Chapter 1

NEGATIVE ASSOCIATION RULES

Luiza Antonie
University of Guelph
lantonie@uoguelph.ca

Jundong Li
University of Alberta
jundong1@cs.ualberta.ca

Osmar Zaïane
University of Alberta
zaiane@cs.ualberta.ca

Abstract

Mining association rules associates events that took place together. In market basket analysis, these discovered rules associate items purchased together. Items that are not part of a transaction are not considered. In other words, typical association rules do not take into account items that are part of the domain but that are not together part of a transaction. Association rules are based on frequencies and count the transactions where items occur together. However, counting absences of items is prohibitive if the number of possible items is very large, which is typically the case. Nonetheless, knowing the relationship between the absence of an item and the presence of another can be very important in some applications. These rules are called negative association rules. We review current approaches for mining negative association rules and we discuss limitations and future research directions.

Keywords: negative association rules

Introduction

Traditional association rule mining algorithms [11] have been developed to find positive associations between items [4, 9, 26, 14]. Positive associations are associations between items existing in transactions (i.e. items that are present and observed). In market basket analysis, we are generally interested in items that were purchased, and particularly in items purchased together. The assumption is that items that appear in transactions are more important than those that do not appear. As opposed to positive associations, we call negative associations, associations that negate presence. In other words, negative association rules are rules that comprise relationships between present and absent items. Indeed, items that are not purchased when others are can be revealing and certainly important in understanding purchasing behaviour. The association "bread implies milk" indicates the purchasing behaviour of buying milk and bread together. What about the following associations: "customers who buy Coke *do not* buy Pepsi" or "customers who buy juice *do not* buy bottled water"? Associations that include negative items (i.e. items absent from the transaction) can be as valuable as positive associations in many applications, such as devising marketing strategies. Aggarwal and Yu [2] discuss some of the weaknesses and the computational issues for mining positive association rules. They observe that current methods are especially unsuitable for dealing with dense datasets, which is exactly the case when one wants to mine negative association rules.

The expensive computation part of association rule mining is the phase enumerating the frequent itemsets (i.e. a set of items). This enumeration takes place in a search space of size 2^k with k being the number of unique items in the data collection. Focusing on only positive associations significantly reduces this prohibitive search space since we only need to count the observed items in the transactions. Moreover, putting the attention on items present in transactions limits the enumeration of relevant itemsets to a depth dictated by the largest available transaction. These advantageous stratagems cannot be used if absent items are also considered.

Although interesting and potentially useful, the discovery of negative association rules is both a complex and computationally expensive problem. We consider a negative association rule either a negative association between two positive itemsets or an association rule that contains at least a negative item in the antecedent or consequent. The mining of negative association rules is a complex problem due to the increase in items when negative items are considered in the mining process. Imagine a transaction in market basket analysis where a customer buys *bread* and *milk*. When mining for positive association rules only those two items are considered (i.e. *bread* and *milk*). However, when negative items are considered (i.e. items/products not present in a basket/transaction) the search space increases exponentially because all the items in the collection, although not present in the transaction have to be considered. Not only is the problem complex, but also large numbers of negative patterns are uninteresting. The research of mining negative association patterns has to take into consideration both the complexity of the problem and the usefulness of the discovered patterns.

1. Negative Patterns and Negative Association Rules

Formally, association rules are defined as follows: Let $\mathcal{I} = \{i_1, i_2, ...i_m\}$ be a set of items. The total number of unique items is m, the dimensionality of the problem. Let $\mathcal{D}$ be a set of transactions, where each transaction T is a set of items such that $T \subseteq \mathcal{I}$. Each transaction is associated with a unique identifier TID. A transaction T is said to contain X, a set of items in $\mathcal{I}$, if $X \subseteq T$. X is called an itemset.

Definition 1. (Association Rule) An *association rule* is an implication of the form "$X \Rightarrow Y$", where $X \subseteq \mathcal{I}, Y \subseteq \mathcal{I}$, and $X \cap Y = \emptyset$.

