International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 1, January - 2014

Density-Subspace Clustering Based on Frequent Data Pattern

Anand Kumar Tripathi [1]
Beerendra Kumar[2]
College of Science and Engineering, Jhansi, Uttar Pradesh

***Abstract*—This paper irrelevant attributes add noise to high dimensional clusters and make traditional clustering techniques inappropriate. Subspace clustering algorithms have been proposed to find the clusters in hidden subspaces. Instead of finding clusters in the full feature space, subspace clustering is an emergent task which aims at detecting clusters embedded in subspaces. Subspace clustering is an efficient approach to clustering high dimensional data. It has recently been developed rapidly. Subspace clustering assumes that different subspace may contain different meaningful cluster. A subspace cluster is a subset of points together with a subset of attributes, such that the cluster points project onto a small range of values in each of these attributes, and are uniformly distributed in the remaining attributes. This paper proposes a subspace clustering algorithm which follows the bottom-up strategy, evaluating each dimension separately and then using only those dimensions with high density in further steps. We realize the analogy between mining frequent item sets and discovering the relevant subspace for a given cluster, proposes a methodology for finding subspaces clusters by mining frequent item sets and present heuristics that improve its quality by regarding the intervals in all dimensions as a set of unique items in frequent item set mining problem, any k-dimensional unit can be regarded as a k-item set, i.e., an item set of cardinality k. Thus, to identify the dense units satisfying the density thresholds in subspace clustering is similar to mine the frequent item sets satisfying the minimum support in frequent item set mining. Efficient algorithms to discover frequent patterns are crucial in data mining research. MFP algorithm converts a transaction database to an MFP tree through scanning the transaction database only once, then prunes the tree and at last mine the tree. Because the MFP algorithm scans a transaction database one time less than the FP growth algorithm, the MFP algorithm is more efficient under certain conditions. Most of previous works in the literature are density-based approaches, where a cluster is regarded as a high-density region in a subspace. However, the identification of dense regions in previous works lacks of considering a critical problem, called "the density divergence problem" in this paper, which refers to the phenomenon that the region densities vary in different subspace cardinalities. To tackle the density divergence problem, in this paper, we devise a novel subspace clustering model to discover the clusters based on the relative region densities in the subspaces, where the clusters are regarded as regions whose densities are relatively high as compared to the region densities in a subspace. Based on this idea, different density thresholds are adaptively determined to discover the clusters in different subspace cardinalities.**

Keywords— Clustering, Density, Subspace Clustering, Frequent Data Pattern.

1. INTRODUCTION

Clustering is the unsupervised classification of patterns (observations, data items, or feature vectors) into groups (clusters). A cluster can be defined as "A cluster is a set of entities which are alike, and entities from different clusters are not alike." A cluster is "an aggregate of points in the test space such that the distance between any two points in the cluster is less than the distance between any point in the cluster and any point not in it." Clearly, a cluster in these definitions is described in terms of internal homogeneity and external separation, i.e., data objects in the same cluster should be similar to each other, while data objects in different clusters should be dissimilar from one another. [1]Clustering algorithms differ among themselves in their ability to handle different types of attributes, numerical and categorical. Clustering can be performed both on numerical data and categorical data. For the clustering of numerical data, the inherent geometric properties can be used to define the distance between the points. But for clustering the categorical data, such a criterion does not exist , on which distance functions are not naturally defined.

Data mining is the process of extracting the data from large databases, used as technology to generate the required information. Data mining methods can be used to predict future data trends, estimate its scope, and can be used as a reliable basis in the decision making process. Functions of data mining are association, correlation, prediction, clustering, classification, analysis, trends, outliers and deviation analysis, and similarity and dissimilarity analysis. One of frequently used data mining method to find patterns or groupings of data is clustering. Clustering is the division of data into objects that have similarities. Showing the data into smaller clusters to make the data becomes much simpler, however, can also be loss of important piece of data, therefore the cluster needs to be analyzed and evaluated. A common approach within the machine learning community involves unsupervised learning of parameters that describe clusters (e.g. the location and scale/shape of the cluster) and partitioning the data by associating every point or region with one or more clusters. In general, data mining tasks can be classified into two categories: descriptive and predictive. Descriptive mining tasks characterize the general properties of the data in the database. Predictive mining tasks perform inference on the current data in order to make predictions. In some cases, users may have no idea regarding what kinds of patterns in their data may be interesting, and hence may like to search for

