Survey on Mining Frequent Item Sets Over Uncertain Databases

Sudhir Lawand
Dept. Computer Engineering
Vidyalankar Polytechnic MS
India

Umesh Kulkarni
Dept. of Computer Engineering
Vidyalankar Institute of
Technology.MS India

***Abstract*— Uncertain data mining frequent itemsets over uncertain databases have attracted much attention due to its wide applications. In uncertain databases, instead of counting fixed occurrence of item sets, we use the support of an item set as random variable. The data modified from sensor tracking system and data integration rigorousness is highly ambiguous. One of the major issues is extracting frequent itemset from a large uncertain database, interpreted under the Possible World Semantics. The mining process can be modeled as a Poisson binomial distribution by observing that an uncertain database contains an exponential number of possible worlds. Mining such manifold itemset from generous ambiguous database illustrated under possible world semantics is a crucial dispute. Approximated algorithm is established to ascertain manifold itemset from generous ambiguous database exceedingly. A data trimming framework is used to improve mining efficiency. The data trimming technique can achieve significant savings in both CPU cost and I/O cost. We focus on the problem of mining frequent itemsets (MFI) over uncertain databases. In order to solve the MFI problem over uncertain databases, we use an efficient mining algorithm, named U-Apriori. Extensive experiments on both real and synthetic datasets verify that this algorithm is effective and efficient.**

Keywords— ***Existential uncertain data, Uncertain probabilistic, aggregation, Regression, Data Mining.***

I. Introduction

Existential Uncertain Data Model:- Association analysis is one of the most important data-mining models. As an example, in market-basket analysis, a dataset consists of a number of tuples each contains the items that a customer has purchased in a transaction. The dataset is analyzed to discover associations among different items. An important step in the mining process is the extraction of frequent itemsets or sets of items that co-occur in a major fraction of the transactions. Besides market-basket analysis, frequent itemsets mining is also a core component in other variations of association analysis, such as association-rule mining [1] and sequential-pattern mining [2]. All previous studies on association analysis assume a data model under which transactions capture doubtless facts about the items that are contained in each transaction. In many applications, however, the existence of an item in a transaction is best captured by a likelihood measure or a probability. As an example, a weather forecast dataset may contain a table of forecast records (tuples), each of which contains a set of attributes cloudy, rainy and sunny. Applying association analysis on such a dataset allows us to discover any potential correlations among the attributes and forecast. In many cases, attributes, being subjective observations, would best be represented by probabilities that indicate their forecast tuples.

Table 1. A Forecast dataset

Country Name	Cloudy	Rainy	Sunny
1	90%	80%	2%
2	50%	30%	5%

Table 1 shows an example forecast dataset. A probability value in such a dataset might be obtained by historical data statistics. (For example, a forecast of country that who shows positive reaction to Test A has a 70% probability of Rain another example of uncertain datasets is pattern recognition applications. Given a satellite picture, image processing techniques can be applied to extract features that indicate the presence or absence of certain target objects (such as bunkers). Due to noises and limited resolution, the presence of a feature in a spatial area is often uncertain and expressed as a probability [3]. Here, we can model a spatial region as an object, and the features (that have non-zero probabilities of being present in a region) as the items of that object. The dataset can thus be considered as a collection of tuples/transactions, each contains a set of items (features) that are associated with the probabilities of being present. Applying association analysis on such a dataset allows us to identify closely-related features. Such knowledge is very useful in pattern classification [4] and image texture analysis [5]. Each record contains a set of items that are associated with existential probabilities. A core step in many association analysis techniques is the extraction of frequent itemsets. An itemset is considered frequent if it appears in a large enough portion of the dataset. The occurrence frequency is often expressed in terms of a support count. For datasets that contain uncertain items, however, the definition of support needs to be redefined. Due to the probabilistic nature of the datasets, the occurrence frequency of an itemset should be captured by an expected support instead of a traditional support count. It contains the Possible Worlds interpretation of an uncertain dataset [6] and also how expected supports can be computed by a simple modification of the well-known Apriori algorithm [1]. The rest of this report is organized as follows. First, it describes the Possible Worlds interpretation of existential uncertain data and defines the expected support measure. Then it discusses a simple modification of the Apriori algorithm to mine uncertain data and explains why such a modification does not lead to an efficient algorithm. It also presents a data trimming technique to improve mining efficiency. It shows some experimental results and discusses some observations. [20]