Definition 2. (Support) The rule $X \Rightarrow Y$ has a *support* s in the transaction set $\mathcal{D}$ if $s\%$ of the transactions in $\mathcal{D}$ contain $X \cup Y$. In other words, the support of the rule is the probability that X and Y hold together among all the possible presented cases.

Definition 3. (Confidence) The rule $X \Rightarrow Y$ holds in the transaction set $\mathcal{D}$ with *confidence* c if $c\%$ of transactions in $\mathcal{D}$ that contain X also contain Y. In other words, the confidence of the rule is the conditional probability that the consequent Y is true under the condition of the antecedent X.

The problem of discovering all association rules from a set of transactions $\mathcal{D}$ consists of generating the rules that have a *support* and *confidence* greater than given thresholds.

Definition 4. (Negative Item and Positive Item) A negative item is defined as $\neg i_k$, meaning that item i_k is absent from a transaction T. The support of $\neg i_k$ is $s(\neg i_k) = 1 - s(i_k)$. i_k, a positive item, is an item that is present in a transaction.

Definition 5. (Negative Association Rule) A negative association rule is an implication of the form $X \Rightarrow Y$, where $X \subseteq \mathcal{I}, Y \subseteq \mathcal{I}$, and $X \cap Y = \emptyset$ and X and/or Y contain at least one negative item.

TID	Original TD	Augmented TD
1	A,C,D	$A, \neg B, C, D, \neg E$
2	B,C	$\neg A, B, C, \neg D, \neg E$
3	C	$\neg A, \neg B, C, \neg D, \neg E$
4	A,B,E	$A, B, \neg C, \neg D, E$
5	A,C,D	$A, \neg B, C, D.\neg E$

Table 1.1. Transactional Database - Positive and Negative Items

Definition 6. Negative Associations between Itemsets A negative association between two positive itemsets X,Y are rules of the following forms $\neg X \Rightarrow Y$, $X \Rightarrow \neg Y$ and $\neg X \Rightarrow \neg Y$.

Table 1.1 shows a toy transactional database with 5 transactions and 5 items. "Original TD" column shows the items present in each transaction, while "Augmented TD" column shows both present and absent items.

Mining association rules from a transactional database that contains information about both present and absent items is computationally expensive due to the following reasons:

1 The number of items in the transactional database swells when their negative counterparts are added to a transactional database. The maximum number of patterns that can be found in a transactional database with d items is $2^d - 1$. The number of items in the "Original TD" in Table 1.1 is $n = 5$. Even for the small set in Table 1.1, the number of itemsets jumps dramatically from 31 to 1023 when the negative items are added.

2 The length of the transactions in the database increases dramatically when negative items are considered. Picture the length of the transaction in a market basket analysis example where all products in a store have to be considered in each transaction. For example, to a basket where bread and milk are bought (i.e. milk and bread are the positive items), all the other products in the store become part of the transaction as negative items.

3 The total number of association rules that can be discovered when negative items are considered is $5^d - 2 \times 3^d + 1$. A detailed calculation for the formula can be found in [18]. The number of association rules for positive items in a transactions is $3^d - 2^{d+1} + 1$. For our small example, it means that we can find up to 180 positive rules and up to 2640 when the negative items are considered as well.

	CM	¬CM	$\sum_{row}$
SM	20	60	80
¬SM	20	0	20
$\sum_{col}$	40	60	100

Table 1.2. Example 1 Data

4 The number of candidate itemsets is reduced when mining positive association rules by the support based pruning. This property is no longer efficient in a transactional database that is augmented with the negative items. Given that the support of a negative item is $s(\neg i_k) = 1 - s(i_k)$, either the negative or the positive item will have a big enough support to pass the minimum support threshold.

Given the reasons above, the traditional association rule mining algorithms can not cope with mining rules when negative items are considered. This is the reason new algorithms are needed to efficiently mine association rules with negative items. Here we survey algorithms that efficiently mine some variety of negative associations from data.

2. Current Approaches

In this section we present current approaches proposed in the literature to discover negative association rules. We illustrate in Example 1 how rules discovered in the support confidence framework could be misleading sometimes and how the negative associations discovered in data can shed a new light on the discovered patterns.

