Mining Both Positive and Negative Association Rules

Xindong Wu
Department of Computer Science, University of Vermont, Burlington, Vermont 05405, USA

Chengqi Zhang and Shichao Zhang
Faculty of Information Technology, University of Technology Sydney,
PO Box 123, Broadway, Sydney, NSW 2007, Australia

KW@EMBA.UVM.EDU
{CHENGQI, ZHANGSC}@IT.UTS.EDU.AU

Abstract

Association rules are traditionally defined as implications of the form \( A \Rightarrow B \), where \( A \) and \( B \) are frequent itemsets in a transaction database. This paper extends this definition to include association rules of forms \( A \Rightarrow \neg B \), \( \neg A \Rightarrow B \), and \( \neg A \Rightarrow \neg B \), which indicate negative associations between itemsets. We call rules of the form \( A \Rightarrow B \) positive rules, and rules of the other forms negative rules. Negative rules are also very useful in association analysis, although they are hidden and different from positive rules. We present a method that mines both positive and negative rules. The negative rules are generated from infrequent itemsets.

1. Introduction

Mining association rules in databases has received much attention recently. Existing work has focused on discovering association rules of the form \( A \Rightarrow B \), whose support (\( \text{supp} \)) and confidence (\( \text{conf} \)) meet some user specified minimum support (\( \text{minsupp} \)) and minimum confidence (\( \text{minconf} \)) thresholds respectively. \( A \) and \( B \) here are disjoint itemsets (\( A \cap B = \emptyset \)). This is the support-confidence framework for association analysis that was first proposed by Agrawal, Imielinski, and Swami (Agrawal, Imielinski, and Swami 1993). In practical applications, the rule \( A \Rightarrow B \) can be used to predict that “if \( A \) occurs in a transaction, then \( B \) will likely also occur in the same transaction”. Such applications are expected to increase product sales and provide more convenience for supermarket customers.

Association rules from the support-confidence framework are positive rules. They indicate that the presence of some itemsets will imply the presence of other itemsets in the same transactions. Below we look at two examples with a different type of association rules.

Example 1. Suppose we have six itemsets, \( A, B, C, D, E \) and \( F \), two association rules, \( A \Rightarrow B \) and \( E \Rightarrow F \), from the support-confidence model, and two aisles in a supermarket to place these six itemsets. We know that \( A \) and \( B \) should be put in the same aisle, as should \( E \) and \( F \). How about \( C \) and \( D \)?

From the two positive rules, \( C \) and \( D \) are not positively associated with any of the other itemsets. What if we have a rule like \( A \Rightarrow \neg C \) which says that the presence of \( A \) in a transaction implies that \( C \) will highly unlikely be present in the same transaction? We call rules of the form \( A \Rightarrow \neg C \) negative rules (in the following sections, we will also have negative rules in the forms of \( \neg A \Rightarrow C \) and \( \neg A \Rightarrow \neg C \)). Negative rules indicate that the presence of some itemsets will imply the absence of other itemsets in the same transactions. In our current example, if we have a negative association between \( A \) and \( C \) (and no other negative associations with \( C \)), we would put \( C \) in the same aisle with \( E \) and \( F \).

Example 2 (adapted from (Brin, Motwani and Silverstein 1997)). Suppose we have a market basket database from a grocery store, consisting of \( n \) baskets. Let us focus on the purchase of tea (denoted by \( t \)) and coffee (denoted by \( c \)).

When \( \text{supp}(t) = 0.25 \) and \( \text{supp}(t \cup c) = 0.2 \), we can apply the support-confidence model for a potential association rule \( t \Rightarrow c \). The support for this rule is 0.2, which is fairly high. The confidence is the conditional probability that a customer who buys tea also buys coffee, i.e., \( \text{conf}(t \Rightarrow c) = \text{supp}(t \cup c)/\text{supp}(t) = 0.2/0.25 = 0.8 \), which is very high. In this case, we would conclude that the rule \( t \Rightarrow c \) is a valid one.

Now consider \( \text{supp}(c) = 0.6 \), \( \text{supp}(t) = 0.4 \), \( \text{supp}(t \cup c) = 0.05 \), and \( \text{minconf} = 0.52 \). The confidence of \( t \Rightarrow c \) is \( \text{supp}[t \cup c]/\text{supp}[t] = 0.05/0.4 = 0.125 \) <
minconf = 0.52 and, supp(t ∪ c) = 0.05 is low. This indicates that t ∪ c is an infrequent itemset and that, t ⇒ c cannot be extracted as a rule in the support-confidence model. However, supp[t ∪ ¬c] = supp[t] − supp[t ∪ c] = 0.4 − 0.05 = 0.35 is high, and the confidence of t ⇒ ¬c is the ratio supp[t ∪ ¬c]/supp[t] = 0.35/0.4 = 0.875 > minconf. Therefore t ⇒ ¬c would be a valid rule from the database.

