



Generating Recurrent Patterns Using Clique Algorithm

Bipin Nair B J

Lecturer in Department of Computer Science
Amrita Vishwa Vidyapeetham, Mysore Campus,
Karnataka, INDIA

Abstract: Clustering is one of the several machine learning techniques to find out frequent patterns. Most of the clustering methods are not designed for high dimensional data. As the dimensions increase, cluster formation becomes a major challenge in data mining. As a solution to this problem, an algorithm called clique is introduced. Clique is a clustering algorithm which helps to find frequent patterns from high dimensional data. In many real world applications, this clustering technique is being used. This report will look into the sales application, where we can find frequent patterns in product sales. This algorithm can be regarded as one of the ways to improve sales.

Keywords: Clustering, Clique

I. INTRODUCTION

Data mining is the process of extracting data from a huge data store and discovering some interesting patterns. Data mining consists mainly of two types of models: predictive and descriptive. The predictive model is to predict the unknown values on the basis of known data. The descriptive model is used to identify patterns. Clustering is an important technique in data mining. It is a grouping process whereby similar objects are grouped together. So clustering is a task that is used for aggregating similar objects inside a cluster and these objects will be dissimilar to objects belonging to other clusters. Clustering is a descriptive model which is used to find frequent patterns from data. Clustering is also known as unsupervised classification. There are different approaches in clustering, such as the one based on the principle of maximizing similarity between objects in the same class, called intra-class similarity and the one based on minimizing similarity between objects of different classes, known as inter-class similarity.

One of the major problems in clustering is the curse of dimensionality [13]. The most relevant aspect of the curse of dimensionality is that on distance or similarity, and requires that objects in cluster. Here, the problem is to find clusters from a full-dimensional space. The clusters of point may only exist in a subset of higher dimensional spaces and the number of possible subspaces is also exponential in the dimensionality of the space. To overcome this problem CLIQUE (Clustering in Quest) is used [2].

Clique represents a grid-based and density-based approach for clustering in high dimensional data space and it is also represented by dimension growth subspace clustering. It is the first high dimensional space clustering algorithm [7]. The dense units are present in subspaces of increasing dimensions. Dimension growth subspace clustering primarily starts with the single dimensional subspace and then it grows upwards to the higher dimensional ones and discovers dense regions in each subspace. Clique partitions each dimension like a grid structure and determines whether a cell is dense, based on the number of points it contains. A unit is dense if the data points in it are exceeding the threshold value. The clique algorithm finds the crowded region from the multidimensional database and discovers patterns.

The application employs the clique algorithm to interpret sales data. Using this algorithm will make it easy to analyse sales details and suggest patterns that will help businesses to make informed and profitable decisions. All details related to sales will be available in the dataset and the clique algorithm will uncover the hidden frequent patterns in it. According to the frequent patterns, sales can be improved by taking necessary decisions and making better plans for the future.

II. OUTLINE OF THE PAPER

Section 3 explains the clique algorithm with a brief discussion about the algorithm. In section 4 implementation of the clique algorithm is explained along with the pseudo code. Moreover, in section 4 there are subsections like preprocessing, customization, base algorithm, post-processing and model. In section 5 related works are explained. Section 6 explains the experimental result of the application. Section 7 explains the goals and challenges of implementing this algorithm. Section 8 corresponds to the literature survey. Finally, section 9 is the conclusion.

III. CLIQUE ALGORITHM

There are mainly three steps in clique algorithm:

- 1) Identification of subspaces that contains clusters.
- 2) Identification of cluster.
- 3) Generation of minimal description for the clusters.

Brief discussion about the algorithm:

- 1) Identification of subspaces that contains clusters: To identify the subspaces that contain clusters, first need to identify the dense units in the different subspaces. The bottom up iterative process is used to find subspace data points.
- 2) Identification of cluster: The dense unit which has selectivity at least tau will form frequent patterns. The frequent patterns which are similar will form as clusters.
- 3) Generation of minimal description for the clusters: First determines the minimal dense regions over the datasets in the subspaces and each cluster then determines the minimal cover for the maximal regions. Same procedure will follows to find cluster if the dimension increase.

