Efficient Mining of Top Correlated Patterns Based on Null-Invariant Measures

Sangkyum Kim[1], Marina Barsky[2], and Jiawei Han[1]

[1] University of Illinois at Urbana-Champaign, Urbana, IL, USA
{kim71, hanj}@illinois.edu
[2] Simon Fraser University, BC, Canada
marina_barsky@sfu.ca

Abstract. Mining strong correlations from transactional databases often leads to more meaningful results than mining association rules. In such mining, null (transaction)-invariance is an important property of the correlation measures. Unfortunately, some useful null-invariant measures such as $Kulczynski$ and $Cosine$, which can discover correlations even for the very unbalanced cases, lack the (anti)-monotonicity property. Thus, they could only be applied to frequent itemsets as the post-evaluation step. For large datasets and for low supports, this approach is computationally prohibitive. This paper presents new properties for all known null-invariant measures. Based on these properties, we develop efficient pruning techniques and design the Apriori-like algorithm NICoMiner for mining strongly correlated patterns *directly*. We develop both the threshold-bounded and the top-k variations of the algorithm, where top-k is used when the optimal correlation threshold is not known in advance and to give user control over the output size. We test NICoMiner on real-life datasets from different application domains, using $Cosine$ as an example of the null-invariant correlation measure. We show that NICoMiner outperforms support-based approach more than an order of magnitude, and that it is very useful for discovering top correlations in itemsets with low support.

1 Introduction

One of the central tasks in data mining is finding correlations in binary relations. Typically, this is formulated as a *market basket* problem [2], where there is a set of baskets (*transactions*), each of which is a set of items purchased together. The goal is to find correlations between items, based on their recurrent co-appearance in the same transaction. The usefulness of the correlations based on the market-basket concept was demonstrated in many different application domains such as climate studies [18], public health [5], or bioinformatics [9, 21]. With the trend of collecting more and more digitized data, the discovery of meaningful correlations offers a new insight into relationships between objects in these large data collections.

In this paper we study the problem of finding groups of items with the top correlations for a given dataset. This implies that we need to rank the correlations. There is no canonical way to assess the degree of the correlation. This seems to be problem-specific and cannot be captured by a single correlation measure which is the best for all cases. As a result, a number of correlation measures has been proposed [8, 16, 17, 19].

Table 1. The same dataset contains coffee c, milk m, popcorn p, and soda s. The total number of transactions is $N = 100,000$. According to $Lift$, correlation(p, s) is significantly stronger than correlation(m, c). Assessed by null-invariant measures, correlation(m, c) is always much stronger than correlation(p, s), which is more meaningful, since cm occur together in much more transactions than ps.

mc	$\bar{m}c$	$m\bar{c}$	$\bar{m}\bar{c}$	$Lift(m, c)$	$Cosine(m, c)$
$10,000$	$1,000$	$1,000$	$88,000$	8.26	0.91
ps	$\bar{p}s$	$p\bar{s}$	$\bar{p}\bar{s}$	$Lift(p, s)$	$Cosine(p, s)$
$1,000$	$1,000$	$1,000$	$97,000$	25.00	0.50

In this work we limit ourselves to *null (transaction)-invariant* [8, 16, 17, 19] correlation measures based on conditional probabilities. They quantify the degree of mutual relationships between items in a group without taking into account the items outside the group in question. For example, if we are computing the correlation between coffee (c) and milk (m), a null-invariant measure does not depend on the number of transactions which contain neither coffee nor milk - *null transactions* with respect to c and m. Thus, these measures are *null (transactions)-invariant*.

The importance of null-invariance for uncovering meaningful relationships between objects was analyzed in [19]. If we use correlation measures which are not null-invariant, the relationships between objects may appear or disappear simply by changing the number of transactions which do not contain items in question.

Even for ranking correlations within *the same* dataset we cannot rely on expectation-based (not null-invariant) measures, since they produce inconsistent and controversial results, as shown in a sample dataset, presented in Table 1. Here the degree of the correlation of two pairs of items is assessed by $Lift$ (not null-invariant) and by $Cosine$ (null-invariant). The items in pair (c, m) are intuitively more correlated than in (p, s), since they occur together in 83% of all transactions with c or m, while (p, s) occur together only in 33%. This is reflected in $Cosine$ values 0.91 and 0.50 respectively. However, according to $Lift$, correlation in pair (p, s) is significantly larger than in (c, m), which contradicts our intuition and the common sence. Hence, in order to produce meaningful and consistent top correlations we require from the correlation measure to be null-invariant.

The five known null-invariant correlation measures are *All Confidence*, *Coherence*, *Cosine*, *Kulczynski* and *Max Confidence* [19]. The degree of the correlation is represented as a real number between 0 and 1.

For different datasets, the strongest correlations may have different values. It is not always appropriate to set a correlation threshold such as 0.5 for all datasets. Hence, it is important to be able to mine the top correlated patterns, instead of patterns with correlation larger than a given threshold. This leads to a problem of mining top-k null-invariant correlations. An example of top-10 correlations, which we extracted from the titles of the database-related publications [15], is shown in Table 2. Note that the correlation here is not expected to be very high, since people use different word combinations to describe even similar ideas. Nevertheless, the top correlated patterns represent quite meaningful concepts.

Table 2. Top-10 highly correlated term groups from the paper titles in the DB-DM-IR subset [15] of the DBLP dataset [1] with minimum support $\theta = 0.02\%$.