The rule t ⇒ ¬c is a negative rule. From Example 2, we need to examine frequent itemsets (such as t ∪ c) to identify negative association rules. Existing algorithms for association analysis have concentrated on identifying frequent (or large) itemsets only in a given database. This indicates that existing algorithms for discovering frequent itemsets are inadequate for mining negative association rules.

The importance of negative association rules was first pointed out in (Brin, Motwani and Silverstein 1997), and related work is introduced in Section 4.4. This paper presents a method for identifying both positive and negative association rules in databases. Our approach is different from existing work in association analysis in two aspects. First, infrequent itemsets in databases are of our interest for mining negative association rules. Second, to design an efficient model for mining both positive and negative association rules in databases, we estimate the confidence of association rules using the increasing degree of the rule’s conditional probability relative to its priori probability.

The paper is organized as follows. We define conditions for itemsets of interest, and present a procedure for identifying both frequent itemsets and infrequent itemsets in Section 2. In Section 3, we construct a model for discovering and measuring positive and negative association rules of interest, and Section 4 presents our experimental results.

2. Frequent and Infrequent Itemsets

A frequent itemset (also called large itemset (Chen, Han, and Yu 1996)) is an itemset that meets the user-specified minimum support. Accordingly, we define an infrequent itemset (or small itemset) as an itemset that does not meet the user-specified minimum support. Suppose we have a market basket database of n baskets with m items. Then there are 2^m possible itemsets in the database. However, only some of them are useful for mining association rules of interest. Clearly, searching for both frequent and infrequent itemsets of interest in a large database is an expensive process.

In this section, we first define the conditions for itemsets of interest in databases, and then present a procedure for identifying both frequent and infrequent itemsets of interest.

Let I = \{i_1, i_2, \ldots, i_N\} be a set of N distinct literals called items, and D a database of variable-length transactions over I. Each transaction contains a set of items \{i_1, i_2, \ldots, i_k\} ∈ I. A set of distinct items from I is called an itemset. The number of items in an itemset is the length (or the size) of the itemset. Itemsets of some length k are referred to as k-itemsets.

Each itemset has an associated statistical measure called support, denoted as supp. For an itemset A ⊆ I, supp(A) = s if the fraction of transactions in D containing A equals to s.

A (positive) association rule in the support-confidence framework is an implication of the form A ⇒ B (or A → B), where A, B ⊆ I, and A ∩ B = ∅. A is the antecedent of the rule, and B is the consequent of the rule.

The support of a rule A ⇒ B is denoted as supp(A ∪ B). The confidence of the rule A ⇒ B is defined as the ratio of the supp(A ∪ B) of itemset A ∪ B over the supp(A) of itemset A. That is, conf(A ⇒ B) = supp(A ∪ B)/supp(A).

2.1. Frequent Itemsets for Positive Rules

Let X, Y ⊆ I be two itemsets, X ∩ Y = ∅, supp(X) ≠ 0, supp(Y) ≠ 0, and the minimum support (minsupp) and minimum confidence (minconf) be given by the user. X ⇒ Y can be extracted as a valid rule if

1. supp(X ∪ Y) = p(X ∪ Y) ≥ minsupp, and
2. conf(X ⇒ Y) = p(Y|X) = \frac{p(X,Y)}{p(X)} ≥ minconf.

However, Piatetsk-Shapiro (1991) argued that a rule X ⇒ Y is not interesting if

\[ support(X ∪ Y) ≈ support(X) × support(Y). \]

This argument proposed that only if supp(X ∪ Y) − supp(X)supp(Y) ≥ mininterest, the rule X ⇒ Y is of interest.

Including Piatetsk-Shapiro’s argument, X ⇒ Y is a valid positive rule of interest if and only if

1. X ∩ Y = ∅,
2. supp(X ∪ Y) ≥ minsupp,
3. supp(X ∪ Y) − supp(X)supp(Y) ≥ mininterest,
4. supp(X ∪ Y)/supp(X) ≥ minconf,

where mininterest is a minimum interest specified by the user, and X ∪ Y is a frequent itemset.
2.2. Infrequent Itemsets of Interest for Negative Rules

To mine negative association rules, all itemsets for possible negative association rules in a given database need to be considered. For example, if \( A \Rightarrow \neg B \) can be discovered as a valid rule, then \( \text{supp}(A \cup \neg B) \geq \text{mnsupp} \) must hold. If \( \text{mnsupp} \) is high, \( \text{supp}(A \cup \neg B) \geq \text{mnsupp} \) would mean that \( \text{supp}(A \cup B) < \text{mnsupp} \) and itemset \( A \cup B \) cannot be generated as a frequent itemset in existing association analysis algorithms. In other words, \( A \cup B \) is an infrequent itemset. However, there are too many infrequent itemsets in databases, and we must define some conditions for identifying infrequent itemsets of interest.

