Tribhuvan University
Institute of Science and Technology

"A Comparative Analysis of Apriori and Clustering Based Apriori Algorithm"

Dissertation

Submitted To

Central Department of Computer Science & Information Technology
Institute of Science and Technology
Tribhuvan University, Kathmandu, Nepal

In Partial Fulfillment of the Requirements for the Degree of Master of Science in Computer Science & Information Technology

Submitted By

Birendra Singh Saud
Roll No. : 19/071

July 29, 2019

Under Supervision of:

Asst. Prof. Nawaraj Paudel

Tribhuvan University

Institute of Science and Technology

Central Department of Computer Science and Information Technology

Date: 2076/04/12

Student's Declaration

I hereby declare that I am the only author of this work and that no sources other than the listed here have been used in this work.

… … … … … … …

Birendra Singh Saud
M.Sc. CSIT
Central Department of Computer Science
And Information Technology,
Institute of Science and Technology,
Kirtipur, Kathmandu, Nepal

Tribhuvan University

Institute of Science and Technology

Central Department of Computer Science and Information Technology

Date: 2076/04/12

Supervisor's Recommendation

I hereby recommend that the dissertation prepared under my supervision by **Mr. Birendra Singh Saud** entitled "**A Comparative Analysis of Apriori and Clustering Based Apriori Algorithm** " be accepted as in fulfilling partial requirement for the completion of Master's Degree of Science in Computer Science & Information Technology. In my best knowledge this is an original work in computer science.

Asst. Prof. Nawaraj Paudel

(Supervisor)

Central Department of Computer Science and Information Technology,
Institute of Science and Technology,
Kirtipur, Kathmandu, Nepal

Tribhuvan University

Institute of Science and Technology

Central Department of Computer Science and Information Technology

Date: 2076/04/12

LETTER OF APPROVAL

We certify that we have read this dissertation work and in our opinion it is appreciable for the scope and quality as a dissertation in the partial fulfillment of the requirements of Master's Degree of Science in Computer Science & Information Technology.

Evaluation Committee

Asst. Prof. Nawaraj Paudel
Head of Department
Central Department of Computer Science
& Information Technology
Tribhuvan University
Kirtipur

Asst. Prof. Nawaraj Paudel
(Supervisor)
Central Department of Computer Science
& Information Technology
Tribhuvan University
Kirtipur

(External Examiner)

(Internal Examiner)

ACKNOWLEDGEMENT

I am deeply indebted to my supervisor Mr. Nawarj Poudel for his admirable cooperation and inspirable guidance that enabled me to bring this thesis to its present form .I am especially grateful to lecturer Mr. Arjun Singh Saud for his valuable suggestion and encouragement throughout the work.

My deep appreciation goes to lecturers Mr. Bikash Balami, Mr. Jagdish Bhatta, Mrs.Laita Sthaphit, Mr. Sarbin Sayami. It would be unwise if I forget my Lecturers Prof. Dr. Subarna Shakya, Prof. Dr. Sashidhar Ram Joshi, Mr. Dhiraj Kedar Pandey, and Mr. Tej Sahi for their inspiration and support.

I would like to extend my acknowledgement to to my friends Mr. Narayan Dhamala, Mr.Bhupendra Lahur, Mr. Kishor Bhatta and Mr. Dipendra Rai for their encouragement and support they provided me to achieve this work in its present form.

I would like to dedicate this thesis to my parents for all their love, inspiration and support, who deserve credit in my success.

Birendra Singh
Saud

19th July, 2019

ABSTRACT

Freqent itemset is the itemset that occurs frequently in a given set of data items. Nowadays, frequent itemset is most popular in developing different marketing strategy. The size of data increases rapidly and to cope with that data a new method is needed that is capable of handling large volume of data. For that purpose, a hybrid clustering based apriori algorithm is used for generating frequent itemset.