	Pattern	Support	$Cosine$
1	$object, orient, database$	748	0.19
2	$sense, word, disambiguation$	26	0.18
3	$support, vector, machine$	122	0.17
4	$enforcement, law, coplink$	7	0.16
5	$nearest, neighbor, search$	74	0.13
6	$reverse, nearest, neighbor$	23	0.13
7	$server, sql, microsoft$	25	0.12
8	$retrieval, cross, language$	187	0.11
9	$model, relationship, entity$	139	0.11
10	$random, field, conditional$	13	0.10

Finding the itemsets with the highest correlations is not trivial. The naïve approach would be to extract all frequent itemsets, and then to rank them based on the correlation within each frequent itemset. Unfortunately, this approach is valid only for itemsets with high support, and in this case the discovered correlations mostly represent the common knowledge. If we are to discover interesting correlations in itemsets with low support, the number of such itemsets can reach several thousands or even millions, thus making the post-evaluation approach computationally infeasible. In addition, the degree of the correlation between items can be higher in itemsets with lower support. This is especially true for such problems as finding correlations between words or finding correlations between authors in a publication database. Therefore, we want to design an efficient framework in which we would be able to find the groups of the top correlated items with low support, without first collecting all frequent itemsets.

The algorithms for the direct mining of interesting null-invariant patterns exist. For example, the direct computation based on *All Confidence* and *Coherence* was proposed in [10]. However, it is applicable only for null-invariant measures which have the *anti-monotonicity* property. Out of five measures, only *All Confidence* and *Coherence* are anti-monotonic. Unfortunately, using only *All Confidence* or *Coherence* may not be appropriate for cases involving unbalanced supports, which was demonstrated in [19]. Strong correlations for such unbalanced cases can be captured if we evaluate the relationships as an *average* of conditional probabilities. For such cases, two measures $Cosine$ and $Kulczynski$ are the most appropriate ones.

Both $Cosine$ and $Kulczynski$ represent the means of conditional probabilities: the geometric mean and the arithmetic mean, respectively. For an itemset $A = \{a_1, \cdots, a_n\}$:

$$Cosine(A) = \sqrt[n]{\prod_{i=1}^{n} P(A|a_i)}, \text{ and } Kulczynski(A) = \frac{1}{n}\sum_{i=1}^{n} P(A|a_i)$$

where $P(A|a_i)$ is a conditional probability of A given a_i.

Being an average, $Cosine$ and $Kulczynski$ do not possess neither monotonicity nor anti-monotonicity properties, and the Apriori principle cannot be applied for efficient pruning based on these measures. Hence, the discovery of all patterns with high $Cosine$ and $Kulczynski$ values poses a great computational challenge, especially for itemsets with low support. To solve this challenging problem, we develop an efficient algorithmic framework based on new pruning properties *common to all null-invariant measures*, but especially valuable for $Cosine$ and $Kulczynski$.

Specifically, our study makes the following contributions.

1. We discover new mathematical properties common to all null-invariant measures.
2. Based on these properties, we design a new pruning strategy which relies mainly on correlation measures rather than on support.
3. We propose new algorithm NICoMiner for *Null Invariant Correlation Mining* and demonstrate its high efficiency on a wide variety of synthetic and real-life datasets.
4. In order to make NICoMiner self-adjustable to the level of the correlations existing in different datasets, and to give user the control over an output size, we develop the top-k version of NICoMiner, which allows us to find the top-k correlated itemsets without specifying the correlation threshold.
5. Finally, we show meaningful correlations discovered by NICoMiner in itemsets with low support. It is hard or somtetimes impossible to find such correlations using the support-based method alone.

The remainder of the paper is organized as follows. In Section 2 we formally define correlated patterns. In Section 3 we describe the new properties of null-invariant measures, and in Section 4 we present our new algorithm. Section 5 is a detailed report on our experiments with synthetic and real datasets. Related work is presented in Section 6, followed by conclusions and future work in Section 7.

We start by introducing a few concepts. Note that for the rest of the paper we use $Cosine$ as a representative of null-invariant correlation measures.

2 Preliminaries

Let $\mathcal{I}$ be a set of items. We define an *itemset* $A = \{a_1, \ldots, a_n\}$ to be a subset of n items from $\mathcal{I}$. Let $\mathcal{T}$ be a set of transactions where each *transaction* is a subset of $\mathcal{I}$. The *support* of an itemset A, $sup(A)$, is defined to be the number of transactions containing all items in A. An itemset A is *frequent* if its support $sup(A)$ is no less than a user-defined *minimum support threshold* θ.

Cosine in terms of supports is explicitly defined as:

$$cos(A) = \frac{sup(A)}{\sqrt[n]{sup(a_1) \times \cdots \times sup(a_n)}}. \tag{1}$$

We define the correlation between items in an itemset as follows:

Definition 1. *An itemset* $A = \{a_1, \ldots, a_n\}$ *is correlated if* $cos(A) \geq \gamma$ *for a given minimum correlation threshold* γ.

The *problem of threshold-based correlation mining* is to find all correlated itemsets for the correlation threshold γ. But, even for the experts, it is sometimes hard to set the proper value of γ. For such cases, it would be helpful to know several patterns with the highest correlation values. This is the *problem of top-k correlation mining*, where only k patterns with the highest correlation values are presented to the user. Note that a minimum correlation threshold γ is not required for top-k correlation mining.

Table 3. A small transactional database of 6 transactions and 6 items.

TID	Transaction
T_1	a_1, a_3, a_4, a_5, a_6
T_2	a_3, a_5, a_6
T_3	a_2, a_4
T_4	a_1, a_4, a_5, a_6
T_5	a_3, a_6
T_6	a_2, a_4, a_5

The lack of the anti-monotonicity property for $Cosine$ poses significant challenges for mining top correlated patterns. This can be illustrated by the following example.