If \( A \) is a frequent itemset and \( B \) is an infrequent itemset with frequency 1 in a large database, then \( A \Rightarrow \neg B \) certainly looks like a valid negative rule, because \( \text{supp}(A) \geq \text{mnsupp}, \text{supp}(B) \approx 0, \text{supp}(A \cup \neg B) \approx \text{supp}(A) \geq \text{mnsupp}, \text{con} f (A \Rightarrow \neg B) = \text{supp}(A \cup \neg B) / \text{supp}(A) \approx 1 \geq \text{minconf} \). This could indicate that the rule \( A \Rightarrow \neg B \) is valid, and the number of this type of itemsets in a given database can be very large. For example, rarely purchased products in a supermarket are always infrequent itemsets. However, in practice, more attention is paid to frequent itemsets, and any patterns mined in databases would mostly involve frequent itemsets only. This means that if \( A \Rightarrow \neg B \) (or \( \neg A \Rightarrow B \), or \( \neg A \Rightarrow \neg B \)) is a negative rule of interest, \( A \) and \( B \) would be frequent itemsets. In other words, no matter whether association rules are positive or negative, we are only interested in frequent itemsets that occur in association rules. We take this observation as one of the main conditions for identifying infrequent itemsets for mining negative association rules.

Based on the conditions for frequent itemsets for mining positive rules, we define the following conditions for a rule \( A \Rightarrow \neg B \) to be a valid negative rule of interest.

1. \( A \cap B = \emptyset \);
2. \( \text{supp}(A) \geq \text{mnsupp}, \text{supp}(B) \geq \text{mnsupp}, \text{supp}(A \cup \neg B) \geq \text{mnsupp} \),
3. \( \text{supp}(A \cup \neg B) - \text{supp}(A) \text{supp}(\neg B) \geq \text{mininterest} \),
4. \( \text{supp}(A \cup \neg B) / \text{supp}(A) \geq \text{minconf} \).

Note that \( \text{supp}(B) \geq \text{mnsupp} \) is required because we are only interested in frequent itemsets that occur in association rules.

We can define conditions for rules of the forms \( \neg A \Rightarrow B \) and \( \neg A \Rightarrow \neg B \) accordingly.

When \( A \Rightarrow \neg B \) is a valid negative rule, \( A \cup B \) is an infrequent itemset of interest. If \( i \) is an infrequent itemset of interest, there is at least one expression \( i = A \cup B \) such that one of the rules \( A \Rightarrow \neg B, \neg A \Rightarrow B, \) and \( \neg A \Rightarrow \neg B \) is a valid negative association rule of interest.

2.3. Identifying Frequent and Infrequent Itemsets of Interest

Based on the constraints specified in Sections 2.1 and 2.2, the following procedure is designed to identify all frequent and infrequent itemsets of interest in databases.

**Procedure 1: AllItemsetsOfInterest**

**Input:** \( D \) - a database; \( \text{mnsupp} \) - minimum support; \( \text{mininterest} \) - minimum interest.

**Output:** \( PL \) - set of frequent itemsets of interest; \( NL \) - set of infrequent itemsets of interest.

1. let \( PL \leftarrow \emptyset; NL \leftarrow \emptyset \);
2. let \( \text{Frequent}_1 \leftarrow \{ \text{frequent 1-itemsets} \}; PL \leftarrow PL \cup \text{Frequent}_1 \); let \( L_1 \leftarrow \text{Frequent}_1; S_1 \leftarrow \emptyset \);
3. for \( (k = 2; L_{k-1} \neq \emptyset; k++) \) do
   begin /*Generate all possible frequent and infrequent k-itemsets of interest in \( D \).*
   (3.1) let \( \text{Tem}_k \leftarrow \text{the k-itemsets constructed from } \text{Frequent}_{k-1} \);
   (3.2) for each transaction \( t \) in \( D \) do
      begin /*Check which k-itemsets are included in \( t \).*
         let \( \text{Tem}_t \leftarrow \text{k-itemsets in both } t \text{ and } \text{Tem}_k \);
         for each itemset \( A \) in \( \text{Tem}_t \) do
            let \( \text{Acount} \leftarrow \text{Acount} + 1 \);
         end
      let \( L_k \leftarrow \text{Frequent}_k \);
      let \( S_k \leftarrow \text{Tem}_k \setminus \text{Frequent}_k \);
   (3.4) /*Prune all uninterested k-itemsets in \( L_k \) for each itemset \( i \) in \( L_k \) do
      if \( i \) is uninterested by the \text{mininterest}
         then let \( L_k \leftarrow L_k \setminus \{i\} \);
      let \( PL \leftarrow PL \cup L_k \);
   (3.5) /*Prune all uninterested k-itemsets in \( S_k \) for each itemset \( i \) in \( S_k \) do
      if \( i \) is uninterested by the \text{mininterest}
         then let \( S_k \leftarrow S_k \setminus \{i\} \);
      let \( NL \leftarrow NL \cup S_k \);
   end
(4) output \( PL \) and \( NL \);

(5) return.