In this research, the comparison of two different frequent itemset generation algorithms (Apriori and Clustering based Apriori) is presented. The main aim of this research is to evaluate the performance of those algorithms based on the parameters like: total number of frequent itemset generated, effect of support percentage on itemset generation and effect of clustering on itemset generation for different dataset with different dimensions. The dataset for this research are chosen such that they are different in size, mainly in terms of number of attributes and number of instances. When comparing the performance it is found that: the clustering based apriori algorithm generates more frequent itemset than the apriori algorithm. In general, by increasing the support percentage both algorithms produces less number of frequent itemset. When the clustering number is balanced then the number of frequent itemset generated is small.

Keywords:

Frequent itemset, Apriori, Clusering based Apriori, Association Rule Mining and K-Means.

Table of Contents

List of Figures

Figure **Page**

List of Tables

Table **Page**

List of Abbreviations

Abbrevations	**Full Form**
ARM	Association Rule Mining
EM	Expectation Maximization
ARWDC	Association Rules mining for Web Document Clustering
MTHFT	Multi-Tire Hashing Frequent Term sets
WEKA	Waikato Environment for Knowledge Analysis
RAM	Random Access Memory
GB	Giga Bytes
HDD	Hard Disk Drive

CHAPTER 1

Introduction

1.1 Introduction to Frequent Itemsets

Data mining plays an important role in the discovery of interesting patterns and knowledge from the huge amount of data that may be stored in relational database, data warehouse, transactional database or any other information repositories. Data mining is also known as knowledge mining from data, knowledge extraction, data/ pattern analysis, data archeology, and data degrading [7]. The data mining can perform different functions like: Classification and prediction, cluster analysis, outlier analysis, trend and evolution analysis, frequent pattern finding and association rule generation etc.

Frequent itemsets are those itemsets that occurs frequently in a dataset. Association rule mining is the process of finding frequent patterns, associations, correlations, or casual structure among set of items or objects in transactional database, relational database and other information repositories. An association rule mining consists of two sub processes: First method is referred as finding frequent item set and second process is known as generating association rule. An association rule can be in the form of implication, A → B, where A is called antecedent and B is called consequent. Both A and B are frequent item sets in a transactional database and A Ç B=∅ (where Ç=Intersection).The rule A → B can be interpreted as "If item set A occurs in a transaction T, then item set B will also be there in the same transaction".Association rule mining is agreat resolution designed for substitute rulemining, since its objects to realize entirelyrules in data and as a result is able to arrange for a whole depiction of associations in a huge dataset[16].There are different association rule mining algorithm like: apriori algorithm, Fp-tree growth algorithm etc. The apriori algorithm is simple and easy to implement but this is favorable only for small database.

Another important function of data mining is clustering. Clustering is concerned with grouping things together. The data items with in the same cluster are similar to each other but they are dissimilar to the data that belongs to other cluster. The different clustering algorithm are K-means, K- Mediod etc.The K-means algorithm is simple, fast for low dimensional data and can find pure sub cluster if large number of cluster is specified.

A method is needed that combines both the concept of Apriori algorithm and K-means clustering algorithm that provides benefits to each other system. A hybrid method, clustering based apriori algorithm, combines the concept of clustering and apriori algorithm. Apriori algorithm cannot handle the large amount of data and K-Means algorithm is fast for low dimensional data. So, the concept of Clustering (K-means) can be used that generate different clusters based on the user provided number of cluster value. Each resultant cluster can be supplied as an input to the apriori algorithm which helps to reduce the size of database and makes apriori algorithm more scalable. Finally, each cluster resultant frequent itemsets are combined together to produce the total set of frequent items generated by the association rule mining.To combine the frequent itemsets take union from all frequent itemsets from each partition and these item sets form the global candidate frequent itemsets for the entire database.

1.2 Association Rule Mining (ARM)

Association rule mining is the process of extracting associations among set of items or products in a transactional database [4]. To measure the rule interestingness in ARM Support and confidence are used [11].

