Graph Ontology Based IAFCM Model for Itemset Mining in Transactional Database: An Optimized Fuzzy Framework

DR P PERUMAL,[A,*], DR P AMUDHAVALLI[B], DR P SHERUBHA[C], AND DR P PREETHI[D]
a Department of CSE, Sri Ramakrishna Engineering College, Tamil Nadu, Coimbatore, India
bDepartment of Computer Applications, School of Computer, Information and Mathematical sciences, BS AbdurRahman crescent Institute of Science and technology, Tamil Nadu,Chennai, India
c Department of IT, Karpagam College of Engineering, Coimbatore, Tamil Nadu, India
d Department of CSE, Kongunadu College of Engineering and Technology, Trichy, Tamil Nadu, India
** corresponding author*

Abstract

HUIM (High Utility Itemset Mining) is one of the most analyzed data mining activities. Product suggestion, e-learning, bioinformatics, text mining, market basket analysis, and web click stream analysis are few of the areas where it can be used. Cost savings, greater competitive advantage, and increased revenue are the advantages gained by pattern analysis. However, because HUIM approaches do not examine the correlation of retrieved patterns, they may uncover false patterns. As a result, a number of technique for mining related HUIs have been presented. These algorithms still have issues with computational cost, both in conditions of period and memory usage. As a result, a method for mining weighted temporal patterns is proposed. The suggested method begins by preprocessing time series-based data into fuzzy itemsets. These are fed into the Improved Adaptive Fuzzy C Means (IAFCM) technique, which is a hybrid of the FCM clustering method and the Graph based Ant Colony Optimization (GACO) technique. The proposed IAFCM technique accomplishes two goals: IAFCM clustering and data reduction in FCM clusters, and ii) optimal itemet placement in clusters using GACO. Using GACO, the suggested technique produces high-quality clusters. On these clusters, weighted sequential pattern mining is used to find the most effective sequential patterns, which take into account knowledge of patterns with low frequency and high weight in a repository that is updated over time. The results of this method show that when compared to other traditional methodologies, the IAFCM with GACO improves execution time. Furthermore, it improves the data representation process by increasing accuracy while using less memory.

Keywords: *Utility Mining, Dimensionality reduction, High utility itemsets, Frequent Pattern, Graph, Fuzzy and Support count*.

Introduction

Data mining is the method of extract the information from huge datasets, and it has become one of the most significant study fields in recent years [1]. Temporal mining refers to data mining that is conducted on time-dependent datasets. This may be done with time series repositories, which identify regularities in the information based on period [2]. Each information element has a start and end time associated with it, which defines the timeframe for which the data item is valid [3]. To deposit temporal classification models of importance to the client, supporting and important decisions values are supplied [4]. Clustering is one of the data mining algorithms. It is a method in which related information items are grouped in a single collection [5]. That is, low intergroup similarity and high intragroup similarity are the goals of clustering. This is first accomplished using a temporal repository based on time. Information has become increasingly complicated in recent years. That is, each data item has a large number of properties, resulting in extremely complicated data. Temporal data mining is time-consuming and complicated. As a result, the amount of data examined for mining must be reduced. The most used clustering algorithm is the fuzzy clustering technique. The data is divided into groupings using fuzzy clustering. With relation to each cluster, member values are assigned to the data points. The fuzzy c means method is a popular cluster algorithm. Clustering using (FCM) is a well-known approach. It uses fuzzy separation to describe that a data item can be a member of any of the groups or clusters with membership degrees ranging from zero to one. However, it must be changed to allow for temporal extraction on information with several dimensions [6]. As a result, new hopeful approaches to grouping temporal data with image compression must be examined. A procedure in which a distinct point belongs to a cluster of position is known as dimensionality reduction. Only relevant qualities are examined for further analysis, whereas non-related attributes are ignored [5].

The clustering process may be optimised to produce high-quality clusters. The highest intragroup similarities and the least intergroup similarity are represented by high-quality clusters. In clustering, locating and evaluating data for similarities is also a significant process that might be improved. By improving the clustering algorithm, misclustering may be prevented. Time spent clustering, memory use, and computation complexity are all aspects that might be addressed for optimization. Clusters of high quality lead to efficient mining of common patterns and classification methods [7]. The process of association rule mining determines the frequently used itemsets that the user prefers [8]. The Apriori technique was utilised as the foundation for determining the frequent items. Because this approach needed a high number of scans of the original data set, various solutions arose to address the algorithm's shortcomings. Because a temporal dataset is dynamic in nature, frequent item mining on it is challenging [9]. As time passes, new information is added to the database, and previous frequently used itemsets must be revised to ensure the present stage of the sequential repository [10]. Furthermore, then the dataset contains multi-scale information, the link rule mining procedure develop into more difficult. Lower complexities, dimensionality reduction algorithms should be used, which resulted in data with fewer characteristics, making it acceptable for association rule on a temporal collection.

Furthermore, every time mining is conducted, new records added to the temporal database must be taken into account [11]. Only then can the frequent itemets that result be precise and efficient. As a result, selecting an effective mining method that takes into account all of the aforementioned characteristics is critical when mining frequent itemsets on a time series

 https://doi.org/10.37896/pd91.4/91422

database [12]. When contrast to frequent prototype mining, weighted temporal pattern extract the uncovers data and it provide more essential information.

The disadvantage of recurrent data set mining, it simply considers the probability of data occasion when detecting frequent patterns. Weighted pattern mining takes into use information of patterns that are both often and essential while discovering frequent patterns [13]. Data processing is conducted on the time series repository ideals to turn the mathematical information into fuzzy information appropriate for cluster in our suggested weighted temporal pattern mining technique. On the altered data, the IFCM technique, which is a hybrid of the FCM and GACO algorithms, is used. The FCM technique reduces dimensionality, and clustering is done using a mix of the FCM and GACO algorithms, evaluated to improved optimised clusters. After that, effective temporal patterns are mined using weighted temporal prototype extract on these clustering. The following are the remaining portions of the proposed work: The research effort relevant to the suggested approach is reviewed in Part II. The suggested technique is explained in Part III, the Experimental and result findings are shown in Part IV, and the study is concluded in Part V.

Related Works

This section examines the HUIM and CHUIM based literature using existing research works from reputed journals.

HUIM based review

In 2004, [14] described the trouble of HUI mining. They created the Mining technique to mine high-utility data sets. Mining is a rough move toward that may not be able to extract all HUIs. As a result, [15] devised a Two-Phase technique to identify the whole collection of HUIs. A fresh upper bound feature called TWU (Transaction Weighted Utilization) has indeed been developed to minimise the explore space in the Two-stage method. The HUIs are mined in two steps via the Two-stage method. It creates aspirant HUIs with a TWU greater than minimal effectiveness requirement in the first stage. The effectiveness of each candidate is then calculated in the second phase by examine the repository again to constrain the HUIs. The Two-stage approach, on the other hand, has issues with time and memory optimization. The fundamental basis is because in the initial phase, a huge number of candidates may be created. HUPtree, a novel approach for extract HUIs support on tree topology, is proposed in [16]. It combines the Two-stage process with the FP-tree idea to create a compression node structure that may be used to exploit the TWU feature. This method extracts HUIs in 3 stage: (1) creates the tree, (2) produces candidate model, and (3) extracts HUIs from the list out of aspirant. The amount of conditionally trees built during the mining process, as well as the traversing cost of each conditioned tree, determine the algorithm's mining performance. As a result of the development of a large no of conditioned node and potential prototype, this approach consumes a lot of instance and remembrance [17].

Despite its many uses, High Utility Pattern Mining has significant drawbacks. As per result, lots of modifications of HUPM have emerge in the literature, such as Incremental Utility Mining [18], which goal to mining HUPs from active datasets, On-Shelf High Utility Pattern Mining [19], which takes into account the projection life of data, and Concise Representations of High Utility Patterns (e.g., Maximal Itemsets [20] and Closed High Utility Itemsets [21]), which needs to be extracted a small list of meaningful

Correlated HUIM based review (CHUIM)

All-confidence, Bond, any-confidence [22], consistency [23], [24] are some of the correlation metrics offered in the information extract literature for connection analysis. Because standard HUIM methods do not take into account the connection of the derived prototype, they may provide uninteresting or deceptive prototype. They frequently find high-utility itemsets in this instance, however these itemsets may comprise weakly connected items.

Gan et al. [25] suggested two methods that incorporate correlation and utility indicators to identify associated buying patterns. CHUIM is the first algorithm [25], whereas Correlated High Utility Pattern Miner [26] is the latter (CUPM). Both methods analyse the interestingness of the intended patterns using the Kulczynsky (abbreviated as Kulc) metric [18] in combination with the effectiveness scale. Fast Associated High Utility Itemset Miner (FCHM) method was developed by [27] for incorporating the notion of correlations in HUIM in order to find valuable prototype that are strongly connected. FCHMbond and FCHMall-confidence are two variations of the method that are based on all-confidence and bond actions that are previously used to measure frequently associated features.