\( \text{AllItemsOfInterest} \) generates all frequent and infrequent itemsets of interest in a given database \( D \), where \( PL \) is the set of all frequent itemsets of interest in \( D \), and \( NL \) is the set of all infrequent itemsets of interest in \( D \). \( PL \) and \( NL \) contain only frequent and infrequent itemsets of interest respectively, and all frequent itemsets in \( \text{Frequent}_i \) (\( i > 0 \)) must be saved for generating future infrequent itemsets of interest.

The initialization is done in Step (1). Step (2) generates \( \text{Frequent}_1 \) of all frequent 1-itemsets in database \( D \) in the first pass of \( D \).

Step (3) generates \( L_k \) and \( S_k \) for \( k \geq 2 \) by a loop, where \( L_k \) is the set of all frequent \( k \)-itemsets of interest in the \( k \)-th pass of \( D \), \( S_k \) is the set of all infrequent \( k \)-itemsets of interest, and the end-condition of the loop is \( L_{k-1} = \emptyset \). For each pass of the database in Step (3), say pass \( k \), there are five substeps as follows.

Step (3.1) generates \( \text{Tem}_k \) of all \( k \)-itemsets in \( D \), where each \( k \)-itemset in \( \text{Tem}_k \) is the union of two frequent itemsets in \( \text{Frequent}_{k-1} \). Each itemset in \( \text{Tem}_k \) is counted in \( D \) by a loop in Step (3.2). \( \text{Frequent}_k \), \( L_k \), and \( S_k \) are generated in Step (3.3). Both \( \text{Frequent}_k \) and \( L_k \) are sets of all frequent \( k \)-itemsets in \( \text{Tem}_k \) that meet \( \text{minsupp} \). That is, \( L_k \) is the set of all frequent \( k \)-itemsets in \( \text{Tem}_k \), \( S_k \) is the set of all infrequent \( k \)-itemsets in \( \text{Tem}_k \), whose supports do not meet \( \text{minsupp} \), and \( S_k = \text{Tem}_k - \text{Frequent}_k \). \( S_k \) is the set of all possible infrequent \( k \)-itemsets in \( \text{Tem}_k \).

Steps (3.4) and (3.5) select all frequent and infrequent \( k \)-itemsets of interest. In Step (3.4), if an itemset \( i \) in \( L_k \) does not satisfy \( |\text{supp}(X \cup Y) - \text{supp}(X)\text{supp}(Y)| \geq \text{mininterest} \) for any expression \( X \cup Y = i \), then \( i \) is an uninteresting frequent itemset, and is pruned from \( L_k \). After all uninteresting frequent itemsets are pruned from \( L_k \), \( L_k \) is appended into \( PL \). In Step (3.5), all infrequent itemsets of interest in \( S_k \) are selected. If an itemset \( i \) in \( S_k \) does not satisfy \( |\text{supp}(X \cup Y) - \text{supp}(X)\text{supp}(Y)| \geq \text{mininterest} \) for any \( X \cup Y = i \), then \( i \) is an uninteresting infrequent itemset, and is pruned from \( S_k \). After all uninteresting infrequent itemsets are pruned from \( S_k \), \( S_k \) is appended into \( NL \).

Step (4) outputs the frequent and infrequent itemsets of interest in \( PL \) and \( NL \) respectively. The procedure ends in Step (5).

**Theorem 1** Algorithm AllItemsOfInterest finds all interesting (frequent and infrequent) itemsets that satisfy our constraints in Sections 2.1 and 2.2.

**Proof:** Since the frequent itemsets are generated in the same way as in Apriori (Agrawal and Srikant 1994), AllItemsOfInterest can generate all frequent itemsets that satisfy our constraints in Section 2.1. The pruning in Step (3.4) verifies the minimum interest requirement, and improves Apriori by avoiding generating uninteresting frequent itemsets.

For any itemset \( c \) in \( \text{Tem}_k \) in Step (3.3), if \( \text{supp}(c) < \text{minsupp} \), \( c \) is not included in \( \text{Frequent}_k \), but in \( S_k \). This means that \( S_k \) is the set of all possible infrequent \( k \)-itemsets in \( \text{Tem}_k \). In Step (3.5), all infrequent itemsets in \( S_k \) are checked against the minimum interest requirement, and those infrequent itemsets selected from \( S_k \) satisfy our constraints for interesting infrequent itemsets, which means that all interesting infrequent \( k \)-itemsets are kept. Therefore, AllItemsOfInterest can also generate all infrequent itemsets that satisfy our constraints for interesting infrequent itemsets. \( \nabla \)

### 3. Extracting Positive and Negative Association Rules

In this section, we present a definition of four types of association rules based on Piatetsky-Shapiro’s argument and probability theory, and design an algorithm for mining both positive and negative association rules of interest in databases.

