A Novel Acceleration Technique to Improve the Speed of Mining Frequent U2 Patterns

K. S. Kalaivani
PG Scholar
Department of CSE, Kongu Engineering College
Erode, TamilNadu, India

S. Kuppuswami
Principal
Kongu Engineering College
Erode, TamilNadu, India

ABSTRACT
Frequent pattern mining is the method of finding patterns like itemsets, subsequences and substructures that repeatedly occur in a dataset. In Univariate Uncertain data, each attribute present in a transaction is represented by a quantitative interval and a probability value. U2P-Miner algorithm is used to mine frequent patterns from U2 data. The number of intervals has a great impact on the time taken for mining frequent patterns. A novel acceleration technique which compares the expected support with the user specified threshold is introduced to minimize the number of intervals thereby improving the speed of the mining process. The runtime of the modified U2P-Miner algorithm is compared with the existing U2P-Miner algorithm.

General Terms
Frequent pattern mining.

Keywords
U2P-tree, Univariate Uncertain data, modified U2P-Miner.

1. INTRODUCTION
Frequent pattern mining plays an essential role in finding correlations, associations, and many other interesting relationships present among data. A number of mining algorithms such as Apriori algorithm [1], FP- growth algorithm [2] and H-mine algorithm [3] were developed to discover frequent itemsets in a database. Frequent pattern mining can be performed on two types of data namely precise data (where users definitely know whether the data is present or absent in a transaction) and uncertain data (where users are not sure about the presence or absence of data items and they only know that data items probably occur).

Uncertain data can be classified into two types, namely itemset uncertain data and Univariate Uncertain data. In U2 data, each attribute present in a transaction is represented by a quantitative interval and a probability value. This value is called as the existential probability. It is the possibility of the value to appear in the transaction. If a low sensitivity sensor is used to compute the quality of air, it may record a quantitative interval to represent the amount of different gases present in the air every day. Table 1 shows an example database consisting of five transactions. Here the attributes A1, A2 and A3 represent the different gases like carbon monoxide, sulphur dioxide and nitrogen dioxide. A quantitative interval is used to record the values for the attributes of each transaction. The "–" symbol shows the absence of an attribute resulting from a sensor malfunction. In U2 data, finding the support count of a pattern is more difficult than in itemset uncertain data. Since, the fundamental element that makes up a pattern in U2 data is not well-defined, the algorithms employed for mining frequent patterns from precise data cannot be used. So, the algorithm must be totally restructured for mining patterns from U2 data.

Table 1. An example database of univariate uncertain data

Attribute / Transaction	A1	A2	A3
T1	[51,75]	[85,100]	[32,54]
T2	[51,62]	[70,110]	[30,54]
T3	[51,62]	-	[32,42]
T4	-	[60,75]	-
T5	[51,75]	-	[32,42]

U2P-Miner algorithm [4] is used for mining frequent patterns from U2 data. The implementation of the algorithm is done in two steps. In the first step, all the transactions in the target database are compressed to construct a U2P-tree. Each and every branch present in the U2P-tree is made up of atomic sub-intervals. In the second step, the U2P-tree is then used to mine the frequent U2 patterns. The base intervals are finally combined to determine the potential frequent U2 patterns and checked by traversing the U2P-tree.

The remainder of this paper is organized as follows. In Section 2, the related works on uncertain and quantitative data mining methods are discussed. Section 3 describes the proposed method in detail. Section 4 deals with the experimental setup and analysis. Finally the paper is concluded in Section 5.

2. LITERATURE REVIEW
Chui et al. [5] proposed the U-Apriori algorithm to mine frequent patterns from uncertain data. Since, it uses candidate generate-and-test methodology it takes a longer time to mine frequent items from large datasets. Subsequently, Leung et al. [6, 7] developed the UF-growth algorithm which is the modification of FP-growth algorithm. Since it uses a tree structure, it is faster than the U-Apriori algorithm. In real life situations, not all the patterns mined are interesting to the user. So, finding all the frequent patterns would be redundant and wastes a lot of computation. This situation results in constrained mining [8] which aims at mining only interesting

patterns. Aggarwal et al. [9] made a comparison on the broad classes of mining algorithms and concluded that the extension of the H-mine algorithm is more efficient than the other algorithms. The H-mine algorithm differs from the FP-growth method by using a hyperlinked structure, which uses the linkage behavior among transactions corresponding to a branch of the FP-tree without actually creating a projected database. All of the above methods are based on the possible world configuration [10].The expected support is defined as the number of transactions that is expected to contain the pattern.

