STRATIFIED SAMPLING FOR ASSOCIATION RULES MINING

Yanrong Li and Raj P. Gopalan
Department of computing, Curtin University of Technology Kent st. Bentley, WA 6102, Australia

Abstract: It is well recognized that mining association rules in a very large database is usually time consuming due to the I/O overhead in scanning the disk resident database. As one of the techniques for reducing the I/O overhead, sampling for mining association rules has been actively investigated during the last few years. Each sampling method and algorithm proposed in the literature has its own merits and demerits in terms of effectiveness and efficiency and none of them can claim to be the best. Which sampling method to use and how big the sample size should be for a given database are key issues in sampling for particular data mining tasks. In this paper a transaction size based stratified sampling method has been proposed, tested and compared with the simple random sampling method for mining association rules. It opens up the questions of how to stratify the datasets so that it can better suit the problem of association rule mining.

Key words: stratified random sampling, association rules, effectiveness.

1. INTRODUCTION

Since the concept of association rules was first introduced in 1993 by Agrawal et al[1], many algorithms and techniques for mining association rules in a static large database has been proposed in the literature. However mining a very large database for association rules is usually time consuming and the I/O overhead in scanning the database plays an important role in it.

While many algorithms have been developed to reduce the number of database scans from as many as the size of the longest frequent itemsets in [2, 3] to as small as two in [4] and one in [5](see[6] for more details of the up-to-date association rules mining algorithms and their performance on typical datasets), sampling is another approach to reduce the I/O cost for processing very large databases. Random sampling of large databases for association rules was first proposed in [3] and was followed by more studies in [7-11].

Toivenon et al performed theoretical analyses of sampling large databases for association rules based on binomial distribution and Chernoff bound in [3, 7]. They sampled a database using simple random sampling with replacement and determined the sample size based on Chernoff bounds where the sample size is a function of the desired error bound and the confidence level. A sample was used to find frequent itemsets that probably hold in the entire database and the results were verified with the rest of the database. Zaki et al applied simple random sampling without replacement to some datasets and mined association rule by using only samples [8]. Since the sample size was empirically chosen as a certain percentage of the original database, which is independent of the error bound, and confidence level, it is difficult to quantify the quality of the results for a chosen sample size. A two-phase sampling based algorithm for association rules, FAST-trim, was presented in [9]. A large initial random sample is selected in Phase I to estimate the support of each distinct item in the database and these supports are used in phase II to trim out the outliers in the initial sample to form a small final sample based on the distance function,

$$\boldsymbol{Dist}(\boldsymbol{S}_0, \boldsymbol{S}) = \frac{|\boldsymbol{L}_1(\boldsymbol{S}) - \boldsymbol{L}_1(\boldsymbol{S}_0)| + |\boldsymbol{L}_1(\boldsymbol{S}_0) - \boldsymbol{L}_1(\boldsymbol{S})|}{|\boldsymbol{L}_1(\boldsymbol{S}_0)| + |\boldsymbol{L}_1(\boldsymbol{S})|}$$

where S and S_0 denote the original sample and the final sample, and $L_1(S)$ and $L_1(S_0)$ denote the sets of frequent 1-itemsets in S and S_0. However, the question of how to determine the initial sample size to ensure its subset, i.e. the small final sample, can effectively approximate the complete frequent itemsets in the original database remains. On the other hand, since the goal of the trimming algorithm is to minimize the Dist (S0, *S)*, the accuracy of the final sample is expected to be nearly the same as the original sample. Zhang et al [10] and Li et al[12] obtained the sufficient sample sizes based on binomial distribution and central limit theorem that were smaller than in [7] for a given error bound and confidence level. A database is sampled by random sampling without replacement in [10] and with replacement in [12] .

To the best of our knowledge, the sampling methods used for mining association rules in the literature are simple random sampling either with replacement or without replacement. In this paper, we explore the feasibility

of using stratified random sampling for association rules mining. Ideally, if a dataset can be partitioned into groups with distinct features appropriate for a particular data mining task, then proportionally sampling each group will give the exact result as with the whole dataset. In this work, a dataset is partitioned into strata according to the size of each transaction, and simple random sampling is applied to each stratum. The accuracy of the proposed stratified sampling method is compared with that using the simple random sampling method. The results open up the question of how to stratify a dataset for a particular data mining task.

