Mining Approximate Frequent Itemsets from Noisy Data

[1]Jinze Liu, [1]Susan Paulsen, [1]Wei Wang, [1,2]Andrew Nobel, [1]Jan Prins
[1]Department of Computer Science
[2]Department of Statistics and Operations Research
University of North Carolina, Chapel Hill, NC 27599
{liuj, paulsen, weiwang, nobel, prins}@cs.unc.edu

Abstract

Frequent itemset mining is a popular and important first step in analyzing data sets across a broad range of applications. The traditional, "exact" approach for finding frequent itemsets requires that every item in the itemset occurs in each supporting transaction. However, real data is typically subject to noise, and in the presence of such noise, traditional itemset mining may fail to detect relevant itemsets, particularly those large itemsets that are more vulnerable to noise.

In this paper we propose approximate frequent itemsets (AFI), as a noise-tolerant itemset model. In addition to the usual requirement for sufficiently many supporting transactions, the AFI model places constraints on the fraction of errors permitted in each item column and the fraction of errors permitted in a supporting transaction. Taken together, these constraints winnow out the approximate itemsets that exhibit systematic errors. In the context of a simple noise model, we demonstrate that AFI is better at recovering underlying data patterns, while identifying fewer spurious patterns than either the exact frequent itemset approach or the existing error tolerant itemset approach of Yang et al. [10].

1 Introduction

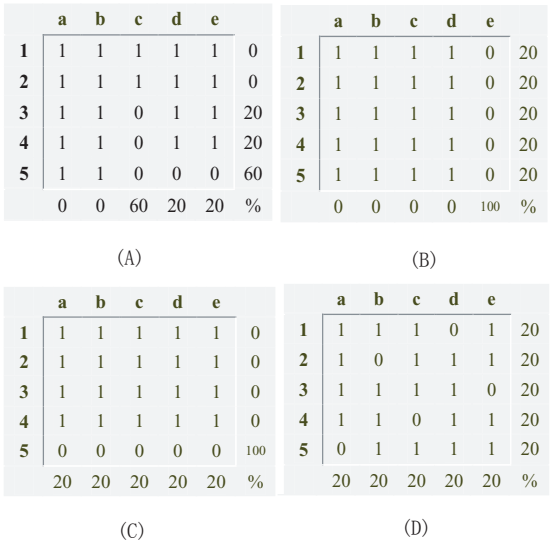
Relational databases are ubiquitous, cataloging everything from market-basket data [1] to gene-expression data [4]. Frequent itemset mining [1] is a key technique in the analysis of such data, providing the basis for deriving association rules, for clustering data, and for building classifiers.

The frequent itemset problem is generally characterized in the following form: The available data take the form of an $n \times m$ binary matrix D. Each row of D corresponds to a transaction t and each column of D corresponds to an item i. The t, i-th element of D, denoted $d_{t,i}$, is one if transaction t contains item i, and zero otherwise. Let $T_0 = \{t_1, t_2, \ldots, t_n\}$ and $I_0 = \{i_1, i_2, \ldots, i_m\}$ be the set of transactions and items associated with D, respectively. Under exact frequent itemset mining a transaction supports an itemset if it contains a '1' under each item in the itemset. An itemset is deemed frequent if the number of its supporting transactions exceeds the "support threshold," a user determined percentage of the total number of transactions.

While the classic exact frequent itemset definition and the algorithms designed to generate such itemsets have been well studied, the problems created by imperfect data have not. Error can be introduced when an item fails to be recorded, or not purchased at all because it was out of stock. In the presence of such "noise" (i.e. actual errors as well as incorrect imputation of measurements), classical frequent itemset algorithms will find a large number of small fragments of the true itemset, and may miss a pattern altogether if the frequency criterion is not satisfied. This failure to detect the full pattern compromises the usefulness of classic frequent itemset mining for detecting associations, clustering items, or building classifiers when such errors are present. As a solution, we present here a noise-tolerant approach to frequent itemset mining.

One natural approach for handling errors is to relax the requirement that a supporting transaction contain only 1's under the items in the itemset. Instead, a small fraction of 0's is tolerated, e.g. the "presence" signal of [10]. However, stipulating a small fraction of 0's row-wise alone may not be sufficient: we would also like to ensure that distribution of 0's is globally reasonable, *e.g.* that they are also not concentrated in a small number of columns.

