CHAPTER 1

INTRODUCTION

1.1 DATA MINING PROCESS

The extraction of meaningful and non-negligible facts from the enormous quantity of data from the diverse field, business, and engineering is Data Mining (DM). The process of valid identification of innovative, valuable and logical patterns from the large and complex data sets referred as Knowledge Discovery in Database (KDD) [34] and it is
regarded as the specific part of the DM. The DM techniques and algorithms utilize the tools that analyse the unknown and associated patterns to support the various applications such as banking, customer relationship management, targeted marketing, fraud detection, manufacturing and production, medicine, technical data inquiry, and web assortment scheme as shown in figure 1.1 below.

![Diagram of Data Mining Process](image)

**Figure 1.1: The Data Mining Process in Different Applications**

The DM faces many challenges such as large size data, diverse data types, mining methodologies and user interactions in the research community. Hence, the new methods are necessary to mine interesting and specific material from the collected data. The high dimensional data cause more space occupation and time complexities. In order to minimize these problems, many algorithms are described in the research studies. Mining algorithms [25] provide the graphical data visualization for the discovery of different types of patterns. The patterns discovered from the dataset are utilized to predict and classify the new data arrived in the decision-making process. Finding hidden patterns in the database includes several tasks such as frequent, weighted and high utility pattern
mining. Among these mining methods, the frequent pattern mining is the best one and utilized for different types of databases such as transactions, streaming databases, etc.

The DM concept includes the parameters such as association rules and the path analysis. The association rules [105] predict the patterns in which the events are inter-related with one another. The mining of complex databases along the transaction path refers path analysis. In association rule mining, the frequent itemset mining is the time conserving process with the utility factor inclusion.

1.2 HIGH UTILITY ITEMSET MINING

The itemset utility is based on the interestingness, status and effectiveness of a particular item for the user with profitability. High utility itemset mining [7] is specified as realizing the itemsets with great earnings. The categorization of the utility of the item set is based on the type of database used. The databases available in real-time are transactional, trajectory and time series. The utility of the itemset in trajectory database is defined based on the following constraints:

- Labeling of specific essential entities as extraneous utility.
- The presence of an essential entity in transaction refers constitutional utility.

The utility of the transaction database is based on two aspects. They are external utility and internal utility. The external utility itemset holds the importance of the separate itemsets and internal utility holds the importance of transaction in various items. The utility of the itemset in time-series database is based on the invention of both outer and inner utility. The utility itemset having the value is higher than the user specified utility threshold value called as high utility itemset.

1.2.1 Association Rule Mining
The Association Rule Mining (ARM) [14] is the vital part that is controlled by the DM. ARM searches the interesting patterns from dense items in databases with the set of association rules. The demand for the discovery of the frequent pattern from huge databases increases the importance of the ARM. The utilization of traditional ARM in the transaction set depends on the occurrence of the utility itemsets. The association rules are widely utilized in various areas such as networks, markets, risk managements and inventory controls.

1.2.1.1 Weighted Association Rule Mining

The Weighted Association Rule Mining (WARM) [10] is implemented in the databases only with attributes, focusing on the significant relationship which results in a combinatorial explosion. WARM utilizes the importance of each itemset within the transaction and it is related to some special promotion and profit on different items. The weighted association rule is based on the traditional rule of association rule mining. It is challengeable in using the weighted rule as the iterative process of generating item-set. The problems raised in the downward closure properties are solved by adapting the traditional model in association rule mining. The developed algorithm is both scalable and efficient in determining itemsets are simulated.

1.2.1.2 Multi-level association rules

For finding the frequent patterns [4] using minimum support thresholds decided by the user has failed in the weighted association rule. The items present at the minimum level in data and the rules associated with the itemsets at suitable levels are helpful in creating the multiple-level association rules. The transaction database is encoded based on the scopes and levels. The Figure 1.2 shows the data flow of utility mining process to identify all actual high utility itemset.
1.2.2 EFFICIENCY OF THE HIGH UTILITY ITEMSET MINING

The target of the association rule mining [13] is to find an interesting association or relations among the different itemsets in the database. The interestingness measure plays the vital role in KDD intended for selecting and ranking patterns based on the potential interest of the user. The factors to improve the efficiency of the high utility Itemset Mining are as follows

- Original database, the number of scans are reduced.
- Minimize memory utilization by reducing the search space.
- Reducing the computation and total execution time.
- Reducing the source utilization.
- Based on time and space complexity, the performance is increased.