Support: Support is the probability of item or item sets in the given transactional database.

Support(X) = n(X) / n where n is the total number of transactions in the database and n(X) is the number of transactions that contains the item set X .

Therefore, **support (X=>Y) = support (XUY)**

Confidence: confidence is conditional probability, for an association rule X=>Y confidence is defined as:

Confidence(X=>Y) = support (XUY) / support(X).

The association rules are considered to be interesting if they satisfy both minimum support and minimum confidence criteria. These criteria are specified by users or experts. The rules having support and confidence greater than or equal to the user specified criteria are extracted by association rule mining task. There are different association rule mining algorithms like: a priori, FP-tree growth algorithm, Border algorithm etc.Apriori algorithm plays an important role in deriving frequent item sets and then extracting association rules out of it. In ARM, the number of generated rules grows exponentially [12] with the number of items or products.

Association rule mining consists of two sub-processes:- finding frequent item sets and generating association rules from those item sets.

Frequent itemset: Frequent itemset is a set of items whose support is greater than the user specified minimum support. An itemset X in A (i.e.,X is a subset of A) is said to be a frequent item set in T with respect to σ, if $support(X)_T \geq \sigma$.

Association rule: An association rule is an implication or if-then-rule which is supported by data and can be represented in the form X→Y. An association rule must satisfy user-set minimum support (min_sup) and minimum confidence (min_conf).The rule X→Y is called a strong association rule if support ≥ min_sup and confidence ≥ min_conf.

The overall performance of the association rule mining is determined by generating frequent itemsets because after discovering frequent itemsets the association rules can be generated in a straight forward manner [15].

1.2.1 Apriori algorithm

Apriori algorithm finds all combinations of items that have transaction support above minimum support and call those combinations as frequent item sets. After that use the frequent item sets to generate the desired rules.

Apriori Algorithm

The apriori algorithm works as follows

Step 1: Initially scan each cluster dataset to get frequent 1- item set.

Step 2: Generate length (K+1) candidate item sets from length K cluster item set.

Step 3: Test the candidate against each cluster.

Step 4: Terminate when no frequent or candidate set can be generated.

Pseudo code:

C_k: Candidate itemset of size k

L_k : frequent itemset of size k

L_1 = {frequent items};

for(k = 1; L_k !=∅; k++) do begin

C_{k+1} = candidates generated from L_k;

For each transaction *t* in database do

Increment the count of all candidates in C_{k+1}that are contained in *t*

L_{k+1} = candidates in C_{k+1} with min_support

End

Return$\cup_k L_k$;

1.3 Clustering

Data clustering is a 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. Clustering partitions the data set into groups based on the similarity, and then assigns labels to each group according to its characteristics. Therefore clustering is some times referred as unsupervised classificati-on. Clustering is different from classification because clustering is concerned with grouping things together whereas classification is concerned with placing things where they belong. There are different clustering algorithms like: K means, K mediod etc.

1.3.1 K-Means Algorithm

The *k*-means algorithm is partitioning method of clustering where each cluster's center is represented by the mean value of the objects in the cluster.

K-Means Clustering Algorithm

The K-Means clustering algorithm works as follows

- Choose K as the number of clusters to be determined.
- Choose K objects randomly as the initial cluster centers.
- Repeat
  - Assign each object to their closest cluster.
  - Compute new clusters, calculate mean points.
- Until

- No change in cluster entities **OR**
- No object change in it's clusters.

Pseudo code:

Randomly choose k points for forming number clusters

  For every point assigned to the centroid

    Calculate the distance between the centroid and the point

    Assign the point to the cluster with the lowest distance

  For every cluster calculate the mean of the points in the cluster

    Assign the centroid to the mean

While any point has changed cluster assignment

Repeat until convergence

1.4 Problem Statement

Data mining plays an important role in finding frequent itemsets which is useful for generating association rules. The association rules helps to generate correlation among the set of transactional data items. A large number of algorithms exist for generating association rules. The traditional association rule mining algorithm like: Apriori cannot handle the large amount of data.