A variety of techniques had been presented to extract CHUI by employing together usefulness and correlated metrics in categorize to extracting more interesting prototype and prevent mislead prototype emerging from classic technique of HUIs. [28] Presented the HUIPM method, which has a significant occurrence attraction for generating unique patterns in high utility item sets with relevant item relationships. The HUIPM method established a novel tree diagram called UTFA as an efficient information for storing the necessary dataset for pattern extraction. The HUIPM technique repeatedly build a number of conditional trees to produce candidates and then derives intriguing patterns, while a novel pruning feature called KWU has been suggested in this approach to minimise the research scope. This operation takes a long time. Lin et al. [29] improved HUIPM by developing a new approach called quick technique for extracting discriminative high utility patterns. Two information design have been suggested in the FDHUP method to hold necessary information for mined the DHUP efficiently: the Element Information table (EI) and the FU table. To decrease the search space, a new pruning attribute based on the outline of affinitive effectiveness and the residual affinitive effectiveness has been implemented.

Vo et al. [30] proposed the CHUIMiner technique for effectively extracting Correlated High Utility Frequent items. To decrease the database size, the CHUI-Miner uses the repository projection Complexity 3 technique. It also presents a novel notion known as the prefix utility of predicted transactions for calculating the utility of data items directly. The CHUIM methods and their properties are summarised in Table 2.

Table 1. Summarization of CHUIM algorithms with attributes

Ref. No.	Phase count	Algorithm	Metrics	Data Structures	Property of Pruning
26	One	CUPM	Bond & Utility	Utility list	(1) TWU (2) Sum of iutil and rutil
28	Two	HUIPM	FA & Utility	UTFA	KWU
25		CHUIM		Projected DB	TWU
39	One	FDHUP	Kulczynsky & Utility	EI table with FU table	(1) TWU (2) Sum of AU and RAU
27		FCHMbond	Bond & Utility	Utility list	(1) TWU (2) Sum of iutil and rutil (3) Antimonotonicity of bond
30		CHUI-Miner		Prefix utility along with projected DB	(1) TWU (2) Sum of iutil and rutil

Proposed Methodology

The proposed section is described with the steps for anticipated weighted temporal pattern mining method are as follows.

- Time series analysis preprocessing: As stated in section 3.3 of this chapter, time series data are preprocessed in order to turn them into fuzzy itemsets that facilitate fuzzy clustering.
- The weather forecast information is being investigated for testing purposes. From 2005 to 2015, this collection contains 10 years of weather forecast data. It is the weather forecast prediction data for Chennai, which includes eight parameters. The data were obtained

 https://doi.org/10.37896/pd91.4/91422

ISSN: 0369-8963

from Chennai, India's India Meteorological Department. Preprocessing is done in a similar way for each characteristic independently. Tables 1a and 1b demonstrate a sample weather prediction dataset with numeric values and fuzzy data, respectively.

- Clustering with IAFCM: Clustering is done with FCM on the itemsets. The information in these clusters is reduced in size. CS performs the optimal placement of the itemsets in the clusters, producing in optimal groupings.

- Weighted pattern recognition mining: This method uses a tree-based weighted pattern mining method to identify effective temporal patterns from optimised groupings by taking into account knowledge about structures with lower frequencies but high probability (weight) in a dataset that is updated over time.

Table 2 (a). Weather forecast dataset: numerical data

District: Chennai	Rain-fall (mm)	Max Tempera-ture (deg C)	Min Tempera-ture (deg C)	Total cloud cover (octa)	Max Relative Humidity (%)	Min Relative Humidity (%)	Wind speed (kmph)	Wind direc-tion (deg)
5Sep'15	3	35	26	5	86	63	4	250
6Sep'15	2	35	26	5	88	65	4	250
7Sep'15	2	35	26	5	88	65	4	250
8Sep'15	2	35	26	5	86	67	4	250
9Sep'15	2	35	26	5	88	67	4	250

Table 2 (b). Fuzzy data for the weather forecast dataset

District: Chennai	Rain-fall (mm)	Max Tempera-ture (deg C)	Min Tempera-ture (deg C)	Total cloud cover (octa)	Max Relative Humidity (%)	Min Relative Humidity (%)	Wind speed (kmph)	Wind direc-tion (deg)
5Sep'15	VL	VL	VL	VL	L	VL	VL	VH
6Sep'15	VL	VL	VL	VL	L	L	VL	VH
7Sep'15	VL	VL	VL	VL	L	L	VL	VH
8Sep'15	VL	VL	VL	VL	L	L	VL	VH
9Sep'15	VL	VL	VL	VL	L	L	VL	VH

 https://doi.org/10.37896/pd91.4/91422

ISSN: 0369-8963

Clustering using IAFCM

The numerical data from the entire data is turned into itemsets in the proposed approach. The consequence of data preprocessing is these itemsets. These are sent into the IAFCM approach, which combines FCM and CS techniques. FCM performs clustering and reduces the amount of the data in these clusters dependent on the qualities in the clusters in each time zone. CS optimises these groupings carrying data of lower size. As a result, the IAFCM approach generates high-quality clusters.

Reduction of data size in FCM clusters

The FCM algorithm creates clustering. FCM is the same as K-means in that it permits a single data item to belonging to many clusters. Depending on how closely the data resembles a certain class, a class membership value is assigned to it. Figure 1 shows the technique for lowering the data size in FCM clusters. Data points indicate the value of a property in this process, whereas dataset contains the itemsets. The detailed process is depicted below.

- The information is organized into time-based segments.
- A specified number of clusters is deployed to each block using FCM.
- The FCM algorithm selects the relevant information for each cluster.
- For each cluster, a centroid value was determined, and these centroid values serve as the clusters' reference points.
- The time property is used to choose a projecting cluster that only comprises a subclass of the centroids.
- Apart from time, the important features are those that are appropriate to the specific group, while the extra are unnecessary.
- The calculation is based on a 3D matrices in which instance are shared by both the relevant qualities and the centroids. As seen in Figure 1, characteristics represent columns and centroids represent rows in the matrix.
- The range of the attribute supplied by Equation 1 is determined as the maximum and minimum value for each feature represented by each column in Figure 1.

$$range_c = max^a_c - min^a_c \qquad (1)$$

where, $range_c$ - reflects the range of data for a certain property 'a' in column 'c'.

max^a_c - in column 'c', reflects the maximum value of attribute 'a'

min^a_c - indicates the 'a' attribute's minimum value in column 'c'

 https://doi.org/10.37896/pd91.4/91422

ISSN: 0369-8963

The relevant information items for each characteristic are the values that fall inside the range for that feature. For the time period under consideration, the other data sets are deemed irrelevant.

- Outliers are identified and deleted when data points are inappropriate. These will not be processed any further.

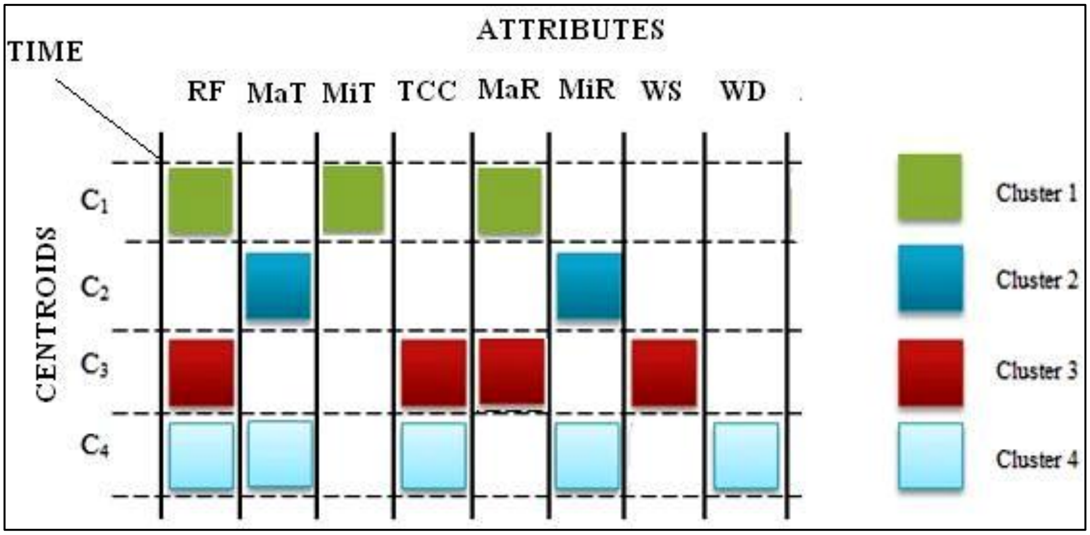


Figure 1. Data size reduction in FCM

The properties are represented by the columns. RF- Rainfall, MaT- Max Temperature, MiT- Min Temperature, TCC- Total cloud cover, MaR- Max Relative Humidity, MiR- Min Relative Humidity, WS- Wind Speed, and WD- Wind Direction are all parameters from the weather prediction dataset shown in Table 1a. The centroids of the fixed number of clusters indicated by C1, C2, C3, and C4 are defined by the rows. Clusters 1, 2, 3, and 4 are predicted clusters that do not have all properties. These are the characteristics that will be linked with each planned cluster. Values for the specified property that fall outside this range are deleted as outliers for the time period. These aren't taken into account for subsequent processing. These clusters with fewer features and results that fall within the range are sent into CS for optimisation. As the number of characteristics decreases, the attribute set's dimensionality decreases. In chapter 3, section 3.4.1, the process for cluster optimization using CS is given. As a result, optimal clusters are created utilising the IAFCM approach.