Example 1. Consider small database of transactions shown in Table 3.
Correlation value for 2-itemset $X = \{a_4, a_6\}$ is $cos(X) = 0.50$. 3-itemset $X' = \{a_1, a_4, a_6\}$ is a superset of X, and its correlation is $cos(X') = 0.63$. Thus, $Cosine$ is *not anti-monotonic*. For the correlation threshold $\gamma = 0.60$, we cannot prune all supersets of X, even though the correlation in X is below γ.
Correlation value for 2-itemset $Y = \{a_1, a_4\}$ is $cos(Y) = 0.71$. 3-itemset $Y' = \{a_1, a_4, a_5\}$ is a superset of Y, and its correlation is $cos(Y') = 0.63$. Thus, $Cosine$ is also *not monotonic*. Knowing that Y is a correlated itemset, we cannot assume that all supersets of Y are also correlated. This shows that finding that $cos(X) < \gamma$ or that $cos(Y) \geq \gamma$ does not tell us anything about the correlation value in their supersets, and hence we cannot stop the extension of X or Y to larger itemsets. ■

3 New Properties of Null-invariant Measures

In this section, we describe useful mathematical properties, common to all known null-invariant measures. These properties are the basis for an efficient pruning used in the NICoMiner algorithm. Our framework is based on the level-wise Apriori algorithm, where each level n corresponds to itemsets of n items.

3.1 Level-based properties

The relationships between $Cosine$ of n-itemset A and $Cosine$ values of all its subsets of size n-1 are captured by the following lemma:

Lemma 1. *For any n-itemset $A = \{a_1, \cdots, a_n\}$ and a set S of all A's (n-1)-subitemsets:*

$$cos(A) \leq \max_{B \in S}(cos(B)). \tag{2}$$

Proof. Since the maximum is not smaller than the geometric mean:

$$\max_{B \in S}(\cos(B)) \geq \sqrt[n]{\cos(a_1, \cdots, a_{n-1}) \times \cdots \times \cos(a_2, \cdots, a_n)}. \tag{3}$$

Then by the definition of $Cosine$ and from the anti-monotonicity of support:

$$
\begin{aligned}
&\max_{B\in S}(\cos(B)) && (4)\\
&\geq \sqrt[n]{\frac{\mathrm{sup}(a_1,\cdots,a_{n-1})}{\sqrt[n-1]{\mathrm{sup}(a_1)\times\cdots\times\mathrm{sup}(a_{n-1})}}\times\cdots\times\frac{\mathrm{sup}(a_2,\cdots,a_n)}{\sqrt[n-1]{\mathrm{sup}(a_2)\times\cdots\times\mathrm{sup}(a_n)}}} && (5)\\
&\geq \frac{sup(a_1,\cdots,a_n)}{\sqrt[n]{sup(a_1)\times\cdots\times sup(a_n)}} && (6)\\
&= \cos(A). && (7)
\end{aligned}
$$

■

Lemma 1 presents an upper bound of *Cosine* in terms of *Cosine* values of subitemsets. A simple corollary follows from Lemma 1: once $Cosine$ values of all (n-1)-subitemsets of $A = \{a_1, \cdots, a_n\}$ are less than γ, $cos(A) < \gamma$. However, this does not mean that A and its supersets can be pruned. There might be a superset of A, $A' = \{a_1, \cdots, a_n, a_{n+1}\}$ with $cos(A') \geq \gamma$, because the condition of the lemma may not be satisfied due to the newly added item a_{n+1}.

Nevertheless, Lemma 1 leads to a simple condition for the termination of correlation pattern growth. Even though *Cosine* for individual patterns is not anti-monotonic, there is a level-based property which we for convenience call *level-anti-monotonicity*. Namely, if all patterns at level n have $Cosine$ values less than γ, then all their supersets have $Cosine$ less than γ.

Let $\mathcal{I}_n$ be set of all n-itemsets at level n. We denote the maximum cosine value for all itemsets in $\mathcal{I}_n$ by $maxCos(\mathcal{I}_n)$. We prove that:

Theorem 1. *Cosine* is *level-anti-monotonic.*

Proof. Let $\mathcal{I}_{n+1}$ be set of all (n+1)-itemsets at level n+1, and let A' be an itemset from $\mathcal{I}_{n+1}$ with maximum cosine value. Let A be an n-subitemset of A' whose cosine value is the maximum from all n-subitemsets of A'. Then, by Lemma 1,

$$maxCos(\mathcal{I}_n) \geq cos(A) \geq cos(A') = maxCos(\mathcal{I}_{n+1}). \qquad (8)$$

■

From Theorem 1 follows:

Corollary 1. Termination of pattern growth (TPG)
If all itemsets at level n are not correlated, then all itemsets at level n' are not correlated for any $n' \geq n$.

Note that TPG holds for all five null-invariant correlation measures. The proofs are essentially similar to that of Cosine, and we omit them due to the page limit.
To demonstrate the termination of pattern growth, consider the following example.

Example 2. For a database described in Table 3 with the *minimum support threshold* $\theta = 2$, there exist 5 frequent 3-itemsets shown in Table 4. Assuming the *minimum correlation threshold* $\gamma = 0.75$, all 3-itemsets have correlation below the threshold. Then, based on TPG, we do not need to mine n-itemsets for $n \geq 3$, and therefore pattern growth terminates. ■

Table 4. $Cosine$ values for all five frequent 3-itemsets from the database in Table 3 ($\theta = 2$). If $\gamma = 0.75$, we can terminate correlation pattern growth according to TPG.

Pattern	a_1, a_4, a_5	a_1, a_4, a_6	a_1, a_5, a_6	a_3, a_5, a_6	a_4, a_5, a_6
$Cosine$	0.63	0.63	0.63	0.55	0.5

3.2 Properties based on a single item

Since *Cosine* is not anti-monotonic, we cannot prune n-itemset A even if A is not correlated. But, in the following, we claim that for some item a from $\mathcal{I}$, knowing correlation values of all (n-1)-itemsets containing a allows to prune n-itemsets containing a.