So, the concept of Clustering (K-means) can be used that generate different clusters based on the user provided number of cluster value. The clustering partitions the large amount of transaction into a smaller group of each cluster. Each resultant cluster can be supplied as an input to the apriori algorithm. Finally, the resultant frequent itemsets are combined together to generate the global frequent itemset. Thus, the larger amount of transaction can be reduced and supplied as an input to the apriori algorithm, which helps to reduce the size of database and makes apriori algorithm more scalable.

1.5 Objective of Thesis

The main objective of this research is

- To make the traditional Apriori algorithm more scalable by applying the concept of clustering (K-means) on Apriori algorithm.
- To compare the performance of Apriori algorithm and Clustering based Apriori algorithm based on parameters like: Number of frequent itemsets generated, Effect of support percentage on frequent itemset generation and effect of clustering number on itemset generation.

1.6 Thesis Organization

The flow of thesis goes on this manner.

Chapter 1 consists of introduction, problem statement and objectives.

Chapter 2 describes about the literature review of the related work by different authors.

Chapter 3 includes the overview of the methodology of apriori and clustering based apriori for frequent itemset generation.

Chapter 4 contains the implementation overview of the apriori and clustering based apriori for frequent itemset generation in java platform along with the empirical analysis of different performance parameters of methodology.

Finally chapter 5 concludes with the main theme of the work and future recommendation.

CHAPTER 2

2. Literature Review

The author [1] used cluster based association rule mining for heart attack prediction. The author presented the new method for generating association rule mining which is based on sequence number generation and clusters the resultant transactional database for heart attack prediction. The different attributes affecting heart attack are considered and transform that medical data into binary form and applied the proposed method on the reduced binary transactional database. The entire medical database is partitioned into an equal number of size to make each cluster value balanced. Each cluster is loaded into memory and calculated the frequent item set.

The author [6] performed clustering based association rule mining to discover user behavioral pattern in web log mining. Association rule mining produces a large number of association rules which makes difficult for the user to analyze the result. So, clustering based association rules are generated to solve the problem. The fuzzy algorithm is used for clustering the users with similar surfing patterns. Then apriori algorithm is applied to discover interesting relationship between the clustered users in a large database. For experiment, the UCI database was used. The experimental result shows that each of the clusters contains some common characteristics and apply the association rules result more frequent pattern specifically.

The author [9] performed a comparative study of association rule mining techniques and predictive mining approaches for association classification. The author integrated association rule mining and classification to make competitive classifier model. The number of association classification rules can be minimized by implementing predictive mining approaches. The association classification system begins with processing of training data by discovering frequent item sets followed by generation of association classification rules. The rules are then filtered to obtain the significant rules to build the classifier. Finally, the classifier will be applied to test dataset which is already in preprocessed format. The proposed association classification method builds more accurate classifiers then the traditional classifier.

The author [3] presented clustering based multi-objective rule mining using genetic algorithm. Multi-objective genetic algorithm is a new approach used for association rule mining which optimizes the support counting phase by clustering the database. The author tested the proposed algorithm on different data sets and founds that the speedup highly depends on the distribution of transactions in the cluster tables.

The author [10] proposed a novel approach for hierarchical document clustering, Fuzzy association rule mining algorithm to generate candidate cluster. The hierarchical clustering is used because it does not suffer from high dimensionality, scalability, accuracy and meaningful cluster labels. To generate, candidate cluster, fuzzy association rule mining is used. The membership functions are used to convert term document matrix into fuzzy set.