These optimised clusters are sent into a weighted sequential prototype extraction method, which produces useful sequential prototype. Each itemset (transaction) existing in the resultant clustered information on which weighted temporal pattern mining is to be done is allocated a transaction id following the clustering procedure. The following section goes with weighted temporal mining techniques.

https://doi.org/10.37896/pd91.4/91422

Weighted Temporal Pattern Mining

The IAFCM approach considers the itemsets in the clusters to be activities, and they are given a transaction id. A tree-based weighted frequent prototype extraction technique based on instance is used to execute weighted numerous prototype extraction. Every transaction item is given a weight that indicates its relevance. The weight of an dataset is specified as a non-negative real integer between 0.2 and 0.6 that is assigned to represent the relevance of each item in the data structure.

Each item is given a weight to represent its relevance in the transaction database. Within the weight category provided by, weight (W) is assigned. $W_{min} \leq W \leq W_{max}$, where W_{min} = 0.2 and W_{max}. = 0.6. An item's weight is allocated in proportion to the minimal support. If an itemset's supporting is lesser than the minimal supporting and its weight is likewise lesser than the lowest confidence, it is considered unusable. This aids in the pruning of lighter objects (Yun & Leggett 2005).

For a set of data $J = \{j_1, j_2 \ldots j_n\}$, weight of a prototype $P\{y_1, y_2 \ldots y_m\}$

is given by Equation (5.2) as

$$Weight\ (P) = \sum_{q=1}^{length(P)} Weight(yq)/length(P) \qquad (2)$$

The weighted supporting of a patterns is considered as the result of develop the pattern's supporting by the pattern's weight, as shown in Equation (3).

$$W_{support}(P) = Weight(P)\ x\ Support\ (P) \qquad (3)$$

If the weighted support of a structure is more than or equal to the minimal support threshold, it is called a weighted frequent structure.

The weighted frequent pattern mining technique is used only to new data that is dynamically introduced to the time series dataset. This is accomplished using the Continuous and Interactively Weighted Frequent Pattern Mining approach and the Incremental Weighted Frequent Pattern Tree with Weight Ascending order tree structure (Ahmed et al. 2008a). It uses the data structures and mining results achieved before to minimise wasteful computations on database updates or changes to the mining threshold. A single database scan is adequate for managing data that is updated frequently on a temporal basis without the need for recurring labour. The things are ordered in ascending sequence of weight, with the heaviest dataset at the bottom. This aids in the efficient creation of potential temporal information.

Allow operations with distinct transaction ids (T1 through T6) to be included in the optimal clusters for the weather forecast dataset. db_1^+ and db_2^+ denotes the position of recently

 https://doi.org/10.37896/pd91.4/91422

additional connections with the original repository (T7, T8, & T9, T10, respectively) depending on time, as illustrated in Table 3a. The tree-based weighted frequent pattern mining approach is divided into two parts: tree creation and searching using GACO as in figure 2. The following is the technique for tree building and mining.

Procedure 1: Tree construction
1. The original database with increments db_1^+ and db_2^+ (*incremental database*), the *updated database* and the *weight table* containing the weight of items, are considered for tree construction and are shown in Table 3a, b and c. 2. In the suggested tree-based weighted frequent prototype extraction method, the items are organised in a header table. 3. Each entry's frequency, item id and weight are reserved in the header table. 4. The tree is constructing for the original database in Table 3a as the first phase. 5. Each transaction is read one by one, and the contents are sorted in the order of the header table (ascending order of weight). For example, the first transaction T1 has the entries RF, MaT, MiT, TCC, WS, and WD. After sorting, the elements are in the following order: MiT, TCC, WD, WS, MaT, and RF. This is placed within the tree. 6. As shown in Tree/table h, all of the connections from the original repository are placed into the node in the same way. 7. As illustrated in Table 3a, db_1^+ and db_2^+ are two increases to the original database. 8. Tree/table i and Tree/table j, correspondingly, illustrate the tree after inserting db_1^+ and db_2^+. 9. As indicated in Table 3b, the original repository is changed by removing T4 and T7 and altering T5. 10. In the tree depicted in Tree/table h, the path 'MiT TCC MaT RF' represents T4. 11. . In the new database, T4 is deleted by lowering the occurrence of all tree in that way by one. As a result of this, the ranges of nodes *RF* and *MaT* become 0 in that path and these are deleted. *T7* is also removed in the same manner. Modification is done to *T5* by return item *WD* by dataset *WS* in the way '*TCC WD MaT RF*'.

Table (with tree) 3. Weight based updation in transaction database

(a) Incremental DB

TID	Itemset
T1	RF, MaT, MiT, TCC, WS, WD
T2	RF, MaR, MiR
T3	MaT, MaR, MiR, WS, WD
T4	RF, MaT, MiT, TCC

ISSN: 0369-8963

T5	RF, MaT, TCC, WD
T6	RF, MaT, TCC, MaR
T7	RF, MaT, MiT
T8	RF, MaT, MiT, TCC, WS
T9	RF, MaT, WS
T10	RF, MiT

(b) Updated DB

TID	Itemset
T1	RF, MaT, MiT, TCC, WS, WD
T2	RF, MaR, MiR
T3	MaT, MaR, MiR, WS, WD
T5	RF, MaT, TCC, WS
T6	RF, MaT, TCC, MaR
T8	RF, MaT, MiT, TCC, WS
T9	RF, MaT, WS
T10	RF, MiT

(c) Weight table

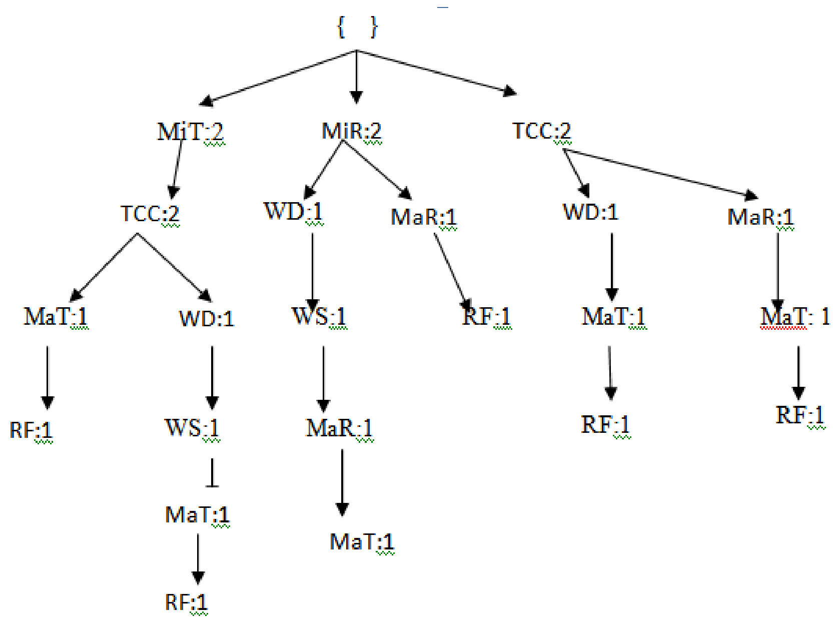
Items	Weight(W)
RF	0.6
MaT	0.5
MiT	0.2
TCC	0.35
MaR	0.5
MiR	0.3
WS	0.4
WD	0.38

 https://doi.org/10.37896/pd91.4/91422

(d) Inserting the transactions of the original database (Tl - T6)

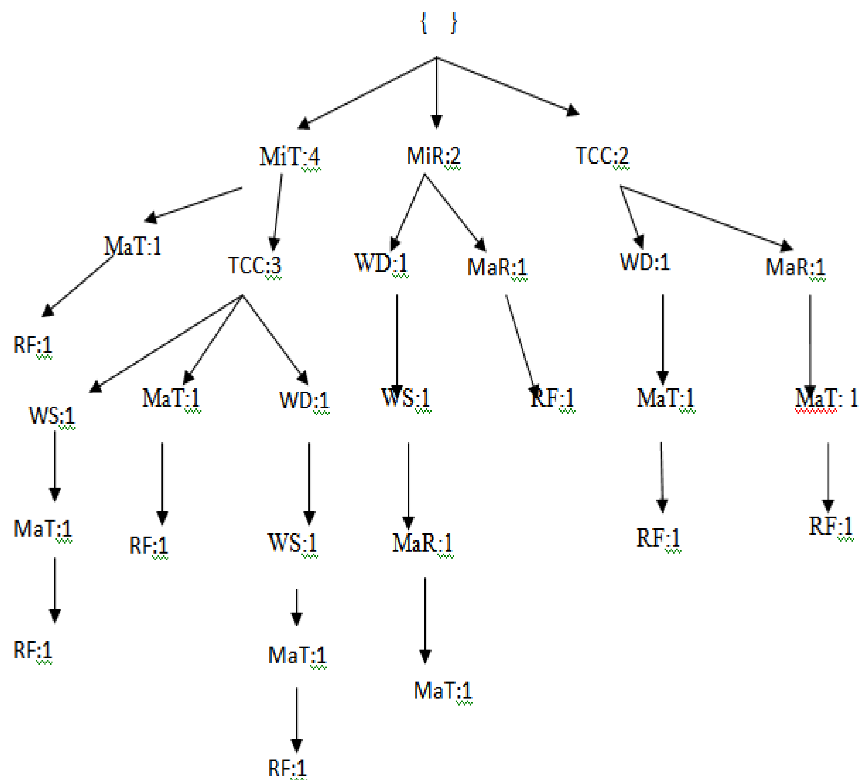
Header Table

Item	W eight	Frequency
MiT	0.2	2
MiR	0.3	2
TCC	0.35	4
WD	0.38	3
WS	0.4	2
MaR	0.5	3
MaT	0.5	5
RF	0.6	5