Lemma 2. *For n-itemset $A = \{a_1, \cdots, a_n\}$, and all its subsets of size n-1 which share the same single item a, if (1) all these subsets are not correlated and (2) the support of at least one item $a_i \neq a$ in A is greater than or equal to $sup(a)$, then A cannot be correlated.*

Proof. Assume $a_1 = a$ and $sup(a_n) = \max\{sup(a_1), \cdots, sup(a_n)\}$, without loss of generality. By simple algebra, we can show that

$$\sqrt[n-1]{sup(a_1) \times \cdots \times sup(a_{n-1})} \leq \sqrt[n]{sup(a_1) \times \cdots \times sup(a_n)}. \tag{9}$$

Then

$$\begin{aligned}
\cos(A) &= \frac{sup(A)}{\sqrt[n]{sup(a_1) \times \cdots \times sup(a_{n-1}) \times sup(a_n)}} && (10)\\
&\leq \frac{sup(A)}{\sqrt[n-1]{sup(a_1) \times \cdots \times sup(a_{n-1})}} && (11)\\
&\leq \frac{sup(A - \{a_n\})}{\sqrt[n-1]{sup(a_1) \times \cdots \times sup(a_{n-1})}} && (12)\\
&\leq \cos(A - \{a_n\}) && (13)\\
&< \gamma, && (14)
\end{aligned}$$

where $A - \{a_n\}$ represents the (n-1)-subitemset of A which does not contain an item a_n with the maximum support. ■

In other words, if we know that all sub-itemsets containing item a are not correlated, we know that adding another item cannot make any of them correlated, given this new item has support not less than $sup(a)$.
Based on Lemma 2, we can claim the following theorem:

Theorem 2. *Let item a be an item with the smallest support among all single items in the database. If all itemsets at level n containing a are not correlated, then all n'-itemsets containing a are not correlated for all $n' \geq n$.*

Proof. Each $(n + 1)$-itemset A' which contains a can be thought of as an extension of some n-itemset containing a with an item a_{n+1}, which has the largest support among all the items in A' (since we know that support of a is not the largest). Then, by Lemma 2, $cos(A') < \gamma$. Since all n-itemsets containing item a have $Cosine$ value less than γ, all

$(n+1)$-itemsets containing item a have $Cosine$ value less than γ. Iteratively applying Lemma 2, now to extension of $(n+1)$-itemsets into $(n+2)$-itemsets, containing a, we conclude that none of the n'-itemsets containing a is correlated, for $n' \geq n$ ■

Based on Theorem 2, we can derive a condition for pruning patterns which contain the same single item a. For convenience, we call the pruning of a non-promising single item and its supersets at level n the *single-item-based pruning* (SIBP).

Corollary 2. Single-Item Based Pruning (SIBP)
If the maximum $Cosine$ value for n-itemsets containing item a is less than γ, and a has the smallest support between single items existing in the database, then all n'-itemsets containing a can be pruned for $n' \geq n$.

For the level-wise processing, which we use here, such an item can be removed from the database. After removing it, we have a new, smaller database, and we can apply the same principle to the next item, which has the smallest support in this new database.

Again, SIBP holds for all null-invariant correlation measures. We skip the proofs due to the page limit, but the proofs are very similar or easier than that for $Cosine$.

The application of the SIBP principle can be illustrated on the following example.

Example 3. Consider the sample database in Table 3 ($\theta = 2$, $\gamma = 0.75$). First, all single frequent items $a_1 \ldots a_6$ are sorted by support. Then, while counting itemsets at level 2, the maximum $Cosine$ value of 2-item supersets of each a_i is recorded. For this example, we have: a_1 (sup:2, maxCos:0.71), a_2 (sup:2, maxCos:0.71), a_3 (sup:3, maxCos:0.87), a_4 (sup:4, maxCos:0.75), a_5 (sup:4, maxCos:0.75), and a_6 (sup:4, maxCos:0.86). Now, based on the SIBP principle, we can safely prune all 2-itemsets containing item a_1 (or item a_2), and we do not need to generate the following 3-itemsets in Table 4: $\{a_1, a_4, a_5\}$, $\{a_1, a_4, a_6\}$, and $\{a_1, a_5, a_6\}$. ■

Table 5. Frequent 2-itemsets from the database in Table 3 ($\theta = 2$). For $\gamma = 0.75$, all supersets of a_1 and a_2 are not correlated according to SIBP.

Pattern	a_1, a_4	a_1, a_5	a_1, a_6	a_2, a_4	a_3, a_5	a_3, a_6	a_4, a_5	a_4, a_6	a_5, a_6
$Cosine$	0.71	0.71	0.71	0.71	0.58	0.87	0.75	0.5	0.75

4 NICoMiner Algorithm

The general framework of NICoMiner is an Apriori-like level-wise (breadth-first) computation. The candidate itemsets for level n are generated from the itemsets on level n-1. Then the support and $Cosine$ are computed for all candidate n-itemsets, and they are pruned based on support and SIBP. The remaining n-itemsets are the candidates for the next level $n+1$. If all patterns at level n are not correlated, the algorithm terminates (TPG).

4.1 Threshold-based correlation mining

Here we present the correlation mining algorithm (Algorithm 1) for the case when a minimum correlation threshold γ is given. The pruning properties developed in the previous section allow to prune uncorrelated patterns in addition to the non-frequent patterns. In practice, the pruning power of TPG and SIBP is extremely high, which allows setting very low support thresholds.