Example 1. Let us consider an example from market basket data. In this example we want to study the purchase of cow's milk (CM) versus soy milk (SM) in a grocery store. Table 1.2 gives us the data collected from 100 baskets in the store. In Table 1.2 "CM" means the basket contains cow's milk and "¬ CM" means the basket does not contain cow's milk. The same applies for soy milk.

In this data, let us find the positive association rules in the "support-confidence" framework. The association rule "SM $\Rightarrow$ CM" has 20% support and 25% confidence (support(SM $\wedge$ CM)/support(SM)). The association rule "CM $\Rightarrow$ SM" has 20% support and 50% confidence (support(SM $\wedge$ CM)/support(CM)). The support is considered fairly high for both rules. Although we may reject the first rule on the confidence basis, the second rule seems a valid rule and may be considered in the data analysis. However, when a statistical significance test is considered, such as statistical correlation between the *SM* and *CM* items, one would find that the two items are actually negatively correlated. This shows that the rule "CM $\Rightarrow$ SM" is misleading. This example shows not only the importance of considering negative association rules, but also the importance of statistical significance of the patterns discovered.

The problem of finding negative association rules is complex and computationally intensive as discussed in Section 1. A common solution to deal with the complexity is to focus the search on special cases of interest. Some techniques employ domain knowledge to guide the search, some are focusing on a certain type of rules of interest, while others are considering interestingness measures to mine for statistically significant patterns. We give more details about some approaches that have been proposed in the literature for mining association rules with negations.

Brin et al. [8] mentioned for the first time the notion of negative relationships in the literature. They proposed to use the chi-square test between two itemsets. The statistical test verifies the independence between the two itemsets. To determine the nature (positive or negative) of the relationship, a correlation metric is used. The negative association rules that could be discovered based on these measures are the following: $\neg X \Rightarrow Y$, $X \Rightarrow \neg Y$ and $\neg X \Rightarrow \neg Y$. One limitation for this method is that the computation of the χ^2 measure can become expensive in large and dense datasets.

Aggarwal and Yu [3, 1] introduced a new method for finding interesting itemsets in data. Their method is based on mining strongly collective itemsets. The collective strength of an itemset I is defined as follows:

$$C(I) = \frac{1 - v(I)}{1 - E[v(I)]} \times \frac{E[v(I)]}{v(I)} \tag{1.1}$$

where $v(I)$ is the violation rate of an itemset I and it is the fraction of violations over the entire set of transactions and $E[v(i)]$ is its expected value. An itemset I is in a violation of a transaction if only a subset of its items appear in that transaction. The collective strength ranges from 0 to ∞, where a value of 0 means that the items are perfectly negatively correlated and a value of ∞ means that the items are perfectly positively correlated. A value of 1 indicates that the value is exactly the same as its expected value, meaning statistical independence. The advantage of mining itemsets with collective strength is that the method finds statistical significant patterns. In addition, this model has good computational efficiency, thus being a good method in mining dense datasets. This property, along with the symmetry of collective strength measure, makes this method a good candidate for mining negative association rules in data.

In [19] the authors present a new idea to mine strong negative rules. They combine positive frequent itemsets with domain knowledge in the form of a taxonomy to mine negative associations. The idea is to reduce the search space, by constraining the search to the positive patterns that pass the minimum support threshold. When all the positive itemsets are discovered, candidate negative itemsets are considered based on the taxonomy used. They are considered interesting if their support is sufficiently different than the expected support. Association rules are generated from the negative itemsets if the interestingness measure of the rule exceeds a given threshold. The type of the rules discovered with this method are implications of the form $A \Rightarrow \neg B$. The issue with this approach is that it is hard to generalize since it is domain dependant and requires a predefined taxonomy. However, it should be noted that taxonomies exist for certain applications, thus making this method useful. A similar approach is described in [25].