II. Possible World Interpretation of Existential Uncertain Data

Since the existence of an item in a transaction is indicated by a probability, an advantage of the existential uncertain data model is that it allows more information to be captured by the dataset. Consider again the example patient dataset. If we adopt a binary data model, then each symptom/illness can either be present (1) or absent (0) in a patient record. Under the binary model, data analysts will be forced to set a threshold value for each symptom/illness to quantize the probabilities into either 1 or 0. In other words, information about those (marginally) low values is discarded. The uncertain data model, however, allows such information be retained and be available for analysis. The disadvantage of retaining such information is that the size of the dataset would be much larger Mining Frequent itemsets from Uncertain Data than that under the quantized binary model. This is particularly true if most of the existential probabilities are very small. Consequently, mining algorithms will run a lot slower on such large datasets. This is an efficient technique for mining existential uncertain datasets, which exploit the statistical properties of low-valued items. Through experiments, the proposed technique is very efficient in terms of both CPU cost and I/O cost. [20]

Expected support measure

Expected Support is calculated by summing up the weighted support counts of ALL the possible worlds.

Expected support $(X) = \sum_{i=1}^{2[|D|]X[|U|]} likehood\,(Di) Support(X, Di)$

Where |D| is the number of transactions |U| is the number of attributes
Instead of enumerating all Possible Worlds to calculate the expected support, it can be calculated by scanning the uncertain dataset once only.

Expected Support $= \sum_{i=1}^{|p|} \prod_{j=1}^{|x|} pi(xj)$

III. The Apriori Algorithm

Apriori algorithm is a classical algorithm in data mining. It is used for mining frequent itemsets and relevant association rules. It is devised to operate on a database containing a lot of transactions, for instance, items brought by customers in a store. It is very important for effective Market Basket Analysis and it helps the customers in purchasing their items with more ease which increases the sales of the markets. It has also been used in the field of healthcare for the detection of adverse drug reactions. It produces association rules that indicate what all combinations of medications and patient characteristics lead to ADRs.

A. Association rules

Association rule learning is a prominent and a well-explored method for determining relations among variables in large databases. Let us take a look at the formal definition of the problem of association rules given by Rakesh Agrawal, the President and Founder of the Data Insights Laboratories. Let I={i1,i2,i3,…,in} be a set of n attributes called items and D={t1,t2,…,tn} be the set of transactions. It is called database. Every transaction, ti in D has a unique transaction ID. A rule can be defined as an implication, X⟶Y where X and Y are subsets of I(X, Y⊆I), and they have no element in common, i.e., X∩Y. X and Y are the antecedent and the consequent of the rule, respectively. Let's take an easy example from the supermarket sphere. The example that we are considering is quite small and in practical situations, datasets contain millions or billions of transactions. The set of item sets,

I ={Onion Burger, Potato, Milk, Beer}

and a database consisting of six transactions. Each transaction is a tuple of 0's and 1's where 0 represents the absence of an item and 1 the presence. An example for a rule in this scenario would be {Onion, Potato} => {Burger}, which means that if onion and potato are bought, customers also buy a burger.

Table 2: Indicates transactions of different data

Transaction ID	Onion	Potato	Burger	Milk	Beer
t1	1	1	1	0	0
t2	0	1	1	1	0
t3	0	0	0	1	1
t4	1	1	0	1	0
t5	1	1	1	0	1
t6	1	1	1	1	1

There are multiple rules possible even from a very small database, so in order to select the interesting ones, we use constraints on various measures of interest and significance. We will look at some of these useful measures such as support, confidence, lift, and conviction.