Bernecker et al. [11] suggested the probabilistic framework for mining frequent itemsets based on possible world semantics. Srikant and Agrawal [12] introduced techniques for mining association rules from quantitative and categorical data. The values of the attribute are fine-partitioned to form non-overlapping intervals and are used to mine the association rules called as the quantitative association rules (QAR).

Even though the U2P-miner algorithm deals with the calculation of existential probability, partial expected support and expected support, it is concerned with U2 data, in which basic elements are combined to form a merged pattern. It deals with the data, where the attribute present in a transaction is represented by a quantitative interval and a probability value. Hence, it is totally different from uncertain and quantitative mining algorithms. The algorithm also constructs a tree instead of using an array or a linked list. Therefore it performs better than the other existing algorithms. Since, the unwanted base intervals are not excluded, the algorithm takes a longer time to find the frequent U2 patterns. A new acceleration technique is proposed to delete as well as merge the base intervals to improve the speed of the U2P-miner algorithm.

3. THE PROPOSED METHOD

3.1 The Proposed Framework

Fig.1 represents the framework of the proposed work. It starts with the formation of base intervals followed by the construction of the U2P-tree. The next step is the application of the acceleration technique to remove the unwanted base intervals, followed by the modification of the U2P-tree.The last step calculates the frequent U2 patterns by constructing the pattern base and the corresponding conditional U2P-tree.

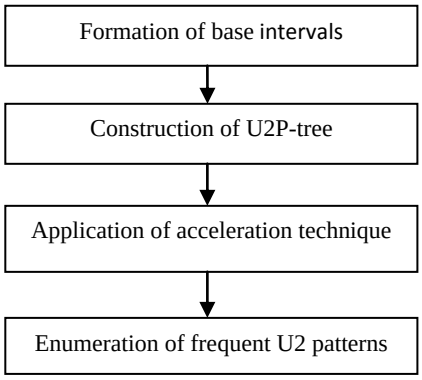

Fig 1: The Proposed Framework

3.2 Formation of Base Intervals

To explain the base interval formation scheme, the attribute A1 in Table 1 is used as an example. The values 51, 62 and 75 shown as lower/ upper bounds of A1, constitute the base intervals for A1 as [51, 62] and [62, 75].. Consider the user specified minimum support as 2.5. If the intervals are not decomposed, then there are no frequent patterns. On the other hand, if they are decomposed, then the U2 pattern [A1:[51, 62]] is frequent. The expected support is calculated as 11/24+11/11+11/11+11/24=2.917. Therefore, as new transactions arrive, decomposition of the intervals of each attribute should be performed if necessary.

3.3 Construction of U2P-tree

All the transactions in the target database are compressed to construct a U2P-tree. This is done by decomposing each base interval into atomic sub-intervals. Each node present in all the branches of the tree denotes a base element and its occurrence frequency. The occurrence frequency is defined as the number of transactions in which the base element is present. The partial expected support of the node (which is the product of the existential probability and occurrence frequency) is also recorded. This method scans the database only once to construct a U2P-tree. Consider the example database in Table 1.The resulting U2P-tree after inserting all the five transactions is shown in figure 2.

3.4 Application of Acceleration Technique

The acceleration technique is used to minimize the number of base intervals, thereby improving the mining speed. The technique works as follows: It deletes those base intervals whose expected support is below half of the user specified threshold and merges those intervals whose expected support is above half of the threshold value. For example, consider the threshold value to be 0.5.The technique deletes the intervals less than 0.25(that is 0.5/2) and merges the intervals between 0.25 and 0.5.In the example database all the intervals for attribute A1 is above 0.5, so they are used as such in the mining process. For attribute A2, BI_4 and BI_5 are merged since their expected support lies between 0.25 and 0.5.The interval BI_7 is left without merging since it has no other intervals to be merged. For attribute A3, the interval BI_8 is less than 0.25 and so it is deleted. Now, the modified U2P-tree as shown in figure 3 is constructed and the base intervals are renamed and used for the mining process.