Wu et al. [24] derived another algorithm for generating both positive and negative association rules. The negative association discovered in this paper are of the following forms: $\neg X \Rightarrow Y$, $X \Rightarrow \neg Y$ and $\neg X \Rightarrow \neg Y$. They add on top of the support-confidence framework another measure called *mininterest* for a better pruning of the frequent itemsets generated (the argument is that a rule $A \Rightarrow B$ is of interest only if $supp(A \cup B) - supp(A)supp(B) \geq mininterest$). *"Mininterest"* parameter is used to assess the dependency between the two itemsets considered, A and B are not independent if they satisfy the condition. The authors consider as itemsets of interest those itemsets that exceed minimum support and minimum interest thresholds. Although [24] introduces the *"mininterest"* parameter, the authors do not discuss how to set it and what would be the impact on the results when changing this parameter.

The algorithm proposed in [20, 21], named SRM (substitution rule mining), discovers a subset of negative associations. The authors develop an algorithm to discover negative associations of the type $X \Rightarrow \neg Y$. These association rules can be used to discover which items are substitutes for others in market basket analysis. Their algorithm discovers first what they call *concrete items*, which are those itemsets that have a high chi-square value and exceed the expected support. Once these itemsets are discovered, they compute the correlation coefficient for each pair of them. From those pairs that are negatively correlated, they extract the desired rules (of the type $X \Rightarrow \neg Y$, where Y is considered as an atomic item). Although interesting for the substitution items application, SRM is limited in the kind of rules that it can discover.

Antonie and Zaïane [7] proposed an algorithm to mine strong positive and negative association rules based on the Person's ϕ correlation coefficient. For the association rule $X \Rightarrow Y$, its ϕ correlation coefficient is as follows:

$$\phi = \frac{s(XY)s(\neg X \neg Y) - s(X \neg Y)s(\neg XY)}{\sqrt{(s(X)s(\neg X)s(Y)s(\neg Y))}} \tag{1.2}$$

In their algorithm, itemset and rule generation are combined and the relevant rules are generated on-the-fly while analyzing the correlations within each candidate itemset. This avoids evaluating item combinations redundantly. For each generated candidate itemset, all possible combinations of items are computed to analyze their correlations. In the end, only those rules generated from item combinations with strong correlations are considered. The strength of the correlation is indicated by a correlation threshold, either given as input or by default set to 0.5. If the correlation between item combinations X and Y of an itemset XY, where X and Y are itemsets, is negative, negative association rules are generated when their confidence is high enough. The produced rules have either the antecedent or the consequent negated: ($\neg X \Rightarrow Y$ and $X \Rightarrow \neg Y$), even if the support is not higher than the support threshold. However, if the correlation is positive, a positive association rule with the classical support-confidence idea is generated. If the support is not adequate, a negative association rule that negates both the antecedent and the consequent is generated when its confidence and support are high enough. They define the negative associations as *confined negative association rules*. A confined negative association rule is one of the following: $\neg X \Rightarrow Y$ or $X \Rightarrow \neg Y$, where the entire antecedent or consequent is treated as an atomic entity and the entire entity is either negated or not. These rules are a subset of the entire set of generalized negative association rules.

In [22], authors extend an existing algorithm for association rule mining, GRD (generalized rule discovery), to include negative items in the rules discovered. The algorithm discovers top-k positive and negative rules. GRD does not operate in the support confidence framework, it uses leverage and the number of rules to be discovered. The limitation of the algorithm is that it mines rules containing no more than 5 items (up to 4 items in the left hand side of the rule and 1 item in the right hand side of the rule).

Cornelis et al. [10] proposed a new Apriori-based algorithm (PNAR) that exploits the upward closure property of negative association rules that if support of $\neg X$ meets the minimum support threshold, then for every $Y \subseteq \mathcal{I}$ such that $X \cap Y = \emptyset$, $\neg(XY)$ also meets the support threshold. With this upward closure property, valid positive and negative association rules are defined in the form of $C_1 \Rightarrow C_2$, $C_1 \in \{X, \neg X\}, C_2 \in \{Y, \neg Y\}$, $X, Y \subseteq \mathcal{I}$, $X \cap Y = \emptyset$, if it meets the following conditions: (1) $s(C_1 \Rightarrow C_2) \geq minsup$; (2) $s(X) \geq minsup$, $s(Y) \geq minsup$; (3)