1.2.3 APPROACHES OF HIGH UTILITY ITEMSET MINING

Over the dense dataset, utility mining is the challenging area of research for the data mining investigators. In the utility, mining framework, the downward closure
property is maintained with the difficult task. The fastest and the most popular algorithms for high utility itemset mining is the FP-growth. The growth on FP is based on the prefix tree representation in database transactions. The FP-tree compresses the database into the tree structure which stores only the larger itemsets. In utility mining, each item may have different importance, such as the profits and degree of user interest and the importance is known as the utility. The high utility pattern mining reserves all itemsets in the transaction database with the utility value that is superior to the user specified utility threshold in minimum value. It discovers importance among various items in the mining process.

1.3 CLUSTERING

Clustering [94] is the process of grouping similar data or dissimilar objects in different groups. In cluster analysis, a set of data is partitioned in a group based on the data similarity. The clustering in data mining concept is based on scalability and has the capability to deal with different features and the discovery of clusters with qualities. The main advantage of the clustering is flexible to change and helps to distinguish different groups of clusters. The applications of clustering are utilized in pattern recognition, data analysis, market research and image processing. The following Figure 1.3 depicts the concepts of clustering formation process.

![Clustering formation process](image)

**Figure 1.3: Clustering formation process**

Functions that should be satisfied by the clustering algorithm
The main requirements that a clustering algorithm should satisfy are as follows:

**Dealing with different types of attributes:**

There are different attributes of data that any clustering algorithm has to satisfy. The most common taxonomy being in public use differentiates among numeric (continuous), ordinal, and nominal variables. A numeric variable can assume any value and ordinal variable assigns a small number of discrete states, and these states can be compared.

**Scalability to large datasets:**

The datasets can be in any range, which varies between large extremes and required to be normalized by the clustering algorithm.

**Ability to work with high dimensional data:**

The data could be multidimensional varying from 1, 2…… n., which, relies on the application data on which clustering is being practical.

**Ability to find clusters of irregular or arbitrary shape:**

The shape of clusters can be arbitrary. The Euclidean distance is utilized to attain a circular shape of the clusters, but the shape of clusters cannot be exactly defined.

**Handling outliers:**

The data points, which lie on the boundary of clusters have to be handled effectively. This is done in a hierarchical way of associating the boundary points to one of the clusters. While in fuzzy clustering, the membership is associated with functions to the points residing on the boundary of clusters.

**Time complexity:**

The complexity of the data points is computed in terms of time, which has to be taken care of during the clustering process.

**Data order dependency:**
The dependency of data points on another variable can affect the clustering of data and thereby the cluster centres too, so it has to be taken care of beforehand.

1.3.1 HEAP TREE FORMATION

The heap tree [7] is a special case for the balanced binary data tree structure based on the parent and child node. The root node key is compared with the constructed children and arranged in order. It is divided into two types on a value basis as Max heap and Min heap. The Max heap is defined as the value of the parent greater than the child and if the value of the parent is less than the child it is considered as the Min-heap.

1.3.2 TREE-BASED SEARCH

The tree based search [63] is also defined as the tree traversal. The tree search is a kind of graph traversal and referred to as the process of visiting or updating an individual node in the tree data structure. The traversal is classified by the nodes visited in order. The depth–first search algorithm is described for constructing the binary tree. In a depth-first search, the tree traversed in pre-order depends on each child before reaching the sibling node. The common recursive pattern for traversing a binary tree is initiated with the root node and the left and right subtree traverse is implemented. The following Figure 1.4 shows the arrangement of different types of nodes in Tree-Based Search.

![Figure 1.4: Types of nodes in Tree-Based Search](image)

31
The following pre-order searching process is used to form the heap tree formation in the research work.

- Display the data part of the current node or the root node.
- Traverse the left subtree by implementing the pre-order function
- Traverse the right subtree by implementing the pre-order function.

1.3.2.1 DEPTH FIRST SEARCH

The Depth First search is a tree-based structure with parent and child nodes. The nodes of the search tree are expanded in the pre-order and searched in depth in the first order. One branch from the root node to a leaf node is explored and the search backups the next shallowest node to have the unexplored success. The exploration desires to keep only the single path from the root to a leaf node along with the enduring in lengthened sibling nodes for each node on the path.