Algorithm 1: **Pseudo code**

- 1) Based on m, the input feature space is split.
- 2) The input is quantized to a particular grid.
- 3) Initialize count (of elements) to 0 across all the grids in the feature space.
- 4) As the input record is read from the file, line by line, the count of grid to which it is mapped is increased by 1.
- 5) For every attribute activate the regions of high density, i.e., store the levels of a particular level which are important.
- 6) Now take two attributes at a time and check for dense regions in the intersection of the dense regions in the individual attribute.
- 7) Repeat the step 6 by adding one dimension with each iteration, and choosing all their possible combinations, until all the dimensions (attributes) in the data set are covered.
- 8) Label all the connected clusters with a label value.

IV. IMPLEMENTING CLIQUE

When the data set has a lot of dimensionality, this leads to wastage of time in looking for clusters in highly sparse regions. To counter this problem, clique uses a very intelligent concept. When there is a cluster, assume k-dimensional subspace, then it can be reasoned that it will have a subspace of k-1 dimension which will also be dense. Now taking this concept in reverse, we start looking for dense regions in each of the k dimension and then look upwards higher and higher dimensions. This is precisely the intuition behind the algorithm.

A. Preprocessing:

Preprocessing is nothing but discretization of the datasets. Discretization is the process of putting values into buckets so that there are a limited number of possible states. In datasets some columns may contain so many values that the algorithm cannot easily identify interesting patterns in the data so discretization is used in it.

B. Customization:

There are two customization parameters:

1. The number of levels into which a particular dimension is divided into, and this step usually taken care in the pre-processing step, or in the data preparation stage itself.
2. The other parameter that can be customized is tau, which can be called as selectivity. If the number of data points within a grid falls below tau then that particular grid is not considered for further processing.

C. Base algorithm

Input: For a dataset of size $m \times d$, each record is of dimension d. The dimension is split into m bins (after preprocessing), if each dimension is split $m_1, m_2, \dots, m_d$, then the total number of bins/grids in the subspace is going to be $m_1 \times m_2 \times \dots \times m_d$. Either a vector of $(m)_d \times 1$ or a single number m is used for splitting all dimensions equally into m bins, giving m^d grids.

Output: Cluster labels.

D. Post-Processing

In this algorithm we are considering only discretized inputs. The input will not have any changes in the algorithm. So there is no specific post processing in this technique because the input is not changed.

E. Model

There is not a model as such for the algorithm. Since this by itself is an unsupervised learning procedure, the cluster labels are the output of this procedure.

V. RELATED WORKS

The working of the Clique algorithm proceeds from a lower to higher dimension. From lower dimension subspaces to higher dimension subspaces dense regions are discovered. There are many clustering

algorithms available in data mining. Algorithms like DBSCAN, OPTICS, and PROCLUS cluster high dimensional data. DBSCAN is a density based clustering method [6]. This algorithm has two parameters: tau, the distance metric that will search the neighbourhood of a point to find other prospective candidates for the same cluster, and min_point, a parameter that specified the algorithm to give importance to a point only if it has other min_points in the vicinity of this particular point. This can be increased to accommodate dense clusters and discard outliers. If the points in a cluster are close to each other and a required number of neighbours are present within that boundary, then the points are classed into a cluster. The cluster will satisfy two properties: all the points within the cluster are mutually density-connected and if a point is density-connected to any other point of the cluster, it is part of the cluster as well. OPTICS is a density based clustering method [3]. If the points in the cluster are close to each other and a required number of neighbours are present within that boundary, then the points are classed into a cluster. In optics there are mainly two parameters, tau and min_point, similar to dbscan. However in optics, min_point can be increased to accommodate dense clusters and leave outliers. The cluster satisfies two properties: all points within the cluster are mutually density-connected and if a point is density-connected to any other point of the cluster, it is part of the cluster as well. In these two algorithms only numerical values are used as inputs. If not, only the numerical attributes of the dataset are sent as input to the algorithm. This is the case because the metric that is used to measure the closeness is the euclidean distance. PROCLUS (PROjected CLUstering) is a typical dimension-reduction subspace clustering method [1]. Proclus algorithm, instead of starting from single-dimensional spaces, starts by finding an initial approximation of the clusters in the high-dimensional attribute space. The proclus algorithm consists of mainly three phases: initialization, iteration, and cluster refinement. Initialization uses a greedy algorithm to select a set of initial medoids that are far apart from each other. This will ensure that each cluster is represented by at least one object in the selected set. Iteration choose a random set of k medoids from the reduced set of medoids and progressively improves the quality of medoids by iteratively replacing the bad medoids in the current set with new points from medoids. In refinement, one more pass over the data will improve the quality of the clustering. Overall it will compute new dimensions for each medoid based on the clusters found and reassigns points to medoids. Finally it will remove all the outliers. In proclus algorithm only numerical values will be taken as input. The attributes can be continuous or discrete valued variables. The algorithm has two parameters: Number of clusters k and Average dimensions I . There is not a model as such for these algorithms. Since this by itself is an unsupervised learning procedure, the cluster labels are the output of this procedure same as clique. In clique it is not necessary to give input as only numerical values; other values also can be given, but only discretized values will be taken as input.