$conf(C_1 \Rightarrow C_2) \geq minconf$; (4) If $C_1 = \neg X$, then there does not exist $X' \subseteq X$ such that $s(\neg X' \Rightarrow C_2) \geq minsup$ (analogously for C_2). Then, the algorithm of mining both positive and negative valid association rules is built up around a partition of the itemset space by 4 steps: (1) generate all positive frequent itemsets $L(P_1)$; (2) for all itemsets I in $L(P_1)$, generate all negative frequent itemsets of the form $\neg(XY)$; (3) generate all negative frequent itemsets of the form $\neg X \neg Y$; (4) generate all negative frequent itemsets of the form $X \neg Y$ or $\neg XY$. The complete set of valid positive and negative association rules are derived after frequent itemsets are generated. No additional interesting measures are required in this support-confidence framework. Wang et al. [23] gave a more intuitive way to express the validity of both positive and negative association rules, the mining process is very similar to PNAR.

MINR [15] is a method that uses Fisher's exact test to identify itemsets that do not occur together by chance, i.e. with a statistical significant probability. Let X and Y denote the disjoint itemsets in the antecedent and consequent part of a rule, respectively. The probability that X and Y occur together with c times by chance is:

$$Pcc(c|n, s(X), s(Y)) = \frac{\binom{s(X)}{c}\binom{n-s(X)}{s(Y)-c}}{\binom{n}{s(Y)}} \tag{1.3}$$

where n is the total number of transactions. The chance threshold is calculated independently for each candidate itemset:

$$chance(n, s(X), s(Y), p) = min\{t|\sum_{i=0}^{i=t} Pcc(i|n, s(X), s(Y)) \geq p\} \tag{1.4}$$

Normally, for a positive association, p-value is set to be very high (usually 0.9999), on the other hand, for a negative association, p-value is set to be very low (usually 0.001). The whole algorithm develops in an iterative way with rule generation and rule pruning. An itemset with a support greater than the positive chance threshold is considered for positive rule generation, while itemset with a support less than the negative chance threshold is considered for negative rule generation. In this way, the algorithm discovers three different types of negative association rules in the form of $X \Rightarrow \neg Y$, $\neg X \Rightarrow Y$ and $\neg X \Rightarrow \neg Y$. The first two types $X \Rightarrow \neg Y$, $\neg X \Rightarrow Y$ can be generated from the negative itemsets if the rule $X \Rightarrow Y$ satisfies the negative chance threshold and minimum confidence threshold. On the other hand, the rules in the form of $\neg X \Rightarrow \neg Y$ are derived from the positive itemsets if they meet the positive chance threshold and minimum confidence threshold.

Kingfisher [12, 13] is an algorithm developed to discover positive and negative dependency rules. The dependency rule can be formulated on the basis of association rule, that the association rule $X \Rightarrow Y$ is defined as a dependency rule if $P(X,Y) \neq P(X)P(Y)$. The dependency is positive, if $P(X,Y) > P(X)P(Y)$; and negative, if $P(X,Y) < P(X)P(Y)$. Otherwise, the rule is an independent rule. The author concentrated on a specific type of dependency rules, the rules with only one single consequent attribute. It can be noticed that the negative dependency for the rules $X \Rightarrow Y$ or $\neg X \Rightarrow \neg Y$ are the same as the positive dependency for the rules $X \Rightarrow \neg Y$ and $\neg X \Rightarrow Y$, therefore, it is enough to only consider the positive dependency rules $X \Rightarrow \neg Y$ or $\neg X \Rightarrow Y$. The statistical dependency of the rule $X \Rightarrow Y$, is measured by Fisher's exact test, the p-value, can be calculated:

$$p_F(X \Rightarrow Y = y) = \sum_{i=0}^{min\{s(XY \neq y), s(\neg X, Y = y)\}} \frac{\binom{s(X)}{s(XY=y)+i}\binom{s(\neg X)}{s(\neg XY \neq y)+i}}{\binom{n}{s(Y=y)}} \tag{1.5}$$