several different kinds of patterns in parallel. Therefore, divisive clustering is not a common choice in practice. However, the divisive clustering algorithms do provide clearer insights of the main structure of the data since the larger clusters are generated at the early stage of the clustering process and are less likely to suffer from the accumulated erroneous decisions, which cannot be corrected by the successive process[9]. Heuristic methods have been proposed, such as the algorithm DIANA (Divisive ANAlysis) [9], based on the earlier work of which consider only a part of all possible divisions. According to [9], at each stage, DIANA consists of a series of iterative steps in order to move the closer objects into the splinter group, which is seeded with the object that is farthest from the others in the cluster to be divided.

Thus it is important to have a data mining system that can mine multiple kinds of patterns to accommodate different user expectations or applications. Furthermore, data mining systems should be able to discover patterns at various granularities (i.e., different levels of abstraction). Data mining systems should also allow users to specify hints to guide or focus the search for interesting patterns. Because some patterns may not hold for all of the data in the database, a measure of certainty or "trustworthiness" is usually associated with each discovered pattern. Finding such frequent patterns plays an essential role in mining associations, correlations, and many other interesting relationships among data. Moreover, it helps in data classification, clustering, and other data mining tasks as well. Thus, frequent pattern mining has become an important data mining task and a focused theme in data mining research. Frequent pattern mining searches for recurring relationships in a given dataset. This section introduces the basic concepts of frequent pattern mining for the discovery of interesting associations and correlations between itemsets in transactional dataset. Imagine that we have given a set of data objects for analysis where, unlike in classification, the class label of each object is not known. This is quite common in large databases, because assigning class labels to a large number of objects can be a very costly process. Clustering is the process of grouping the data into classes or clusters, so that objects within a cluster have high similarity in comparison to one another but are very dissimilar to objects in other clusters. Dissimilarities are assessed based on the attribute values describing the objects. Often, distance measures are used. Clustering has its roots in many areas, including data mining, statistics, biology, and machine learning [2]. Clustering is a key to data mining problem. Density and grid based technique is a popular way to mine clusters in a large multi-dimensional space wherein clusters are regarded as dense regions than their surroundings.

1.1 Subspace Clustering

By Carlotta Domeniconi from George Mason University and Dimitris Papadopoulos from University of California Riverside[7] discussed the subspace clustering methods for the high dimensional data. Subspace clustering is an extension of traditional clustering that seeks to handle clusters in different subspaces within a dataset. Often in high dimensional data, many dimensions are irrelevant and can mask existing clusters in noisy data. Feature selection removes irrelevant and redundant dimensions by analyzing the entire dataset. Subspace clustering algorithms localized the search for relevant dimensions allowing them to handle clusters that exist in multiple, possibly overlapping subspaces. There are two major branches of subspace clustering based on their search strategy. Top- down algorithms find an initial clustering in the full set of dimensions and evaluate the subspaces of each cluster, iteratively improving the results. Bottom-up approaches find dense regions in low dimensional spaces and combine them to form clusters. This paper presents a survey of the various subspace clustering algorithms along with a hierarchy organizing the algorithms by their defining characteristics. Subspace clustering is an extension of feature selection that attempts to find clusters in different subspaces of the same dataset. Just as with feature selection, subspace clustering requires a search method and an evaluation criteria. In high dimensional data, the number of possible subspaces is huge, requiring efficient search algorithms.

1.2 Cluster Analysis

Cluster analysis is a quite popular method of discretizing the data [3]. Cluster analysis performed with multivariate statistics, identifies objects that have similarities and separate from the other object, so the variation between objects in a group smaller than the variation with objects in other groups. Cluster analysis consists of several stages, beginning with the separation of objects into a cluster or group, followed by appropriate to interpret each characteristic value contained within their objects, and labelled of each group. The next stage is to validate the results of the cluster, using discriminant function.

