IJESRT
INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH

A SURVEY ON MINING UNCERTAIN FREQUENT ITEM SET EFFECTIVELY USING PATTERN GROWTH APPROACH

Ankita B.Ramanandi[*], Amit H.Rathod
[*] IT Department ,Parul Institute of Engineering And Technology,India

ABSTRACT

The Frequent Itemset Mining (FIM) is well-known problem in data mining. The FIM is very useful for business intellisense, weather forecasting etc. Many frequent pattern mining algorithms find patterns from traditional transaction databases, in which the content of each transaction namely, items is definitely known and precise. However, there are many real-life situations in which the content of transactions is uncertain.

There are two main approaches for FIM: the level-wise approach and the pattern-growth approach. The level-wise approach requires multiple scans of dataset and generates candidate itemsets. The pattern-growth approach requires a large amount of memory and computation time to process tree nodes because the current algorithms for uncertain datasets cannot create a tree as compact as the original FP-Tree. In this literature the proposed method modifies the tree construction strategy in AT-Mine (Array based Tail node Tree) algorithm. The main goal of the proposed approach is to reduce the total time taken to mine the uncertain Frequent Item set using AT-Mine algorithm.

KEYWORDS: Frequent Item Set Mining (FIM), AT-Mine,Pattern Growth Approach .

INTRODUCTION

Data Mining is the process of extracting information from large data sets through the use of algorithms and techniques drawn from the field of Statistics, Machine Learning and Data Base Management Systems. Traditional data analysis methods often involve manual work and interpretation data that is slow, expensive and highly subjective. Data Mining, popularly called as knowledge discovery in large data, enables firms and organizations to make calculated decisions by assembling accumulating, analyzing and accessing corporate data. It uses variety of tools like query and reporting tools, analytical processing tools, and Decision Support System tools.

Data Mining, also popularly known as Knowledge Discovery in Databases (KDD), refers to the nontrivial extraction of implicit, previously unknown and potentially useful information from data in databases. While data mining and knowledge discovery in databases (or KDD) are frequently treated as synonyms, data mining is actually part of the knowledge discovery process. The Knowledge Discovery in Databases process comprises of a few steps leading from raw data collections to some form of new knowledge. The iterative process consists of the following steps:

- Data cleaning: also known as data cleansing, it is a phase in which noise data and irrelevant data are removed from the collection.
- Data integration: at this stage, multiple data sources, often heterogeneous, may be combined in a common source.
- Data selection: at this step, the data relevant to the analysis is decided on and retrieved from the data collection.
- Data transformation: also known as data consolidation, it is a phase in which the selected data is transformed into forms appropriate for the mining procedure.
- Data mining: it is the crucial step in which clever techniques are applied to extract patterns potentially useful.
- Pattern evaluation: in this step, strictly interesting patterns representing knowledge are identified based on given measures.
- Knowledge representation: is the final phase in which the discovered knowledge is visually represented to the user. This essential step uses visualization techniques to help users understand and interpret the data mining results.

Frequent Item set mining is very useful for many fields like business intelligence and many more. We state the problem to generate frequent item set mining over transactional dataset. Item in transactional database is described with existential probability in an uncertain transactional database. There exists an algorithm to solve this problem. We need more efficient and accurate approach to solve this problem. There is a need of an algorithm which can mine uncertain frequent item sets in efficient time.

MATERIAL AND METHODOLOGY

Data Mining Techniques

Association analysis: Association analysis is the discovery of what are commonly called association rules. It studies the frequency of items occurring together in transactional databases, and based on a threshold called support, identifies the frequent item sets. Association analysis is commonly used for market basket analysis.

Classification: Classification analysis is the organization of data in given classes. Also known as supervised classification, the classification uses given class labels to order the objects in the data collection. Classification approaches normally use a training set where all objects are already associated with known class labels.

Clustering: Similar to classification, clustering is the organization of data in classes. However, unlike classification, in clustering, class labels are unknown and it is up to the clustering algorithm to discover acceptable classes. Clustering is also called unsupervised classification, because the classification is not dictated by given class labels.