where $y \in \{0,1\}$ denotes the presence or absent of Y, and n is the total number of transactions. It can also be observed that $p_F(X \Rightarrow \neg Y) = p_F(\neg X \Rightarrow Y)$, therefore, it is enough to consider the negative rules in the form of $X \Rightarrow \neg Y$. An important task of rule mining is to find the non-redundant rules. Rules are considered as redundant when they do not add new information to the remaining rules. In order to reduce the number of discovered rules, Kingfisher focused on finding non-redundant rules. The rule $X \Rightarrow Y = y$ is non-redundant, if there does not exist any rules in the form of $X' \Rightarrow Y = y$ such that $X' \subsetneq X$ and $p_F(X' \Rightarrow Y = y) < p_F(X \Rightarrow Y = y)$, otherwise, the rule is considered as redundant. However, the statistical dependency is not a monotonic property, it is impossible to do some frequency-based pruning as Apriori-like algorithms. A straightforward solution is to list all possible negative rules in the form of $X \Rightarrow \neg Y$ in the whole search space via an enumeration tree, and then calculate their p_F-values to see if they are significant. The items are ordered in an ascending order (by frequency) in the enumeration tree and the tree is traversed by a breadth-first manner. In this way, more general rules are checked before their specializations, therefore, it is possible that redundant specializations can be pruned without checking. If the task is to search for the top K rules, the threshold of p_F-value, needs to be updated consistently, when a new K-th top rule is found with a lower p_F-value. However, in both cases, the size of the whole search space is $|\mathcal{P}(\mathcal{I})|$, where $\mathcal{P}(\mathcal{I})$ is the power set of $\mathcal{I}$, it grows exponentially with the size of attributes. In order to reduce the search space, the author fully exploits the property of p_F-value, and describes the basic branch-and-bound search by introducing three lower bounds for the measure of p_F-value, therefore, some insignificant rules can be pruned without further checking. Apart from the three lower bounds of p_F-value, anther two pruning strategies (pruning by minimality and pruning by principles of Lapis philosophorum) are also introduced to speed up the search.

3. Associative Classification and Negative Association Rules

Associative classifiers are classification models that use association rules discovered in the data to make predictions [17, 16, 5]. Training data is transformed into transactions and constrained association rules are discovered from these transactions. The constraints limit the frequent itemsets to those including a class label, and limit the rules to those with a class label as the consequent. After a pruning phase to remove noisy and redundant rules, the remaining rules, classification rules, are used as a learned classification model. Negative association rules have been used for associative classifiers [6] and it was shown that the performance of the classifiers improved when negative association rules were employed in the training and the classification process. The negative association rules generated and used in addition to the positive rules are of the form $\neg X \Rightarrow Y$ (if feature X absent then class Y) or $X \Rightarrow \neg Y$ (if feature X present then cannot be class Y), where $|Y| = 1$ and Y is a class label.

4. Conclusions

In this chapter we have surveyed some methods proposed in the literature for mining association rules with negations. Although the problem of mining these types of rules is an interesting and challenging one there is a limited body of work. None of the existing methods find all the possible negative association rules. This is due to the complexity and size of the problem. A user should choose the algorithm that is most useful for the application considered. If a taxonomy is available or substitution rules are useful, the algorithms in [19] and [20, 21] are good candidates. If a user is interested in all the negative associations between pairs of itemsets, the methods proposed in [24] and [7] should be considered. Another research direction that can be useful in some situations is the mining of top-K rules with positive and negative items. This is investigated in [22] and [12, 13] which may be of interest to users who want to investigate and use a limited number of rules.

References

[1] C. C. Aggarwal and P. S. Yu. Mining associations with the collective strength approach. *IEEE Trans. on Knowl. and Data Eng.*, 13(6):863–873, November 2001.

[2] Charu C. Aggarwal and Philip S. Yu. Mining large itemsets for association rules. *IEEE Data Eng. Bull.*, 21(1):23–31, 1998.

[3] Charu C. Aggarwal and Philip S. Yu. A new framework for itemset generation. In *Proceedings of the Seventeenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, PODS '98, pages 18–24, 1998.

[4] Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. Mining association rules between sets of items in large databases. In *Proc. of SIGMOD*, pages 207–216, 1993.

[5] M-L Antonie and Osmar R Zaïane. Text document categorization by term association. In *Proc. of ICDM*, pages 19–26, 2002.