VI. EXPERIMENTAL RESULT

The real world application used in this algorithm is sales data. The competition in this field is very high. Most of them have loss more than profit in business, So they need a better plan to improve their sales. Clique will give a fine solution for this problem. In sales there is some pattern we need to follow, if customer is buying one product frequently then we need to discover which frequent pattern is they follow. But this is possible only when there is a small dataset. When it comes in large datasets finding the frequent pattern is quite difficult. So to deal with the high dimensional data clique algorithm is used. This algorithm will help to find the frequent pattern from the high dimensional dataset easily. The result will depend upon the datasets.

Product	Product_in	Product_i	Product_in	Product	Product_in	Product	Product_in	Product_in	Product_in
ice	maggi	toyota	nails	book	bag	mango	diarymilk	chain	laptop
cone	knorr	i20	hammer	book	box	orange	kitkat	ring	mouse
cone	knorr	toyota	hammer	book	bottle	mango	snickers	chain	cpu
cone	topramen	toyota	hammer	pencil	bottle	mango	diarymilk	bangles	cpu
cream	topramen	ford	nails	ink	box	orange	diarymilk	ring	laptop
ice	topramen	toyota	bolt	ink	bag	banana	kitkat	ring	mouse
cone	knorr	ford	bolt	pencil	bag	banana	snickers	chain	laptop
cream	knorr	i20	nails	ink	box	orange	kitkat	bangles	mouse
cone	maggi	ford	hammer	book	bottle	mango	snickers	bangles	cpu
ice	maggi	toyota	nails	book	bag	mango	diarymilk	chain	laptop

Figure 1: Dataset

Figure 1 will give a brief idea about how the dataset looks like. This is a sample sales dataset, the products are the items which people buy from the shop. The detailed description about the dataset is given below.

Table I will give a brief explanation about the dataset. And Table II will give an overall idea about the attributes present in the dataset.

Table I: Description about the Dataset

Dataset	Sales.csv
No. of attributes	10
No. of rows	500

To explain this application, we are using sales dataset with an extension of .csv file. In this dataset there are total 10 attributes and 500 instances. Attributes are considered as the dimensions.

Table II: Attributes Names and Details

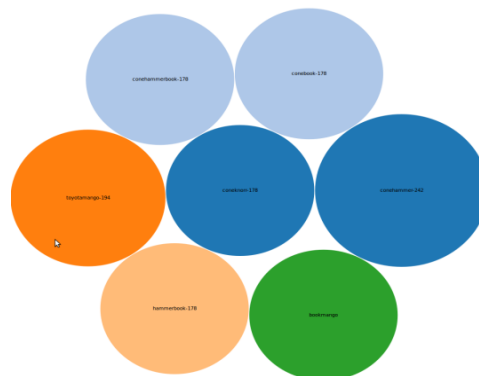
Name of attributes	Products available in each section
Product_in_sectionA	Ice, Cone and Cream
Product_in_sectionB	Maggi, Knorr and Top_ramen
Product_in_sectionC	Toyota, i20 and Ford
Product_in_sectionD	Hammer, Nails and Bolt
Product_in_sectionE	Book, Pencil and Ink
Product_in_sectionF	Bag, Box and Bottle
Product_in_sectionG	Mango, Orange and Banana
Product_in_sectionH	Diarymilk, Kitkat and Snickers
Product_in_sectionI	Chain, Ring and Bangles
Product_in_sectionJ	Laptop, Mouse and CPU