The rest of this paper is organized as follows. Section 2 provides the definitions of association rules. The stratified random sampling method for association rules is presented in section 3 and the experimental evaluation of the sampling method is shown in section 4. Section 5 contains the conclusion and pointers for further work.

2. TERMS AND DEFINITIONS

Let $I=\{I_1, I_2, ..., I_m\}$ be a set of m distinct items. A transaction T is a non-empty subset of I. A database D is a set of N transactions. A set of items is called an itemset, and an itemset with k items is called a k-itemset. The support p of an itemset X in D, is the proportion of the transactions that contains X as subset. An itemset is called a *frequent itemset* if its support $p \geq p_t$, where p_t is the *minimum support* specified by users. Otherwise, the itemset is *not frequent*.

The *transaction size* is defined as the number of distinct items in the transaction, and its values lies in [1, m]. The *density* of a dataset is defined as the average transaction size of the dataset divided by the total number of distinct items in the dataset. The domain of density is [0, 1]. A dataset with longer average transaction size has higher density and vice versa.

3. SAMPLING LARGE DATASET FOR ASSOCAIATION RULES

In the context of sampling large databases for association rules, a transaction database D of size N is the population that we want to study, and a sample is a subset of D that consists of n transactions selected from the population D. The sample size selection and transaction size based stratified sampling method are described below.

3.1 Sample Size Selections

The size of a sample will affect both efficiency and the effectiveness of the sampling based algorithms. If we sample a transaction database with replacement, for a given error bound e and a given confidence level $1-\alpha$, the sufficient sample size for estimating an itemset's support is [12] :

$$n \geq \frac{z_{\alpha/2}^2 p(1-p)}{e^2} \tag{1}$$

where p is the support of the itemset in the population and $z_{\alpha/2}$ is the Z value above which the area under the standard normal curve is $\alpha/2$. Since $p(1\text{-}p)$ has the maximum value of 1/4 when p = 1/2, if we choose

$$n \geq \frac{z_{\alpha/2}^2}{4e^2} \tag{2}$$

then we can simultaneously estimate the support of each itemset in the dataset with confidence $1-\alpha$ that the error will not exceed e (see [12]for more details).

3.2 Transaction Size Based Stratified Random Sampling

In stratified random sampling, the population of size N is partitioned into strata and a sample is selected by simple random sampling within each stratum [13]. Given a total sample size n, if the strata differ in size, proportional allocation could be used to maintain a steady sampling fraction throughout the population [13]. If stratum k has N_k units, the sample size allocated to it would be

$$n_k = \frac{n}{N} N_k \tag{3}$$

The principle of stratification is to partition the population in such a way that the units within a stratum are as similar as possible. For example, in the survey of a human population, stratification may be based on the geographic region, sex or socio-economic factors. In this paper, we partition a dataset according to transaction sizes.

The difference between the size of the longest transactions and the shortest transactions in a stratum is called the width of the stratum, which is denoted as w. If we partition the data in such a way that each stratum has the same width, then the partition method is called *equal-width partition*,

otherwise it is called *unequal-width partition.* In equal-width partition, there will be $\lceil L_{max}/w \rceil$ strata for a dataset whose longest transactions' size is L_{max}. Transactions with size s will be partitioned to kth stratum, where $k = \lceil s/w \rceil$. For example, when $w = 2$, transactions with size s = 1 and s = 2 will go to 1st stratum, transactions with size s = 3 and s = 4 will go to 2nd stratum, and so on.

Given a minimum support, an error bound e, a confidence level $1-\alpha$, and the desired width of a stratum w, the transaction size based stratified sampling algorithm proceed as follows:

1. Initialize a sample $S = \{\}$;
2. Calculate the sufficient sample size n using equation (2);
3. Partition the dataset into $\lceil L_{max}/w \rceil$ strata based on the size of transaction;
4. Calculate sample size for each stratum using equation (3), sample the stratum by simple random sampling without replacement and add to S;
5. Run a standard association rules mining algorithm on S.