[6] M-L Antonie and Osmar R Zaïane. An associative classifier based on positive and negative rules. In *Proc. of DMKD*, pages 64–69, 2004.

[7] Maria-Luiza Antonie and Osmar R Zaïane. Mining positive and negative association rules: An approach for confined rules. In *Proc. of PKDD*, pages 27–38, 2004.

[8] Sergey Brin, Rajeev Motwani, and Craig Silverstein. Beyond market basket: Generalizing association rules to correlations. In *Proc. SIGMOD*, pages 265–276, 1997.

[9] Sergey Brin, Rajeev Motwani, Jeffrey D Ullman, and Shalom Tsur. Dynamic itemset counting and implication rules for market basket data. In *Proc. of SIGMOD*, pages 255–264, 1997.

[10] Chris Cornelis, Peng Yan, Xing Zhang, and Guoqing Chen. Mining positive and negative association rules from large databases. In *Proc. of CIS*, pages 1–6, 2006.

[11] Bart Goethals and Mohammed J Zaki. Fimi03: Workshop on frequent itemset mining implementations. In *Third IEEE International Conference on Data Mining Workshop on Frequent Itemset Mining Implementations*, pages 1–13, 2003.

[12] Wilhelmiina Hamalainen. Efficient discovery of the top-k optimal dependency rules with fisher's exact test of significance. In *Proc. of ICDM*, pages 196–205, 2010.

[13] Wilhelmiina Hamalainen. Kingfisher: an efficient algorithm for searching for both positive and negative dependency rules with statistical significance measures. *Knowl. Inf. Syst.*, 32(2):383–414, 2012.

[14] Jiawei Han, Jian Pei, and Yiwen Yin. Mining frequent patterns without candidate generation. In *Proc. of SIGMOD*, pages 1–12, 2000.

[15] Yun Sing Koh and Russel Pears. Efficiently finding negative association rules without support threshold. In *Proc. of Australian AI*, pages 710–714, 2007.

[16] Wenmin Li, Jiawei Han, and Jian Pei. CMAR: Accurate and efficient classification based on multiple class-association rules. In *Proc. of ICDM*, pages 369–376, 2001.

[17] Bing Liu, Wynne Hsu, and Yiming Ma. Integrating classification and association rule mining. In *Proc. of SIGKDD*, pages 80–86, 1998.

[18] Paul David McNicholas, Thomas Brendan Murphy, and M ORegan. Standardising the lift of an association rule. *Comput. Stat. Data Anal.*, 52(10):4712–4721, 2008.

[19] Ashok Savasere, Edward Omiecinski, and Shamkant Navathe. Mining for strong negative associations in a large database of customer transactions. In *Proc. of ICDE*, pages 494–502, 1998.

[20] Wei-Guang Teng, Ming-Jyh Hsieh, and Ming-Syan Chen. On the mining of substitution rules for statistically dependent items. In *Proc. of ICDM*, pages 442–449, 2002.

[21] Wei-Guang Teng, Ming-Jyh Hsieh, and Ming-Syan Chen. A statistical framework for mining substitution rules. *Knowl. Inf. Syst.*, 7(2):158–178, 2005.

[22] D. R. Thiruvady and G. I. Webb. Mining negative rules using grd. In *Proc. of PAKDD*, pages 161–165, 2004.

[23] Hao Wang, Xing Zhang, and Guoqing Chen. Mining a complete set of both positive and negative association rules from large databases. In *Proc. of PAKDD*, pages 777–784, 2008.

[24] Xindong Wu, Chengqi Zhang, and Shichao Zhang. Efficient mining of both positive and negative association rules. *ACM Trans. on Inf. Syst.*, 22(3):381–405, 2004.

[25] Xiaohui Yuan, Bill P Buckles, Zhaoshan Yuan, and Jian Zhang. Mining negative association rules. In *Proc. of ISCC*, pages 623–628, 2002.

[26] Mohammed J Zaki. Parallel and distributed association mining: A survey. *IEEE Concurrency: Special Issue on Parallel Mechanisms for Data Mining*, 7(4):14–25, 1999.