International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2019 IJSRCSEIT | Volume 5 | Issue 5 | ISSN : 2456-3307
DOI : https://doi.org/10.32628/CSEIT195518

A Weighted Frequent Itemset Mining Algorithm for Intelligent Decision in Smart System

A. Kowsalya, S. Uma Parameswari, N. Kokila
Assistant Professor, K. S. Rangasamy College of Arts and Science, Tiruchengode, Namakkal, Tamil Nadu, India

ABSTRACT

Identifying the frequent item set is the challenging task in data mining as data is increased day by day in all fields. To analyze the accurate item set in that data like market basket is the key factor of improving the economical strategy of the marketing management. Frequent itemset mining, as an imperative of association rule examination, one of the mainly essential study fields in data mining. Weighted frequent itemset mining in vague databases equally the current prospect and significance of items into version in order to discover frequent itemsets of great importance to users. But many data are inconsistency because of the incomplete field in the collected data. This brings less stability in predicting the accurate information in the data which has the many fields. Many existing research have developed many technique or algorithm to bring the stable procedure to predict the data. But achieving the 100% accurate data from the collected dataset is still not completed. In this thesis, the proposed system will bring various parameters that will analyze dataset with Apriori and weighted Downwards Frequency Itemset Mining (WDFIM). In this analysis the minimum support, confidence level and time consumption are the parameters that analyzed where WDFIM is analyzing more accurate result when compared to Apiori algorithm.

Keywords : Data mining, WDFIM, Apiori Algorithm

I. INTRODUCTION

Data mining methods helps to extract accurate data from a large datasets. It is used to filter the needed data in the large set of data and picking out relevant information through certain convenient algorithms. Data mining tool become the important which help to gather large data in recent year. As more information is gathered, with the amount of data doubling every three year, data mining is becoming an ever more important tool to modify this data into information. Data mining is developed for the field in the medical community because it is supported by various technologies that are now sufficiently mature.

Data mining technique are the result of a time-consuming process of study and product development. In the data mining the type of task performed are Classification, Clustering, Regression, Dependence Modeling, Prediction Regression, and Association. The value of a previously defined goal attribute based on other attributes is often represented by IF-THEN rules are searched knowledge that is able to calculate. We can say the Dependence modeling as a generalization of classification. The goal of dependence modeling is to discover rules that are able to calculate the attribute value, from the values of calculated attributes. There are more goal attribute in dependence modeling. Clustering is the process of partitioning the item set in a set of significant sub-classes.

In the field of data mining, the Association rule mining is developed to spot the unfamiliar essentials in huge datasets and portrayal inferences on how a subset of items influences the incidence of another subset. Let T= {T1, T2, T3..........Tn} and S= {S1, S2, S3............. Sn} be a universe of Items is a set of transactions. Then the expression X => Y is an association rule where X and Y are itemsets and X ∩ Y=Φ. Here antecedent and consequent are X and Y called the rule respectively. In this rule, support is a set of transactions in set T which contain both X and Y and confidence is percentage of transactions in T containing X that contain Y. An association rule satisfies the user-set minimum confidence (minconf) and minimum support (minsup) such as confidence ≥ minconf and support ≥ minsup. An association rule is a frequent if its support ≥ minsup

II. LITERATURE SURVEY

R. Ishita and A. Rathod [1] proposed a decremental pruning (DP) approach for efficient mining of frequent itemsets from existential uncertain data. Experimental results showed that DP achieved significant candidate reduction and computational cost savings. Compared with LGS-Trimming, DP had the advantages of not requiring a trimming threshold and its performance was relatively stable over a wide range of low-probability-item population. In particular, it outperformed data trimming when the dataset contained few low-probability items. It argued that the Trimming approach and the DP approach were orthogonal to each other. It showed that the two approaches could be combined leading to a generally best overall performance.

L. Yue [2] research surveys that the broad areas of work in this rapidly expanding field. It presented the important data mining and management techniques in this field along with the key representational issues in uncertain data management. While the field will continue to expand over time, it is hoped that this survey will provide an understanding of the foundational issues and a good starting point to practitioners and researchers in focusing on the important and emerging issues in this field.