Figure 1.5: Depth First Search
In the above Figure 1.5, the depth-first search in pre-order is described. The nodes with H to O are placed in the tree structure and A is the root node, the other nodes are siblings. The result from the tree is ordered with H D B A C F E G L J I K N M O.

1.4 BIT MASK SEARCH

The bit search provides compaction or compression mechanism to increase the density in bit vector region. The bit search region will not increase the projection length but increases the CPU performance. From the root node, a search representation for each frequent item is created and provides the sufficient improvements in the BMS. The graph-based approach for searching frequent dataset are with primitive association rules and generalized association rules and the Bit Mask search initiated with the array list. The bit mask search is the novel approach that the input file is first converted into numerical data.

The transaction file is compressed into an array for further processing. In data mining, the association rule extraction is the important issue in discovering regular itemset. The stock and the weighted association rule mining are not processed on the databases with binary structures. The application introduces the requirement for the mining algorithm to be scalable based on the number of records with respect to the domain size. The steps to find the frequent itemsets and the association rules are

- All the items are assigned as integer number in numbering phase.
- Large item and records related information are generated by large item generation phase.

The item wise representation format is converted from the transactions. The sparse matrix form is derived from the items representation and masked into sparse bit form based on the transaction. The sparse matrix form is referred as the medium that is initiated primarily to zero and warehoused using a special data structure. It allows to store only non-zero elements and conserve memory with time on working based on the matrix.
Based on the features of the sparse data source when mining association rules, a linked list unit and some strategies is designed to store data in the matrix. The sparse matrix mining is utilized to search large itemset in the database.

The sparse matrix mining maps the database to the binary sparse matrix and thus stores compressed data into the linked list to discover the huge itemsets. To convert the association rule mining algorithm, the vertical form of scarce bit representation is utilized. The itemset combinations are implemented through candidate key generation. The itemsets are bitwise and processed based on the transactions of itemset at first scan. The bit array transactions are operated with the subset bit array and the result is compared with the corresponding itemset bit array structure. The count is increased by one based on the items present in the itemsets and compare the next transaction to find the next support level of itemsets.

The bit Mask search is superior to the bit search, Apriori tree and FP-growth based on the advantages such as

- The Bit Mask reduces the search space in each iteration.
- The execution time is minimized and the BMS execution time is minimum when compared with the other algorithm and the size of the data sets is also large.

Frequent itemsets have been generated with the help of Apriori-based bit search technique and is defined as the Sparse Bit mask Search. A pattern of the binary value is combined with the bit value 1 for the presence of an item and 0 for the absence of an item. The transaction with 0 and 1 combination for the searching process is known as the Bit Mask. The new data structure built for the frequent itemset mining is implemented. One of the important contributions of this work is a novel searching technique used for special data structure, called Sparse Bit Matrix.

**LOSSY ALGORITHM**
The lossy algorithm is introduced to identify the elements in the data stream where the threshold value is defined by the user. The time, space required to run and the problem are inversely proportional to the error threshold specification. It tracks the items and merges with the already stored items with counts. The lossy algorithm is utilized for compressing the data in order to avoid the data loss.

**COUNTING ALGORITHM**

The counting algorithm is a collection of objects based on the keys with small integers. It is initiated by counting the objects having the key value with the arithmetic count to determine the position of output sequence. The number of items with the higher and lower key value difference is based on the linear running time of counting sort. The counting algorithm is utilized to reduce the number of processing times in the source dataset that scanned in the frequent pattern query processing.

**GRAPH-BASED MINING (GBM)**

The efficient Graph Based Mining (GBM) is utilized in mining the frequent patterns in the spatial-temporal database. This method results in two phases and they are:

- The trajectories available in the database are plotted into a graph, which is used to generate the TI list. It consists of trajectories are to be created for all the vertices of the graph.
- To mine the frequent patterns, the GBM traverses the mapping graph in the DFS method.

The GBM maintains counting and pattern extension with the small number of information list using the mapping graph. The GBM utilizes the adjacency property to reduce the search space.
**K-NEAREST NEIGHBOUR (K-NN)**

K-NN is one of the simplest classifiers that govern the unidentified data point by using the previously known data points (nearest neighbour) and categorized data points according to the voting system. K-NN classifies the data points using more than one nearest neighbour. It includes a number of applications in diverse areas such as image field, cluster analysis, pattern recognition, health datasets, online marketing, etc.