1) Support

The support of an itemset X, supp(X) is the proportion of transaction in the database in which the item X appears. It signifies the popularity of an itemset. Supp(X)=Number of transaction in which X appears Total number of transactions. In the example above, supp (Onion) =46=0.66667. If the sales of a particular product (item) above a certain proportion have a meaningful effect on profits, that proportion can be considered as the support threshold. Furthermore, we can identify itemsets that have support values beyond this threshold as significant itemsets.

2) Confidence

Confidence of a rule is defined as follows:

conf(X⟶Y)=supp(X∪Y)supp(X)

It signifies the likelihood of item Y being purchased when item X is purchased. So, for the rule {Onion, Potato} =>{Burger}, this implies that for 75% of the transactions containing onion and potatoes, the rule is correct. It can also be interpreted as the conditional probability P(Y|X), i.e., the probability of finding the itemset Y in transactions given the transaction already contains X. It can give some important insights, but it also has a major drawback. It only takes into account the popularity of the itemset X and not the popularity of Y. If Y is equally popular as X then there will be a higher probability that a transaction containing X will also contain Y thus increasing the confidence. To overcome this drawback there is another measure called lift.

3) Lift

The lift of a rule is defined as:

$\text{lift}\ (X \longrightarrow Y) = \text{supp}(X \cup Y)\ \text{supp}(X) * \text{supp}(Y)$

This signifies the likelihood of the itemset Y being purchased when item X is purchased while taking into account the popularity of Y. In our example above, Undefined control sequence implies

If the value of lift is greater than 1, it means that the itemset Y is likely to be bought with itemset X, while a value less than 1 implies that itemset Y is unlikely to be bought if the itemset X is bought.

4) Conviction

The conviction of a rule can be defined as:

$\text{conv}(X \longrightarrow Y) = 1 - \text{supp}(Y) 1 - \text{conf}(X \longrightarrow Y)$

For the rule {onion, potato} =>{burger}

Undefined control sequence \implies

The conviction value of 1.32 means that the rule {onion, potato} =>{burger} would be incorrect 32% more often if the association between X and Y was an accidental chance.

5) General Process of the Apriori algorithm

The entire algorithm can be divided into two steps:

Step 1: Apply minimum support to find all the frequent sets with k items in a database.

Step 2: Use the self-join rule to find the frequent sets with k+1 items with the help of frequent k-itemsets. Repeat this process from k=1 to the point when we are unable to apply the self-join rule. [19]

This approach of extending a frequent itemset one at a time is called the bottom-up approach.

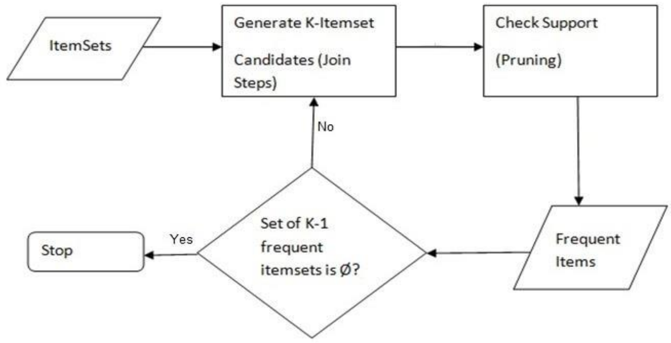

Fig.1 General Process of the Apriori algorithm

B. Pros of the Apriori algorithm

1. It is an easy-to-implement and easy-to-understand algorithm.
2. It can be used on large itemsets.[19]

C. Cons of the Apriori Algorithm

1. Sometimes, it may need to find a large number of candidate rules which can be computationally expensive.
2. Calculating support is also expensive because it has to go through the entire database. [19]

III. Data Trimming Technique

More specifically, the data trimming technique works under a framework that consists three modules: the trimming module, pruning module and patch up module. As shown in figure, the mining process starts by passing an uncertain dataset D into the trimming module. It first obtains the frequent items by scanning D once. A trimmed dataset DT is constructed by removing the items with existential probabilities smaller than a trimming threshold t in the second iteration. Depending on the trimming strategy, t can be either global to all items or local to each item. Some statistics such as the maximum existential probability being trimmed for each item is kept for error estimation. DT is then mined by U-Apriori. Notice that if an itemset is frequent in DT, it must also be frequent in D. On the other hand, if an itemset is infrequent in DT, we cannot conclude that it is infrequent in D. [12]