In paper [2], a novel hybrid algorithm for recommendation based on clustering and association rule mining is proposed. Clustering is used to form the user clusters based on the similarity. Once the similar users form a cluster we use these clusters to find items strongly associated with other. This information is used while recommendi-ng items to new test users. In this approach, use collaborative filtering technique to form the user cluster. Each cluster contains a set of users who are similar to each other and dissimilar to the users in the other clusters so each cluster is considered and form transactional database for that cluster. The extended FP-tree algorithm is used to find frequent item sets and that frequent itemsets is used to form association rules and recommend the item. The hybrid recommendation system which combines clustering and association rule mining can be used to address one of the most challenging issues of recommendation systems- Cold-start Problem.

The author [14] considered an efficient approach for text clustering based on the frequent item sets. A renowned method, called Apriori algorithm is used for mining the frequent item sets. The mined frequent item sets are then used for obtaining the partition, where the do cuments are initially clustered without overlapping.Furthermore, the resultant clusters are effe -ctively obtained by grouping the documents within the partition by means of derived keywords. Finally, for experimentation, any of the dataset can be used and thus the obtained o utputs can ensure that the performance of the proposed approach has been improved effective -ly.

The Negm Noha et.al [13] investigates the performance of document clustering approach based on association rule mining on the large data set routers, examine the efficiency and scalability of the algorithm. Based on Association Rules mining, an efficient approach for Web Document Clustering (ARWDC) has been devised. An efficient Multi-Tire Hashing frequent termsets algorithm (MTHFT) has been used to improve the efficiency of mining association rules by targeting improvement in mining of frequent termset. Then, the documents are initially partitioned based on association rules.Since a document usually contains more than one frequent term set, the same document may appear in multiple initial partitions, i.e., initial partitions are overlapping. After making partitions disjoint, the documents are grouped within the partition using descriptive keywords, the resultant clusters are obtained effectively. The performance of algorithm is evaluated with the help of evaluation measures such as, Precision, Recall and F-measure compared to the existing clustering algorithms like Bisecting K-means and FIHC. The experimental results show that the efficiency, scalability and accuracy of the ARWDC approach has been improved significantly.

The author [8] proposed a simple method for mining cluster in large various data sets which describes information about products purchased by customers. The result of experiment allows one to find informative clustering. The restriction is made on the type of data set where the objects are the binary attributes in vector form. The effectiveness of the algorithm is when quantitative as well as descriptive attributes are used.

The paper [17] proposed a new hybrid algorithm for music recommendation system based on clustering and association rule mining. At first, the clustering is done based on the similarity of users. For similarity calculation the users listening history is taken. Secondly, find the items which are strongly associated with each other by using association rule mining and finally the strong association rules are generated to recommend the items. The performance of the recommender system is evaluated on the basis of parameters like: precision, recall and F-measure. This hybrid approach solves the cold-start problem of recommender system.

The author [5] used apriori and clustering algorithm in WEKA tools to to mine data set of traffic accidents. WEKA tools were used to analysing traffic dataset, which is composed of 946 instances and 8 attributes. Apriori algorithm and EM cluster were implemented for traffic dataset to discover the factors, which causes accidents. The apriori algorithm result shows that most of the incidents happened during the day; the most common types of accidents were a collision with another vehicle. The highest accidents appeared in highway the drivers who caused the accidents were non-Saudis. Most accidents happened between two or more vehicles. The EM algorithm results also shows that the most of the incidents happen during the day; collision with other vehicle in highway. The highest accidents appeared in highway is by the drivers who caused the accidents were both side non-Saudis. The apriori algorithm outperforms the EM algorithm because of its effectiveness for finding frequent itemsets. Thus, the apriori algorithm is better than the EM clustering algorithm.

CHAPTER 3

3. Research Methodology

3.1 Data Collection

The input data for the experiment are collected from online machine repository. The data set have been chosen such that they are differing in size, mainly in terms of number of instances and number of attributes. The collected data types are of Numeric type.