#### 3.1. Four Types of Association Rules

Recall the relationship between \( p(Y|X) \) and \( p(Y) \) (or \( \text{supp}(Y) \)) for a possible rule \( X \Rightarrow Y \) in Section 2.1. Based on Piatetsky-Shapiro’s argument, we can write the interest of an association between \( X \) and \( Y \) in the form of their statistical dependence,

\[
\text{Interest}(X, Y) = \frac{p(X \cup Y)}{p(X)p(Y)} = \frac{p(Y|X)}{p(Y)}
\]

This formula is referred to as the interest of \( Y \) given \( X \). When \( \text{Interest}(X, Y) = 1 \), \( p(Y|X) = p(Y) \), and \( X \) and \( Y \) are independent of each other. When \( \text{Interest}(X, Y) > 1 \), \( p(Y|X) > p(Y) \), \( Y \) is positively dependent on \( X \), and the probability that \( Y \) occurs is increased when \( X \) occurs. When \( \text{Interest}(X, Y) < 1 \), \( \text{supp}(Y|X) < p(Y) \), \( Y \) is negatively dependent on \( X \), and the probability that \( Y \) occurs is decreased when \( X \) occurs.

Consider the above relationships between \( p(Y|X) \) and \( p(Y) \), \( \text{Interest}(X, Y) \) has the following three possible cases.

1. If \( \text{Interest}(X, Y) = 1 \) or \( p(Y|X) = p(Y) \), then \( Y \) and \( X \) are independent.
(2) If \( \text{Interest}(X, Y) > 1 \) or \( p(Y|X) > p(Y) \), then \( Y \) is positively dependent on \( X \), and the following holds,
\[
0 < p(Y|X) - p(Y) \leq 1 - p(Y)
\]
In particular, we have
\[
0 < \frac{p(Y|X) - p(Y)}{1 - p(Y)} \leq 1
\]
The bigger the ratio \( (p(Y|X) - p(Y))/(1 - p(Y)) \), the higher the positive dependence.

(3) If \( \text{Interest}(X, Y) < 1 \) or \( p(Y|X) < p(Y) \), then \( Y \) is negatively dependent on \( X \) (or \( \neg Y \) is positively dependent on \( X \)), and the following holds,
\[
0 > p(Y|X) - p(Y) \geq -p(Y)
\]
In particular, we have
\[
0 < \frac{p(Y|X) - p(Y)}{-p(Y)} \leq 1
\]
The bigger the ratio \( (p(Y|X) - p(Y))/(-p(Y)) \), the higher the negative dependence.

In the first case, the rule \( X \Rightarrow Y \) and possible negative rules between \( X \) and \( Y \) are not of interest because \( X \) and \( Y \) are independent. A small neighborhood of 1, i.e., \( p(Y|X) - p(Y) < \text{mininterest} \), would also indicate that \( X \Rightarrow Y \) and possible negative rules between \( X \) and \( Y \) are not of interest either.

The second case has been widely explored in association analysis for positive rules, which indicates that the rule \( X \Rightarrow Y \) may be an association rule of interest. The last case has received little attention. In this case, because \( Y \) is negatively dependent on \( X \), \( X \Rightarrow \neg Y \) may be a negative association rule of interest.

To reflect the above relationships between \( p(Y|X) \) and \( p(Y) \), we adopt the certainty factor model in (Shortliffe 1976), denoted by \( PR \) (probability ratio) in this paper, which uses the ratio of the conditional probability and the priori probability to describe the increased degree of \( p(Y|X) \) relative to \( p(Y) \) as follows.
\[
PR(Y|X) = \begin{cases} 
\frac{p(Y|X) - p(Y)}{1 - p(Y)}, & \text{if } p(Y|X) \geq p(Y), p(Y) \neq 1 \\
\frac{p(Y|X) - p(Y)}{p(Y)}, & \text{if } p(Y) > p(Y|X), p(Y) \neq 0 
\end{cases}
\]
To discover and measure both positive and negative association rules, we can take \( PR(Y|X) \) as the confidence of the association rule between itemsets \( X \) and \( Y \). Clearly, \( \text{confidence}(X \Rightarrow Y) \) has several special cases as follows.

- If \( p(Y|X) = p(Y) \), \( Y \) and \( X \) are independent in probability theory. The confidence of the association rule \( X \Rightarrow Y \) would be assigned as
  \[
  \text{confidence}(X \Rightarrow Y) = PR(Y|X) = 0
  \]
- If \( p(Y|X) - p(Y) > 0 \), \( Y \) is positively dependent on \( X \). When \( p(Y|X) = 1 \) which is the strongest possible condition, the confidence of the association rule \( X \Rightarrow Y \) would be assigned as
  \[
  \text{confidence}(X \Rightarrow Y) = PR(Y|X) = 1
  \]
- If \( p(Y|X) - p(Y) < 0 \), \( Y \) is negatively dependent on \( X \). When \( p(Y|X) = 0 \), the confidence of the association rule \( X \Rightarrow \neg Y \) would be assigned as
  \[
  \text{confidence}(X \Rightarrow \neg Y) = PR(\neg Y|X) = 1
  \]
We take the first half of the \( PR(Y|X) \) definition,
\[
PR(Y|X) = \frac{p(Y|X) - p(Y)}{1 - p(Y)}, \text{or}
\]
\[
PR(Y|X) = \frac{\text{support}(X \cup Y) - \text{support}(X) \cdot \text{support}(Y)}{\text{support}(X)(1 - \text{support}(Y))}
\]
as a metric for confidence of the rule \( X \Rightarrow Y \) in the following discussion when \( \text{support}(X \cup Y) \geq \text{support}(X) \cdot \text{support}(Y) \) and \( \text{support}(X)(1 - \text{support}(Y)) \neq 0 \), where \( \text{support}(Y|X) \) in the certainty factor model is replaced with \( \text{support}(X \cup Y)/\text{support}(X) \) for the convenience of mining association rules. We now present a definition for association rules of interest by this metric.