**1.4.1 STATISTICAL ANALYSIS**

The statistical analysis is the most common method to extract the exact information from the website. Many web circulation exploration tools yield the statistical material based on the frequently accessed pages, average view time of the page and the average distance of the pathway through the site. The depth of the analysis can be potentially useful to improve the system performance and enhance the security of the system fascinating the site modification assignment and giving support for marketing conclusion.

**1.4.2 SEQUENTIAL PATTERN**

The technique of sequential pattern discovery attempts intersession patterns with time ordered in the presence of the set of items followed by another item in the time ordered set of sessions. The other types of the analysis that are performed on the sequential patterns includes the analysis, change point of the detection and similarity analysis, etc.

**1.4.3 CLUSTERING BASED ON PREDICTION**

Clustering is the technique to group the set of items based on the similar characteristics. In the web field, there are two types of clusters such as usage clusters and page clusters. The cluster is utilized to create a group of users on similar patterns.
Classifications can be supervised by inductive learning such as decision tree classifiers, naïve Bayesian classifiers, k-nearest neighbour classifiers, support vector machines, etc.

**Figure 1.6: Clustering based on prediction**

In detecting a small and homogeneous group of items, the clustering approaches are utilized. Based on clustering, the precision knowledge is improved and the multiple classifications improve the performance prediction in a system.

### 1.4.3.1 HIERARCHICAL CLUSTERING

Hierarchical clustering decomposes the data points either by using the bottom-up approach or top-down approach. It can be classified into two categories, namely, Agglomerative and Divisive, which relies on the decomposition process. Agglomerative approach primarily considers each data point as a distinct group and further it combines the data points, which have some similarity with each other. This process is repetitive until all the data points are joined into one cluster or class or until it acquires some termination condition. A divisive clustering begins with one cluster of all data points and recursively divides the most appropriate cluster. This technique starts at the top with all patterns in one cluster and then the cluster is divided by utilizing a flat clustering algorithm. This procedure is applied recursively until each pattern is in its own singleton cluster.

### 1.4.3.2 K-MEANS CLUSTERING

K-means clustering is a modest iterative method to partition the given dataset into a user-specified number of K-clusters. This function relies on fuzzy logic in its work and assigns many centers. Later, it identifies the smallest distance from the points to these
centres and then rearranges these points as clusters. The distance of each cluster from fixed centre is less than the distance from another centre.

1.4.4 DEPENDENCY MODELING

The dependency modeling is another useful pattern discovery task in web mining. The target of this modeling is to enrich the capabilities of representing major dependencies among the several variables in the web domain. The motivation behind the analysis is to filter, then on interesting rules or patterns from the set that are present in the pattern discovery phase. The exact exploration methodology is usually applied by the web mining application. The shared pattern contains the knowledge query mechanism such as SQL. And the other method to convert usage data into a data cube is to perform OLAP operations. Visualization techniques based on graphic patterns signing colors to different values highlights from the overall patterns in the data. The content and the structure data are used to filter the patterns and contain data that match the hyperlink structure.

1.4.5 SPACE PARTITIONING

Space partitioning is the hierarchical system organizing the region as a tree structure. This is the process of dividing the space into regions as disjoint subsets. The divided regions structure the tree.

1.4.6 APRIORI ALGORITHM

Apriori [45] is the standard algorithm for auditing association rules and specially implemented to operate transaction based databases. Apriori is utilized in association rule mining for the candidate generation. The Apriori algorithm helps to obtain the relevant itemset from the database. It counts the occurrences to determine the large itemset that requires scanning of the database at each time in the entry level. In the next step, Apriori
predicts the overall value of the candidate itemset by database scanning. From the frequent itemset, the association rules are created.

1.4.7 FP-TREE BASED MINING

The efficient mining method builds the frequent pattern tree structure for handling and storing the critical frequent pattern information into the compressed structure. This pattern growth based an tree structures minimize the scanning process for finding frequent pattern using FP-Growth [63]. The original database is compressed and FP-tree is constructed seems smaller significance when compared with the original database. The expensive database scans are stored in the mining process and the pattern growth method applies avoiding the costly candidate generation.

Apriori tree and FP-growth are started with

- Increase in the density in bit vector region.
- Reduced execution.
- Reduced search space in each iteration.

The FP-Growth performs fewer scans with more energy consumption in long pattern dataset generation and high utility itemset discovery.