For example, the fraction of 1's is 80% in each of the submatrices presented in panels (A)-(D) of Figure 1. However, not all of the transactions in each panel sensibly support the itemset $I = \{a, b, c, d, e\}$. In (A), the row-wise constraint employed by Yang *et. al.* [10] correctly excludes transaction 5 from the support; however,



	a	b	c	d	e	
1	1	1	1	1	1	0
2	1	1	1	1	1	0
3	1	1	0	1	1	20
4	1	1	0	1	1	20
5	1	1	0	0	0	60
	0	0	60	20	20	%

(A)

	a	b	c	d	e	
1	1	1	1	1	0	20
2	1	1	1	1	0	20
3	1	1	1	1	0	20
4	1	1	1	1	0	20
5	1	1	1	1	0	20
	0	0	0	0	100	%

(B)

	a	b	c	d	e	
1	1	1	1	1	0	0
2	1	1	1	1	0	0
3	1	1	1	1	0	0
4	1	1	1	1	0	0
5	0	0	0	0	0	100
	20	20	20	20	20	%

(C)

	a	b	c	d	e	
1	1	1	1	0	1	20
2	1	0	1	1	1	20
3	1	1	1	1	0	20
4	1	1	0	1	1	20
5	0	1	1	1	1	20
	20	20	20	20	20	%

(D)

Figure 1. Itemsets with global density of 80% but different distributions of noise in individual transactions and items.

in (B) enforcing the row-wise constraint alone allows each transaction to support the addition of $\{e\}$ to the itemset. Panel (C) illustrates the problem with a purely column-wise constraint, while (D) exhibits an error distribution where each transaction sensibly lends support to the full itemset. In this latter case, each row and column permits no more than 20% error.

Thus to attain noise-tolerant itemsets free from systematic errors, we propose the joint use of two criteria. We define an *approximate itemset* to be one where the fraction of 0's in each row and each column is restricted to ϵ_r and ϵ_c, respectively. If the approximate itemset has sufficiently many rows, it is an *approximate frequent itemset* (AFI).

Definition 1.1 *Let D be as above, and let $\epsilon_r, \epsilon_c \in [0, 1]$. An itemset $I \subseteq I_0$ is an AFI, if there exists a set of transactions $T \subseteq T_0$ with $|T| \geq minsup|T_0|$ such that the following two conditions hold: (i) for each $t \in T$ the fraction of items in I that appear in t is at least $(1 - \epsilon_r)$ and (2) for each $i \in I$, the fraction of transactions in T that appear in each item i is at least $(1 - \epsilon_c)$.*

Example 1.1 *Consider the transaction database D in Figure 1(A) with AFI parameters $minsup = 0.5$, $\epsilon_r = 1/3$ and $\epsilon_c = 1/3$. Then the maximal AFI contained in D is $I = \{a, b, c\}$, which is supported at least four transactions ($T = \{t_1, t_2, t_3, t_4, t_5\}$). For each item $i \in I$, at least $80\% > 100(1 - \epsilon_c)\%$ of the transactions in T contain it; each transaction $t \in T$ is missing at most one of the items in I, so the fraction of zeros in each row is at most $1/3 = \epsilon_r$.*

	a	b	c	d
1	1	1	1	0
2	1	1	0	0
3	1	0	1	0
4	0	1	1	0
5	1	1	1	1
6	0	0	0	1
7	0	1	0	1
8	1	0	0	0

Table 1. An example dataset

The rest of the paper is organized as follows. Section 2 presents a formal definition of our problem and outlines related work in the area of noise-tolerant itemset minings. Section 3 presents a brute-force algorithm. Evaluation of the AFI algorithm using both synthetic and real datasets is presented in Section 4. Section 5 concludes the paper.

2 Background and Related Work

Noise-tolerant itemsets were first discussed by Yang *et. al* [10], who proposed two error tolerant models, termed weak error-tolerant itemsets (ETIs) and strong ETIs. An itemset is a weak ETI if the fraction of noise in the entire set of supporting transactions is below a certain threshold, with no constraint on where the noise may occur. An itemset is a strong ETI if it satisfies the row, but not necessarily the column, constraint of the AFI definition above. As noted in the discussion of Figure 1, neither of the ETI models precludes columns of zeros. Yang *et. al* [10] describe algorithms for finding weak and strong ETIs based on a variety of heuristics and sampling techniques.

In [7] Seppanen *et. al* seeks weak ETIs by adding the constraint that all of their subsets must also be weak ETIs. The resulting itemsets belong to the category of weak ETIs but their overall characteristics are hard to derive. In some cases, this additional constraint eliminates irrelevant transactions as in Figure 1(B), but in others it permits Figure 1(C).