**Definition 1** Let \( I \) be the set of items in a database \( D, i = A \cup B \subseteq I \) be an itemset, \( A \cap B = \emptyset \), \( \text{support}(A) \neq 0 \), \( \text{support}(B) \neq 0 \), and \( \text{minsupp}, \text{minconf} \) and \( \text{mininterest} > 0 \) be given by the user. Then,

(1) If \( \text{support}(A \cup B) \geq \text{minsupp}, \text{support}(A \cup B) - \text{support}(A) \cdot \text{support}(B) \geq \text{mininterest} \), and \( PR(B|A) \geq \text{minconf} \), then \( A \Rightarrow B \) can be extracted as a positive rule of interest.

(2) If \( \text{support}(A \cup \neg B) \geq \text{minsupp}, \text{support}(A) \geq \text{minsupp}, \text{support}(A \cup \neg B) - \text{support}(A) \cdot \text{support}(\neg B) \geq \text{mininterest} \), and \( PR(\neg B|A) \geq \text{minconf} \), then \( A \Rightarrow \neg B \) can be extracted as a negative rule of interest.
(3) If supp(−A ∪ B) ≥ minsupp, supp(A) ≥ minsupp, supp(B) ≥ minsupp, supp(−A ∪ B) − supp(−A)supp(B) ≥ mininterest, and
PR(B|−A) ≥ minconf, then −A ⇒ B can be extracted as a negative rule of interest.

(4) If supp(−A ∪ −B) ≥ minsupp, supp(A) ≥ minsupp, supp(B) ≥ minsupp, supp(−A ∪ −B) − supp(−A)supp(B) ≥ mininterest, and
PR(−B|−A) ≥ minconf, then −A ⇒ −B can extracted as a negative rule of interest.

This definition shows four types of valid association rules of interest. Case 1 defines positive association rules of interest. Three types of negative association rules are dealt with in Cases 2, 3, and 4. In the above definition, supp(•) ≥ minsupp guarantees that an association rule describes the relationship between two frequent itemsets; the mininterest requirement makes sure that the association rule is of interest; and PR(•) ≥ minconf specifies the confidence condition.

3.2. Algorithm Design

Mining both positive and negative association rules of interest can be decomposed into the following two subproblems, in a similar way to mining positive rules only.

(1) Generate the set PL of frequent itemsets and the set NL of infrequent itemsets.

(2) Extract positive rules of the form A ⇒ B in PL, and negative rules of the forms A ⇒ −B, −A ⇒ B, and −A ⇒ −B in NL.

Let D be a database, and minsupp, minconf and mininterest given by the user. Our algorithm for extracting both positive and negative association rules with the probability ratio model for confidence checking is designed as follows.

Algorithm 1 PositiveAndNegativeAssociations

Input: D - a database; minsupp, minconf, mininterest - threshold values;
Output: association rules;

(1) call procedure AllItemsetsOfInterest;
(2) // Generate positive association rules in PL.
for each frequent itemset A in PL do
for each itemset X ∪ Y = A and X ∩ Y = Ø do
begin
if supp(X∪Y) − supp(X)supp(Y) ≥ mininterest
then
if PR(Y|X) ≥ minconf then
output the rule X ⇒ Y;
if PR(X|Y) ≥ minconf then
output the rule Y ⇒ X;
end;
(3) // Generate all negative association rules in NL.
for each itemset A in NL do
for any X ∪ Y = A and X ∩ Y = Ø do
begin
(3.1) // Generate negative association rules of the forms −X ⇒ Y and Y ⇒ −X.
if supp(X) ≥ minsupp and supp(Y) ≥ minsupp and supp(−X ∪ Y) ≥ minsupp then
if supp(−X ∪ Y) − supp(−X)supp(Y) ≥ mininterest then
begin
if PR(Y|−X) ≥ minconf then
output the rule −X ⇒ Y;
if PR(−X|Y) ≥ minconf then
output the rule Y ⇒ −X;
end;
(3.2) // Generate negative association rules of the forms −X ⇒ −Y and −Y ⇒ −X.
if supp(X) ≥ minsupp and supp(Y) ≥ minsupp and supp(−X ∪ −Y) ≥ minsupp then
if supp(−X ∪ −Y) − supp(−X)supp(−Y) ≥ mininterest then
begin
if PR(−Y|−X) ≥ minconf then
output the rule −X ⇒ −Y;
if PR(−X|−Y) ≥ minconf then
output the rule −Y ⇒ −X;
end;
end;
end;
end;
end;
(4) return.

