Generalized Closed Itemsets for Association Rule Mining

Vikram Pudi * Jayant R. Haritsa *

Abstract

The output of boolean association rule mining algorithms is often too large for manual examination. For dense datasets, it is often impractical to even generate all frequent itemsets. The closed itemset approach handles this information overload by pruning "uninteresting" rules following the observation that most rules can be derived from other rules. In this paper, we propose a new framework, namely, the generalized closed (or g-closed) itemset framework. By allowing for a small tolerance in the accuracy of itemset supports, we show that the number of such redundant rules is far more than what was previously estimated. Our scheme can be integrated into both levelwise algorithms (Apriori) and two-pass algorithms (ARMOR). We evaluate its performance by measuring the reduction in output size as well as in response time. Our experiments show that incorporating g-closed itemsets provides significant performance improvements on a variety of databases.

1. Introduction

The output of boolean association rule mining algorithms is often too large for manual examination. For dense datasets, it is often impractical to even generate all frequent itemsets. Among recent approaches to manage this gigantic output, the *closed itemset* approach [4, 5] is attractive in that both the identities and supports of all frequent itemsets can be derived *completely* from the frequent closed itemsets. However, the usefulness of this approach critically depends on the presence of frequent itemsets that have supersets with *exactly the same support*. This means that even minor changes in the database can result in a significant increase in the number of frequent closed itemsets. For example, adding a select 5% transactions to the mushroom dataset (from the UC Irvine Repository) caused the number of closed frequent itemsets at a support threshold of 20% to increase from 1,390 to 15,541 – a factor of 11 times!

In order to overcome this limitation, we propose in this paper the *generalized closed (or g-closed) itemset framework*, which is more robust to the database contents. In our scheme, although we do not output the *exact* supports of frequent itemsets, we estimate the supports of frequent itemsets within a *deterministic*, user-specified "tolerance" factor. A side-effect of allowing for a tolerance in itemset supports is that the supports of some "borderline" infrequent itemsets may be over-estimated causing them to be incorrectly identified as frequent. Since our typical tolerance factors are much less than the minimum support threshold, this is not a major issue. Further, an extra (quick) database pass can always be made to check these borderline cases.

We provide theoretical arguments to show why the g-closed itemset scheme works and substantiate these observations with experimental evidence. Our experiments were run on a variety of databases, both real and synthetic, as well as sparse and dense. Our experimental results show that even for very small tolerances, we produce *exponentially fewer rules* for most datasets and support specifications than the closed itemsets, which are themselves much fewer than the total number of frequent itemsets.

Our scheme can be used in one of two ways: (1) As a post-processing step of the mining process, or (2) as an integrated solution. Further, our scheme can be integrated into both levelwise algorithms as well as the more recent two-pass mining algorithms. We note that integration into two-pass mining algorithms is a novel and important contribution because two-pass algorithms have several advantages over Apriori-like levelwise algorithms. These include: (1) significantly less I/O cost, (2) significantly better overall performance, and (3) the ability to provide approximate supports of frequent itemsets (with a deterministic bound in error) at the end of the first pass itself. This ability is essential for mining *data streams* as it is infeasible to perform more than one pass over the complete stream.

2. Generalized Closed Itemsets

In addition to the standard boolean association rule mining inputs ($\mathcal{I}$, the set of database columns, $\mathcal{D}$, the set of database rows and $minsup$, the minimum support threshold), the frequent g-closed itemset mining problem also takes as input ϵ, the user-specified tolerance factor. It pro-

*Database Systems Lab, SERC, Indian Institute of Science, Bangalore 560012, INDIA. Email: {vikram, haritsa}@dsl.serc.iisc.ernet.in

Proceedings of the 19th International Conference on Data Engineering (ICDE'03)
1063-6382/03 $ 17.00 © 2003 IEEE

duces as output a set of itemsets (that we refer to as the frequent g-closed itemsets) along with corresponding supports. The frequent g-closed itemsets are required to satisfy the following properties: (1) The supports of all frequent itemsets can be derived from the output within an error of ϵ. (2) If $\epsilon = 0$, the output is precisely the frequent closed itemsets.

2.1. Generalized Openness Propagation

The key concept in the g-closed itemset framework lies in the generalized openness propagation property, which is stated in the following theorem[1]. Here, the supports of itemsets X and Y are said to be approximately equal or ϵ-equal (denoted as $support(X) \approx support(Y)$) iff $|support(X) - support(Y)| \leq \epsilon$.