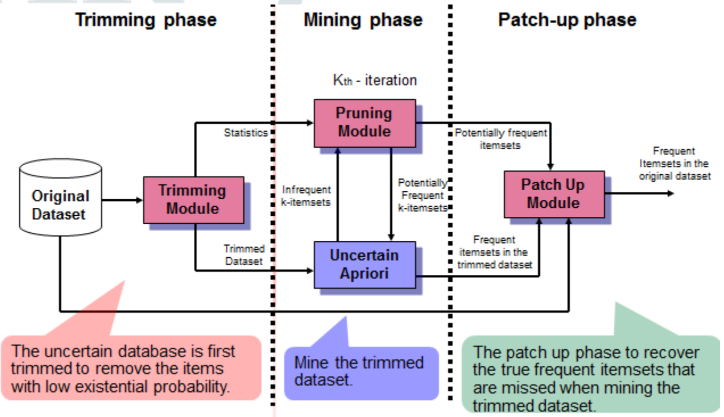
Trimming phase
Mining phase
Patch-up phase
Kth - iteration
Pruning Module
Statistics
Original Dataset
Trimming Module
Infrequent k-itemsets
Potentially Frequent k-itemsets
Potentially frequent itemsets
Frequent Itemsets in the original dataset
Patch Up Module
Trimmed Dataset
Uncertain Apriori
Frequent itemsets in the trimmed dataset
The uncertain database is first trimmed to remove the items with low existential probability.
Mine the trimmed dataset.
The patch up phase to recover the true frequent itemsets that are missed when mining the trimmed dataset.

Fig.2 The Data Trimming Framework [12]

Drawback

The main drawback of Data Trimming is that its performance is sensitive to the percentage of low probability items.

IV. Decremental Pruning Techniques

The decremental pruning technique achieves candidate reduction during the mining process. It estimates the upper bounds of expected supports of the candidate itemsets progressively after each dataset transaction is processed.
Following two methods are used:
1. Aggregate by singleton method (AS)
2. Common Prefix method (CP)

A. Aggregate by singleton method (AS)

It reduces the number of decremental counters to the number of frequent singletons.[23]
If the counter $i_s(a,k)$ drops below the minimum support requirement, candidates that contain *a* must be infrequent and can be pruned.

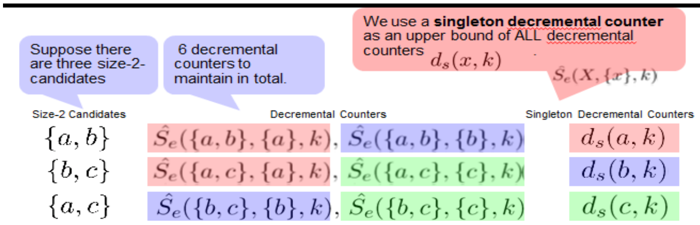
Suppose there are three size-2-candidates
6 decremental counters to maintain in total.
We use a singleton decremental counter as an upper bound of ALL decremental counters
$d_s(x,k)$
$\hat{S}_e(X,\{x\},k)$
Size-2 Candidates
Decremental Counters
Singleton Decremental Counters
$\{a,b\}$ $\hat{S}_e(\{a,b\},\{a\},k)$, $\hat{S}_e(\{a,b\},\{b\},k)$ $d_s(a,k)$
$\{b,c\}$ $\hat{S}_e(\{a,c\},\{a\},k)$, $\hat{S}_e(\{a,c\},\{c\},k)$ $d_s(b,k)$
$\{a,c\}$ $\hat{S}_e(\{b,c\},\{b\},k)$, $\hat{S}_e(\{b,c\},\{c\},k)$ $d_s(c,k)$

Fig 3 : Aggregate singleton decremental counter [12]

Initialization and update of singleton decremental counter [23]

$$\left\{ ds(x,k) = \begin{cases} Se\,(\{x\}) \quad , & \text{if } k = 0 \\ ds(x, k-1) - ptk(x) * \{1 - \max(k)\}, & \text{if } k > 0 \end{cases} \right.$$

Where $max_s(k)$=max {p_{tk}(x')x'€ tk ,x'≠ x}returns maximum existential probability among the items (expected x) in transactions t_k

B. Common Prefix method (CP)

Another method is to aggregate the decremental counters with common prefix.