1.4.8 TWO-PHASE ALGORITHM

The main goal of the mining process is to find the high utility itemset that is drawn with a large portion of the total utility. The challenges in the utility mining are the eradication of candidate size and the simplification of the computed value for utility estimation. With the high utilities, the determination of itemsets is perceived by the high utility itemset mining. The two-phase algorithm effectively trims the number of candidates and set the high utility itemsets. The performance of the mining is efficiently based on both the speed and cost of the memory.
The two-phase algorithm identifies the most efficient, high utility itemset and the transaction weighted utilization mining restricts the search space but covers all the utility itemsets. More than one scan is performed to filter the misjudged itemsets and this algorithm requires less scan and less memory space with less calculation cost when compared with the existing utility mining algorithm.

1.4.9 UP-GROWTH ALGORITHM

The UP-Growth mining [98] method is utilized to overcome the difficulty of creating a huge number of candidate itemset for high utility itemset. The huge number of candidate itemset decreases based on the mining performance of execution time and space requirement. The long transactions in the database lead to the worst condition in obtaining high utility itemsets. For removing high utility itemsets for pruning the candidate itemset with the set of methods is performed in the UP-Growth algorithm. The information regarding the high utility itemset is used to perform the database scanning by using the UP-Tree data structure. The UP-Growth algorithm reduces the execution time of long structure transaction databases handling.

1.5 MOTIVATION

Time and space are the two important factors in the data mining concept. Research studies illustrated many algorithms that are used to enhance the memory space for frequent entity sets prediction and to improve the efficiency. The Apriori algorithm utilization for frequent pattern extraction leads to more candidate itemsets generation and a number of scans in the database. In order to overcome these drawbacks, the FP-tree based mining is utilized for extracting and storing the critical frequent information patterns. But, the excessive energy consumption due to the large data size utilization causes the difficulties in best result prediction.
To overcome the above-mentioned drawbacks in the FP-tree algorithm, the two-phase algorithm is applied. This algorithm obtains the high utility itemset by using multiple scans that lead to time complexity. Hence, the tree based algorithm maintains the frequent pattern information and their values with the worst performance in both time and space utilization that initiates the UP-Growth. UP-Growth searches the high utility itemset values beyond the user-specified threshold in a transaction database with better performance. But, the more memory consumption in UP-Growth leads to the evolution of Bit Mask Search (BSM) algorithm [102]. This fulfills space complexity and memory consumption problems described in earlier studies.

1.6 OBJECTIVES

- To reduce the number of candidates by using an Isolated Item Discarding Strategy (IIDS) method.
- To maintain the information and their attributes by using Incremental High Utility Pattern (IHUP) utilization.
- To obtain the frequent itemset from the particular database by the introduction of Apriori algorithm.
- To store the crucial information about the frequent pattern itemsets into the compressed format by using the FP-tree based mining method.
- To prune down the number of candidates and obtaining the complete set of high utility itemsets, the Two-phase algorithm is implemented.
- To identify the high utility itemset with values outside the user-indicated threshold in a transaction database, the UP-Growth method is introduced.
- To mine the high utility entities from the database by using the UP-Growth + algorithm
- To raise the density of the bit vectors by using the Bit Mask search
1.7 THESIS ORGANIZATION

**Chapter 1** contains the basic introduction of this research work, throws light on the objective, motivation behind this research work and justification of its relevance.

**Chapter 2** describes the literature review related to the existing algorithm and the various strategies are explained. The drawbacks in the existing algorithm are described elaborately. Various algorithms like Apriori, FP-tree, Two-phase, Tree-based, and UP-Growth are described.

**Chapter 3** elaborates, the BM search algorithm for mining the frequent entity sets in trajectory database. The BM search algorithm starts with an array list to raise the density in bit-vector. The BM Search algorithm, the denied patterns will be removed to find the frequently visited places in the trajectory database. This chapter also depicts the comparative analysis of BM search and UP-Growth Algorithm.

**Chapter 4** describes the Tree-based Space Partition of Trajectory Pattern Mining (TSPTPM) algorithm to mine the frequent itemsets. It also explains the frequent item set mining based on Vague-Space Partition (VSP) algorithms under the tree-based structures.

**Chapter 5** illustrates the performance comparison of the suggested BM search algorithm against the UP-Growth+ algorithm. Further, it establishes the superiority of the proposed TSPTPM algorithm over the existing algorithms. The need for the proposed work and the superiority of the proposed method against the existing methods is validated in the upcoming chapters.

**Chapter 6** concludes the research work related to problem defined. It explains future directions of the research work also.