Another alternative, the support envelope[8] identifies regions of the data matrix where each transaction contains at least m' items and each item appears in at least n' transactions, where n' and m' are fixed integers. Support envelope mining can only recover one big submatrix at a time, prohibiting the discovery of multiple embedded dense regions. Furthermore, if the matrix is large, the one approximate itemset found by the support envelope approach tends to be very sparse.

Fault-tolerant frequent itemsets [**?**] allow a fixed number of errors δ within an itemset. This criterion is not consistent with our expectation that the number of errors should be permitted to scale with the size of the result.

3 A Brute Force Algorithm To Discover AFIs

As implied by its definition, an AFI(ϵ_r, ϵ_c) is also an ETI(ϵ_r), where only the row-wise constraint is enforced. Thus, a natural brute-force method to find the set of AFI(ϵ_r, ϵ_c) can be obtained two steps:

1. Generate the set of all ETIs(ϵ_r)
2. For each ETI(ϵ_r), check its validity as an AFI(ϵ_r, ϵ_c).

The first step of the algorithm was studied by Cheng *et.al* [10]. The exhaustive algorithm proposed in their paper starts with single items and develops them into longer itemsets by adding one of the remaining items at each time. The lattice of itemsets is traversed in a breadth-first manner. As may be obvious, the Apriori property of classical frequent itemset mining will not hold for either ETI or AFI. Thus an itemset cannot be pruned if one of its $(k-1)$ is not a valid itemset. Instead, an length-k itemset cannot be eliminated as a valid ETI until it is established that none of its $(k-1)$ subsets is a weak ETI. The second step in the algorithm is a postprocessing step. For each of the submatrices of itemsets and transactions discovered in the first step, determine which transactions meet the AFI column constraint and if the number of qualifying transactions is still large enough to meet the support constraint.

4 Experiments

We performed two experiments to evaluate the performance of AFI . A synthetic data matrix corrupted with noise was used to compare the results of AFI mining to both exact frequent itemset mining and the ETI approach. In addition, we applied AFI to a data set drawn from a real biogeographic problem, where the AFI algorithm identified interesting patterns more succinctly than the competing algorithms.

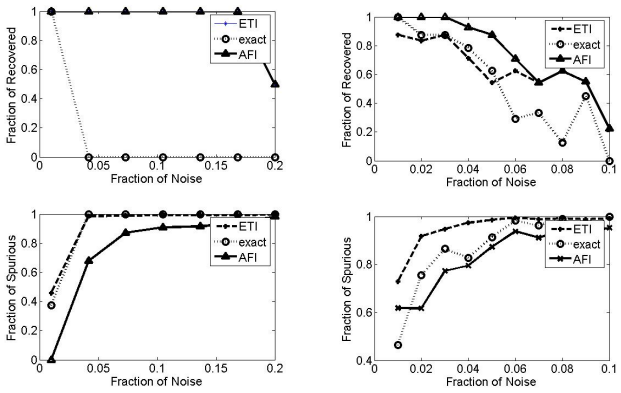
4.1 Quality Testing with Synthetic Data

In order to test the quality of the AFI model, we created data with both embedded patterns and overlaid random errors. By knowing the true patterns, we were able to assess the quality of AFI 's results. To each synthetic dataset created, an exact method, ETI and AFI were each applied.

To evaluate the performance of an algorithm on a given dataset, we employed two measures that jointly describe quality: "recoverability" and "spuriousness" (in the spirit, but not exact detail of [6]). Recoverability is the fraction of the embedded patterns recovered by an algorithm, while spuriousness is the fraction of the mined results that fail to correspond to any planted cluster. A truly useful data mining algorithm should achieve high recoverability with little spuriousness to dilute the results.

Multiple datasets were created and analyzed to explore the relationship between increasing noise levels and the quality of the result. Noise was introduced by bit-flipping each entry of the full matrix with a probability equal to p. The probability p was varied from 0.01 to 0.2. The number of pattern blocks embedded also varied, but the results were consistent across this parameter. Here we present results when 1 or 3 blocks were embedded in the data matrix (Figure 2(A) and (B), respectively).

In both cases, the exact method performed poorly as noise increased. Beyond $p = 0.05$ the original pattern cannot be recovered, and all of the discovered patterns are spurious. In contrast, the error-tolerant algorithms, ETI and AFI , were much better at recovering the embedded matrices at the higher error rates. However, the ETI algorithm reported many more spurious results than AFI . Though it may discover the embedded patterns, ETI generates many more patterns that are not of interest, which may overshadow the real patterns of interest. The AFI algorithm consistently demonstrates higher recoverability of embedded pattern while maintaining a low level of spuriousness.