We only keep those decremental counters where *X'* is a proper prefix of *X*.[18] We use a prefix decremental counter as an upper bound of all decremental counters.

Candidates with common prefix are organized under the same subtree in a prefix tree (hash tree). [18]

If the value of a prefix decremental counter drops below the minimum support requirement, all the candidates in the corresponding subtree can be pruned.

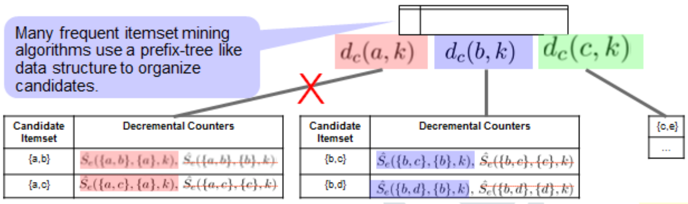
Many frequent itemset mining algorithms use a prefix-tree like data structure to organize candidates.
$d_c(a,k)$
$d_c(b,k)$
$d_c(c,k)$
Candidate Itemset
Decremental Counters
{a,b} $\hat{S}_e(\{a,b\},\{a\},k)$, $\hat{S}_e(\{a,b\},\{b\},k)$
{a,c} $\hat{S}_e(\{a,c\},\{a\},k)$, $\hat{S}_e(\{a,c\},\{c\},k)$
Candidate Itemset
Decremental Counters
{b,c} $\hat{S}_e(\{b,c\},\{b\},k)$, $\hat{S}_e(\{b,c\},\{c\},k)$
{b,d} $\hat{S}_e(\{b,d\},\{b\},k)$, $\hat{S}_e(\{b,d\},\{d\},k)$
{c,e}
...

Fig 4: Common Prefix method (CP)[12]

Initialization and update of prefix decremental counter

$$d_p(X',k)=\begin{cases} s_e(X') & \text{if } k = 0. \\ d_p(X', k-1) - s_e^{tk}(X') * \{1 - max_p(k)\} & \text{if } k > 0. \end{cases}$$

Where $s_e^{tk}(X') = \prod_{x \in X'} p_{tk}(x)$ and max_p (k)=max{p_{tk}(z)| z is after all the Itemset in X' according to the ordering φ}[4]

V. Advantages And Disadvantages

A. Advantages

- This data mining technique calculates the lower and upper bounds that filter out infrequent items but also identifies frequent items.
- It eliminates the redundant frequent itemsets and the noisy influence in uncertain data by data trimming technique.
- Enhances the efficiency of the mining process, by designing Apriori framework.
- The decremental pruning technique achieves candidate reduction during the mining process.

B. Disadvantages

- U-Apriori algorithm encounters high communication cost in the pruning phase.
- Increases the computational cost due to patch-up phase.
- U-Apriori discovers the frequent itemsets with candidate itemset generation.

VI. Conclusion & Future Scope

We studied the problem of mining frequent itemsets under a probabilistic framework. Data trimming technique works well when there are substantial amount of items with low existential probabilities in the dataset. The decremental pruning technique that achieves candidate reduction during the mining process. The Decremental Pruning and Data Trimming techniques can be combined to yield the most stable and efficient algorithm. Thus data trimming technique eliminates noisy influence in uncertain data. It also reduces redundant frequent itemsets from uncertain data. But its drawback is that when there are few low probability items in the dataset then data trimming technique can be counterproductive i.e. its performance is sensitive to the percentage of low probability items. The combined method, which uses both Common Prefix and Data Trimming provides a good balance and gives consistently good performance. A survey is carried to study how an uncertain data can be used for conclusion of predicted values. These data mining techniques and data trimming methods can generate new business opportunities in future.