(e) Inserting of db_1^+ (T7 & T8)

Item	W eight	Frequency
MiT	0.2	4
MiR	0.3	2
TCC	0.35	5
WD	0.38	3
WS	0.4	3
MaR	0.5	3
MaT	0.5	7
RF	0.6	7

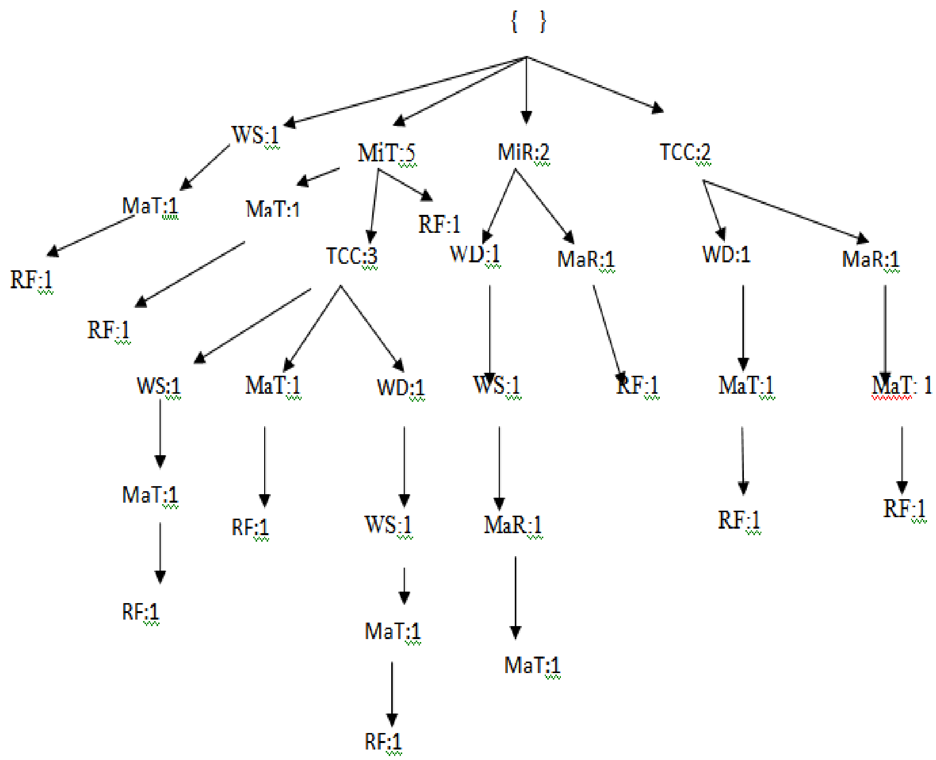


 https://doi.org/10.37896/pd91.4/91422

ISSN: 0369-8963

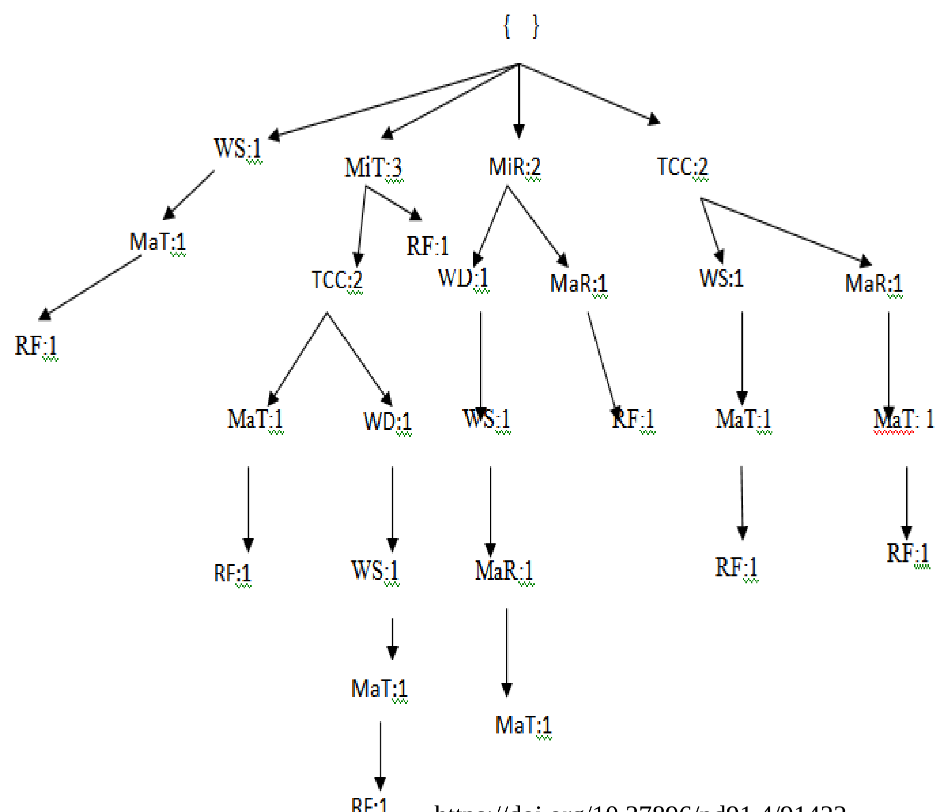
(f)Inserting of db_2^+ (T9 & T10)

Item	Weight	Frequency
MiT	0.2	5
MiR	0.3	2
TCC	0.35	5
WD	0.38	3
WS	0.4	4
MaR	0.5	3
MaT	0.5	8
RF	0.6	9



(g) Updated tree

Item	W eight	Frequency
MiT	0.2	3
MiR	0.3	2
TCC	0.35	4
WD	0.38	2
WS	0.4	5
MaR	0.5	3
MaT	0.5	6
RF	0.6	7



https://doi.org/10.37896/pd91.4/91422

ISSN: 0369-8963

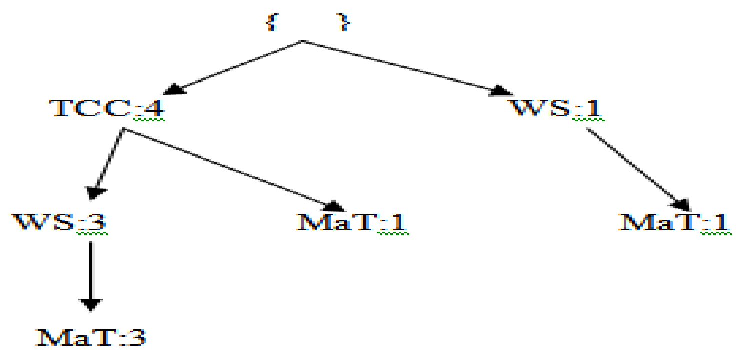
Procedure 2: Mining

1. The revised database (Table 3b), the tree built for it (Tree/table 3d), and the weight table (Table 3c) are used to execute weighted frequent temporal pattern mining.
2. For mining weighted frequent item sets, local maximum weight and global maximum weight are applied.
3. It refers to the total weight of all entries in the repository. For illustration, the item RF in Table 3c has a maximum weight of 0.6 and hence reflects the worldwide maximum weight.
4. The local maximum weight is required for a specific mining activity. Weighted frequent patterns preceded with item RF, for example, are mined first. Following that, those prefixed with item MaT are mined, and item RF will not appear in any conditional tree, as per the FP-Growth tree's premise. In this scenario, the item MaT is deemed to have the highest weight. Local maximum weight is the term for this.
5. Mining is done from the bottom up since the tree is organised in ascending order of item values.
6. Look at Table 3b for the modified database, Tree/table 3d for the tree, and Table 3c for the weight table. Allow for a minimum support threshold of 2. 2. Take, for example, the item RF, which has a global maximum weight of 0.6.
7. The occurrence of each of the other elements is multiplied by the global maximum weight 0.6, yielding MiT: 1.8 MiR: 1.2 TCC: 2.4 WD: 1.2 WS: 3.0 MaR: 1.8 MaT: 3.6 RF: 4.2 > as a weighted frequencies list.
8. TCC, WS, MaT, and RF are the possible items.
9. The conditioned tree for the bottommost item RF is retrieved. This is constructed by examining all branches prefixed with the item RF and eliminating nodes objects or devices that aren't candidate patterns with item RF.
10. By multiplying the frequencies of the other items with 0.6, the local maximum weight for item RF = 0.6 and its weighted frequency list are derived (local maximum weight). The weighted occurrence of the dataset RF, for example, is TCC: 2.4 WS: 3.0 MaT: 3.6 >, and the possible patterns are RF TCC, RF MaT, and RF.
11. A similar procedure is used to create the conditional tree for pattern RF MaT, with the candidate patterns RF MaT TCC and RF MaT WS.
12. The local weight limit for the item MaT is 0.4, and the weighted occurrence record is TCC: 2.0 WS: 2.5 >.MaT WS is formed as the candidate pattern from the conditional tree.
13. The actual weights and normalized frequency of all potential patterns are tested. The weighted frequent patterns that arise are as follow.