1.3 Density Based Clustering

Density-based clustering method calculating the distance to the nearest neighbour object, object measured with the objects of the local neighbourhood, if inter-object close relative with its neighbour said as normal object, and vice versa. In density-based cluster, there are two points to be concerned; first is density-reachable.

1.4 Frequent Patterns-Based Subspace Clustering

In this method of clustering, they first obtain cluster information, and then convert the original data set into a transaction dataset, in which they can find out frequent patterns which imply subspace clusters. The contributions of this paper are as follows:

- It presents a frequent patterns-based algorithm for subspace clustering
- It avoids to generate candidate subspace.
- It provides sufficient experimental results which validate performance of presented method.

This algorithm uses the concept of frequent pattern mining to find subspace clusters. Clustering procedure can be done in two stages. In the first stage, they obtain cluster information of 1-dimensional subspace; in the second stage, they search subspace clusters by using FP-trees. The frequent patterns represent aggregations of X. These aggregations actually imply subspace clusters, so we can find subspace clusters by

searching frequent patterns. The key of subspace clustering problem is to know which subspace contains cluster. To this end, we define the concept of difference degree of distribution based on the results of OPTICS algorithm. OPTICS algorithm analysis cluster in full space. It does not produce a clustering of a data set explicitly, but instead creates an augmented ordering of the database representing its density based clustering structure. It is a versatile basis for both automatic and interactive cluster analysis. As defined in [10], outlier detection is a method of finding objects that are extremely dissimilar or inconsistent with the remaining data.

2. OVERVIEW OF PROPOSED WORK

High dimensionality is a major contributor to data complexity. Technology makes it possible to automatically and systematically obtain a large amount of measurements. Data observations with thousands of features or more are now common, such as genomic data, financial data, web document data, sensor data, and satellite image data. High dimensionality also causes a problem in the separation of data points. Beyer et al. (1999) showed that the distance between the nearest point and a query point is no different from that of other points when the dimensionality of the space is high enough (10 – 15 dimensions). Therefore, algorithms that are based on the distance measure may no longer be effective in a high -dimensional space. Thus in the Proposed approach clustering is done using subspace and density clustering. Density - based approaches rely on the density of data points for clustering and have the advantage of generating clusters with arbitrary shapes and good scalability. In this approach grid structure is used by partitioning the data space S into a number of non-overlapping rectangular units. These rectangular units are derived by partitioning each attribute into ∂equal-length intervals (where ∂ is an input parameter) in such a way that a unit in space S is the intersection of one interval from each of the attributes. Any k-dimensional subspace of S is the space with the k dimensions drawn from the d attributes, where k >= d.

$A = \{A_1, A_2 \ldots\ldots .A_d\}$ be the set of d attributes of the data set
$S = \{A_1 * A_2 * \ldots\ldots A_d\}$ be the corresponding d-dimensional data space

To discover the clusters which are based on the relative region densities in the subspaces, using a concept where the clusters are regarded as regions whose densities are relatively high as compared to the average region density in a subspace. Thus, to identify the dense units satisfying the density thresholds in subspace clustering is similar to mine the frequent item sets satisfying the minimum support in frequent item sets mining. The dense unit discovery is performed by utilizing a novel data structure DMFP-tree (Density MFP-tree)[6][5], which is constructed on the data set to store the complete information of the dense units. From the DMFP tree, here compute the lower bounds and upper bounds of the unit counts for accelerating the dense unit discovery, and these information's are utilized in a divide and conquer scheme to mine the dense units. Proposed algorithm is having two sub phases. In the first phase only consider those nodes which satisfying the thresholds to discover the dense units, these nodes are called as 'Generation of Inherent Dense unit'. In the second phase consider, for the nodes whose node counts do not exceed to threshold, take the nodes carrying the same one dimensional unit together into consideration in discovering the k-dimensional dense units, these nodes we called as 'Generation of Acquired Dense unit'. T_k denote the density threshold for the subspace cardinality k, and N be the total number of data points. The density threshold T_k is defined as