3.5 Enumeration of Frequent U2 Patterns

The calculation of frequent U2 patterns begins with the last interval of the final attribute i.e., with BI_{10} of A_3. The partial expected supports of those nodes that have $[BI_{10}]$ in all the branches are added to calculate the expected support of U2 pattern $[BI_{10}]$.Therefore, the expected support for $[BI_{10}]$ is 0.545+0.5=1.045.This means that the interval $[BI_{10}]$ is not frequent because the expected support is below the minimum support (i.e., given by the user as 2.5). Next, checking the U2 pattern $[BI_9, BI_{10}]$ is done. The expected support of the U2 pattern $[BI_9, BI_{10}]$ is (0.455+0.545)+(0.417+0.5)=1.917. Here also the expected support is below the minimum support. So the interval $[BI_9, BI_{10}]$ is also not frequent. Next, checking the U2 pattern $[BI_8, BI_9, BI_{10}]$ is done and found to be not frequent. Testing all U2 patterns containing BI_{10} is now completed.

The next step checks the U2 patterns containing BI_9. All the patterns, except $[BI_8, BI_9, BI_{10}]$ and $[BI_9, BI_{10}]$ (patterns were already checked) are examined. The expected support of $[BI_9]$ is found to be 0.455+1.0+0.417+1.0=2.872.