https://doi.org/10.37896/pd91.4/91422

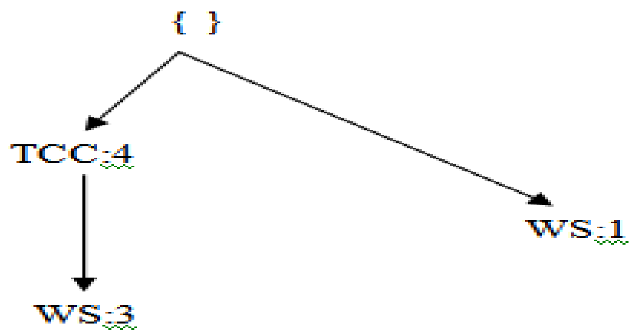
(h) Conditional tree for item RF

Item	Weight (W)	Frequency (F)
TCC	0.35	4
WS	0.4	4
MaT	0.5	5



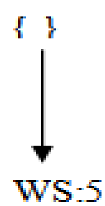
(i) Conditional tree for itemset 'RF MaT'

Item	Weight (W)	Frequency (F)
TCC	0.35	4
WS	0.4	4



(j) Conditional tree for item MaT

Item	Weight (W)	Frequency (F)
WS	0.4	5



Rainfall (RF), Max Temperature (MaT), and Wind Speed (WS) are the characteristics that determine the weather for the time period under consideration, according to the mining findings. It also depicts the link between these variables. As a consequence, the suggested IAFCM approach reduces data size and optimises clustering, resulting in high-quality clusters. Over these clusters, the tree-based weighted frequent pattern mining approach is used, resulting in effective temporal patterns.

 https://doi.org/10.37896/pd91.4/91422

ISSN: 0369-8963

GACO description based on graph ontology

ACO is included in Swarm Intelligence [31]. Pheromones are used by ants to find the quickest path between a food source and their colony in the wild. Consider the graph G = (N, A) [23], where N represents the set of n = |N| nodes and A represents a set of unsupervised arcs linking them. The sending and receiving nodes are the collection of places between which we search the lowest-cost path. As with other minimal level route issues, cost is computed. We may refer to the source as the "nest" and the food supply as the "end" in allusion to actual ants' shortest-path-finding behaviour.”

Sub-graph and graph isomorphisms (formal comparison) are all described. G = (V, E), where V denotes a collection of nodes and E denotes the set of edges connecting the nodes. The graph-based method is simple to expand. The graph is quantifiable and represents transaction itemsets. A graph-based technique is likely to capture common itemsets from a huge database of client transactions. By discovering suitable pathways across graphs, computational issues can be minimised (ACS). The fundamental benefit of graphbased approaches is that they preserve the original document's basic structural integrity.

The following phases are common to both ACO and the graph-based method using association rules:

1. Constructing fully linked graphs from two frequently occurring item sets
2. ACO-ARM parameter settings are initialised.
3. Ant colony establishment
4. Pheromone initialization
5. Pheromones matrices construction
6. In the linked graph, compute heuristic dataset for all vertices and their edges.
7. The heuristic matrix's construction
8. Pheromone and ant evaluation
9. Updating pheromone amounts and graph showing evaporation by tour value
10. Consistent pattern recognition
11. Establishment of a new ant colony
12. Find the best last frequent pattern

 https://doi.org/10.37896/pd91.4/91422

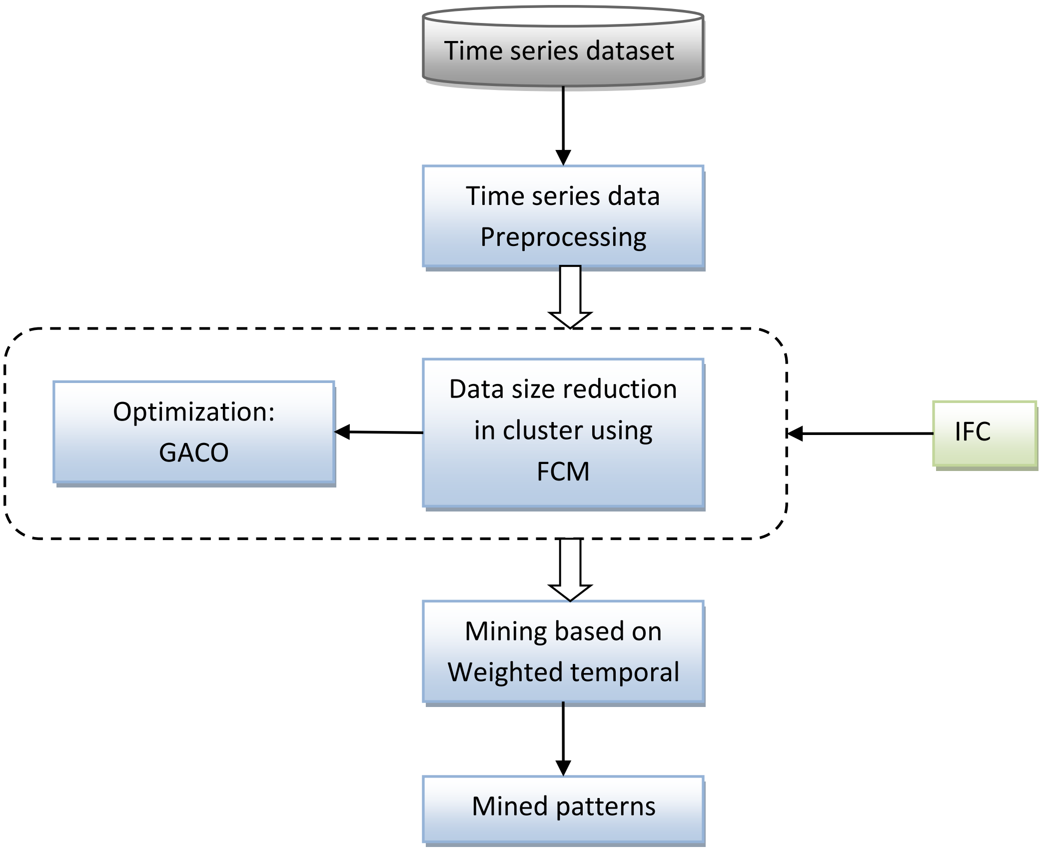


Figure 2. Generic view of the proposed model

Many important aspects are recognised while applying the ACO is an extraction association rules, such as collecting heuristic data for all vertices of the diagram, determining the weight for each edge of vertices, and initializing the cardinality variables [32]. Initializing the number of ants and the no. of repetitions yields the preliminary number of ants in a colony (maximum cycles). These settings are chosen for the network within the greatest no. of nodes. As a outcome, the number of ants and nodes will be equal [33-35]. At each vertex, the ants transit the building graph and make a statistical judgement. The update of pheromones will proceed till the termination criteria are reached. Then common patterns are discovered as depicted in figure 3, where $\eta_{ij}(t)$ is connected graph vertices (i) and edge (j) value computation and q is derived fuzzy support then q_0 is taken as 0.9.

