ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)
Volume 2, Issue 11, November 2013

Incremental Mining for Frequent Item set on Large Uncertain Databases

Mr. Avinash A. Powar (PG Scholar, Department of C.S.E) A.D.C.E.T. Ashta.
Prof. A.S. Tamboli (Department of C.S.E.) A.D.C.E.T. Ashta.

Abstract – In recent years **dealing with uncertainty has gained increasing attention in the past few years in both static and streaming data management and mining. Data uncertainty is inherent in emerging applications such as location-based services, sensor monitoring systems and data integration. There are many existing works on mining frequent item sets from uncertain databases.. There are two main problems with this approach of mining frequent item sets from uncertain databases. The mining process can be done by using Poisson Binomial Distribution. We examine how an existing algorithm that extracts exact item sets, as well as our approximate algorithm, can support incremental mining. All our approaches support both tuple and attribute uncertainty, which are two common uncertain database models. We also perform extensive evaluation on real and synthetic data sets to validate our approaches.**

Keywords - **Frequent item sets, uncertain data set, Incremental mining, PFI, PWS**

I. INTRODUCTION:

The databases used in many important and novel applications are often uncertain. For example, the locations of users obtained through RFID and GPS systems are not precise due to measurement errors. As another example, data collected from sensors in habitat monitoring systems (e.g., temperature and humidity) are noisy. Customer purchase behaviours, as captured supermarket basket databases, contain statistical information for predicting what a customer will buy in the future. Integration and record linkage tools also associate confidence values to the output tuples according to the quality of matching.

In structured information extractors, confidence values are appended to rules for extracting patterns from unstructured data.

To meet the increasing application needs of handling a large amount of uncertain data, uncertain databases have been recently developed. An online marketplace application, which carries probabilistic information. Particularly, the purchase behaviour details of customers Jack and Mary are recorded.

The value associated with each item the chance that a customer may buy that item in the near future. These probability values may be by analysing the users browsing histories. For instance, if Jack visited the marketplace 10 times in the previous week, out of which video products were clicked five times, the marketplace may conclude that Jack has a 50 percent chance of buying videos.

II. RELATED WORK:

A few incremental mining algorithms that work for exact data have been developed. For example, in [1], the Fast Update algorithm (FUP) was proposed to efficiently maintain frequent item sets, for a database to which new tuples are inserted. Our incremental mining framework is inspired by FUP. The FUP2 algorithm was developed to handle both addition and deletion of tuples.

ZIGZAG also examines the efficient maintenance of maximal frequent item sets for databases that are constantly changing. In a data structure, called CATS Tree, was introduced to maintain frequent item sets in evolving databases. Another structure, called Can Tree, arranges tree nodes in an order that is not affected by changes in item frequency. The data structure is used to support mining on a changing database.

To our best knowledge, maintaining frequent item sets in evolving uncertain databases has not been examined before. We propose novel incremental mining algorithms for both exact and approximate PFI discovery. Our algorithms can also support attribute and tuple uncertainty models.so we are motivate to implement the above techniques in parallel manner, and implement new rules for tuple update and deletion.

III. LITERATURE REVIEW:

Mining frequent item sets is often regarded as an important step. Algorithms have been proposed to retrieve frequent item sets , such as Apriori [1] and FP-growth [9]. While these algorithms work well for databases with precise and exact values, it is interesting to extend them to support uncertain data. Our algorithms are based on the Apriori. We believe that they can be used by other algorithms (e.g., FP-growth) to support uncertain data. For uncertain databases [9, 2] developed efficient frequent pattern mining algorithms based on the expected support counts of the patterns. In [7], dynamic-programming-based solutions were developed to retrieve PFIs from attribute-uncertain databases, for both threshold- and rank-based PFIs. However, their algorithms have to compute exact probabilities, and find a PFI in O(n2) time. By using probability models, our algorithms avoid the use of dynamic programming, and can make a PFI much faster (in O(n) time). In [10], approximate algorithms for deriving threshold-based PFIs from tuple-uncertain data streams were developed. While [10] only considered the extraction of singletons (i.e., sets of single items), our solution discovers patterns with more than one item. More recently, [24] developed an exact threshold-based PFI mining algorithm. However, it does not support rank-based PFI discovery. Here we also study the retrieval of rank-based PFIs from tuple uncertain data. To our best knowledge, this has not been examined before.

IV. PREVIOUS WORK:

1. Uncertain Database

In the previous work, Dynamic Programming algorithm is used to extract the frequent item set from large uncertain database. It verifies the dataset and needs O (n2) time to authenticate the item set as PFI (Probabilistic Frequent Item set).This algorithm has so many disadvantages. That is low accuracy and high computational cost. In dynamic Programming approach, the whole algorithm is re-evaluated when a new tuple is inserted to the dataset. Experimental result [8] shows that dynamic programming algorithm takes long time to complete. With a 300k real dataset dynamic programming algorithm takes 30.1 hours to find all PFI" s. Either tuple or attribute ambiguity is supported. This is validated by interpreting both real and synthetic dataset. This dynamic programming algorithm has low performance in discovering PFI. It does not support incremental mining. It requires O (n 2) time to authenticate an itemset as PFI.