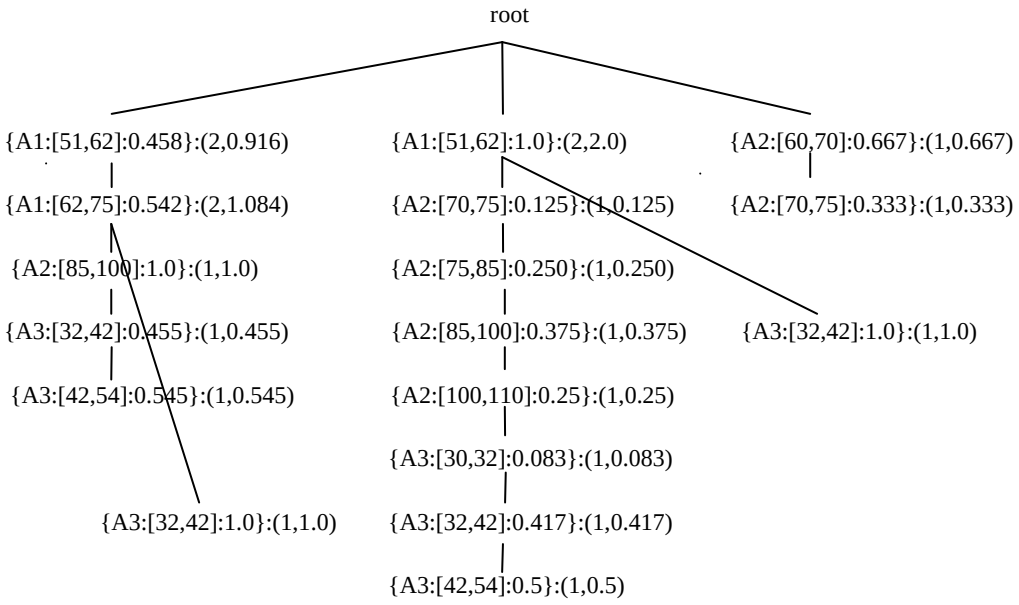

Fig 2: The U2P-tree after the fifth insertion operation

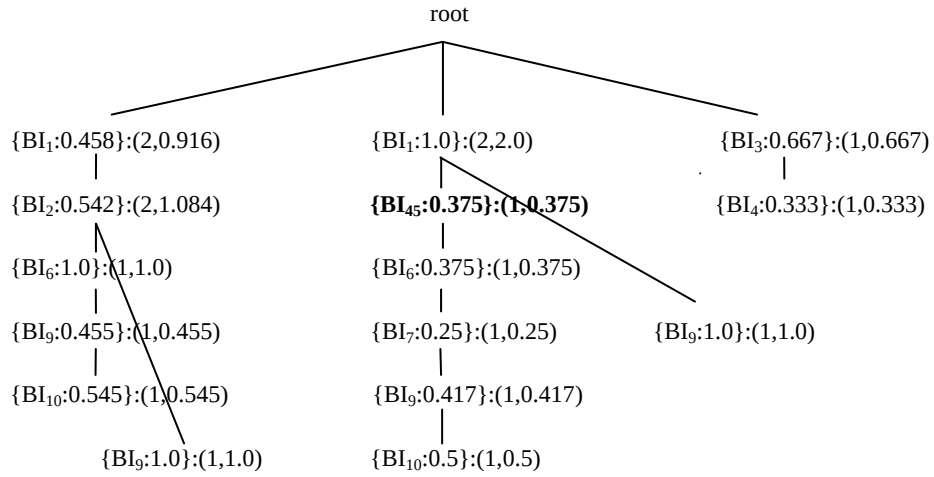

Fig 3: The modified U2P-tree after renaming the intervals

Table 2.The Conditional U2 Pattern base

Prefix Attribute Serial number	A1	A2
CP1	{BI_1:0.458} {BI_2:0.542}	{BI_6:1.0}
CP2	{BI_1:0.458} {BI_2:0.542}	-
CP3	{BI_1:1.0}	**{BI_{45}:0.375}** {BI_6:0.375} {BI_7:0.25}
CP4	{BI_1:1.0}	-

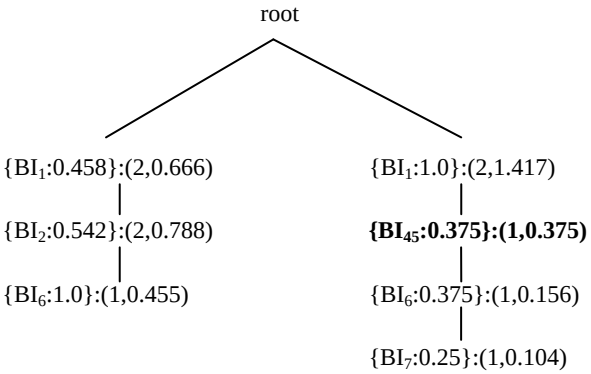

Fig 4: The Conditional U2P-tree

The expected support is above the minimum support i.e., 2.5 and so the pattern [BI_9] is frequent. Now, the conditional U2 pattern base of [BI_9] is derived as given in Table 2. [BI_9] is called a projected base. The conditional U2P-tree of [BI_9] is used to construct the conditional U2 pattern base, as shown in Figure 4.The projected base should be considered, to calculate the partial expected support.

There is a parent path in the original U2P-tree for each path in the conditional U2P-tree. That parent tree path holds the projected base. Here the calculation of the partial expected support is different from the original U2P-tree. It is calculated as the product of the partial expected support of the projected base on the corresponding parent tree path and the existential probability of the base element of the conditional U2P-tree node. The base element {BI_1:0.458} appears two times in the pattern base. So the occurrence frequency of {BI_1:0.458} is 2. The partial expected support is calculated as (0.455×0.458)+(1.0×0.458)=0.666. The process of mining the patterns from conditional U2P-tree is similar to mining patterns from original U2P-tree. Only BI_1 is found to be frequent in the conditional U2P-tree, i.e., [BI_1, BI_9] is frequent.

After examining BI_9, the process of calculating the frequent U2 patterns containing BI_8 is performed and found to be not frequent. Thus, checking all the base intervals of attribute A3 is over. The enumeration process then proceeds with the base intervals of attribute A2.It is found that all the U2 patterns comprised of the base intervals of attribute A2 is not frequent. The set of base intervals of attribute A1 are finally checked. The U2 pattern [BI_1] is found to be frequent. Thus for the example database, only the U2 patterns [BI_1], [BI_9] and [BI_1, BI_9] are frequent.