 https://doi.org/10.37896/pd91.4/91422

ISSN: 0369-8963

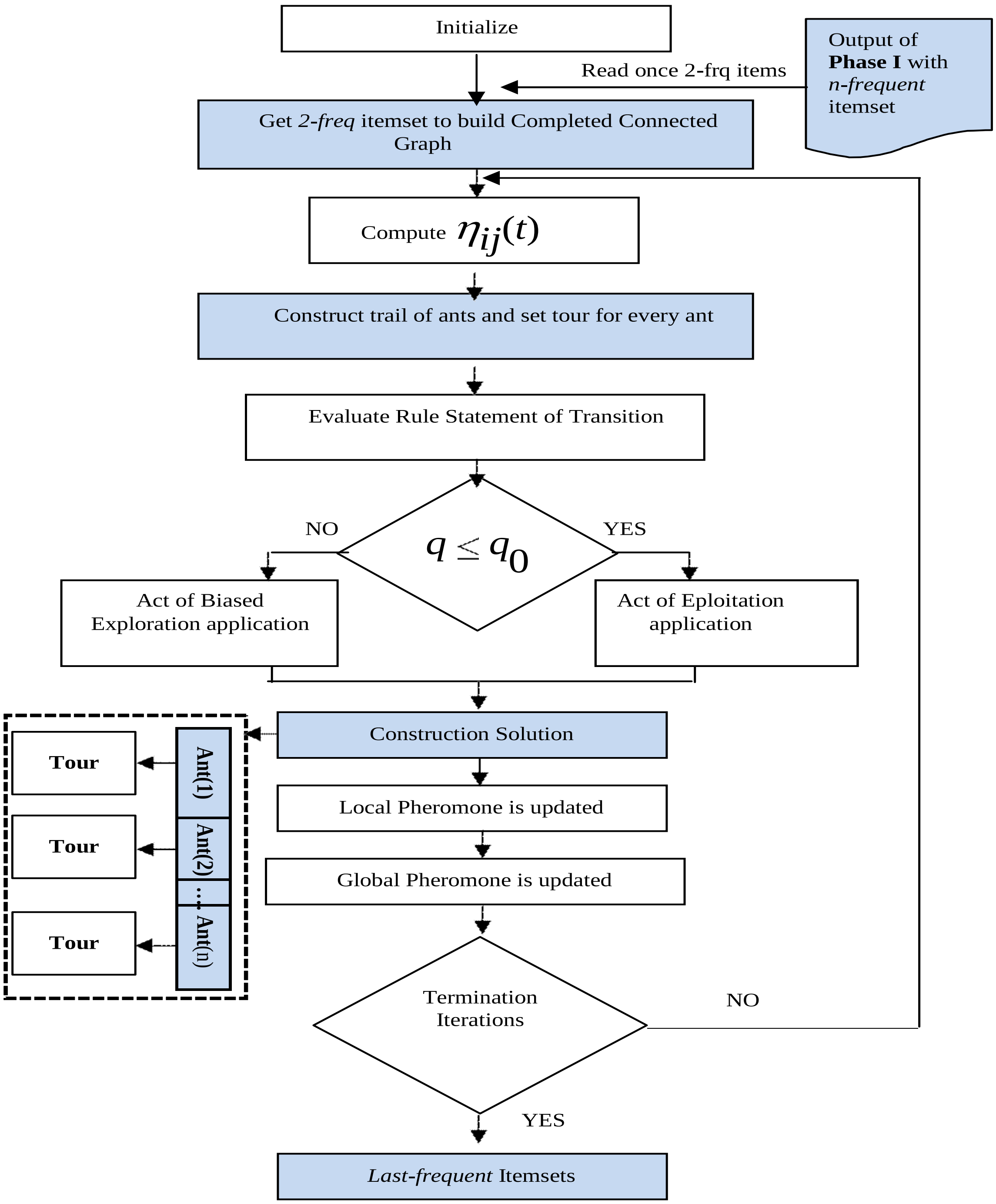


Figure 3: Flow of GACO for frequent itemsets

 https://doi.org/10.37896/pd91.4/91422

ISSN: 0369-8963

Experimental Results

This Part describes the experiment's design for evaluating performance. Research were conceded out on a processor that was running 64-bit Windows 10 Pro and has an CPU @ 2.60 GHz (4 CPUs), 2.8 GHz, Intel CoreTM D7-6600U 8 GB, of RAM. The proposed IAFCM-GACO method was evaluated to the CHUIM and CHUI-Miner method in set of time, accuracy and remembrance utilization.

For evaluating the accuracy, memory handling and implementation time, the experiments are carried out on a dataset namely weather forecast which is taken from UCI Machine learning DB. For performing the experiment, netbeans tool is used. First, collected the datasets and converted them into the binary form by using the weka tool. In the adult dataset 25 attributes and 126 instances are considered. Hybrid data distribution is used. First, two horizontal subsets are made from the original dataset and then each horizontal subset has been partitioned into three vertical subsets. After the partitioning, rules are combined together to form global rules. Then accuracy, storage complexity (memory usage) and execution time has been computed. Thus, proposed research shows better accuracy and has small memory usage and also it has small execution time. Figure 4 depicts the accuracy where the proposed method shows better trade off than CHUIM and CHUIM-Miner approaches because the maximum fuzzy supported values set at 0.9 in proposed. The graph-based technique with ACO relies on the preexisting ACO technique to speed up the retrieval of the most recent frequent dataset. From a repository of client connections, the graph technique can capture frequent data set. The'sparse dataset' is converted to a Boolean matrix using an appropriate information representation strategy. The approach then generates an undirected, fully linked, and weighted graph suited for ACO. In the first phase of implementation, the graph is built by extracting the two most common itemsets. The importance of this study stems from the requirement to minimise time and improve rule quality.

The identification of association rules from data transfers, where every operation consists of a collection of dataset, is one of the demanding aspects of information extraction. The assessment of the frequency of appearances of intriguing subsets is the most time-consuming function in this discovery approach (called candidates). Furthermore, synthetic data is turned into a Boolean matrix, after which the Apriori technique is used to quickly obtain the n-frequent itemset. After creating two frequent itemsets, a fully connected graph is generated. This graph can be analysed using the ACO method. The last frequent itemsets are obtained using the upgraded ACO algorithm. After the database has really been scanned, the ACS will be stimulated to get a good result by a representation of the data. Thereby, proposed system infrastructure employs a probabilistic graph theory-based method for solving computational issues and is depicted in figure 7.

 https://doi.org/10.37896/pd91.4/91422

ISSN: 0369-8963

Hence the accuracy, storage complexity of the models generated by the methods, which are focused on the highest supports and conviction values produced. This is because IAFCM, which is a mix of FCM and the GACO optimization method, creates optimal clusters.