VII. References

[1] Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: Proc. of 20th International Conference on Very Large Data Bases, September 12-15, 1994, Santiago de Chile, Chile, Morgan Kaufmann (1994) 487–499[22]

[2] Agrawal, R., Srikant, R.: Mining sequential patterns. In: Proc. of the 11th International Conference on Data Engineering, March 6-10, 1995, Taipei, Taiwan, IEEE Computer Society (1995) 3–14[22]

[3] Dai, X., Yiu M .L Mamoulis, N., Tao, Y., Vaitis, M.: Probabilistic spatial queries on existentially uncertain data. In: SSTD. Volume 3633 of Lecture Notes in Computer Science., Springer (2005) 400–417[22]

[4] Liu, B., Hsu, W., Ma, Y.: Integrating classification and association rule mining. In: KDD. (1998) 80–86[22]

[5] Rushing, A., Ranganath, S., Hinke, H., Graves, J.: Using association rules as texture features. IEEE Trans. Pattern Anal. Mach. Intell. 23(8) (2001) 845–858[22]

[6] Zim´anyi, E., Pirotte, A.: Imperfect information in relational databases. In: uncertainty management in Information Systems. (1996) 35–88[22]

[7] A. Veloso, W. Meira Jr., M. de Carvalho, S.Parthasarathy, and M.J. Zaki, Mining Frequent Itemsets in Evolving Databases, Proc. Second SIAM Int'l Conf. Data Mining (SDM), 2002.[24]

[8] 8. 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.[24]

[9] C. Aggarwal and P. Yu, A Survey of Uncertain Data Algorithms and Applications, IEEE Trans Knowledge and Data Eng., vol. 21, no. 5, pp. 609-623, May 2009.[24]

[10] R. Agrawal, T. Imielinski, and A. Swami, Mining Association Rules between Sets of Items in Large Databases, Proc. ACM SIGMOD Int'l Conf. Management of Data, 1993.[24]

[11] O. Benjelloun, A.D. Sarma, A. Halevy, and J. Widom, ULDBs: Databases with Uncertainty and Lineage, Proc. 32nd Int'l Conf. Very Large Data Bases (VLDB), 2006.[24]

[12] Chun-Kit Chui, Ben Kao, and Edward Hung, Mining Frequent Itemsets from Uncertain Data Springer-Verlag Berlin Heidelberg 2007.DOI https://doi.org/10.1007/978-3-540-71701-0_8 online ISBN 978-3-540-71701-0

[13] T. Bernecker, H. Kriegel, M. Renz, F. Verhein, and A. Zuefle, Probabilistic Frequent Itemset Mining in Uncertain Databases, Proc. 15th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), 2009.[24]

[14] C.J. van Rijsbergen, Information Retrieval. Butterworth, 1979.[24]

[15] L.L. Cam, An Approximation Theorem for the Poisson Binomial Distribution, Pacific J. Math., vol. 10, pp. 1181-1197, 1960.[24]

[16] H. Cheng, P. Yu, and J. Han, Approximate Frequent Itemset Mining in the Presence of Random Noise, Proc. Soft Computing for Knowledge Discovery and Data Mining, pp. 363-389, 2008.[24]

[17] R. Cheng, D. Kalashnikov, and S. Prabhakar, Evaluating Probabilistic Queries over Imprecise Data, Proc. ACM SIGMOD Int'l Conf. Management of Data, 2003.[24]

[18] Advance in Knowledge Discovery and Data Mining , Springer Nature America, Inc., 2008

[19] www.hackerearth.com

[20] www4.comp.polyu.edu.uk

[21] www.ijcaonline.org

[22] Chun-Kit Chui. Mining Frequent Itemsets from uncertain Data Lecture Notes in Computer Science,2007

[23] Chun-Kit Chui . A Decremental Approach for Mining Frequent Itemsets from uncertain Data Lecture Notes in Computer Science,2008

[24] www.hub.hku.hk