Algorithm PositiveAndNegativeAssociations generates not only all positive association rules in PL, but also negative association rules in NL. Step (1) calls procedure AllItemsetsOfInterest to generate the sets PL and NL with frequent and infrequent itemsets of interest respectively, in the database D.

Step (2) generates all positive association rules of interest of the form: X ⇒ Y, in PL, where supp(X ∪ Y) − supp(X)supp(Y) ≥ mininterest. If PR(Y|X) ≥ minconf, X ⇒ Y is extracted as a valid rule of interest, with confidence PR(Y|X) and support supp(X ∪ Y). If PR(X|Y) ≥ minconf, Y ⇒ X is extracted as a valid rule of interest, with confidence PR(X|Y) and support supp(X ∪ Y).

Step (3) generates all negative rules of interest of the forms −X ⇒ Y, Y ⇒ −X, −X ⇒ −Y, and
\( -Y \Rightarrow \neg X, \text{ in } NL, \text{ in Step (3.1) and Step (3.2) respectively. In Step (3.1), we have } \supp(X) \geq \minsupp, \supp(Y) \geq \minsupp, \supp(\neg X \cup Y) \geq \minsupp, \text{ and } \supp(\neg X \cup \neg Y) \geq \mininter. \text{ If } PR(Y \mid X) \geq \minconf, \neg X \Rightarrow Y \text{ is extracted as a valid rule of interest, with confidence } PR(Y \mid X) \text{ and support } \supp(\neg X \cup Y). \text{ In Step (3.2), we have } \supp(X) \geq \minsupp, \supp(Y) \geq \minsupp, \supp(\neg X \cup \neg Y) \geq \minsupp, \text{ and } \supp(\neg (X \cup \neg Y) - \supp(\neg X) \supp(Y) \geq \mininter. \text{ If } PR(\neg X \mid Y) \geq \minconf, \neg Y \Rightarrow \neg X \text{ is extracted as a valid rule of interest, with confidence } PR(\neg X \mid Y) \text{ and support } \supp(\neg X \cup \neg Y). \)

4. Experimental Results

To study the properties of our model, we have performed several experiments. Our server is Oracle 8.0.3, and the software was implemented on Sun Sparc using Java. JDBC API was used as the interface between our program and Oracle.

The three databases used in our experiments are supermarket basket data from the Synthetic Classification Data Sets on the Internet (http://www.kdnuggets.com/). The main properties of the databases are as follows. The attributes \( R \) are approximately 1000, the average number of attributes per row is 5, 10, and 20 respectively. The number \( |R| \) of rows is approximately 100000. The average size \( I \) of maximal frequent sets is 4, 4, and 6 respectively. Table 2 summarizes these parameters.

| Database name | \( |R| \) | \( |T| \) | \( |I| \) | \( |R| \) |
|---------------|--------|--------|--------|--------|
| T5.12.D100K   | 940    | 5      | 4      | 96953  |
| T10.14.D100K  | 987    | 10     | 4      | 98376  |
| T20.16.D100K  | 976    | 20     | 6      | 99997  |

4.1. A Comparison with Apriori

To evaluate the positive association rules generated by our proposed approach, we compare our approach with Apriori (Agrawal and Srikant 1994) in the support-confidence framework proposed in (Agrawal, Imieliński, and Swami 1993). When mining positive association rules of interest, a rule \( X \Rightarrow Y \) is of interest if and only if it satisfies four conditions: (1) \( X \cap Y = \emptyset \); (2) \( \supp(X \cup Y) \geq \minsupp \); (3) \( \supp(X \cup Y) - \supp(X) \supp(Y) \geq \mininter \); (4) \( \supp(X \cup Y) / \supp(Y) \geq \minconf \). Condition (3) is not required in the support-confidence framework, but for comparison purposes, it was added into the support-confidence framework in our experiments. Also, the domain of \( PR(Y \mid X) \) is \([-1,1]\). In our experiments, we have transformed it into interval \([0,1]\) by using confidence \( Y \mid X = (PR(Y \mid X) + 1) / 2 \). Based on the constraints, the positive association rules in the two models are identical in our experiments.

In the meanwhile, our approach is more efficient than the Apriori algorithm in discovering positive association rules. Table 2 shows the running time of Apriori and our approach for interesting positive association rules (denoted by IPAR) in seconds in generating positive association rules.