$$[\, T_k = \alpha\, N \setminus \partial^k]$$

Where α is user defined variable. the term $N \setminus \partial^k$ is the average unit count of the units in a k dimensional subspace. In proposed DMFP tree algorithm, firstly scans the database only once then prune the tree and at last mine the tree. In the last step which is mining phase we got the dense regions which are later used for producing final clusters. Clustering is a fundamental component of real-world problems in nearly every computational discipline, probably in large part due to the human tendency to use categorization as a tool for understanding data. In clustering process execution time and memory space is essential parameter for measure the efficiency of the methods. In the previous method which is based on dense frequent pattern tree (DFP) has taken much time for generating dense regions. By the experimental results it is observed that the proposed algorithm produce more efficient clusters and gives more accurate results. It gives the results having lower memory consumption and faster execution time.

3. PROBLEM STATEMENT

High-dimensional data are ubiquitous in many areas of machine learning, signal/image processing, computer vision, pattern recognition, bioinformatics, etc. Images consist of millions of pixels, videos have thousands of frames, text and web documents are associated with thousands and millions of features**.** In multi-dimensional spaces, it is highly likely that, for any given pair of points within the same cluster, there exist at least a few dimensions on which the points are far apart from each other. As a consequence, distance functions that equally use all input features may not be effective. Furthermore, several clusters may exist in different subspaces, comprised of different combinations of features. In many real world problems, in fact, some points are correlated with respect to a given set of dimensions, and others are correlated with respect to different dimensions. Each dimension could be relevant to at least one of the clusters. There are many Clustering algorithms such as partitioning and hierarchical clustering algorithms but they are not best suited for multi-dimensional space.

Density based subspace clustering is a method to determine the clusters that form on a different subspaces, this method is better in handling multidimensional data than other methods, but these approaches are suffered by density divergence problem, which refers to the phenomenon that the region densities vary in different subspace cardinalities. Objective of the proposed algorithm is to overcome the problem of varying density in different subspace cardinality.
Density clustering is widely used. But there are still some problems existing in clustering algorithms, such as effective,

contradiction between precision and efficiency, sensitive to noise, arbitrary shaped clusters and so on. The most serious problem, however, is efficiency. All these problems actually suffer from huge computational time requirements. Therefore we use a new algorithm for finding the dense point which is more efficient than previous one DFP algorithm known as MFP tree

- To control the increase in CPU cost, which is incurred by the large number of possible items combinations.
- To avoid disk thrashing caused by the lack of main memory (needed for candidate sets or for auxiliary structures). Also, to control the increase in I/O cost, which is incurred by approach such as database projection.

To achieve the better cluster results, algorithm's performance also based on execution time and memory utilization parameters respectively .Motive of the proposed algorithm is to generate efficient clusters by reducing memory consumption and by improving execution time, also compare the experimental results to previous one approach.

4. PROPOSED ALGORITHM:

Input: Transaction database S with d attributes, density threshold value (count value)

Output: Candidate dense frequent set CF

Algorithm

Step 1:- Scan the dataset S for all transaction 1 to N. Calculate the count value of all items present in the transaction data set.

Step 2:- Partitioning the data space S into a number of non-overlapping rectangular units. Based on their cardinality and count value.

Step 3: Store the min tuple manner in such a way that a unit in space S is the intersection of one interval from each of the d attributes.

Step [6]:- Create the root of the MFP-tree T with label "Null".

Step 5:- Insert items from dataset S as accordingly

```
While(N)
  {
  if ( root node has a sub node labeled with A and the label == first )
  A.count = A.count+1 ;
  else
  {
   create node A;
  A.count=1;
  A.pointer point to root_node ;
  }
```

Step 6:- pruning MFP_tree

```
if ( item_threshold< threshold value)
delete it and update pointers
else
move the pointer to next row
```

Step 7:-Mining MFP tree

```
if (mothers_node count – leaf_node count) >= Threshold value
follows the node path as a dense frequent path.
else
discard it
```

Step 8:- Repeat step 5 to 6 till all nodes are not traversed.