Prediction: Prediction has attracted considerable attention given the potential implications of successful forecasting in a business context. Prediction is however more often referred to the forecast of missing numerical values, or increase/ decrease trends in time related data. The major idea is to use a large number of past values to consider probable future values.

Evolution and Deviation Analysis: Evolution and deviation analysis pertain to the study of time related data that changes in time. Evolution analysis models evolutionary trends in data, which consent to characterizing, comparing, classifying or clustering of time related data. Deviation analysis, on the other hand, considers differences between measured values and expected values, and attempts to find the cause of the deviations from the anticipated values

Well Known Methods for Mining Frequent Item set

Here are some well-known algorithms like Apriori, Eclat and FP-Growth that are used to generate association rules with the help of frequent item sets.

To mine association rules, Apriori is the best-known algorithm. It uses the support of item sets and a candidate generation function which exploits the downward closure property of support .The eclat algorithm is based on the idea that it uses tid set intersections to compute the support of a candidate item set that avoids the generation of subsets. The FP-Growth Algorithm is used to find frequent item sets without using candidate generations, thus it improves the performance. The most important part of this method is the use of a special data structure-frequent-pattern tree (FP-tree), which gives the item set association information.

Uncertain Data

In recent years, many advanced technologies have been developed to store and record large quantities of data continuously. In many cases, the data may contain errors or may only be partially complete. For example, sensor networks typically create large amounts of uncertain datasets. In other cases, the data points may correspond to objects which are only vaguely specified, and are there for considered uncertain in their representation. Similarly, surveys and imputation techniques create data which is uncertain in nature.

The field of uncertain data management poses a number of unique challenges on several fronts. The two broad issues are those of modeling the uncertain data, and then leveraging it to work with a variety of applications. A number of issues and working models for uncertain data are there. The second issue is that of adapting data management and mining applications to work with the uncertain data. The main areas of research in the field are as follows:

Modelling of uncertain data: A key issue is the process of modelling the uncertain data. Therefore, the underlying complexities can be captured while keeping the data useful for database management applications.

Uncertain data management: In this case, one wishes to adapt traditional database management techniques for uncertain data. Examples of such techniques could be join processing, query processing, indexing, or database integration. .

Uncertain data mining: The results of data mining applications are affected by the underlying uncertainty in the data. Therefore, it is critical to design data mining techniques that can take such uncertainty into account during the computations.

RESULTS AND DISCUSSION

First we will see the concept of uncertain dataset.

TID	Transaction item set
T_1	(a: 0.8), (b: 0.7), (d: 0.9), (f: 0.5)
T_2	(c: 0.8), (d: 0.85), (e: 0.4)
T_3	(c: 0.85), (d: 0.6), (e: 0.6)
T_4	(a: 0.9) , (b: 0.85), (d: 0.65)
T_5	(a: 0.95), (b: 0.7), (d: 0.8) , (e: 0.7)
T_6	(b: 0.7), (c: 0.65), (f: 0.45)

Table 1. Uncertain Dataset

As shown in Table 1, an example of uncertain transaction dataset in which each transaction of which represents that a customer might buy a certain item with a certain probability. The value associated with each item is called the existential probability of the item. For example, the first transactionT_1 in Table 1 shows that a customer might purchase “a”, “b”, “d” and “f” with 80%, 70%, 90% and 50% chances in the future respectively. In this paper [1], they have proposed three important concepts for frequent item set mining for uncertain data.

1. A new tree structure named AT-Tree (Array based Tail node Tree) for maintaining important information related to an uncertain transaction dataset is given by researchers.
2. Algorithm named AT-Mine for FIM over uncertain transaction datasets based on AT-Tree was introduced.
3. Both sparse and dense datasets are used in experiments to compare the performance of the proposed algorithm against level-wise approach and pattern-growth approach, respectively.

They have introduced some definition for making algorithm based on survey and as per their requirement for AT tree algorithm. Now we will see definitions which are required. [1][5][6][7]

Suppose D = {T_1, T_2, …,T_n} can be an uncertain transaction dataset which contains n transaction item sets and m distinct items, i.e. I= {i_1, i_2, …, i_m}. Each transaction item set is represented as {i_1:p_1, i_2:p_2, …, i_v: p_v }, where {i_1, i_2, …, i_v} is a subset of I, and p_u ($1 \leq u \leq v$) is the existential probability of item i_u in a transaction item set. The size of dataset D is the number of transaction item sets and is denoted as |D|. An item set X = {i_1, i_2, …,i_k}, which contains k distinct items, is called a k-item set, and k is the length of the item set X.