Algorithm 1: The threshold-based version of the NICoMiner Algorithm.

```
input : a transactional database $\mathcal{D} = \{T_1, T_2, ..., T_n\}$, minimum correlation threshold $\gamma$,
        minimum support threshold $\theta$
output: all patterns with correlation at least $\gamma$
scan $\mathcal{D}$ and find all frequent 1-itemsets $\mathcal{I}_1$;
for $n = 2, \cdots$ do
    generate candidate itemsets $\mathcal{I}_n$ from $\mathcal{I}_{n-1}$;
    scan $\mathcal{D}$ to compute support and $Cosine$ values of itemsets in $\mathcal{I}_n$;
    output frequent $n$-itemsets with $Cosine \geq \gamma$;
    prune itemsets from $\mathcal{I}_n$ based on SIBP and support;
    if ($maxCos(\mathcal{I}_n) < \gamma$) OR (no frequent $n$-itemsets) then break;
end
```

4.2 Top-k correlation mining

Without knowing what is the top level of correlations in a given dataset, it is hard to choose an appropriate correlation threshold γ. Running the top-k version of our algorithm helps in this situation. After this, the information about the top correlations can be used to set a meaningful threshold in order to collect all interesting patterns. Often, the set of the top-k correlated patterns is interesting in its own right.

In order to find top-k correlated patterns, we can iteratively run the threshold-based NICoMiner until it produces at least k patterns. If in the current iteration the size of the output is less than k, we can decrease the correlation threshold γ and run Algorithm 1 with this new parameter. We implemented this iterative top-k approach, halving the correlation threshold in each iteration.

However, guessing the correlation threshold γ which produces close to k patterns is not efficient. Not only we need to repeat the entire computation several times, but if we accidentally set γ too low, we have an expensive computation and a huge output, while we are interested only in k patterns.

Much more efficient approach would be to adjust threshold γ throughout the mining process until we get top-k correlated patterns (Algorithm 2). Here, instead of using a fixed threshold value, we start with $\gamma = 0.0$ and keep top k correlated itemsets from the itemsets processed so far. Once we mine more than k patterns, we set γ to the k-th largest $Cosine$ value, and the pattern growth continues with this new, higher correlation threshold. Since the correlation threshold is constantly increasing, the termination of the pattern growth is reached earlier than in the method with the constant initial correlation threshold.

Algorithm 2: The top-k version of NICoMiner

```
input : a transactional database $\mathcal{D} = \{T_1, T_2, ..., T_n\}$, number $k$, minimum support
        threshold $\theta$
output: set $TOP$ of top-$k$ correlated patterns

$\gamma \leftarrow 0; TOP \leftarrow \emptyset$;
scan $\mathcal{D}$ and find all frequent 1-itemsets $\mathcal{I}_1$;
for $n = 2, \cdots$ do
    generate candidate itemsets $\mathcal{I}_n$ from $\mathcal{I}_{n-1}$;
    scan $\mathcal{D}$ to compute support and $Cosine$ values of all candidate $k$-itemsets;
    $TOP \leftarrow TOP \cup \{$correlated $n$-itemsets$\}$;
    if $|TOP| \geq k$ then
        keep only top-$k$ in $TOP$;
        $\gamma \leftarrow$ minimum $Cosine$ value in $TOP$;
    end
    prune itemsets from $\mathcal{I}_n$ based on SIBP and support;
    if ($maxCos(\mathcal{I}_n) < \gamma$) OR (no frequent $n$-itemsets) then break;
end
```

5 Experiments

In this section, we present experimental results for two versions of NICoMiner: one computes all patterns with the correlation above the minimum correlation threshold and the other finds the top-k correlations. All experiments were performed on a Linux (ver 2.6.18) server with quad core Xeon 5500 processors and 48 GB of main memory.

For the threshold-based version, we used the support-based pruning as the baseline. To evaluate the pruning power of each new technique, we added to the baseline algorithm the pattern growth termination (TPG), and then enhanced it with the single-item-based pruning (SIBP). The latter represents the full version of the threshold-based NICoMiner.

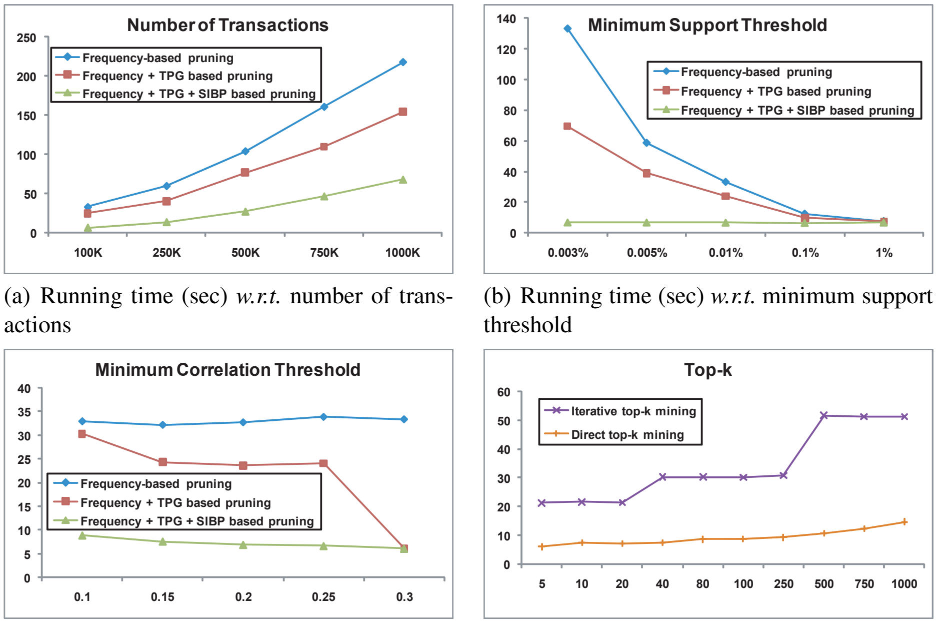
For the top-k version, we compared our direct top-k NICoMiner with the naïve iterative top-k mining, which uses multiple iterations of the threshold-based version, halving the correlation threshold in each iteration, until the output contains at least k patterns.