T. G. Green and V. Tannen [3] intend a novel frequent pattern tree (FP-tree) construction, that is an absolute pre_x- tree configuration for storing condensed, vital information on frequent patterns, and extend an early FP-tree- based mining process, FP-growth, set of frequent patterns by pattern fragment growth. Competence of mining is attained with three techniques: (1) a huge database is squashed into a extremely strong, lesser data structure, that avoids expensive, frequent database scans, (2) FP-tree-based mining adopts a prototype portion growth technique to avoid the expensive generation of a vast number of candidate sets, and (3) a partitioning-based, divide-and-conquer technique is used to decay the mining task into a set of minor tasks for mining conned patterns in restricted databases, which noticeably reduces the search space. It presentation revision shows that the FP-growth technique is competent and scalable for mining on long and short frequent patterns, and is about an order of extent faster than the Apriori algorithm and also closer than newly reported new frequent pattern mining methods.

C. C. Aggarwal and P. S. Yu [4] it proposed the correctness of the proposed algorithm by comparing this with previous state-of-the-art approaches that can mine exact results of uncertain frequent pattern mining. In addition, the results of performance analysis provided in the performance evaluation section showed that the proposed algorithm outperformed the competitors in various aspects such as runtime, memory usage, and scalability. The data structures and mining techniques devised in this research can also be applied in various pattern mining areas such as dynamic pattern mining on data streams and representative pattern mining because they are techniques for fundamental uncertain frequent pattern mining that can be utilized widely as basic frameworks.

D. Suciu [5] proposed the concept of TPC, which tightens the upper bound to the expected support of frequent patterns to be mined from uncertain data. The TPC is computed based on the information captured by the TPC-tree structure. Once such a TPC-tree structure is constructed by the TPC-growth algorithm (after two scans of the uncertain data), all potentially frequent patterns—containing *all* truly frequent patterns (i.e., *no* false negatives) but some false positives (i.e., any pattern *X* with *expSupCap*(*X*) ≥ *minsup* but with *expSup*(*X*) < *minsup*)—can then be mined from the TPC-tree structure. Providentially, the number of false positives is compact as the TPC helps constrict the upper bounds to anticipated supports. To absolute mining process, TPC-growth scans the hesitant data a third time to calculate the true probable carry and to abolish this miniature number of false positives. Estimation results illustrate, even though TPC-tree takes up more space than the obtainable PUF-growth algorithm; it pays off since the tension of these upper bounds twisted by the TPC led to a considerably low number (e.g., 1%) of false positives.

III. METHODOLOGY

3.1 OVERVIEW OF DATAMINING

Data Mining is the innovation of secreted data establish for the large quantities of data and can be viewed as a step in the knowledge discovery process. Data mining definite as a set of computer-assisted technique designed to mechanically mine huge volume of inter combined data for new, hidden or unexpected information, or interesting patterns. The main and uncomplicated systematic step in data mining is to clarify the data review its statistical attribute such as means and standard deviation, visually examining it by the means of charts and graphs, and viewed potentially important links among variables such as values that often occur together.

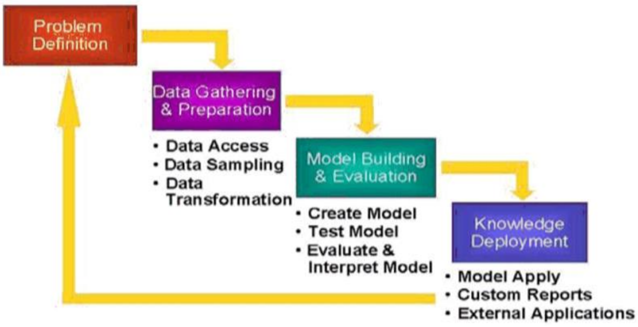


Figure 3.1 An overview of steps that compose KDD process

3.2 Apriori Algorithm

Apriori is intended to work on databases contains transactions. Additional algorithms are premeditated for verdict association rules in data having no transactions or having no timestamps. Every transaction is a set of items. Given a porch$C$, the Apriori algorithm analyzes the item sets that are subsets of at least C transactions in the database. Apriori uses a "bottom up" loom, where frequent subsets are extensive one item at a time (a step known as candidate generation), and groups of candidates are tested against the data. The algorithm terminates when no additional victorious extensions are found.