Definition 1: The support number (SN) of an item set X in a transaction dataset is defined by the number of transaction item sets containing X.

Definition 2: The probability of an item iu in transaction T_d is denoted as p(i_u,T_d) and is defined by $p(i_u,T_d)=p_u$.For example, in Table 2.1, p({a},T1) = 0.8, p({b},T1) = 0.7, p({d},T1) = 0.9, p({f},T1) = 0.5.

Definition 3: The probability of an item set X in a transaction T_d is denoted as p(X, T_d) and is defined by $p(X,T_d)=\prod i_u \in X,X \subset T_d \quad p(i_u,T_d)$.For example, in Table 2.1, p({a, b},T1) = 0.8×0.7 = 0.56, p({a, b},T4)=0.9×0.85=0.765, p({a, b},T5) = 0.95×0.7=0.665.

Definition 4: The expected support number (exp SN) of an item set X in an uncertain transaction dataset is denoted as exp SN(X) and is defined by $exp\ SN(X)=\sum T_d \supseteq X,T_d \in DP(X,T_d)$.

For example, exp SN({a, b}) = p({a, b},T1) + p({a, b},T4) + p({a, b},T5) = 0.56+0.765+ 0.665 = 1.99.

Definition 5: Given a dataset D, the minimum expected support threshold η is a predefined percentage of |D|; correspondingly, the minimum expected support number (min Exp SN) is defined by $min\ Exp\ SN=|D| \times \eta$.

An item set X is called a frequent item set if its expected support number is not less than the value min Exp SN. Mining frequent item sets from an uncertain transaction dataset means discovering all item sets whose expected support numbers are not less than the value min Exp SN.

Definition 6: The minimum support threshold λ is a predefined percentage of |D|; correspondingly, the minimum support number (min SN) in a dataset D is defined by $min\ SN=|D| \times \lambda$.

Definition9: Let item set X= {i_1, i_2 ,i_3, …,i_u} be assorted item set ,and the item i_u is called *tail-item* of *X*. When the item set *X* is inserted in to a tree *T* in accordance with its items’ order, the node *N* on the tree that represents this *tail-item* is defined as ***tail node*** of item set *X*, and other nodes that represent items i_1, $i2$,…, $i_{u\text{-}1}$ are defined as **normal nodes**. The item set *X* is called ***tail-node-item set*** for node N.[1]

Definition10: Let an item set *X* contain item set *Y*. When item set *X* is added to a prefix tree of item set *Y*, the probability of item set *Y* in item set *X*, *p*(*Y*,*X*),is defined as the ***base probability*** of item set *X* on the tree *T*, and is denoted as *BP*(*X*,*Y*):

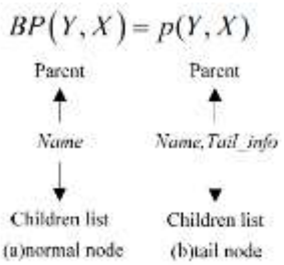


Figure 1. Structure of nodes on an AT-Tree

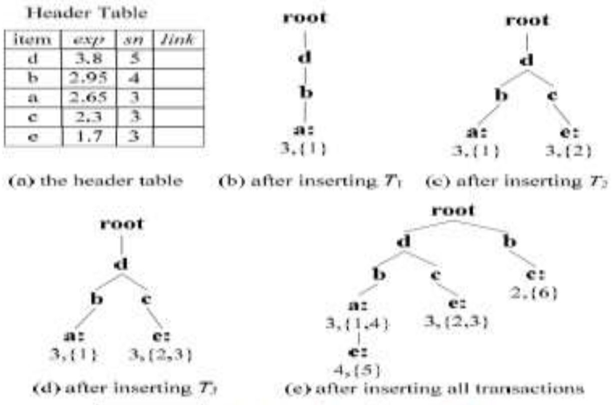
The node structure on an AT-Tree is illustrated in Figure1.There are two types of nodes :one is normal node, as shown in Figure 2.1(a),where *Name* is the item name of each node ;the other type is tail node, as shown in Figure 2.1(b), where ***Tail info*** is the supplemental information that includes 4 fields:(1) ***bp***: a list at keeps *base probability* values of all *tail-node-item sets*;(2) ***len***: the length of the tail-node-item set;(3) ***Arr_ind***: a list of index values of an array each element of which records probability values of items in each sorted transaction item set.[1]