5.1 Synthetic datasets

Synthetic datasets for our experiments were generated by the generator used in [14]. The default parameters are: number of transactions $N = 100K$, average number of items per transactions $W = 5$, number of distinct items $|\mathcal{I}| = 1K$. The default set of thresholds for all experiments is as follows: minimum support threshold $\theta = 0.01\%$, and minimum correlation threshold $\gamma = 0.2$.

For the correlation-based version of NICoMiner we show the dependence of the running time on the following parameters: number of transactions, minimum support threshold, and minimum correlation threshold.

Number of transactions: The results in Figure 1(a) show the comparative performance for 5 different synthetic datasets with number of transactions varying from



(a) Running time (sec) *w.r.t.* number of transactions

(b) Running time (sec) *w.r.t.* minimum support threshold

(c) Running time (sec) *w.r.t.* minimum correlation threshold

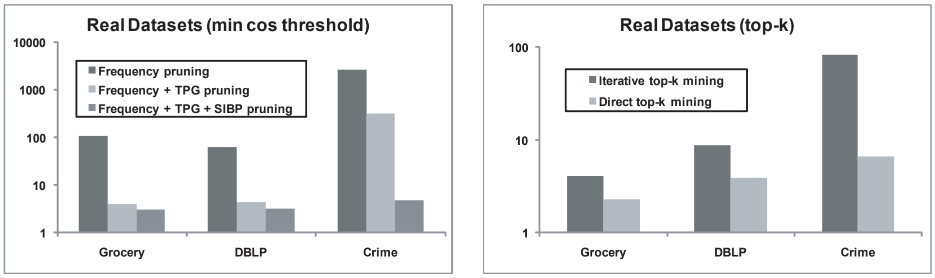
(d) Running time (sec) *w.r.t.* top-k

Fig. 1. Performance results for synthetic datasets.

100K to 1M. For all methods, the running time shows linear dependency on N, which means that the size of a dataset is not the limiting parameter for the performance of NICoMiner.

Minimum support threshold: In Figure 1(b), we evaluated the performance of our algorithm for various minimum support threshold values. As the threshold becomes lower, frequency-based pruning deteriorates exponentially. Adding TPG makes the baseline algorithm about two times faster, but the performance still degrades for low support thresholds. On the other hand, the full version of NICoMiner demonstrates consistently high performance. For the lowest minimum support threshold 0.003%, our algorithm is more than an order of magnitude faster than two other methods. This demonstrates the main power of our algorithm, which is meant for finding correlated patterns with low supports.

Minimum correlation threshold: In Figure 1(c), we show the effect of the minimum correlation threshold. Frequency-based pruning does not depend on the minimum correlation threshold, since there is no pruning based on correlation values. The termination of pattern growth (TPG) cannot be applied before all correlations at some level has been evaluated. For the largest correlation threshold $\gamma = 0.3$, the algorithm terminates after level 2 (all 2-itemsets are below threshold), while for the lowest correlation threshold $\gamma = 0.1$, it continues up to level 4. This explains the difference in the running time. For $\gamma = 0.1$, the full NICoMiner also stops at level 4, however it generates much less candidates due to the high pruning power of SIBP.



(a) Running time (sec) for threshold-based version.

(b) Running time (sec) for top-k version (k=100).

Fig. 2. Performance results for real datasets.

Top-k: In Figure 1(d), we compare the iterative and the direct top-k correlation mining for various values of k. Both approaches used all pruning properties for maximum performance. As expected, the direct approach was faster than the iterative approach. The gap in performance becomes bigger as k grows. This is because more iterations are performed by the iterative method before the output contains at least k patterns.

5.2 Real datasets

We tested NICoMiner applying the market basket concept to three real-life datasets. The performance results are presented in Figure 2. In Figure 2(a) we compare the efficiency of different pruning methods with the baseline pruning by support, and in Figure 2(b) we compare the direct top-k version with the iterative top-k mining.

1. The GROCERIES dataset [6, 7] ($9,800$ transactions) represents 1-month of the point-of-sale transactions in the local grocery store. This dataset is comparatively sparse: the number of frequent itemsets is low even for the minimum support threshold as low as 0.05%. Nevertheless, for $\theta = 0.05\%$ and $\gamma = 0.10$ our algorithm is 35 times faster than the baseline support-based computation. This performance gain for such relatively small dataset shows the potential of our method for typical market basket applications.
2. The DBLP dataset [1] is a set of computer science bibliography. In our experiments, we used its subset DBLP AUTHORS ($72K$ citations) generated in [15], with publications in fields of databases, data mining and information retrieval. We regard each paper as a transaction and each author as an item. The correlation here describes the degree of the collaboration inside the group of authors. For $\theta = 0.007\%$ and $\gamma = 0.3$, NICoMiner is 20 times faster than the baseline method.
3. The COMMUNITIES dataset [12, 13] is a publicly available dataset, which represents the demographic summarization for $1,980$ US communities. Each attribute value is a normalized numeric value between 0 and 1, which characterizes the relative presence of this attribute in a given community. We discretized each value into 5 equal-sized buckets: with ≤ 0.2 be very low and with > 0.8 be very high. Each community can be considered as a transaction, and each attribute-value pair

Table 6. Examples of top correlated patterns for each dataset.