4. EFFECTIVENESS OF STRATIFIED SAMPLING

4.1 Measurement of Accuracy and Errors

The complete frequent itemsets discovered from the original database and its sample are denoted as $L(D)$ and $L(S)$, *respectively.* If an itemset exists in $L(S)$ but not in $L(D)$ then we call the itemset a *false positive.* If an itemset exists in $L(D)$ but not in $L(S)$, then we call this itemset a *false negative.* Therefore the number of false positives is $|L(S) - L(D)|$ and the number of false negatives is $|L(D) - L(S)|$. We use the same measure as in [9] to obtain the accuracy of sampling:

$$accuracy = 1 - \frac{|L(D) - L(S)| + |L(S) - L(D)|}{|L(D)| + |L(S)|} \tag{4}$$

This measurement is sensitive to both false positives and false negatives. We also use the following two measurements to quantify the errors of sampling:

$$fp = |L(S) - L(D)| / |L(S)| \tag{5}$$

which represents the proportion of the false frequent itemsets in a sample, and

$$fn = | L(D) - L(S) | / | L(D) | \quad (6)$$

which represents the proportion of the frequent itemsets in the original dataset that is missing in a sample.

4.2 Datasets Studied

We perform experiments on three datasets that are available at FIMI'03 (Frequent Itemsets Mining Implementations Repository)[14]. They are: (1) BMSPOS dataset, provided by Blue Martin Software, (2) Retail dataset, donated by Tom Brijs, which contains the (anonymized) retail market basket data from an anonymous Belgian retail store, and (3) Accidents dataset, donated by Karolien Geurts and contains (anonymized) traffic accident data. The density of the Accidents dataset is relatively higher than for the other two. The characteristics of these datasets are summarized in table 1, where N is the number of transactions in a database, T is the average transaction length, $|R|$ is the number of distinct items in the database and $L_{\max}$ is the size of the longest transactions. The histogram for each dataset is shown in Figure 1. To save space, transactions with size > 20 in BMSPOS and transactions with size > 27 in Retail, which only count for less than 5% of transactions, are not shown.

Table 1. Summaries of characteristics of datasets

Dataset Name	N	$\|R\|$	T	$L_{\max}$	*Density(%)*
BMSPOS	515596	1657	7.5	164	0.45
Retail	88162	16570	13	76	0.08
Accidents	340184	468	34	51	7.26

4.3 Experimental Results

In this section we show the experimental results of the proposed stratified sampling method and compare its accuracy with the simple random sampling method in [12]. The sample size chosen is 16513, which corresponds to error bound of 0.01 and confidence level of 99%.

As mentioned before, a dataset is partitioned based on the transactions sizes. Given a w value, the dataset is partition into $k = \lceil L_{max} / w \rceil$ strata. The width of each stratum in the resulting strata is w except that the width of kth stratum may be less than w. Each stratum is sampled according to equation (2). The accuracy and errors of the proposed stratified sampling method were compared with that of the simple random sampling method. Figures 2-4 show some of the testing results on different datasets for different

minimum support levels and different widths of stratum. In each figure, SR represents the result of simple random sampling.