The pseudo code for the algorithm is given below for a transaction database$T$, and a support threshold ofϵ. Usual set theoretic notation is employed; though note that $T$is a multiset. C_k is the candidate set for level k. At every step, the algorithm is unspecified to create the candidate sets on the huge item sets of the previous level, heeding the descending conclusion lemma. $count[c]$ Accesses a pasture of the data structure that indicates candidate set c, which is originally unspecified to be zero. More information are lost below, frequently the vital element of the completion is the data structure used for storing the candidate sets, and counting their frequencies.

```
Apriori($T, \epsilon$)
    $L_1 \leftarrow \{\text{large } 1 - \text{itemsets}\}$
    $k \leftarrow 2$
    while $L_{k-1} \neq \emptyset$
        $C_k \leftarrow \{a \cup \{b\} \mid a \in L_{k-1} \wedge b \notin a\} - \{c \mid \{s \mid s \subseteq c \wedge |s| = k-1\} \nsubseteq L_{k-1}\}$
        for transactions $t \in T$
            $C_t \leftarrow \{c \mid c \in C_k \wedge c \subseteq t\}$
            for candidates $c \in C_t$
                $count[c] \leftarrow count[c] + 1$
        $L_k \leftarrow \{c \mid c \in C_k \wedge \ count[c] \geq \epsilon\}$
        $k \leftarrow k + 1$
    return $\bigcup_k L_k$
```

3.3 WDFIM Algorithm

The binary association rule has the form X ⇒ Y. A binary weighted association rule has a similar form.

Definition 1 weighted association rule has the form of X ⇒ Y, for the set of items T ={i_1, . . . , i_m} in a set of transactions D, and X ⊂ I, Y ⊂ I,and X ∩ Y = ø.

Definition 14 The (unnormalized) weighted support of the binary weighted rule X⇒ Y is the adjusting ratio of the support, or mathematically,

$$\text{wsupport}(X, Y) = \left(\sum_{i_j \in (X \cup Y)} w_j \right) \text{support}(X; Y) \qquad (4.1)$$

Where the weights of the items {i_1, . . .,i_n} are {w_1, . . ., w_n} respectively. In order to find the interesting rules, two thresholds, minimum weighted support (*wminsup*) and minimum confidence (*minconf*) must be specified.

Definition 2 An item set X is called a large weighted item set if the (unnormalized) weighted support of the item set X is greater than or equal to the weighted support.

$$\text{wsupport}(X) \geq \text{wminsup}$$

Definition 3 A binary weighted association rule X⇒ Y is called an interesting rule if the confidence of item set (X ∪ Y) is greater than or equal to a minimum confidence threshold, and (X ∪ Y) is a large weighted item sets.

Weights and Counts

In this problem, a balance between the two measures, which are weights and supports introduce. A computation of a new support, which is the multiplication of weight and support of an item set, is applied. There are different possible solutions to solve the conflict between the support and the weights.

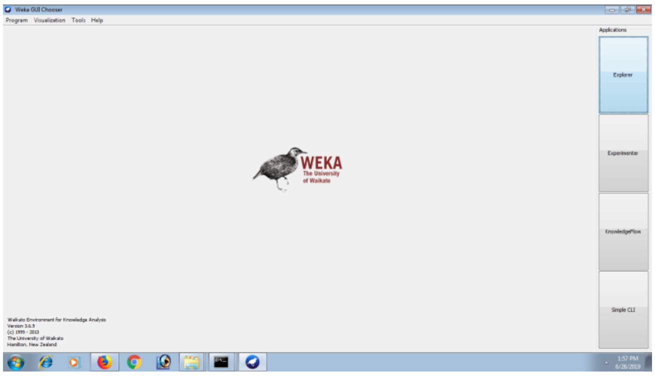

Weka GUI Chooser
Program Visualization Tools Help
Applications
Explorer
Experimenter
KnowledgeFlow
Simple CLI
WEKA
The University of Waikato
Waikato Environment for Knowledge Analysis