Dataset	Pattern	sup	cos
GROCERIES	{*butter milk, yogurt*}	84	0.14
	{*salty snack, popcorn*}	22	0.14
	{*chocolate, candy*}	49	0.13
	{*frankfurter, brown bread*}	70	0.12
	{*sausage, white bread*}	71	0.12
DBLP AUTHORS	{*Steven M. Beitzel, Eric C. Jensen*}	25	1.00
	{*In-Su Kang, Seung-Hoon Na*}	20	0.98
	{*Ana Simonet, Michel Simonet*}	16	0.94
	{*Caetano Traina Jr., Agma J. M. Traina*}	35	0.92
	{*Claudio Carpineto, Giovanni Romano*}	15	0.91
COMMUNITIES	{*People with social security income:* $> 80\%$, *Age* ≥ 65*:* $> 80\%$}	47	0.76
	{*Large families* (≥ 6)*:* $\leq 20\%$, *White:* $> 80\%$}	1017	0.75
	{*In dense housing* (≥ 1 *per room*)*:* $> 80\%$, *Hispanic:* $> 80\%$, *Large families* (≥ 6)*:* $> 80\%$}	53	0.64
	{*People with Bachelor or higher degree:* $> 80\%$, *Median family income: very high* }	60	0.63
	{*People with investment income:* $> 80\%$, *Median family income: very high* }	66	0.61

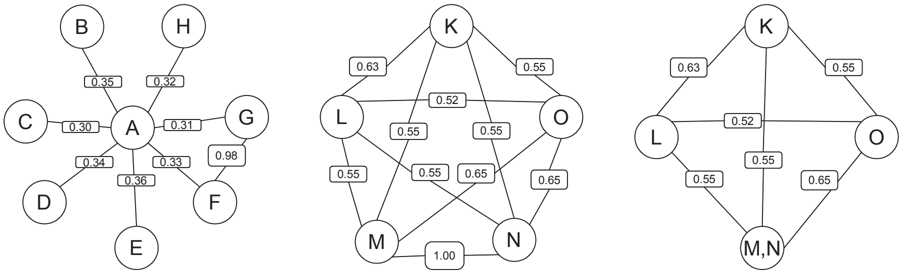
as an item. The correlation here describes which demographic characteristics appear together in the same communities. COMMUNITIES is an example of a very dense dataset. The results in Figure 2(a) are for $\theta = 10\%$ and $\gamma = 0.60$. Even for this very high support threshold, the total number of frequent candidates exceeded the memory capacity of $40GB$, available in our experiments, and the results show the time before memory crashed: NICoMiner is more than 500 times faster than the baseline method. Note that using our new algorithm, we were able to lower the minimum support threshold for this dataset to 1% and obtain the results in just 12 seconds. This demonstrates the ability of NICoMiner to produce highly correlated patterns with low support, which for some datasets is even impossible using the frequency-based pruning alone.

In Table 6 we show some examples of patterns for each dataset, found among the top-20 correlations. These examples show that top correlations at low support can be used not only for such classic applications as product marketing, but also for the demographics analysis, or for the study of social networks.

For illustration, consider strong correlations extracted from the DBLP AUTHORS dataset (Figures 3(a)[3] and 3(b)[4]), where the edges label the degree of the pairwise corre-

[3] The letters in Figure 3(a) correspond to the following researchers: [A] *Hsinchun Chen*, [B] *Homa Atabakhsh*, [C] *Siddharth Kaza*, [D] *Jennifer Jie Xu*, [E] *Daniel Dajun Zeng*, [F] *Jialun Qin*, [G] *Yilu Zhou*, [H] *Chunju Tseng*.

[4] The letters in Figure 3(b) correspond to the following researchers: [K] *David A. Grossman*, [L] *Ophir Frieder*, [M] *Eric C. Jensen*, [N] *Steven M. Beitzel*, [O] *Abdur Chowdhury*.



(a) Pairwise correlations for typical advisor-advisee co-authorships.

(b) Pairwise correlations for advanced mutual co-authorships.

(c) Redundancy in (b) can be removed without information loss.

Fig. 3. Strong pairwise correlations in DBLP AUTHORS dataset.

lation between authors. The nodes represent authors with 60 - 70 papers ($\theta = 0.001\%$). The pairwise correlations in Figures 3(a) and 3(b) are typical examples of (a) advisor-advisee relationships and (b) advanced mutual collaboration in an established collaborative group. Hence, such correlations can be used in studying evolving collaborations. Note that such strong correlations as in Figure 3(b) rarely take place in groups of authors with very high support. In general, for all datasets used in our experiments, the most interesting non-trivial correlations are found in the itemsets with low support.

Even though the number of correlated patterns is significantly smaller than the number of frequent itemsets, some of these patterns carry redundant information. As an extreme case, consider correlation value 1.00. The set of pairwise correlations in Figure 3(b) can be compressed without losing any information by replacing two authors M and N which co-authored in 100% of their papers by the joined item (MN). This removes significant amount of redundant correlations, as shown in Figure 3(c).

In addition, if the correlation values of the itemset and all its subsets are similar, they may be considered redundant. However in general, the correlation computed for a superset is not a redundant information, as can be shown on example in Figure 3(c). Based on values of pairwise correlations, we expect the correlation {K,M,N,O} to be at least as strong as {K,L,M,N}, while after computing actual correlations we find out that *Cosine*{*K,L,M,N*}$= 0.52$, while *Cosine*{*K,M,N,O*} is less than 0.1. This shows that information about mutual relationships of 3 or more objects cannot be deduced from pairwise correlations, and thus is not a redundant information. The distinction between redundant and non-redundant information represents the problem which requires special attention.

6 Related Work

The extension of association rules to correlations was introduced in the pioneering work of Brin et al. [3]. Since then, dozens of correlation measures have been proposed to assess the degree of the correlation. The comprehensive comparison of 21 different correlation measures can be found in [16], where the *null invariance* was introduced among other properties such as scaling-invariance and inversion-invariance. The importance of null-invariance for capturing meaningful correlations in large transactional databases was demonstrated later in [8, 17, 19]. In [19], the authors provide a unified definition of existing null-invariant correlation measures.