The structure of AT-Tree [1] is designed to store the related information on tail nodes. It is constructed by two scans of dataset.

In the first scan, a header table is created to maintain sorted frequent items.

In the second scan, the probability values of frequent items In each transaction item sets are stored to a list according to the order of the header table; the list is then added to an array (and its corresponding sequence number in the array is denoted as *ID*);the frequent items in each transaction item set are inserted to an AT-Tree according to the order of the header table ;the length of the item set and the number *ID* are stored to the corresponding tail node. When the transaction item sets are added to an AT-Tree, they are rearranged in descending order of support numbers of items, and share the same node/nodes if their prefix items/item sets are identical. Thus the AT-Tree is as compact as the original FP-Tree. Moreover, AT-Tree does not lose probability information with respect to the distinct probability values of the transaction item sets. [1]

Here, example is given for understanding algorithm for mining.



Construction of an AT-Tree

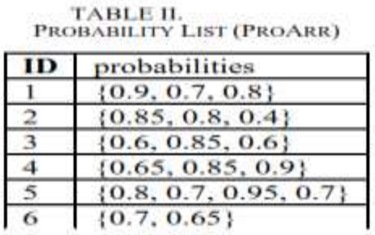
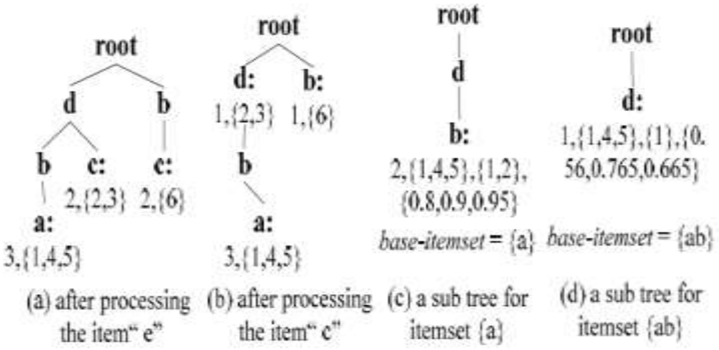


TABLE II.
PROBABILITY LIST (PROARR)

ID	probabilities
1	{0.9, 0.7, 0.8}
2	{0.85, 0.8, 0.4}
3	{0.6, 0.85, 0.6}
4	{0.65, 0.85, 0.9}
5	{0.8, 0.7, 0.95, 0.7}
6	{0.7, 0.65}



(a) after processing the item" e" (b) after processing the item" c" (c) a sub tree for itemset {a} (d) a sub tree for itemset {ab}

Figure 2. An Example of mining frequent item sets from uncertain dataset [1]

Algorithm steps for Creating AT-Tree [1]:

Step 1: Calculate the minimum expected support number min Exp SN.

Step 2: Put those items whose expected support numbers are not less than min Exp SN to a header table, and sort the items in the header table according to the descending order of their support numbers; finish the algorithm if the header table is null.

Step 3: Initially set the root node of the AT-Tree T as null.

Step 4: Remove the items that are not in the header table from each transaction item set, and sort the remaining items of each transaction item set according to the order of the header table, and get a sorted item set X.

Step 5: If the length of item set X is 0, process the next transaction item set.

Step 6: Process the next transaction item set.

Algorithm steps for Mining [1]:

Step 1: Process the items in the header table one by one from the last item by the following steps.

Step 2: Append item Z to the current base-item set (which is initialized as null); each new base-item set is a frequent item set.