4. EXPERIMENTS

For experiment, a real dataset named AirQuality is used. It contains values of the different gases present in the air in Taiwan for the year 2008. The dataset can be downloaded from the Taiwan Environmental Protection Administration website [13]. The original AirQuality dataset contains number of indices, out of which five indices namely, carbon monoxide, suspended particulates, sulphur dioxide, nitrogen dioxide, and ozone are selected. The original dataset contains readings for each observation station (measured every hour for all the 365 days). The base intervals are obtained by using minimum and maximum values of each index as lower and upper boundary of the interval. There are totally 26,527 transactions.

The first set of experiments can be performed to analyze the runtime by varying the minimum support and the dataset size. It is expected that the modified U2P-miner algorithm will outperform the other existing algorithms.

The next experiment can be used to compare the performance of the modified U2P-miner algorithm with the existing U2P-miner algorithm. It is expected that the modified U2P-miner algorithm performs better than the existing algorithm. This is because the speed of the mining process is improved by deleting the unwanted base intervals (i.e., by applying the acceleration technique).

5. CONCLUSION

This paper proposes the modified U2P-Miner algorithm for mining frequent patterns from U2 data in an efficient manner. The implementation of the modified U2P-Miner algorithm is done in three steps. In the first step, all the transactions in the target database are compressed to construct a U2P-tree. This requires a single scan of the database. As new transactions arrive, decomposition of the intervals of each attribute should be performed if necessary. .In the second step, the acceleration technique is applied to improve the speed of mining by removing the unwanted base intervals. In the third step, frequent patterns are discovered by comparing the expected supports with the minimum support count given by the user. Since the modified U2P-Miner algorithm uses a tree structure, it provides better speedup and scalability than the existing algorithms.

6. REFERENCES

[1] Agrawal, R., & Srikant, R. 1994, September. Fast algorithms for mining association rules. In Proc. 20th Int. Conf. Very Large Data Bases, VLDB (Vol. 1215, pp. 487-499).

[2] Han, J., Pei, J., & Yin, Y. 2000, May. Mining frequent patterns without candidate generation. In ACM SIGMOD Record (Vol. 29, No. 2, pp. 1-12). ACM.

[3] Pei, J., Han, J., Lu, H., Nishio, S., Tang, S., & Yang, D. 2001. H-mine: Hyper-structure mining of frequent patterns in large databases. In Data Mining, 2001. ICDM 2001, Proceedings IEEE International Conference on (pp. 441-448). IEEE.

[4] Liu, Y. H. 2012. Mining frequent patterns from univariate uncertain data. Data & Knowledge Engineering, 71(1), 47-68.

[5] Chui, C. K., Kao, B., & Hung, E. 2007. Mining frequent itemsets from uncertain data. Advances in Knowledge Discovery and Data Mining, 47-58.

[6] Leung, C. S., Carmichael, C. L., & Hao, B. 2007, October. Efficient mining of frequent patterns from uncertain data. In Data Mining Workshops, 2007. ICDM Workshops 2007. Seventh IEEE International Conference on (pp. 489-494). IEEE.

[7] Leung, C. K. S., & Brajczuk, D. A. 2009, June. Efficient algorithms for mining constrained frequent patterns from uncertain data. In Proceedings of the 1st ACM SIGKDD Workshop on Knowledge Discovery from Uncertain Data (pp. 9-18). ACM.

[8] Aggarwal, C. C., Li, Y., Wang, J., & Wang, J. 2009, June. Frequent pattern mining with uncertain data. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 29-38). ACM.

[9] Zimányi, E., & Pirotte, A. 1996. Imperfect information in relational databases. Uncertainty Management in Information Systems, 35-88.

[10] Bernecker, T., Kriegel, H. P., Renz, M., Verhein, F., & Zuefle, A. 2009, June. Probabilistic frequent itemset mining in uncertain databases. In Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 119-128). ACM.

[11] Srikant, R., & Agrawal, R. 1996, June. Mining quantitative association rules in large relational tables. In ACM SIGMOD Record (Vol. 25, No. 2, pp. 1-12). ACM.

[12] http://taqm.epa.gov.tw/taqm/zh-tw/default.aspx.