In Table II the attributes of sales dataset is explained. The products which is present each section is explained in the table. In each section different products are available. The algorithm will work in such a way that, only discretized datasets will take as an input. If we consider a high dimensional dataset it will not be easy to identify interesting patterns from that data. Discretization is the process of putting the values into buckets. In this process there are a limited number of possible states will be present. There are several methods to discretize the data. The number of buckets to use for grouping data by setting the value of discretization Bucket Count property. The default bucket count is 5. This is the first customized parameter taken care in the pre-processing step. This algorithm working is depends on the customized parameter tau value or selectivity value. The whole algorithm working is, taking the count attribute wise and removing the count that fall below tau value and taking the comparison of attributes in all the ways and there should not be any repetition in comparison and checking the count again removing the count that falls below the tau value and finding the frequent patterns. All this work is depends on the customized tau value.

Table III: Frequent Pattern

Products in frequent pattern	No. of items purchased	Count of items
Cone, Knorr	2 items	178
Cone, Hammer	2 items	242
Cone, Book	2 items	178
Toyota, Mango	2 items	194
Hammer, Book	2 items	178
Book, Mango	2 items	178
Cone, Hammer, Book	3 items	178

Table III is the frequent patterns from the sales dataset. This will give a clear idea of which products are buying frequently. Using clique we can easily identify which all products is having more marking value. This algorithm will help to reduce the work load of finding patterns from such a large datasets. The result of this application is, combination of cone and knorr purchased together 178 times, combination of cone and hammer purchased together 242 times, combination of cone and book purchased together 178 times, combination of toyota and mango purchased together 194 times, combination of hammer and book purchased together 178 times, combination of book and mango purchased together 178 times and combination of cone, hammer and book purchased together 178 times. By analyzing this result the sales can be improved by keeping more stock on the products which are frequently purchased. Finally we will get five clusters as result based on the customized tau value. This is one application where we can use clique algorithm, likewise there are many areas where we can use clique algorithm for the easy analysis of data.

**Figure 2: Clusters**

The visualization of the clusters is shown in Figure 2. Based on the customized tau value five clusters are formed for this dataset. The visualization is done based on the final frequent patterns. In Table III the final frequent patterns is explained. Each frequent pattern is separately visualized for a better understanding. In Figure 2 each cluster is represented in different colors. The cluster which contains more than one frequent pattern is visualized in same color, so this is considered as one cluster.

The result of this application is represented in the form of a multibar chart is shown in Figure 3. This chart is used to visualize the frequent patterns in each cluster separately. For this multibar chart the result will convert to D3.js. In this multibar chart total five clusters are charted. In chart the frequent patterns are shown like first bar from first group and second bar from the second group is belongs to one cluster. And the second bar from second first group and the second bar from second group belong to one cluster. Remaining bars from first group is showing clusters with single frequent pattern. For the details of the frequent pattern refer Table III. Table III is giving the details about the frequent patterns mentioned in the multibar chart.

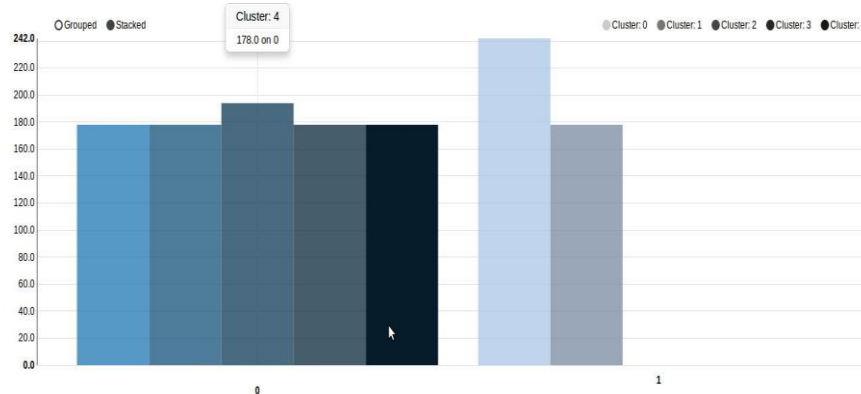


Figure 3: Multibar chart

In multibar chart, it is plotted based on the frequent pattern values. The advantage of this chart is, it will give the details separately about the cluster when we select the cluster labels in the top of the chart. And it also show the count of frequent pattern when we point the cursor on the top of the bar. The benefit of visualizing the result in multibar chart is it will show the details of cluster very clearly rather than visualizing it in the cluster form.