Stop

5. RESULT ANALYSIS

In the research work, we have evaluated response time, memory space and accuracy of the proposed algorithm. To measure these performance parameters we have used Thyroid disease dataset, which contains 18,152thyroide diagnoses with four numerical attribute from UCI machine learning repository [4]. As experimental result, the proposed algorithm find clusters from all subspaces with less execution time and less memory space and also with great accuracy. The main purpose of the proposed algorithm is to improve execution time and take less memory.

5.1 PERFORMANCE PARAMETERS

We measure the performance of our algorithm in the form of following parameters:

A. Response Time

Response time or Execution time is the total amount of time which is spend or the total amount of time that an algorithm is take to generate the final results. This is also called the Time complexity of an algorithm. The algorithm that takes less response time is more efficient in nature.

B. Memory Space

Memory space is the total amount of memory taken by an algorithm in the generation of result. Memory is used to store the global and temporary variables in which the temporary and final result is stored. It is the second important parameter on the basis we find out how efficient an algorithm is.

The algorithm that take less memory is more efficient in nature.

5.2 Following are the result for the DMFP Tree:

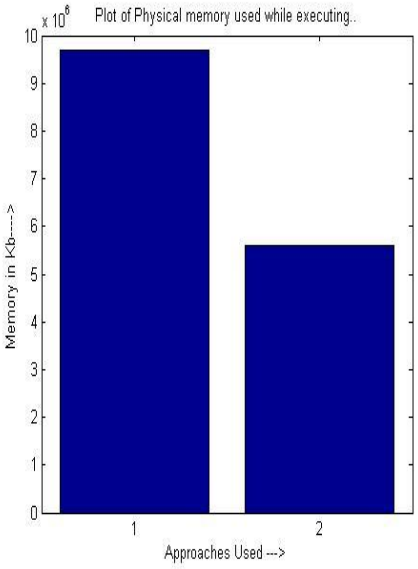

Figure 5.1: Bar graph show the memory utilization difference between previous DFP approach and proposed DMFP approach.

Here '1' is previous approach and '2' is proposed approach.

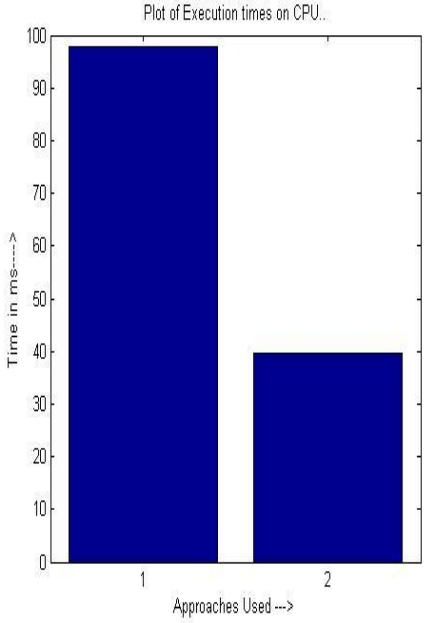

Figure 5.2: Bar graph show the execution time difference between previous DFP approach and proposed DMFP approach.

Here '1' is previous approach and '2' is proposed approach.

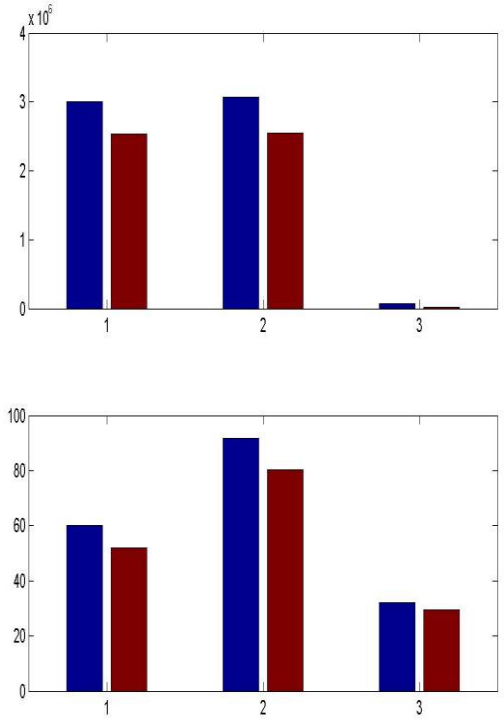

Figure 5.3: Bar graph show the execution time and memory difference between previous DFP approach and proposed DMFP approach at three different time instances.

