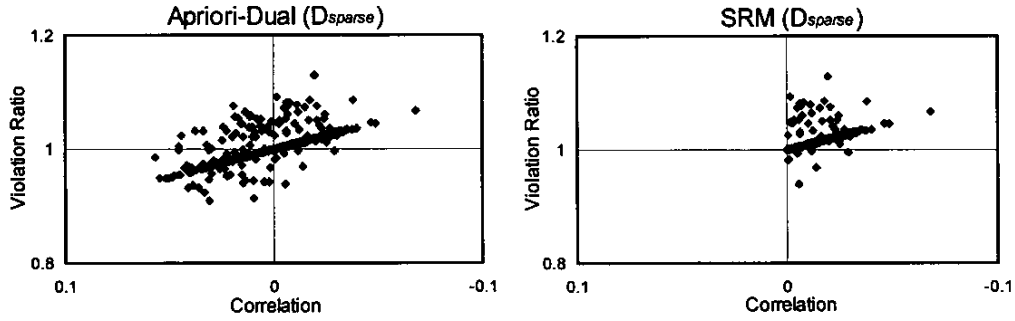


Figure 3. Quality matrix in the sparse dataset

Experiments on the two datasets are conducted with MinSup=15% (D_{dense}), MinSup=10% (D_{sparse}) and MinConf=50%. The distribution curves for results of both sparse and dense datasets are depicted in Figure 1. Note that the percentage of population rather than the actual number of rules is used as the measurement for vertical axes in both charts. Also, to provide a remarkable index for evaluating the quality of rules, proportions of interesting rules, i.e., whose violation ratios are larger than 1, to uninteresting ones are also presented as pie charts in Figure 1. Note that more than half of rules generated by algorithm Apriori-Dual in both datasets have a violation ratio less than one. In contrast, more than 98% of rules generated by algorithm SRM are interesting for both datasets. Also note that algorithm Apriori-Dual favors dense databases while algorithm SRM performs well in each dataset, showing that algorithm SRM is more adaptive and robust.

The resulting rules by Apriori-Dual and SRM for both datasets are plotted in Figure 2 and Figure 3, where each point corresponds to a rule produced. The y-axis indicates the violation ratio and the x-axis shows the correlation of the antecedent and the consequent itemsets of the rule. Each figure is divided into four areas. *In the upper right area, the rules are the most interesting ones among those in all areas* due to the negative correlation of the antecedent and the consequent of each rule and high violation ratios.

Note that rules generated by algorithm Apriori-Dual and algorithm SRM are subsets of negative association rules. It can be seen from Figure 2 and Figure 3 that algorithm SRM can generate the most appropriate ones on the basis of negative association rules.

5. Conclusions

In this paper, a new mining capability, called mining of substitution rules, is explored. The notion of evaluating the dependency among items in a concrete itemset proposed in this paper offers another dimension for itemset selection (in addition to the one of using the support threshold), thereby being able to lead to more interesting results in the subsequent rule derivation based on these itemsets. We have derived theoretical properties for the model of substitution rule mining and devised a technique on the induction of positive itemset supports to improve the efficiency of support counting for negative itemsets. In light of these properties, algorithm SRM is proposed to discover the substitution rules

efficiently while attaining good statistical significance. It is shown by empirical studies that algorithm SRM not only has very good execution efficiency but also produces substitution rules of very high quality.

Acknowledgement

The authors are supported in part by the National Science Council, Project No. NSC 91-2213-E-002-034 and NSC 91-2213-E-002-045, Taiwan, Republic of China.

References

[1] C. C. Aggarwal and P. S. Yu. A New Framework for Itemset Generation. *Proceedings of the 17th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, pages 18–24, June 1998.

[2] R. Agrawal, T. Imielinski, and A. Swami. Mining Association Rules between Sets of Items in Large Databases. *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pages 207–216, May 1993.

[3] J.-F. Boulicaut, A. Bykowski, and B. Jeudy. Towards the Tractable Discovery of Association Rules with Negations. *Proceedings of the 4th International Conference on Flexible Query Answering Systems*, pages 425–434, October 2000.

[4] S. Brin, R. Motwani, and C. Silverstein. Beyond Market Baskets: Generalizing Association Rules to Correlations. *Proceedings of the 1997 ACM SIGMOD International Conference on the Management of Data*, pages 265–276, May 1997.

[5] M.-S. Chen, J. Han, and P. S. Yu. Data Mining: An Overview from Database Perspective. *IEEE Transactions on Knowledge and Data Engineering*, 8(6):866–883, December 1996.