(A) Single Cluster (B) Multiple Clusters

Figure 2. Algorithm quality versus noise level

4.2 An Application in Biogeography

One novel, but natural, application of frequent itemset mining is in the field of biogeography, the study of the geographical distributions of organisms. The patterns discovered in species distributions are used to infer either connections or barriers between regions, which in turn lead to hypotheses concerning the biogeographic tracks of organisms in historical time. Here we apply AFI to data from a study of freshwater fish across Australia (from Unmack [9]). The presence or absence of 167 species was recorded for each of 31 regions covering the continent. This type of data is subject to error in its collection, and "soft" (i.e. approximate) patterns are of interest.

Application of exact frequent itemset mining using $minsup = 5$ produced a total of 31 itemsets and a reasonable result: the broadest cluster in terms of regions covered corresponds to one of data author's results. Its 11 regions form a contiguous coastal band across Northern and Eastern Australia (shown as the dark-colored provinces in Figure3). However, application of AFI not only recovered the exact result (with fewer spurious blocks), but at $\epsilon_c = \epsilon_r = 0.2$, it adds two more regions to the item-wise largest block. These regions have been acknowledged by Unmack as sensible additions: they are contiguous with the previously identified cluster in the northern portion of Australia, and appear to be the next most closely related regions in Unmack's analysis. These regions appear in Figure 3 as the light gray regions.

Figure 3. Map of Australia with shading representing provinces in the cluster

5 Conclusion

In this paper we have defined criteria for mining approximate frequent itemsets from noisy data. The AFI model places constraints on the fraction of noise in each row and column, and so ensures a relatively reasonable distribution of error in any patterns found. According to investigation, AFI generates more reasonable and useful itemsets than classical frequent itemset mining and existing noise-tolerant frequent itemset mining.

Several computational challenges remain unsolved, however, and are currently under investigation. Noise tolerance creates substantial algorithmic challenges not present in exact frequent itemset mining. First, the AFI criteria do not have the anti-monotone (Apriori) property enjoyed by exact frequent itemsets. Second, one cannot derive the support set of an AFI from the common support sets of its sub-patterns, as is done in exact frequent itemset mining. Both of these considerations make the traditional breadth-first, and the projection-based depth-first algorithm hard for the generation of approximate frequent itemsets. Development of an efficient algorithm and pruning method will be the main focus of our future work.

This research was partially supported through NIH Integrative Research Resource grant 1-P20-RR020751-01, NSF grant DMS-0406361 and NSF grant IIS-0448392.

References

[1] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases In SIGMOD 1993.

[2] R. Agrawal, H. Mannila, R. Srikant, H. Toivonen, and A Verkamo. Fast discovery of association rules. In U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, Advances in Knowledge Discove and Data Mining, chapter 12, pages 307328. AAAI Pre 1996.

[3] C. Becquet , S. Blachon, B. Jeudy, J.F. Boulicaut, Gandrillon O. Strong-association-rule mining for large-scale gene-expression data analysis: a case study on humaMining gene expression databases for association rules. n SAGE data. Genome Biol. 2002

[4] C. Creighton, S. Hanash .Mining gene expression databases for association rules. Bioinformatics. 2003 Jan;19(1):79-86.

[5] J. Liu, S. Paulsen, W. Wang, A. Nobel, J. Prins. "Mining Approximate Frequent Itemset from Noisy Data". Technical Report(TR05-015) of Department of Compuater Science, UNC-Chapel Hill, 2005 Jun.

[6] H.C. Kum, S. Paulsen, W. Wang, Comparative Study of Sequential Pattern Mining Models, Studies in Computational Intelligence, Volume 6, Aug 2005, Pages 43-70.

[7] J. K. Seppanen, H. Mannila. Dense Itemsets. In SIGKDD 2004.

[8] M. Steinbach, P. N. Tan, V. Kumar. Support envelopes: a technique for exploring the structure of association patterns. In SIGKDD 2004.

[9] P. J. Unmack. Biogeography of Australian freshwater fishes. Journal of Biogeography. Vol. 28: pages 1053–1089. Blackwell Science Ltd. 2001.

[10] C. Yang, U. Fayyad, P. S. Bradley. Efficient discovery of error-tolerant frequent itemsets in high dimensions. In SIGKDD 2001.