Figure 3.1.1 Main Page

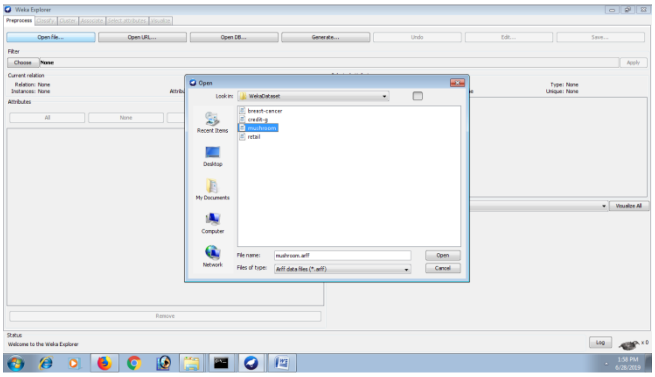


Figure 3.1.2 Dataset Uploading

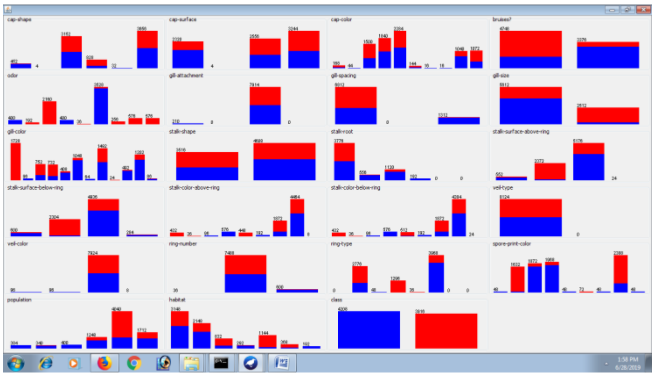
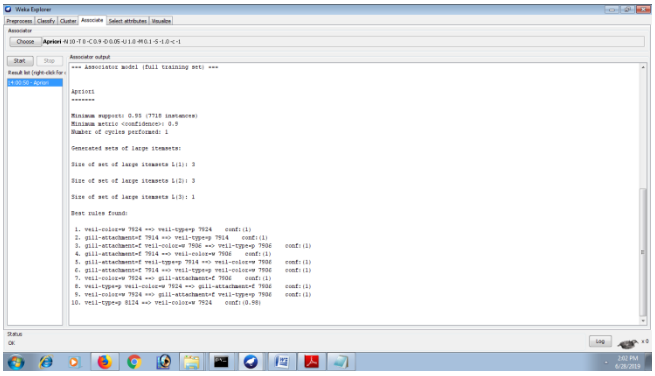


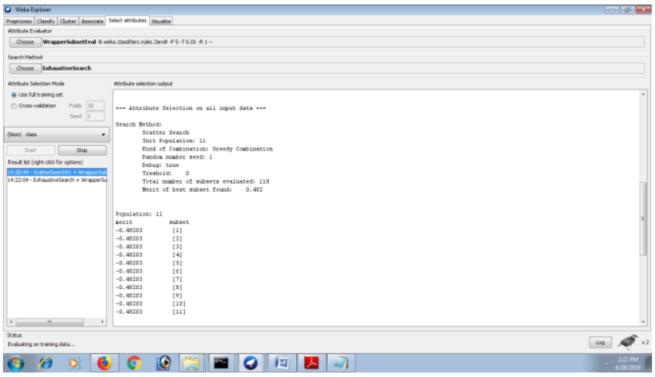
Figure 3.1.3 Mushroom dataset overall view



Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1

```
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.95 (7718 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 1

Generated sets of large itemsets:

Size of set of large itemsets L(1): 3

Size of set of large itemsets L(2): 3

Size of set of large itemsets L(3): 1

Best rules found:

 1. veil-color=w 7924 ==> veil-type=p 7924    conf:(1)
 2. gill-attachment=f 7914 ==> veil-type=p 7914    conf:(1)
 3. gill-attachment=f veil-color=w 7906 ==> veil-type=p 7906    conf:(1)
 4. gill-attachment=f 7914 ==> veil-color=w 7906    conf:(1)
 5. gill-attachment=f veil-type=p 7914 ==> veil-color=w 7906    conf:(1)
 6. gill-attachment=f 7914 ==> veil-type=p veil-color=w 7906    conf:(1)
 7. veil-color=w 7924 ==> gill-attachment=f 7906    conf:(1)
 8. veil-type=p veil-color=w 7924 ==> gill-attachment=f 7906    conf:(1)
 9. veil-color=w 7924 ==> gill-attachment=f veil-type=p 7906    conf:(1)
10. veil-type=p 8124 ==> veil-color=w 7924    conf:(0.98)
```