[6] W. DuMouchel and D. Pregibon. Empirical Bayes Screening for Multi-Item Associations. *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 67–76, August 2001.

[7] J. Han and M. Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, 2000.

[8] R. V. Hogg and E. A. Tanis. *Probability and Statistical Inference, 6/e*. Prentice-Hall International, Inc., 2001.

[9] J. C. Hosseini, R. R. Harmon, and M. Zwick. An Information Theoretic Framework for Exploratory Multivariate Market Segmentation Research. *Decision Sciences*, 22:663–677, 1991.

[10] C. Jermaine. The Computational Complexity of High-Dimensional Correlation Search. *Proceedings of the 1st IEEE International Conference on Data Mining*, pages 249–256, November 2001.

[11] R. A. Johnson and D. W. Wichern. *Applied Multivariate Statistical Analysis, 5/e*. Prentice-Hall International, Inc., 2002.

[12] M. G. Kendall and G. U. Yule. *Journal of Royal Statistical Society 115*, pages 156–161, 1952.

[13] B. Liu, W. Hsu, and Y. Ma. Identifying Non-Actionable Association Rules. *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 329–334, August 2001.

[14] S. Ma and J. L. Hellerstein. Mining Mutually Dependent Patterns. *Proceedings of the 1st IEEE International Conference on Data Mining*, November 2001.

[15] A. Savasere, E. Omiecinski, and S. Navathe. Mining for Strong Negative Associations in a Large Database of Customer Transactions. *Proceeding of the 14th International Conference on Data Engineering*, pages 494–502, February 1998.

[16] R. Srikant and R. Agrawal. Mining Generalized Association Rules. *Proceedings of the 21th International Conference on Very Large Data Bases*, pages 407–419, September 1995.

Appendix: A Counterexample to Theorem 1 in [4]

Theorem 1 in [4]: In the binomial case, the chi-square statistic is upward closed.

This theorem means that "if S is correlated with significance level α, any superset of S is also correlated with significance level α." Consider a contingency table which is slightly modified from the one provided in [4].

Table 6. An example contingency table of market basket data for coffee (c), tea (t), and doughnuts (d)

d	c	$\bar{c}$	Σrow
t	8	2	10
$\bar{t}$	40	2	42
Σcol	48	4	52

$\bar{d}$	c	$\bar{c}$	Σrow
t	10	2	12
$\bar{t}$	34	2	36
Σcol	44	4	48

From Theorem 1 in this paper, we have

$$Chi(\{c,t\}) = \frac{(8+10)^2}{(100)\frac{(48+44)}{100}\frac{(10+12)}{100}} + \frac{(40+34)^2}{(100)\frac{(48+44)}{100}\frac{(42+36)}{100}} + \frac{(2+2)^2}{(100)\frac{(4+4)}{100}\frac{(10+12)}{100}} + \frac{(2+2)^2}{(100)\frac{(4+4)}{100}\frac{(42+36)}{100}} \cdot 100 = 3.98, \text{ and}$$

$$Chi(\{d,c,t\}) = \frac{8^2}{(100)\frac{52}{100}\frac{(48+44)}{100}\frac{(10+12)}{100}} + \frac{40^2}{(100)\frac{52}{100}\frac{(48+44)}{100}\frac{(42+36)}{100}} + \frac{2^2}{(100)\frac{52}{100}\frac{(4+4)}{100}\frac{(10+12)}{100}} + \frac{2^2}{(100)\frac{52}{100}\frac{(4+4)}{100}\frac{(42+36)}{100}} + \frac{10^2}{(100)\frac{48}{100}\frac{(48+44)}{100}\frac{(10+12)}{100}} + \frac{34^2}{(100)\frac{48}{100}\frac{(48+44)}{100}\frac{(42+36)}{100}} + \frac{2^2}{(100)\frac{48}{100}\frac{(4+4)}{100}\frac{(10+12)}{100}} + \frac{2^2}{(100)\frac{48}{100}\frac{(4+4)}{100}\frac{(42+36)}{100}} \cdot 100 = 4.49.$$

As mentioned in Section 2.2, the corresponding degrees of freedom should increase with k, i.e., df({c, t})=1 and df({d, c, t})=4, respectively. Given a significance level α=0.05, it can be verified that Chi({c, t}) = 3.98 > $\chi^2_{1,0.05}$ = 3.84 and Chi({d, c, t}) = 4.49 < $\chi^2_{4,0.05}$ = 9.49. Note that {c, t} passed the chi-square test and {d, c, t} did not, meaning that the chi-square test is not upward closed. This leads to a counterexample to Theorem 1 in [4].