VII. LITERATURE SURVEY

A method for speeding up the step of clique algorithm is described in [12]. Clique doesn't have any model because it is unsupervised learning. Clustering is unsupervised learning of a hidden data concept [5] and the resulting system represents a data concept. Clustering is a division of data into groups of similar objects. To find clusters within subspaces of a dataset, clique was one of the first algorithms proposed to attempt that [11]. Subspace clustering is an extension of feature selection that attempts to find clusters in different subspaces of the same dataset. The subspace clustering requires a search method and evaluation criteria as with feature selection. Clique algorithm combines density and grid based clustering and it uses an apriori technique to find cluster able subspaces.

IX. CONCLUSION

In this paper a brief explanation of data mining is provided along with clustering and the challenges of clustering in high dimensional data. This application has been successfully applied in some areas and needs more study to understand its strength and limitations. There is no reason to expect that this specific clustering algorithm will give clusters for all sorts of high dimensional data. However, almost all high dimensional data clustering problems can be solved by applying this clustering algorithm.

REFERENCES

- [1] Charu C. Aggarwal, Joel L. Wolf, Philip S. Yu, Cecilia Procopiuc, and Jong Soo Park. Fast algorithms for projected clustering. *SIGMOD Rec.*, 28(2):61–72, June 1999.
- [2] Rakesh Agrawal, Johannes Gehrke, Dimitrios Gunopulos, and Prabhakar Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. *SIGMOD Rec.*, 27(2):94–105, June 1998.
- [3] Mihael Ankerst, Markus M. Breunig, Hans-Peter Kriegel, and Jörg Sander. Optics: Ordering points to identify the clustering structure. *SIGMOD Rec.*, 28(2):49–60, June 1999.
- [4] Christian Baumgartner, Claudia Plant, Karin Kailing, Hans-Peter Kriegel, and Peer Kröger. Subspace selection for clustering high- dimensional data. In *ICDM*, pages 11–18, 2004.
- [5] p.Berkhin. A survey of clustering data mining techniques. *Grouping Multidimensional Data*, Pages 25-71, 2006.
- [6] Martin Ester, Hans peter Kriegel, Jörg S, and Xiaowei Xu. A density- based algorithm for discovering clusters in large spatial databases with noise. pages 226–231. AAAI Press, 1996.
- [7] Jiawei Han. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2005.
- [8] Karin Kailing, Hans-Peter Kriegel, and Peer Kröger. Density- connected subspace clustering for high-dimensional data. In *IN: PROC. SDM*. (2004, pages 246–257, 2004.
- [9] Hans-Peter Kriegel, Peer Kröger, Irene Ntoutsis, and Arthur Zimek. Density based subspace clustering over dynamic data. In

- SSDBM, pages 387–404, 2011.
- [10] Hans-Peter Kriegel, Peer Kröger, and Arthur Zimek. Detecting clusters in moderate-to-high dimensional data: subspace clustering, pattern-based clustering, and correlation clustering. *PVLDB*, 1(2):1528–1529, 2008.
 - [11] Lance Parsons, Ehtesham Haque, and Huan Liu. Subspace clustering for high dimensional data: A review. *SIGKDD Explor. Newsl.*, 6(1):90–105, June 2004.
 - [12] Jyoti Pawar and P. R. Rao. An attribute based storage method for speeding up clique algorithm for subspace clustering. In *Proceedings of the 10th International Database Engineering and Applications Symposium, IDEAS '06*, pages 309–310, Washington, DC, USA, 2006. IEEE Computer Society.
 - [13] Michael Steinbach, Levent Ertoz, and Vipin Kumar. The challenges of clustering high-dimensional data. In *New Vistas in Statistical Physics: Applications in Econophysics, Bioinformatics, and Pattern Recognition*. Springer-Verlag, 2003.