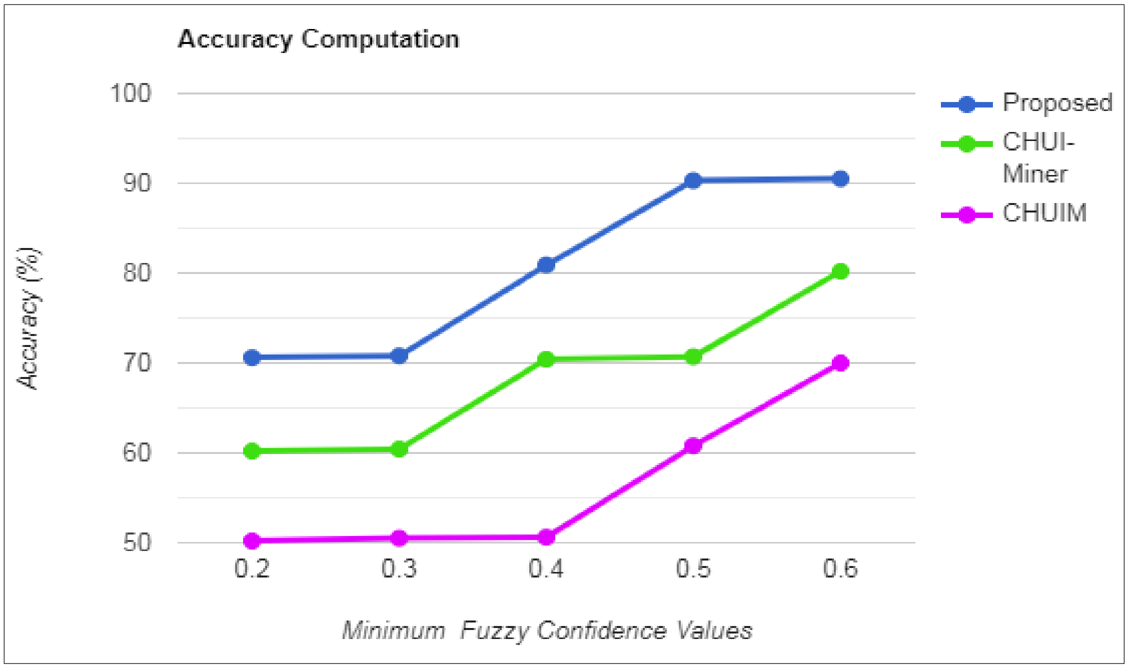


Figure 4. Comparison of the accuracy

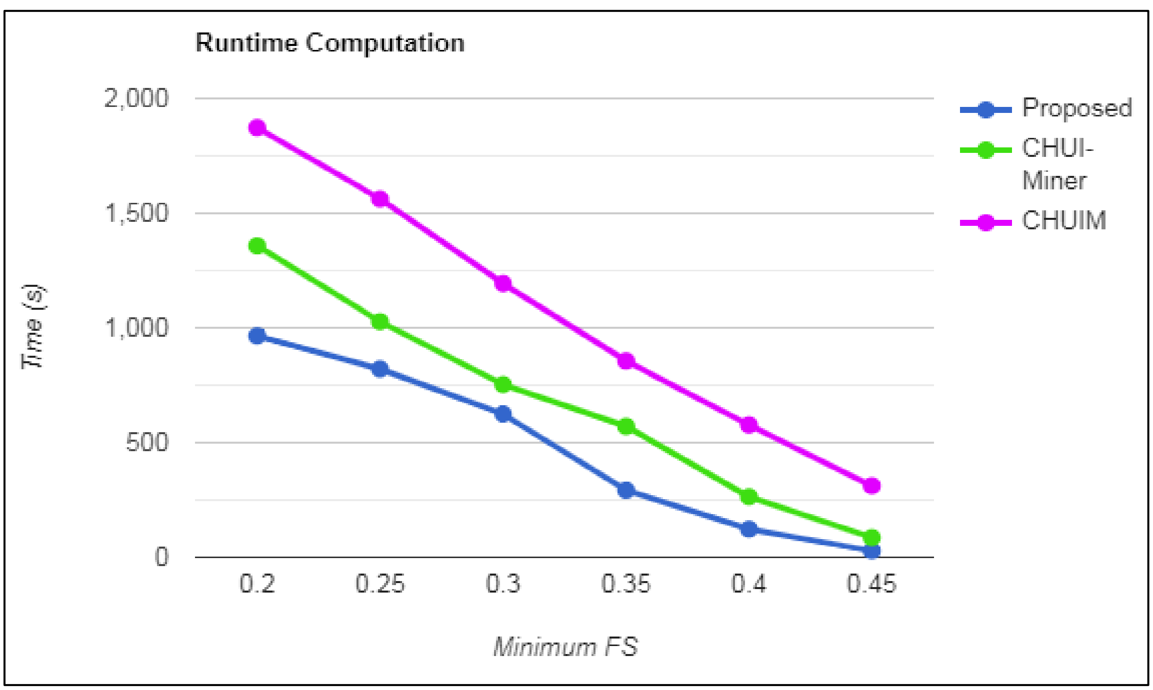


Figure 5. Runtime computation

 https://doi.org/10.37896/pd91.4/91422

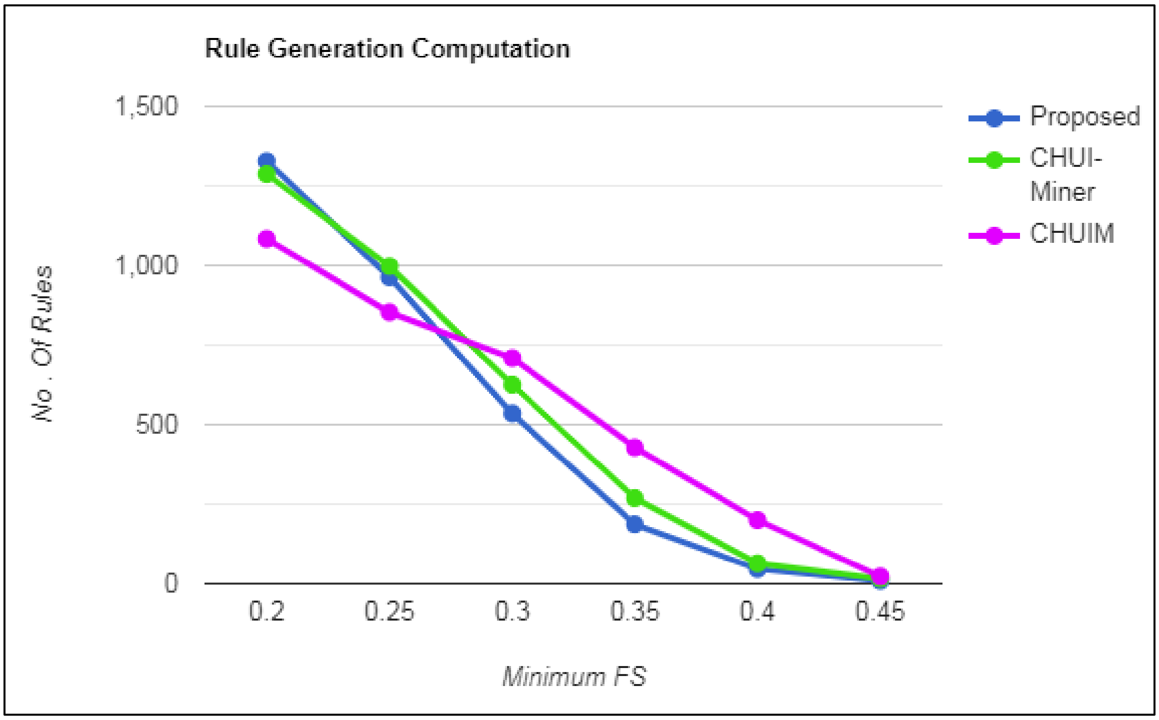


Figure 6. Rule generation computation

Table 4. The Experiment Results for run time and rule generation

Minimum Fuzzy Support	The proposed method		CHUIM		CHUIM-Miner	
	Time(s)	Rule(s)	Time(s)	Rule(s)	Time(s)	Rule(s)
0.20	966	1328	1359	1287	1874	1084
0.250	822	963	1028	998	1563	853
0.30	626	535	754	625	1195	709
0.350	293	187	572	269	857	428
0.40	124	46	265	64	578	200
0.450	29	9	87	16	313	23

For various minimal fuzzy level of confidence, the correctness of the suggested technique is superior than that of CHUIM and CHUIM-Miner as in figure 6. Furthermore, table 4 shows that the minimal fuzzy support is reduced, resulting in more clustering algorithms. The suggested technique outperforms the previous process in terms of evaluate time also, with a reduced baseline fuzzy supporting. However, a lesser minimum inaccurateness support has no effect on the overall evaluate time. The execution instance will become increasingly apparent when the minimal fuzzy supporting grows.

 https://doi.org/10.37896/pd91.4/91422

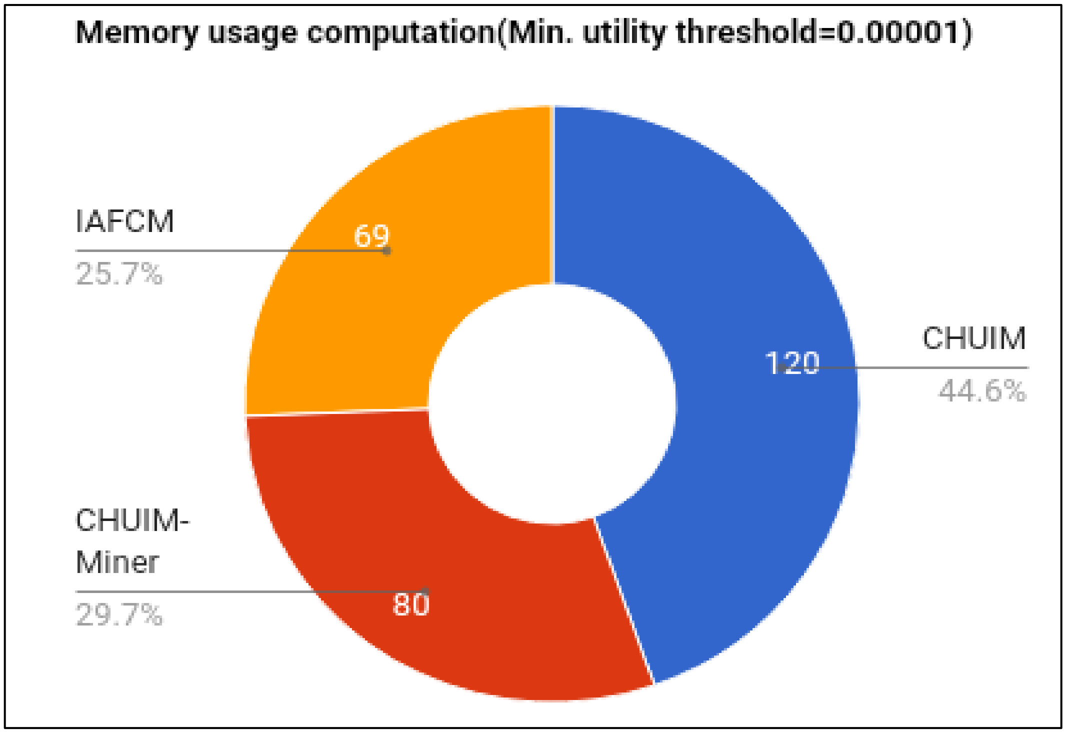


Figure 7. Memory usage computation

Figure 7 shows an analysis of proposed method memory use vs CHUIM and CHUIM-Miner. The pie chart represents the storage for minimum utility threshold of 0.00001 by the Python memory size library.

Figure 5 shows the suggested technique takes lesser instance to compute than PCFA and FCM-based mining methods for different numbers of transactions. For time-based increments, a single examine of the record is adequate in the suggested technique. It employs previously discovered data structures and mining results to prevent wasteful computations on database changes or minimum support thresholds. As a consequence, computation time is reduced. Thus, the suggested weighted temporal pattern mining approach (tree based weighted frequent pattern mining) has resulting in effective temporal patterns with decreased computing costs due to a reduction in data size and optimal clustering utilising MoAFCM.

It is observed, the hypothesized IAFCM-GACO method is found to use less memory than the other algorithms. In particular, the space use of the CHUIM and CHUIM-Miner are 2 and 0.5 times larger that of the suggested method correspondingly.