An example uncertain database is shown in Fig. 1. It shows an uncertain database in an online marketplace application, which contains statistical information. In this the purchasing behaviour of users Alice, Bob, Eve and Peter are stored. Each item value denotes the chance that a user may buy that item in the near future. These probability values are obtained by analysing users browsing histories. For example assume that Bob visited the marketplace ten times in the previous week, and he clicked camera items five times. As a result Bob has 50% chance of buying cameras in near future.

TABLE I
UNCERTAIN DATABASE

User	Purchase items
Rogger	(camera:1/2), (chocolate:1/2)
David	(cloth:3/4), (pencil:1/4)
Nick	(chocolate:1/2), (pen:1/2)
Sun	(camera:1/3), (chocolate:1/3), (pencil:1/3)

Table I, consider two tuples such as {camera} and {pencil} for Rogger and David respectively. {camera} tuple occurs with a probability of 1/2 * (1-1/2)=1/4 (i.e., probability of customer Rogger buying the item camera is multiplied by probability of customer Rogger not buying the item chocolate) likewise, the probability of {pencil} is, (1-3/4) *1/4=1/16. Then the value of possible world w is 1/4 * 1/16=1/64.

The frequentness probability is used to define the support pmf of an item set using closed frequent item sets in uncertain databases. This proposed method effectively and accurately extracts the PFI. The proposed algorithms also verify PFI in O(n) time and are thus more suitable for large databases.

2. Probabilistic Frequent Item set

A set of items is referred to as an item set. An item set that contains K items is a K-Item set. For example, a set {camera, pencil} is a 2-itemset. The occurrence frequency of an item set is the number of transactions that contain the item set. This is also known, simply as the frequency, support count or count of the item set. An item set satisfies minimum support if the occurrence frequency of the item set is greater than or equal to the product of min_sup and the total number of transactions in D. If an item set satisfies minimum support, then it is a frequent item set.

V. PROPOSED WORK:

.

We propose incremental mining algorithms, which enable Probabilistic Frequent Item set (PFI) results to be refreshed. This reduces the need of re-executing the whole mining algorithm on the new database, which is often more expensive and unnecessary. We examine how an existing algorithm that extracts exact item sets, as well as our approximate algorithm, can support incremental mining.

Our approach makes use of parallel and incremental techniques to generate frequent item sets in the presence of data updates without examining the entire database, and imposes minimal communication overhead when mining distributed databases.

We define new rule mining algorithms to generate both local and global frequent item sets. This ability permits our approach to identify high-contrast frequent item sets, which allows one to examine how the data is skewed over different site. We also examine how to use the model based approach to develop other mining algorithms (e.g. Clustering and classification).

Our approach is able to generate efficient mining algorithms for handling tuple updates and deletion.

Another interesting work is to investigate PFI mining algorithms for probability models that capture correlation among attributes and tuples. Traditional methods for frequent item set mining typically assume that data is centralized and static. Such methods impose excessive communication overhead when data is distributed, and they waste computational resources when data is dynamic.

Our approach makes use of parallel and incremental techniques to generate frequent item sets in the presence of data updates without examining the entire database, and imposes minimal communication overhead when mining distributed databases.

A. Incremental Algorithm

This algorithm handles the uncertainty by adding more transactions to the existing dataset. In existing system if new transactions are added the whole dataset is processed again from the scratch. So to avoid this we can use this algorithm to extract PFI's only with the new transactions and update the existing result with the new PFI. It also reduce the time and cost to extract the PFI.

B. Rule Mining Algorithm

Incremental Mining algorithm can handle only insertion of new transactions to the existing dataset; it won't support update or delete operation. To overcome the above said problem a rule mining algorithm is used to support Insert, Update & Delete operation. This will refresh the result if any of the above operation is done. The result is updated instead of re-executing the whole dataset from the scrap.

VI. IMPLEMENTATION STEPS:

There are three main steps of implementation implementation .which are mainly, modifying the Dataset, Extracting the Threshold Based PFI and Performance Evaluation. In the first step (Modifying the Dataset) , includes modification of the existing Dataset by updating or deleting the existing items in dataset from which the Probabilistic Frequent item sets are extracted.

Second step (Extracting the Threshold Based PFI), involves updating the result of Threshold based Probabilistic Frequent Item set extracted from old dataset with respect to the modification done with the dataset by using the rule mining algorithm which only refresh the result of threshold based PFI. In incremental Mining the result can be updated only when some new items are inserted to the existing dataset but it cannot handle update or delete operations. But the rule mining algorithm can update the result of PFI with respect to update or delete operations instead of processing the dataset from the scrap, which takes more time to extract the Probabilistic frequent item set from scratch. It also reduce the processing cost by just refreshing the result.