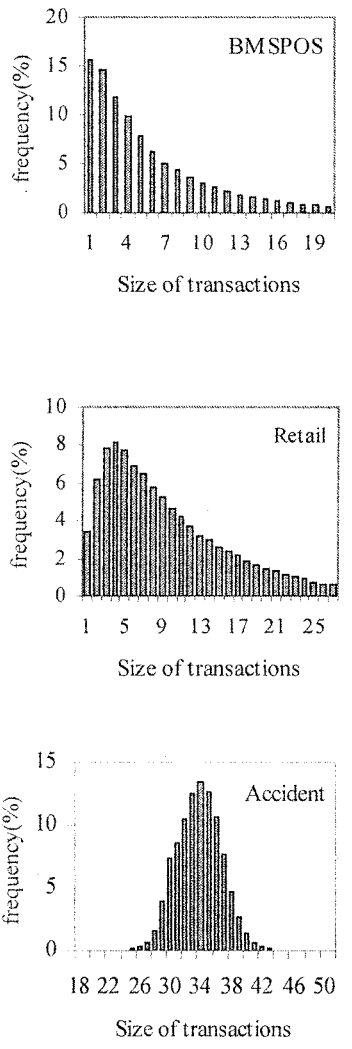

Figure 1. Histogram of transactions

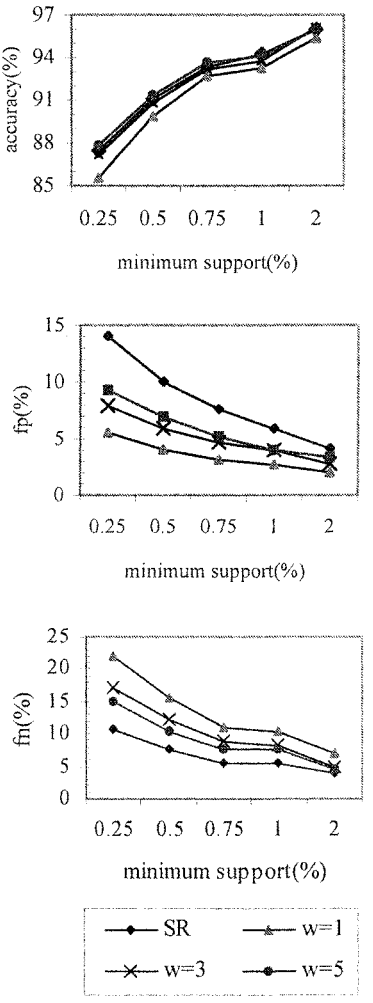

Figure 2. Testing results for BMSPOS

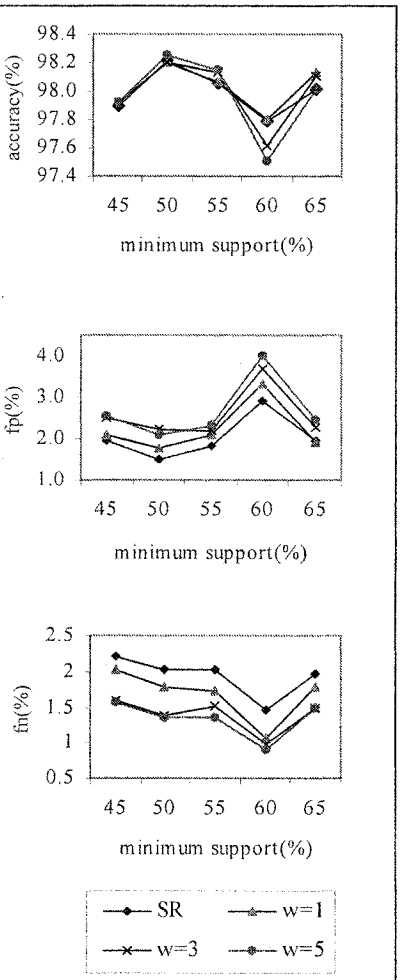

Figure 3. Testing results for Accidents

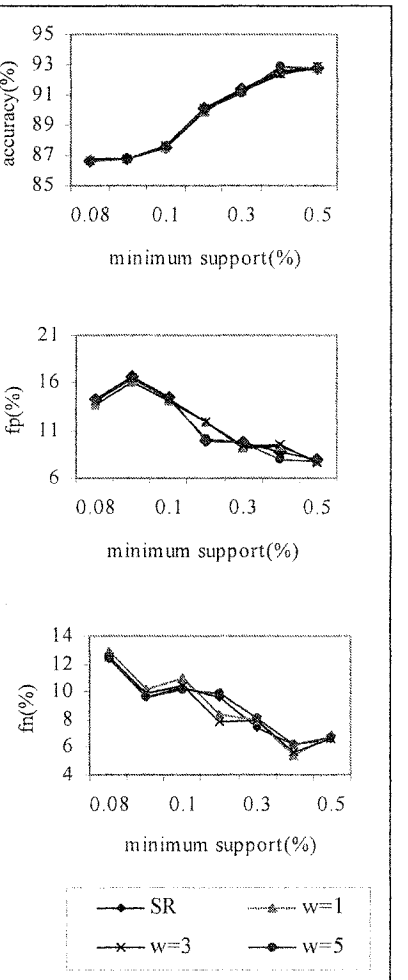

Figure 4. Testing results for Retail

For BMSPOS (Fig. 2), the accuracy of stratified sampling method increases while w increases. When $w = 5$, the accuracy of stratified sampling is slightly higher than that of simple random sampling. When we inspect the errors closely, we can see that *fp* increases as w increases and the *fn* decreases when w increases. The *fp* of stratified sampling is lower than that of simple random sampling and the *fn* of stratified sampling is higher than that of simple random sampling for all w values.