Conclusion

The IAFCM methodology is combined with a tree-based weighted rule extraction method in the proposed weighted sequential information retrieval strategy. The IAFCM technique combines FCM and GACOS algorithms. The IAFCM approach creates more productive clusters as compared to earlier approaches. By examining patterns with lower frequencies but high priority, the tree-based weighted frequent pattern mining approach applied to these graph onotology optimized clusters produces effective temporal patterns (weight). Because time is such an important factor in sequential information, the IAFCM approach is suitable for time-based relevance, and weighted sequential prototype extraction may be used to temporal datasets that are updated in a time-based way. Thus, when combined with IAFCM, the weighted temporal pattern mining approach is ideally suitable for time series of item set and may be used in a concurrent setting. Prior to frequent itemset mining, it is necessary to cluster time-based data. The effectiveness of the often received itemsets is determined by the cluster quality. As a result, the future work has to be concentrated over a strategy for cluster refining before undertaking frequent itemset mining using learning algorithms.

References

[1] Jyothi Pillai, and Vyas O.P., “Overview of Itemset Utility Mining and its Applications”, International Journal of Computer Applications, Vol. 5, No.11, pp. 9-13, 2010.

[2] Praseeda Manoj, “Emerging Database Models and Related Technologies”, International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 3, No. 2, pp. 264-268, 2013.

[3] Manziba Akanda Nishi, Chowdhury Farhan Ahmed, Md. Samiullah, and Byeong-SooJeong, “Effective periodic pattern mining in time series databases”, Expert Systems with Applications, Vol. 40, No. 8, pp. 3015-3027, 2013.

[4] Thodeti Srikanth, “Data Mining in Sequential Pattern for Asynchronous Periodic Patterns”, International Journal of Computer Science Issues, Vol. 8, No. 6, pp. 313-316, 2011.

[5] Illango Murugappan, and Mohan Vasudev, “PCFA: Mining of Projected Clusters in High Dimensional Data Using Modified FCM Algorithm”, The International Arab Journal of Information Technology, Vol. 11, No. 2, pp. 168 – 177, 2014.

[6]Duraimutharasan N., and Sarukesi K., “Study on Event Matching In Temporal Database Using AGT Approach”, International Journal Advanced Networking and Applications, Vol. 4 No. 3, pp. 1640- 1644, 2012.

[7] Priya Vaijayanthi, Xin-She Yang, Natarajan A.M., and Raja Murugadoss, “ High Dimensional Data Clustering Using Cuckoo Search Optimization Algorithm”, International Journal of Advanced Computer Engineering and Communication Technology, Vol. 3, Issue-3, pp. 31 – 35, 2014.

[8] Vedant Rastogi, and Vinay Kumar Khare, “Apriori Based: Mining Positive and Negative Frequent Sequential Patterns”, International Journal of Latest Trends in Engineering and Technology, Vol. 1, No. 3, pp. 24-33, 2012.

[9] Deshpande S.P., and Thakare V.M., “Data Mining System and Applications: A Review”, International Journal of Distributed and Parallel systems, Vol. 1, No. 1, pp. 32-44,2010

[10]Madan Kumar K.M.V., Srinivas P.V.S., and Raghavendra Rao C., “Sequential Pattern Mining With Multiple Minimum Supports by MS-SPADE”, International Journal of Computer Science Issues, Vol. 9, No. 5, pp. 285-292, 2012.

[11]Yong Joon Lee, Jun Wook Lee, Duck Jin Chai, Bu Hyun Hwang, and Keun Ho Ryu, “Mining temporal interval relational rules from temporal data”, Journal of Systems and Software, Vol. 82, No. 1, 155–167, 2009.

[12] Sunil Joshi, Jadon R.S., and Jain R.C., “Sequential Pattern Mining Using Formal language Tools”, International Journal of Computer Science Issues, Vol. 9, No 2, pp. 316-325, 2012.

[13] Ahmed, C.F., Tanbeer, S.K., Jeong, B.-S., and Lee, Y.- K., “Mining Weighted Frequent Patterns in Incremental Databases”, In: Ho T.-B., and Zhou Z.-H. (eds.) PRICAI 2008, LNCS (LNAI), Springer, Heidelberg, Vol. 5351, pp. 933–938. 2008.

[14]H. Yao, H. J. Hamilton, and C. J. Butz, “A foundational approach to mining itemset utilities from databases,” in Proceedings of the 2004 SIAM International Conference on Data Mining, pp. 482–486, SIAM, Lake Buena Vista, FL, USA, 2004

[15] Y. Liu, W.-K. Liao, and A. Choudhary, “A two-phase algorithm for fast discovery of high utility itemsets,” Advances in Knowledge Discovery and Data Mining, Springer, Berlin, Germany, pp. 689–695, 2005

[16] C.-W. Lin, T.-P. Hong, and W.-H. Lu, “An effective tree structure for mining high utility itemsets,” Expert Systems with Applications, vol. 38, no. 6, pp. 7419–7424, 2011.

[17] W. Song, Y. Liu, and J. Li, “Mining high utility itemsets by dynamically pruning the tree structure,” Applied Intelligence, vol. 40, no. 1, pp. 29–43, 2014.

[18] P. Fournier-Viger, J. C.-W. Lin, T. Gueniche, and P. Barhate, “Efficient incremental high utility itemset mining,” in Proceedings of the ASE Big Data & Social Informatics, pp. 1–6, Kaohsiung, Taiwan, 2015.

[19] T.-L. Dam, K. Li, P. Fournier-Viger, and Q.-H. Duong, “An efficient algorithm for mining top-k on-shelf high utility itemsets,” Knowledge and Information Systems, vol. 52, no. 3, pp. 621–655, 2017.

[20]C. Manike and H. Om, “Modified GUIDE (LM) algorithm for mining maximal high utility patterns from data streams,” International Journal of Computational Intelligence Systems , vol. 8, no. 3, pp. 517–529, 2015.

[21]J. C.-W. Lin, Y. Djenouri, and G. Srivastava, “Efficient closed high-utility pattern fusion model in large-scale databases,” Information Fusion, vol. 76, pp. 122–132, 2021.

[22]S. Bouasker and S. Ben Yahia, “Key correlation mining by simultaneous monotone and anti-monotone constraints checking,” in Proceedings of the 30th Annual ACM Symposium on Applied Computing, pp. 851–856, New York, NY, USA, 2015.

[23] M. Barsky, S. Kim, T. Weninger, and J. Han, “Mining flipping correlations from large datasets with taxonomies,” 2011, https://arxiv.org/abs/1201.0233.

[24] T. Wu, Y. Chen, and J. Han, “Re-examination of interestingness measures in pattern mining: a unified framework,” Data Mining and Knowledge Discovery, vol. 21, no. 3, pp. 371–397, 2010.

[25] W. Gan, J. C.-W. Lin, P. Fournier-Viger, H.-C. Chao, and H. Fujita, “Extracting non-redundant correlated purchase behaviors by utility measure,” Knowledge-Based Systems, vol. 143, pp. 30–41, 2018

[26] W. Gan, J. Chun-Wei, H.-C. Chao, T.-P. Hong, and S. Y. Philip, “CoUPM: correlated utility-based pattern mining,” in Proceedings of the IEEE International Conference on Big Data (Big Data), pp. 2607–2616, IEEE, Seattle, WA, USA, 2018.

[27] P. Fournier-Viger, Y. Zhang, J. C.-W. Lin, D.-T. Dinh, and H. Bac Le, “Mining correlated high-utility itemsets using various measures,” Logic Journal of the IGPL, vol. 28, no. 1, pp. 19–32, 2020.

[28] C. F. Ahmed, S. K. Tanbeer, B.-S. Jeong, and H.-J. Choi, “A framework for mining interesting high utility patterns with a strong frequency affinity,” Information Sciences, vol. 181, no. 21, pp. 4878–4894, 2011.

[29] J. C.-W. Lin, W. Gan, P. Fournier-Viger, T.-P. Hong, and H.-C. Chao, “FDHUP: fast algorithm for mining discriminative high utility patterns,” Knowledge and Information Systems, vol. 51, no. 3, pp. 873–909, 2017

[30] B. Vo, L. V. Nguyen, V. V. Vu et al., “Mining correlated high utility itemsets in one phase,” IEEE Access, vol. 8, pp. 90465–90477, 2020.

[31]. Zhu, W.; Wang, J.; Cao, H.; Zhang, Y.: A novel association rule decision algorithm based on ant colony optimization algorithm for ball mill pulverizing system. In: International Conference on Computer Science and Software Engineering, IEEE (2008).

[32]. Preethi, P., Asokan, R., Thillaiarasu, N., & Saravanan, T. (2021). An effective digit recognition model using enhanced convolutional neural network based chaotic grey wolf optimization. Journal of Intelligent & Fuzzy Systems, (Preprint), 1-11.

[33]. Punarselvam, E., & Suresh, P. (2011, December). Edge detection of CT scan spine disc image using canny edge detection algorithm based on magnitude and edge length. In 3rd International Conference on Trendz in Information Sciences & Computing (TISC2011) (pp. 136-140). IEEE.

[34]. Preethi, P., & Asokan, R. (2021). Modelling LSUTE: PKE Schemes for Safeguarding Electronic Healthcare Records Over Cloud Communication Environment. Wireless Personal Communications, 117(4), 2695-2711.

[35]. Hemalatha, E., Dhamodaran, M., & Punarselvam, E. (2019). Robust Data Collection with Multiple Sink Zone in 3-D Underwater Sensor Networks. International Journal on Applications in Basic and Applied Sciences, 5(1), 8-14.