Figure 3.1.4 Mushroom Dataset in Apriori Algorithm



WrapperSubsetEval -B weka.classifiers.rules.ZeroR -F 5 -T 0.01 -R 1 --

ExhaustiveSearch

```
=== Attribute Selection on all input data ===

Search Method:
        Scatter Search
        Init Population: 11
        Kind of Combination: Greedy Combination
        Random number seed: 1
        Debug: true
        Treshold:    0
        Total number of subsets evaluated: 118
        Merit of best subset found:    0.482


Population: 11
merit          subset
-0.48203        [1]
-0.48203        [2]
-0.48203        [3]
-0.48203        [4]
-0.48203        [5]
-0.48203        [6]
-0.48203        [7]
-0.48203        [8]
-0.48203        [9]
-0.48203        [10]
-0.48203        [11]
```

Figure 3.1.5 Mushroom Dataset in WD-FIM Algorithm

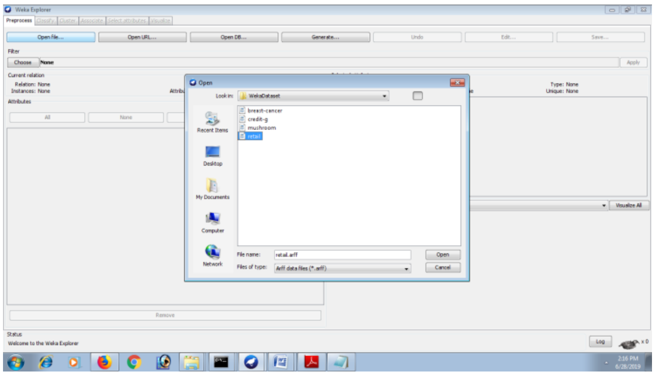


Figure 3.1.6 Retail Dataset Uploading

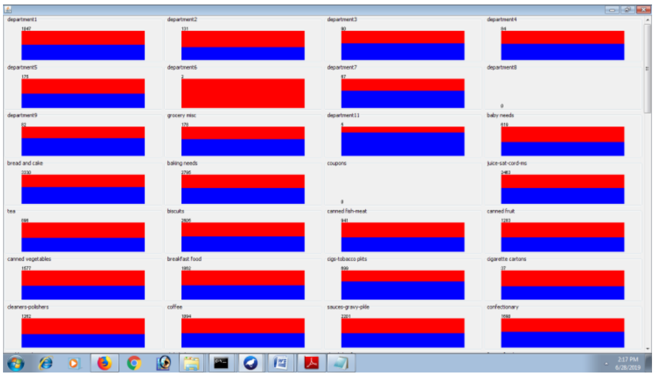
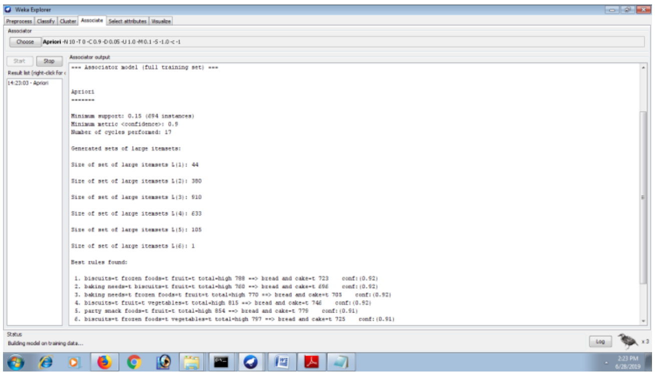


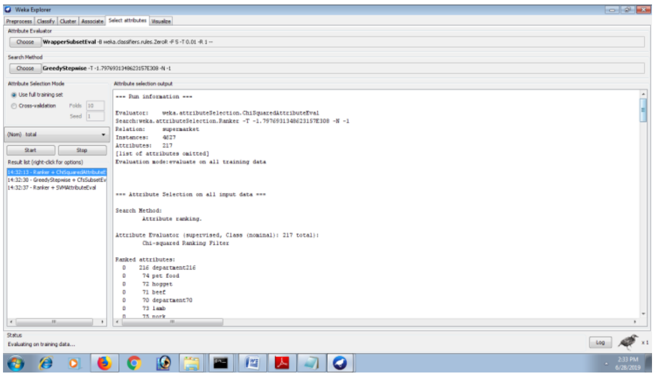
Figure 3.1.7 Retail dataset overall view



Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1

```
=== Associator model (full training set) ===


Apriori
=======

Minimum support: 0.15 (694 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 17

Generated sets of large itemsets:

Size of set of large itemsets L(1): 44

Size of set of large itemsets L(2): 380

Size of set of large itemsets L(3): 910

Size of set of large itemsets L(4): 633

Size of set of large itemsets L(5): 105

Size of set of large itemsets L(6): 1

Best rules found:

 1. biscuits=t frozen foods=t fruit=t total=high 788 ==> bread and cake=t 723    conf:(0.92)
 2. baking needs=t biscuits=t fruit=t total=high 760 ==> bread and cake=t 696    conf:(0.92)
 3. baking needs=t frozen foods=t fruit=t total=high 770 ==> bread and cake=t 705    conf:(0.92)
 4. biscuits=t fruit=t vegetables=t total=high 815 ==> bread and cake=t 746    conf:(0.92)
 5. party snack foods=t fruit=t total=high 854 ==> bread and cake=t 779    conf:(0.91)
 6. biscuits=t frozen foods=t vegetables=t total=high 797 ==> bread and cake=t 725    conf:(0.91)
```

Figure 3.1.8 Retail Dataset in Apriori Algorithm



WrapperSubsetEval -B weka.classifiers.rules.ZeroR -F 5 -T 0.01 -R 1 --

GreedyStepwise -T -1.7976931348623157E308 -N -1

```
=== Run information ===

Evaluator:    weka.attributeSelection.ChiSquaredAttributeEval
Search:weka.attributeSelection.Ranker -T -1.7976931348623157E308 -N -1
Relation:     supermarket
Instances:    4627
Attributes:   217
[list of attributes omitted]
Evaluation mode:evaluate on all training data



=== Attribute Selection on all input data ===

Search Method:
        Attribute ranking.

Attribute Evaluator (supervised, Class (nominal): 217 total):
        Chi-squared Ranking Filter

Ranked attributes:
  0    216 department216
  0     74 pet food
  0     72 hogget
  0     71 beef
  0     70 department70
  0     73 lamb
  0     75 pork
```

Figure 3.1.9 Retail Dataset in WD-FIM Algorithm

IV. RESULTS AND DISCUSSION

4.1 MINIMUM SUPPPORT

A mushroom, retail dataset has been used with 119 items each for analysis. A set of association rules that are obtained by applying Apriori and WDFIM algorithm. By analyzing the data, and giving different support and confidence values, it can obtain different number of rules. During analysis it found that WDFIM is much faster for large number of transactions as compare to Apriori. To generate frequent item sets it takes less time. It works on mushroom data which contains 8124 transactions. Entire results are gathered from Pentium Dual core processor with 1. 73GHz speed and 1 - GB RAM

TABLE 4.1.1 Dataset Details

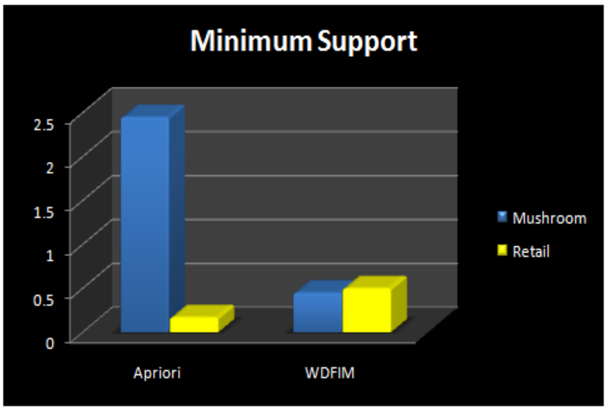
Data Sets	No of Attribute	Attribute Types	Instance	Classes