For the Accidents dataset (Fig. 3), the accuracy is very high since it is a relatively dense dataset and many transactions have the similar items. The accuracy does not vary too much while *w* changes (less than 1% for all minimum support levels). The trends of changes of *fn* and *fp* while w changes are the same as that for BMSPOS. In contrast to the results for BMSPOS, the *fp* of stratified sampling for Accidents is higher than that of simple random sampling while *fn* of stratified sampling is lower.

For Retail (Fig. 4), the accuracy and the errors (*fp* and *fn*) of stratified sampling method are almost the same as for the simple random sampling method at different *w* values. The values of *fp* and *fn* are not too different for most of the minimum support levels except 0.2%. It is noticed that sampling ratio of Retail is high, about 6 times higher than that of BMSPOS. much higher that of

We want to point out that transaction size based stratified sampling method will not be any different from the simple random sampling method for the dense datasets like Connect-4 that is also used for comparing algorithms at FIMI'03. In Connect-4, all the transactions have the same transaction size, and the proposed stratified sampling method will give exactly the same result as simple random sampling since all the transactions will be partitioned to a single stratum.

5. CONCLUSIONS AND DISCUSSIONS

Just as with other sampling methods, the proposed transaction size based stratified sampling may not suit all applications. The choice of a sampling method should depend on the characteristics of the dataset to be mined and the cost of errors in a given application. For a dataset like BMSPOS, if lower *fp* is desirable, the proposed stratified sampling will be a better choice than simple random sampling. Similarly, for applications where the lower *fn* is crucial, stratified sampling will perform better than the simple random sampling for dataset like Accidents. For some datasets such as Retail, both simple random sampling and stratified sampling are suitable. The open question for stratified sampling is how to partition the dataset so that each stratum has similar properties in relation to the association rules mining problem. We consider that the accuracy of stratified sampling may be improved if the stratification scheme is based on the similarity of transactions, i.e., the number of common items between transactions. We will continue our work to find better stratification schemes to improve the accuracy of stratified sampling.

REFERENCES

[1] R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules between sets of items in large databases," presented at ACM SIGMOD Conference on Management of Data, Washington, D.C, 1993.

[2] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," presented at the 20th VLDB Conference, Santiago, Chile, 1994.

[3] H. T. Heikki Mannila, Inkeri Verkamo, "Efficient Algorithms for Discovering Association Rules," presented at AAAI Workshop on Knowledge Discovery in Databases (KDD-94), Seattle, Washington, 1994.

[4] J. Han, J. Pei, and Y. Yin, "Mining Frequent Patterns without Candidate Generation," presented at Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data (SIGMOD'00), Dallas, TX, 2000.

[5] W. Cheung and O. R. Zaïane, " Incremental Mining of Frequent Patterns Without Candidate Generation or Support Constraint," presented at Seventh International Database Engineering and Applications Symposium (IDEAS 2003), Hong Kong, China, 2003.

[6] B. Goethals and M. J. Zaki, "Advances in Frequent Itemset Mining Implementations," *ACM SIGKDD Explorations*, vol. 6, pp. 109-117, 2004.

[7] H. Toivonen, "Sampling large databases for association rules," presented at 22th International Conference on Very Large Databases (VLDB'96), Mumbay, India, 1996.

[8] M. J. Zaki, S. Parthasarathy, W. Li, and M. Ogihara, "Evaluation of sampling for data mining of association rules," presented at 7th International Workshop on Research Issues in Data Engineering (RIDE '97) High Performance Database Management for Large-Scale Applications, Birmingham, UK, 1997.

[9] B. Chen, P. Haas, and P. Scheuermann, "A New Two Phase Sampling Based Algorithm for Discovering Association Rules," presented at SIGKDD '02, Edmonton, Alberta, Canada, 2002.

[10] C. Zhang, S. Zhang, and G. I. Webb, "Identifying Approximate Itemsets of Interest in Large Databases." *Applied Intelligence*, vol. 18, pp. 91-104, 2003.

[11] S. Parthasarathy, " Efficient Progressive Sampling for Association Rules," presented at IEEE International Conference on Data Mining, 2002.

[12] Y. Li and R. P. Gopalan, "Effective Sampling for Mining Association Rules," presented at 17th Australian Joint Conference on Artificial Intelligence, Cairns, Australia, 2004.

[13] S. K. Thomson, *Sampling*: John Wiley & Sons Inc., 1992.

[14] "Frequent Itemset Mining Dataset Repository," http://fimi.cs.helsinki.fi/data/.