Theorem 2.1 *If X and Y are itemsets such that $Y \supseteq X$ and $support(X) \approx support(Y)$, then for every itemset Z : $Z \supseteq X$, $support(Z) \approx support(Y \cup Z)$.*

Here Y can be considered redundant because its support can be estimated (within an error of ϵ) from that of X. This theorem implies that if Y can be considered redundant in such a fashion, then *all* supersets of Y can also be considered redundant. This result suggests a general pruning technique to incorporate into mining algorithms, which we refer to as *ϵ-equal support pruning*: If an itemset X has an immediate superset Y[2], with ϵ-equal support, then prune Y and avoid generating any candidates that are supersets of Y. The support of any of these pruned itemsets, say W, will be ϵ-equal to one of its subsets, namely, $(W - Y) \cup X$.

2.2. Approximation Error Accumulation

A direct application of the ϵ-equal support pruning technique outlined above will not produce correct results: the supports of all frequent itemsets will not be derivable even approximately from the output. This is because Theorem 2.1 considers for any itemset X, only *one* superset Y with ϵ-equal support. If X has more than one superset (say $Y_1, Y_2, \ldots, Y_n$) with ϵ-equal support then a naive interpretation of the generalized openness propagation property would seem to indicate the following: Every itemset $Z : Z \supset X \bigwedge Z \not\supseteq Y_k, k = 1 \ldots n$, also has a proper superset $\bigcup_{k=1}^{n} Y_k \cup Z$ with ϵ-equal support. Although this is valid when $\epsilon = 0$, in the general case, it is not necessarily true. However, the following theorem reveals an upper bound on the difference between the supports of $\bigcup_{k=1}^{n} Y_k \cup Z$ and Z.

Theorem 2.2 *If $Y_1, Y_2, \ldots, Y_n, Z$ are supersets of itemset X, then $support(Z) - support(\bigcup_{k=1}^{n} Y_k \cup Z) \leq \sum_{k=1}^{n} (support(X) - support(Y_k))$.*

[1] Proofs of theorems are available in the full version of this paper [2].

[2] An immediate superset of X is a superset of X with cardinality $|X + 1|$. Likewise, an immediate subset of X is one with cardinality $|X - 1|$.

Therefore, in our approach we solve the problem of approximation error accumulation in the following manner: Whenever an itemset X having more than one immediate superset $Y_1, Y_2, \ldots, Y_n$, with ϵ-equal support is encountered, we prune each superset Y_k only as long as the sum of the differences between the supports of each pruned superset and X is within ϵ. While performing this procedure, at any stage, the sum of the differences between the support counts of each pruned superset and X is denoted by $X.debt$. Also, for each such superset Y, we include $Y - X$ in a set denoted by $X.pruned$, which along with $X.debt$ needs to be propagated to all unpruned supersets of X due to Theorem 2.1.

For any itemset X, $X \cup X.pruned$ is referred to as its corresponding g-closed itemset and will have ϵ-equal support. In the exact closed itemset case ($\epsilon = 0$), $X \cup X.pruned$ would be the closed itemset corresponding to X.

3. Rule Generation

We show that given the frequent g-closed itemsets and their associated supports, it is possible to generate association rules with approximate supports and confidences. This is stated in the following theorem:

Theorem 3.1 *Given the g-closed itemsets and their associated supports, let $\hat{c}$ and $\hat{s}$ be the estimated confidence and support of a rule $X_1 \rightarrow X_2$, and c and s be its actual confidence and support. Then, $\hat{c} \times \lambda \leq c \leq \hat{c}/\lambda$ where $\lambda = (1 - \epsilon/minsup)$; and $\hat{s} - \epsilon \leq s \leq \hat{s}$.*

Further, it has been shown earlier [4, 5] that it suffices to consider rules among adjacent frequent closed itemsets in the itemset lattice since other rules can be inferred by transitivity. This result carries over to frequent g-closed itemsets.

4. Incorporation in Mining Algorithms

As mentioned in the Introduction, our scheme can be used in one of two ways: (1) As a post-processing step of the mining process, or (2) as an integrated solution. Further, our scheme can be integrated into both levelwise algorithms as well as the more recent two-pass mining algorithms. We chose the classical Apriori and the recently-proposed ARMOR [3] as representatives of these two classes of algorithms. Integration into Apriori yields a new algorithm, **g-Apriori** and into ARMOR, yields **g-ARMOR**.

Integrating the g-closed scheme into Apriori is fairly straightforward. However, g-Apriori utilizes a novel technique for generating frequent g-closed itemsets from their *generators* [1] that avoids the costly additional pass required in the A-Close algorithm [1] for mining frequent closed itemsets. Integrating the g-closed scheme into ARMOR is

Proceedings of the 19th International Conference on Data Engineering (ICDE'03)
1063-6382/03 $ 17.00 © 2003 IEEE

non-trivial because the complete supports of candidate itemsets are not available during algorithm execution. Details of integrations are in [2].