<table>
<thead>
<tr>
<th>Database name</th>
<th>Apriori 0.001</th>
<th>Apriori 0.005</th>
<th>IPAR 0.001</th>
<th>IPAR 0.005</th>
</tr>
</thead>
<tbody>
<tr>
<td>T5.12.D100K</td>
<td>143</td>
<td>207</td>
<td>65</td>
<td>91</td>
</tr>
<tr>
<td>T10.14.D100K</td>
<td>351</td>
<td>590</td>
<td>133</td>
<td>189</td>
</tr>
<tr>
<td>T20.16.D100K</td>
<td>702</td>
<td>1201</td>
<td>238</td>
<td>325</td>
</tr>
</tbody>
</table>

4.2. Efficiency

To assess the efficiency of our proposed approach, we use two algorithms to generate all (frequent and infrequent) itemsets of interest. The first algorithm is Apriori-like, which also generates infrequent itemsets \( (A \cup B) \) that satisfy conditions: (1) \( A \cap B = \emptyset \); (2) \( \supp(A) \geq \minsupp \) and \( \supp(B) \geq \minsupp \); and (3) \( \supp(A \cup \neg B) \geq \minsupp \) or \( \supp(\neg A \cup B) \geq \minsupp \). This algorithm does not have any specific pruning facility, and we denote it by \( MNP \) (Mining with No-Pruning). The other algorithm is our \( AllItemsetsOfInterest \) procedure with the pruning strategies in Steps (3.4) and (3.5) in Section 2.3 that remove all uninteresting itemsets that do not satisfy our interestingness constraints.

We denote our \( AllItemsetsOfInterest \) procedure with the above pruning strategy by \( MBP \) (Mining By Pruning). Table 3 shows the running time of \( MNP \) and \( MBP \) in seconds in generating frequent itemsets.

<table>
<thead>
<tr>
<th>Database Name</th>
<th>MBP 0.0015</th>
<th>MBP 0.001</th>
<th>MNP 0.0015</th>
<th>MNP 0.001</th>
</tr>
</thead>
<tbody>
<tr>
<td>T5.12.D100K</td>
<td>121</td>
<td>238</td>
<td>725</td>
<td>1987</td>
</tr>
<tr>
<td>T10.14.D100K</td>
<td>388</td>
<td>651</td>
<td>2171</td>
<td>4278</td>
</tr>
<tr>
<td>T20.16.D100K</td>
<td>759</td>
<td>1094</td>
<td>4582</td>
<td>7639</td>
</tr>
</tbody>
</table>
4.3. Analysis

The results from our proposed approach for mining both positive and negative association rules of interest are promising. Firstly, as shown in Section 4.1, the positive association rules mined by the proposed model are identical to that by the support-confidence framework proposed in (Agrawal, Imielinski, and Swami 1993) when the condition $supp(X \cup Y) - supp(X)supp(Y) \geq min\text{interest}$ is added into the support-confidence framework in our experiments. However, our proposed approach can also discover negative association rules.

In terms of efficiency, it is evident from the experimental results in Section 4.2 that our proposed approach is more efficient than Apriori in generating itemsets of interest. This is because many frequent itemsets and infrequent itemsets are pruned if they don’t occur in any association rules.

4.4. Related Work

The Chi-square test based model in (Brin, Motwani and Silverstein 1997) first mentioned negative relationships between two frequent itemsets. The chi-square value for itemsets $X$ and $Y$ can be used to determine whether $X$ and $Y$ are independent or not, and if they are not independent, a metric is needed to determine whether the correlation between $X$ and $Y$ is positive or negative. However, if the correlation is negative, other methods must be applied to determine which of $X \Rightarrow \neg Y$, $\neg X \Rightarrow Y$, and $\neg X \Rightarrow \neg Y$ can be extracted as a valid rule and, to compute the support, confidence and interest for such a rule. Therefore, this model has not addressed how to mine negative association rules.

(Savasere, Omiecinski, and Navathe 1998) addresses the issue of negative rule mining, called strong negative association mining. Previously discovered positive associations are combined with domain knowledge in the form of a taxonomy for mining association rules. This model is knowledge-dependent, and can discover negative associations of the form $A \not\Rightarrow B$. However, it is not clear in this model which one of $A \Rightarrow \neg B$, $\neg A \Rightarrow B$, and $\neg A \Rightarrow \neg B$ is the actual relationship between $A$ and $B$. Our model in this paper is different from the strong negative association mining model. First, our model does not require domain knowledge. Second, our negative association rules are given in more concrete expressions to indicate actual relationships between different itemsets. Third and most importantly, we have designed a general framework for mining both positive and negative association rules at the same time.

5. Conclusions

Decision making in many applications such as product placement and investment analysis often involves a number of factors, some of which play advantage roles and others play disadvantage roles. We need to minimize the disadvantage impacts as well as maximize possible benefits. Negative association rules such as $A \Rightarrow \neg C$ are very important in decision making because $A \Rightarrow \neg C$ can tell us that $C$ (which may be a disadvantage factor) rarely occurs when $A$ (which may be an advantage factor) occurs.

In this paper, we have constructed a new method for mining both positive and negative association rules in databases. Our approach is novel and different from existing research efforts on association analysis. Some infrequent itemsets are of interest in our method but not in existing research efforts. We have defined a set of conditions for frequent itemsets and infrequent itemsets to be of interest, and have used the increasing degree of the conditional probability relative to the prior probability to estimate the confidence of positive and negative association rules. Our experimental results have demonstrated that the proposed approach is efficient and promising.

References