An efficient algorithm for correlation mining based on *All Confidence* and *Coherence* was proposed in [10, 11]. In both papers, authors use the downward closure (or, anti-monotonicity) property for pruning. In [19], authors derive an upper bound of *Kulczynski*, which was shown to be effective only for the comparatively high minimum support thresholds. The techniques based on sampling were recently proposed in [4], which are much faster, but at the cost of the incompleteness of results. Our approach works well for all null-invariant measures including $Kulczynski$ and $Cosine$, which did not have efficient algorithms for low support, and it produces the complete results.

Top-k correlated pattern mining was mostly developed only for 2-itemsets [22, 23]. Our algorithm produces top-k correlations among itemsets with any number of items.

7 Conclusions & Future Work

In this paper, we addressed the problem of efficient mining of the top correlated patterns, based on any known null-invariant measure. We used *Cosine* correlation measure as an example, because it is one of the most widely-used, and at the same time, one of the most computationally challenging correlation measures. Even though it does not have the (anti)-monotonicity property, we developed two pruning methods that enabled an order of magnitude faster running time than the frequent pattern mining approach. We have shown experimentally that new pruning methods have high efficiency for discovering correlations in the itemsets with low support.

The top-k version of our new algorithm presents a valuable new tool to find top correlations. It can be easily extended to the problem of finding top-k correlations containing a particular item or pattern of interest (query pattern). This can be achived by maintaining a min heap data structure that keeps the top-k supersets of the query pattern.

In the future, we plan to address the problem of redundancy. If the correlation in the itemset is close to the correlation in its superset, it might be enough to output only the maximal superset pattern instead of reporting all patterns. One way to do it is to define a summary (or compressed) pattern for correlated patterns as in [20]. It would be interesting to incorporate the redundancy removal into the mining process, instead of performing it in a post-processing step.

Acknowledgement

The work was supported in part by the U.S. National Science Foundation grants IIS-09-05215 and OCI-07-25070, the Network Science Collaborative Technology Alliance Program (NS-CTA / INARC) of U.S. Army Research Lab (ARL) under the contract number W911NF-09-2-0053, and the Postdoctoral Fellowship of NSERC (Natural Science and Engineering Research Council) of Canada. Any opinions, findings, and conclusions expressed here are those of the authors and do not necessarily reflect the views of the funding agencies.

References

1. Dataset: Dblp. `http://www.informatik.uni-trier.de/~ley/db/`, 2006.
2. R. Agrawal, T. Imieliński, and A. Swami. Mining association rules between sets of items in large databases. In *SIGMOD*, 1993.
3. S. Brin, R. Motwani, and C. Silverstein. Beyond market baskets: generalizing association rules to correlations. In *SIGMOD*, 1997.
4. A. Campagna and R. Pagh. Finding associations and computing similarity via biased pair sampling. In *ICDM*, 2009.
5. J. Cohen, S. G. West, P. Cohen, and L. Aiken. *Applied Multiple Regression Correlation Analysis for the Behavioral Sciences*. Lawrence Erlbaum Assoc Inc, 3rd edition, 2002.
6. M. Hahsler. Groceries dataset. `http://rss.acs.unt.edu/Rdoc/library/arules/data/`, 2007.
7. M. Hahsler, K. Hornik, and T. Reutterer. Implications of probabilistic data modeling for mining association rules. In *Proceedings of the 29th Annual Conference of the Gesellschaft für Klassifikation*, 2006.
8. J. Han and M. Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, 2nd edition, 2006.
9. W. P. Kuo, T.-K. Jenssen, A. J. Butte, L. Ohno-Machado, and I. S. Kohane. Analysis of matched mrna measurements from two different microarray technologies. *Bioinformatics*, 18(3):405–412, 2002.
10. Y.-K. Lee, W.-Y. Kim, Y. D. Cai, and J. Han. Comine: Efficient mining of correlated patterns. In *ICDM*, 2003.
11. E. R. Omiecinski. Alternative interest measures for mining associations in databases. *IEEE Trans. on Knowl. and Data Eng.*, 15:57–69, 2003.
12. M. Redmond. Communities and crime dataset. `http://archive.ics.uci.edu/ml/datasets/Communities+and+Crime`, 2009.
13. M. A. Redmond and A. Baveja. A data-driven software tool for enabling cooperative information sharing among police departments. *European Journal of Operational Research*, 141:660–678, 2002.
14. R. Srikant and R. Agrawal. Mining generalized association rules. In *VLDB*, 1995.
15. Y. Sun, J. Han, J. Gao, and Y. Yu. iTopicModel: Information network-integrated topic modeling. In *ICDM*, 2009.
16. P.-N. Tan, V. Kumar, and J. Srivastava. Selecting the right interestingness measure for association patterns. In *KDD*, 2002.
17. P.-N. Tan, M. Steinbach, and V. Kumar. *Introduction to Data Mining*. Addison-Wesley Longman Publishing Co., Inc., 1st edition, 2005.
18. H. von Storch and F. W. Zwiers. *Statistical analysis in climate research*. Cambridge University Press, 2002.
19. T. Wu, Y. Chen, and J. Han. Re-examination of interestingness measures in pattern mining: a unified framework. *Data Min. Knowl. Discov.*, 21(3):371–397, 2010.
20. D. Xin, J. Han, X. Yan, and H. Cheng. Mining compressed frequent-pattern sets. In *VLDB*, 2005.
21. H. Xiong, X. He, C. H. Q. Ding, Y. Zhang, V. Kumar, and S. R. Holbrook. Identification of functional modules in protein complexes via hyperclique pattern discovery. In *Pacific Symposium on Biocomputing*, 2005.
22. H. Xiong, W. Zhou, M. Brodie, and S. Ma. Top-k ϕ correlation computation. *INFORMS Journal on Computing*, 20(4):539–552, 2008.
23. S. Zhu, J. Wu, H. Xiong, and G. Xia. Scaling up top-k cosine similarity search. *Data Knowl. Eng.*, 70:60–83, January 2011.