The third step (Performance evaluation), involves evaluation of the performance of proposed rule mining algorithm by computing the recall and precision value for the PFI extracted from the dataset by comparing the result obtained from dynamic programming approach and model based approach. Finally comes across the conclusion that the rule mining algorithm can effectively deal with update or delete operation which cannot be done with incremental mining algorithm. So these three main steps are involved in our proposed model

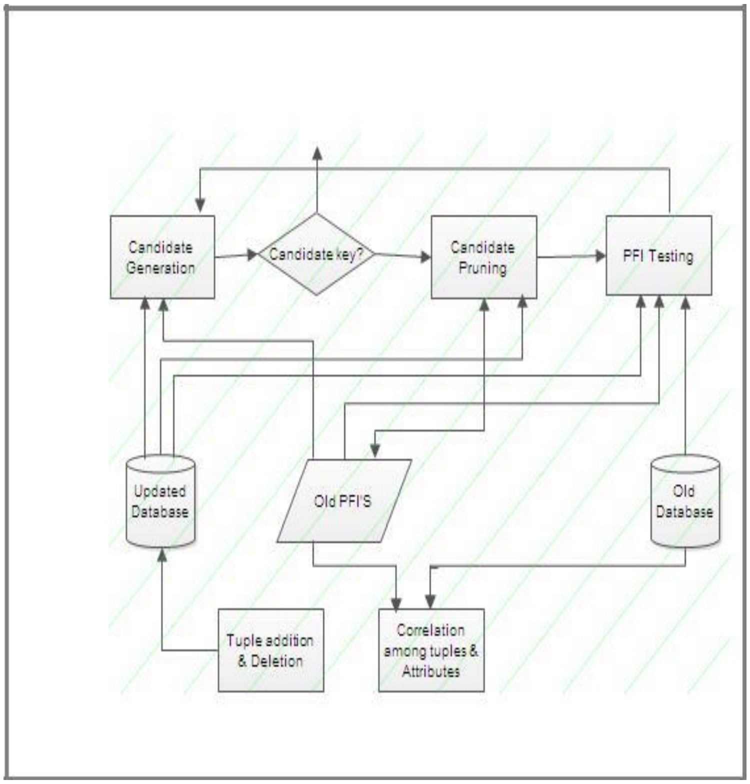

Fig. I
System Architecture

VII. CONCLUSION:

In previous a model-based approach to extract threshold-based PFIs from large uncertain databases. Its main idea is to approximate the s-pmf of a PFI by some common probability model, so that a PFI can be verified quickly. We also study two incremental mining algorithms for retrieving PFIs from evolving databases, but it does not support the Update or delete operation.

To overcome the above problem, this paper proposes a rule mining algorithm which supports Update, Insert and Delete Operations in Evolving Database. We will also examine how to use the model based approach to develop other mining algorithms (e.g., clustering and classification) on uncertain data. Another interesting work is to investigate. PFI mining algorithms for probability models that capture correlation among attributes and tuple.

The efficiency of the proposed method will be evaluated by computing the recall and precision values & results of dynamic programming and model based approach will be compared for finding better approach.

References

[1] A. Veloso, W. Meira Jr., M. de Carvalho, B. Po ssas, S. Parthasarathy, and M.J. Zaki, Mining Frequent item sets in Evolving Databases, Proc. Second SIAM Intl Conf. Data Mining (SDM), 2002.

[2] C. Aggarwal, Y. Li, J. Wang, and J. Wang, Frequent Pattern Mining with Uncertain Data, Proc. 15th ACM SIGKDD Intl Conf. Knowledge Discovery and Data Mining (KDD), 2009.

[3] T. Bernecker, H. Kriegel, M. Renz, F. Verhein, and A. Zuefle, Probabilistic Frequent Itemset Mining in Uncertain Databases, Proc. 15th ACM SIGKDD Intl Conf. Knowledge Discovery and Data Mining (KDD), 2009.

[4] R. Agrawal, T. Imielinski, and A. Swami, Mining Association Rules between Sets of Items in Large Databases, Proc. ACM SIGMOD Intl Conf. Management of Data, 1993.

[5] O. Benjelloun, A.D. Sarma, A. Halevy, and J.Widom, ULDBs: Databases with Uncertainty and Lineage, Proc. 32nd Intl Conf. Very Large Data Bases (VLDB), 2006.

[6] M. E. Otey C. Wang S. Parthasarathy, Computer and Information Science Dept.The Ohio-State University fotey, wachao, srinig@cis.ohio-state.edu

[7] X. Yan and J. Han"CloseGraph: Mining Closed Frequent Graph Patterns" Proc. Ninth ACM Int'l Conf. Knowledge Discovery and Data Mining, pp. 286-295, 2003.

[8] M. E. Otey, S. Parthasarathy ,Computer and Information Science Dept. The Ohio-State University fotey, wachao, srinig@cis.ohio-state.edu A. Veloso W. Meira Jr. Computer Science Dept. Universidade Federal de Minas Gerais.

[9] Efficient Mining of Frequent Item Sets on Large Uncertain Databases Liang Wang, David Wai-Lok Cheung, Reynold Cheng, Member, IEEE, Sau Dan Lee, and Xuan S. Yang

[10] C. Aggarwal, Y. Li, J. Wang, and J. Wang, "Frequent Pattern Mining with Uncertain Data," Proc. 15th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), 2009