Mushroom	23	Nominal	8124	6
Retail	217	Nominal	4627	3

TABLE 4.1.2 Minimum Support in Dataset

Dataset	Apriori	WDFIM
	Minimum Support	
Mushroom	2.45	0.45
Retail	0.17	0.5

Chart 4.1.1 Minimum Support in Dataset



1) *4.2 CONFIDENCE*

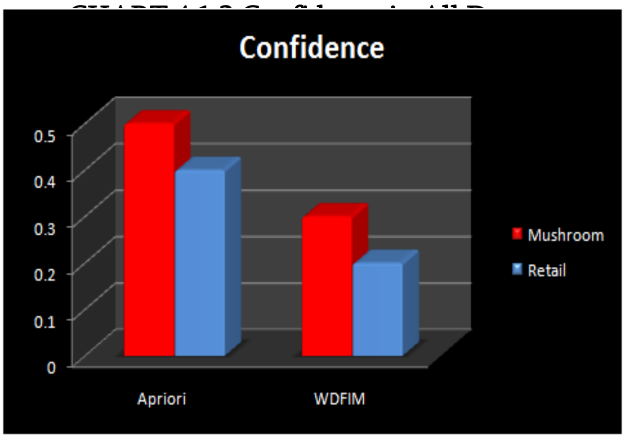
Confidence is an indication of how often the rule has been found to be true. The *confidence* value of a rule, X➔Y, with respect to a set of transactions T, is the proportion of the transactions that contains X which also contains Y.

Confidence is defined as:

$$\mathrm{conf}(X \Rightarrow Y) = \mathrm{supp}(X \cup Y)/\mathrm{supp}(X)$$

TABLE 4.1.3 Confidence in All Dataset

Dataset	Apriori	WDFIM
	Confidence	
Mushroom	0.5	0.3
Retail	0.4	0.2



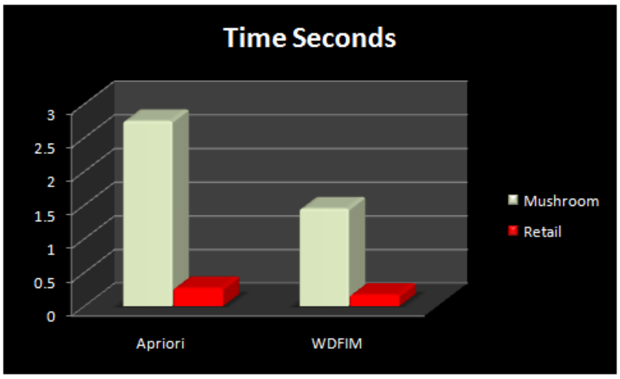
4.3 TIME CONSUMPTION

Experiments are performed on the entire datasets. Machine with configuration of windows 7 system and 2-GB of RAM is used. The results were compared to experiments with WEKA implementations of K-Means and GA the techniques run to ensure that the results.

Table 4.1.4 Execution Time Second

Dataset	Apriori Time Taken (in secs.)	WDFIM Time Taken (in secs.)
Mushroom	2.75	1.45
Retail	0.27	0.17

CHART 4.1.3 Execution Time in All Dataset



V. CONCLUSION

In order to realize intelligent decision making in smart systems, a weight judgment downward closure property based frequent itemset mining algorithm is proposed in this thesis to compare the dataset weighted frequent itemsets and improve the time efficiency. The weight judgment downward closure property for weighted frequent itemsets and property of weighted frequent subsets are introduced. Based on these two properties, the WD-FIM algorithm is described in detail. Moreover, the minimum support and time efficiency of WD-FIM algorithm are analyzed. Finally, the performance of the proposed WD-FIM algorithm is verified on both synthetic and real-life datasets that performs well.

VI. REFERENCES

[1]. R. Ishita and A. Rathod, “Frequent Itemset Mining in Data Mining: A Survey,” International Journal of Computer Applications, vol. 139, no. 9, pp.15-18, April 2016.

[2]. L. Yue, “Review of Algorithm for Mining Frequent Patterns,” International Journal of Computer Science and Network Security, vol. 15, no.6, pp.17-21, June 2015.

[3]. T. G. Green and V. Tannen, “Models for incomplete and probabilistic information,” Lecture Notes in Computer Science, vol. 29, no.1, pp.278-296, Oct. 2006.

[4]. C. C. Aggarwal and P. S. Yu, “A Survey of Uncertain Data Algorithms and Applications,” IEEE Transactions on Knowledge & Data Engineering, vol.21, no. 5, pp. 609-623, May, 2009.

[5]. D. Suciu, “Probabilistic databases,” Acm Sigact News,vol.39, no.2, pp.111-124, Feb. 2011.

Cite this article as :

A. Kowsalya, S. Uma Parameswari, N. Kokila, "A Weighted Frequent Itemset Mining Algorithm for Intelligent Decision in Smart System", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 5 Issue 5, pp. 249-255, September-October 2019. Available at doi : https://doi.org/10.32628/CSEIT195518
Journal URL : http://ijsrcseit.com/CSEIT195518