Step 3: Let Z links in the header table H contain k nodes whose item name is Z; we denote these k nodes as N1, N2… Nk; because item Z is the last one in the header table, all these k nodes are tail nodes, i.e., each of these nodes contains a Tail info.

Step 4: Remove item Z from the base-item set.

Step 5: For each of these k nodes (which we denote as Ni, 1≤i≤k), modify its Tail info.

Step 6: Process the next item of the header table H.

CONCLUSION

AT-Mine algorithm for mining uncertain frequent item sets is studied. The algorithm uses pattern growth approach. The AT-Tree (Array based Tail node Tree) is new structure proposed.

Proposed method reduces number of database scan. At first database scan it simply creates tree and count the expected support and probability of items. Thus there is no need to scan the database second time to construct tree. This will reduce the total time for whole process.

In future we will implement our method and we will compare our result with AT-Mine algorithm. We will implement such a method which can use minimum scanning and filtering with same structure as in AT-Mine algorithm to generate pattern but efficiently.

ACKNOWLEDGEMENTS

Special thanks to parul Institute of Engineering and Technology for supporting in this research. I am grateful to Prof. Amit H. Rathod for helpful discussion and comments.

REFERENCES

1. Wang, Le, Lin Feng, and Mingfei Wu. "AT-Mine: An Efficient Algorithm of Frequent Item set Mining on Uncertain Dataset." *Journal of Computers* 8.6 (2013): 1417-1426.
2. Leung, Carson Kai-Sang, Mark Anthony F. Mateo, and Dale A. Brajczuk. "A tree-based approach for frequent pattern mining from uncertain data." *Advances in Knowledge Discovery and Data Mining*. Springer Berlin Heidelberg, 2008. 653-661.
3. Lu, Qiwei, et al. "Integrity Verification for Outsourcing Uncertain Frequent Item set Mining." *arXiv preprint arXiv:1307.2991* (2013).
4. Leung, Carson Kai-Sang, and Syed KhairuzzamanTanbeer. "PUF-tree: a compact tree structure for frequent pattern mining of uncertain data." *Advances in Knowledge Discovery and Data Mining*. Springer Berlin Heidelberg, 2013. 13-25.
5. C.W. Lin and T.P. Hong, "A new mining approach for uncertain databases using CUFP trees," *Expert Systems with Applications*, Vol.39 (4), pp.4084–4093, 2012.
6. C.K. Leung, C.L. Carmichael and B. Hao, Efficient mining of frequent patterns from uncertain data, in *International Conference on Data Mining Workshops (ICDM Workshops2007)*. 2007, pp.489-494.
7. Y. Liu, "Mining frequent patterns from univariate uncertain data," *Data and Knowledge Engineering*, Vol.71, no.1,pp.47-68, 2012.
8. L. Wang, D.W. Cheung, R. Cheng, S. Lee, and X. Yang," Efficient Mining of Frequent Item sets on Large Uncertain Databases," *IEEE Transactions on Knowledge and Data Engineering*, no.99(Pre Prints), 2011
9. C. Chui, B. Kao and E. Hung, Mining frequent item sets from uncertain data, in *11th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2007)*.2007, pp.47-58.
10. X. Sun, L. Lim and S. Wang, "An approximation algorithm of mining frequent item sets from uncertain data set," *International Journal of Advancements in Computing Technology*, Vol.4, no.3, pp.42-49, 2012.
11. C.K. Leung, M.A.F. Mateo and D.A. Brajczuk, A tree based approach for frequent pattern mining from uncertain data, in *12th Pacific-Asia Conference on KnowledgeDiscovery and Data Mining (PAKDD 2008)*. 2008, pp.653-661.
12. C.C. Agarwal, Y. Li, J. Wang, and J. Wang, Frequent pattern mining with uncertain data, in *15th ACM SIG KDD International Conference on Knowledge Discovery and Data Mining (KDD '09)*. 2009, pp.29-37

AUTHOR BIBLOGRAPHY

	Amit H.Rathod IT Department, Assistant Professor at Parul Institute of Engineering And Technology.Vadodara,India
	Ankita B.Ramanandi IT Department, Student at Parul Institute of Engeneering And Technology.Vadodara,India.