5. Performance Study

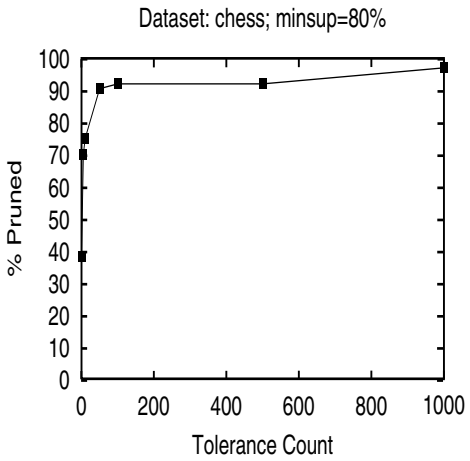

Figure 1. Output Size Reduction

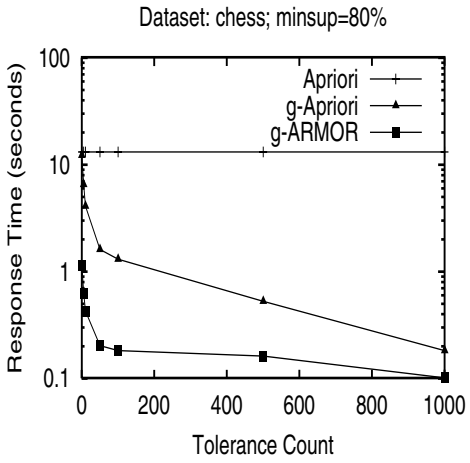

Figure 2. Response Times

We have conducted a detailed study to assess the utility of the g-closed framework in reducing both the output size and the response time of mining operations. Our experiments cover a range of databases and mining workloads including the real datasets from the UC Irvine Machine Learning Database Repository, the synthetic datasets from the IBM Almaden generator, and the real dataset, BMS-WebView-1 from Blue Martini Software.

Our experimental results for output size reduction, a sample of which is shown in Figure 1, show that even by allowing for a very small tolerance, we produce *exponentially fewer rules* for most datasets and support specifications than the closed itemsets, which are themselves much fewer than the total number of frequent itemsets. For example, on the chess dataset (Figure 1) for a minimum support threshold of 80%, the percentage of pruned itemsets is only 38% at zero tolerance (closed itemset case). For the same example, at a tolerance count of 50 (corresponding to a maximum error of 1.5% in itemset supports), *the percentage of pruned itemsets increases to 90%!* Further, the pruning achieved by our scheme is often significant even on *sparse* datasets.

Our experimental results for measuring response time performance, a sample of which is shown in Figure 2, show that g-Apriori performs significantly better than Apriori solely because the frequent g-closed itemsets are much fewer than the frequent itemsets. Finally, g-ARMOR is observed to perform over an order of magnitude better than Apriori over all databases and support specifications used in our experimental evaluation.

6. Conclusions

In this paper we proposed the g-closed itemset framework in order to manage the information overload produced as the output of frequent itemset mining algorithms. This framework provides an order of magnitude improvement over the earlier closed itemset concept. This is achieved by relaxing the requirement for exact equality between the supports of itemsets and their supersets. Instead, our framework accepts the supports of two itemsets to be equal if their difference is within a user-specified tolerance factor.

The complete details of the issues involved in the design, implementation and evaluation of the g-closed itemset framework and of the g-Apriori and g-ARMOR algorithms are available in the full version of this paper [2].

References

[1] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Discovering frequent closed itemsets for association rules. In *Proc. of Intl. Conference on Database Theory (ICDT)*, Jan. 1999.

[2] V. Pudi and J. Haritsa. Generalized closed itemsets: Improving the conciseness of rule covers. Technical Report TR-2002-02, DSL, Indian Institute of Science, 2002.

[3] V. Pudi and J. Haritsa. On the efficiency of association-rule mining algorithms. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, May 2002.

[4] R. Taouil, N. Pasquier, Y. Bastide, and L. Lakhal. Mining basis for association rules using closed sets. In *Intl. Conf. on Data Engineering (ICDE)*, Feb. 2000.

[5] M. J. Zaki. Generating non-redundant association rules. In *Intl. Conf. on Knowledge Discovery and Data Mining (KDD)*, Aug. 2000.

Proceedings of the 19th International Conference on Data Engineering (ICDE'03)
1063-6382/03 $ 17.00 © 2003 IEEE