5.[6] Result comparison:

Approach Used	**Execution time (in ms)**	**Memory Used (in byte)**
Previous approach	56.515691878 97.858044015 30.218010699	6955008 7118848 81920
Proposed approach	50.0322899144 81.2812322250 28.6694904379	3698688 2535424 1122345

Table 5.1: Table shows the comparative results of both approach DFP and DMFP at three different time instances, in term of execution time and memory space respectively.

The Above results are compared with DFP tree Algorithm and which shows how Dense Modified Frequent Pattern Tree Algorithm produces far much better results than dense frequent pattern tree algorithm.

6. CONCLUSION AND FUTURE WORK

In this paper tried to improve the solution quality of clusters which are produced by method called Modified frequent pattern tree. By the experimental result, it is observed that the proposed algorithm gives faster execution time and improvement in memory space utilization for data sets. It is very important to have a data mining algorithm with high efficiency because transaction database usually are very large. Analyzing FP growth algorithm fully, formalized a subspace clustering and proposed a frequent pattern algorithm for it called MFP. MFP algorithm can complete the mining process through scanning the transaction database only one time. MFP algorithm can be applied to any situation where FP growth or Apriori algorithm can be used. Algorithm identifies the dense regions (clusters) in a subspace by discovering the regions which have relatively high densities as compared to the average region density in the subspace. Different thresholds are utilized to discover the clusters in different subspace cardinalities. From the experimental result on the dataset shows that the D- MFP tree algorithm serves a good subspace clustering algorithm which identifies cluster in all subspaces quickly. Finally it is concluded from the results that the algorithm is a significantly outperforms previous works, thus demonstrating its practicability for subspace clustering.

Some future work may also include: For finding Frequent patterns, instead of using tree based structure, graph based structure may also used. Graph based algorithm takes lesser memory space.

REFERENCES

1. Jain & Dubes, (1988) JAIN, ANIL K., & DUBES, RICHARD C. "Algorithms for clustering data". Prentice Hall,1988.
2. Incrementally fast updated frequent pattern trees" , Tzung-Pei Hong a, Chun-Wei Lin b, Yu-Lung Wu b, Department of Electrical Engineering, National University of Kaohsiung, Kaohsiung 811, Taiwan, ROC Department of Information Management, I-Shou University, Kaohsiung 8[6]008, Taiwan, ROC, Expert Systems with Applications 3[6] (2008) 2[6]2[6]–2[6]35 Elesvier.
3. Han, Jiawei, Micheline Kamber, "Data Mining: Concepts and Techniques, 2nd Edition", 2006, p.25-26, Morgan Kaufmann.
4. MacQueen, J. , "Some methods for classification and analysis of multivariate observations" In Proceedings of the Fifth Berkeley Symposium , Vol. 1 , pp. 281 – 297,1967.
5. lun Gao Dept. of Computer Science Shanghai Institute of Technology "Realization of New Association Rule Mining Algorithm**"**Shanghai 200235, China 2007 3rd International Conference on Advanced Computer Theory and Engineering(ICACTE).
6. R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, "Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications," Proc. ACM SIGMOD Int'l Conf. Management of Data, 1998.
7. "Subspace Clustering of High Dimensional Data" Carlotta Domeniconi George Mason University Dimitris Papadopoulos Dimitrios Gunopulos University of California Riverside.Sheng Ma IBM T. J. Watson Research Center.
8. Shao, Yuanhai, Yining Feng, Jing Chen, Naiyang Deng, "Density Clustering Based SVM and Its Application to Polyadenylation Signals", The Third International Symposium on Optimization and Systems Biology (OSB'09), 2009
9. "Sparse Subspace Clustering: Algorithm, Theory, and Applications" Ehsan Elhamifar, Student Member, IEEE, and Ren´e Vidal, Senior Member, IEEE.
10. Journal of computing, volume 2, issue 11, november 2010, issn 2151-9617 "Cluster Evaluation of Density Based Subspace Clustering" Rahmat Widia Sembiring, Jasni Mohamad Zain.