3.1.1 Dataset 1

The first data set is small iris data set. The data set contains 5 attributes (4 numeric and 1 predictive class attribute) including class attribute and 150 numbers of instances. The attributes are sepal length in cm, sepal length in cm, petal length in cm, petal width in cm, and class attribute which may be iris setosa, iris versicolor and iris virginica. But in this research only 100 instances have been taken and only two class attributes: iris-setosa (representing 1) and iris-versicolor (representing 0) have been considered.

3.1.2 Dataset 2

The second data set is comparatively large Pima Indian diabetes data set. The data set contains 9 attributes including class attribute and the number of instances in the dataset is 768. The attributes are number of times pregnant, plasma glucose concentration of 2 hours in an oral glucose tolerance test, diastolaic blood pressure (mm Hg), triceps skin fold thickness (mm), 2- hour serum insulin (mu U/ml) , body mass index (weight in kg/(height in m)^2), Diabetes pedigree function, age (in years) and class (either yes or no). But in this research only 268 instances have been taken all of which are diabetes present class and only 8 attributes (all numeric) are considered.

3.2 Tools Used

To implement this thesis, the following hardware and software configurations will be used

Hardware Requirement

System Type: Personal Computers

Processor Type: Pentium or Intel

RAM: 2 GB or Higher

HDD: 500 GB or Higher

Software Requirement

Operating System: Windows XP/ Windows 7

Developed In: Java Programming Language

Front-end: Eclipse

3.3 Data Preprocessing

The numeric data in the data set can be converted into binary form. In the first iris data set, the given numeric attributes can be converted into binary form by calculating mean value of all corresponding attribute instance value and comparing the mean value with each instance value. If the instance value is greater than or equal to mean value then binary value 1 is set otherwise binary value 0 is set. For class attribute, the presence of attribute iris-setosa is represented by binary 1 and the presence of attribute iris-versicolor is represented by binary 0 as only two classes are taken out of three.

Similarly in the second diabetes data set, only the data that shows patients have diabetes are considered. The given numeric attributes value can be converted into binary form by calculating mean value of all corresponding attribute instance value and then converted those value into ceiling or flooring (if value is greater than or equal to 5 after decimal then ceiling is done otherwise flooring is done) and comparing the converted mean value with each instance value. If the instance value is greater than or equal to mean value then binary value 1 is set otherwise binary value 0 is set.

3.4 Clustering Based Apriori Algoirthm

Algorithm

Step 1: Transform the transactional data into its binary form.

Step 2: Apply K-means clustering algorithm on binary data.

- Choose K as the number of clusters to be determined.
- Choose K objects randomly as the initial cluster centers.
- Repeat
  - Assign each object to their closest cluster.
  - Compute new clusters, calculate mean points.

Until

  - No change in cluster entities

OR

  - No object change in it's clusters.

Step 3: Apply Association rule mining algorithm (Apriori) to each resultant K clusters obtain from Step 2.

- Initially scan each cluster dataset to get frequent 1- item set.
- Generate length (K+1) candidate item sets from length K cluster item set.
- Test the candidate against each cluster.
- Terminate when no frequent or candidate set can be generated.

Step 4: Finally generate frequent itemsets from each cluster and combine the resultant frequent itemsets from each cluster to obtain the final itemsets that are frequent.

3.5 Comparison Criteria

The comparative analysis or the result is made on the basis of the following criteria.

1) Total number of frequent item sets generation

Here, the number of frequent itemset generated by the apriori algorithm and the clustering based apriori algorithm obtained by counting in total is analyzed.

2) Effect of support percentage on itemsets generation

Here, how the changes in support percentage affect for generating item set for different cluster value are analyzed.

3) Effect of clustering number on itemsets generation

Here, how changes in clustering number affect the number of frequent itemset generation is analyzed.

CHAPTER 4

4. RESULT, ANALYSIS AND COMPARISONS

4.1 Result, Analysis and Comparison

In this study, the two algorithms mentioned in chapter 3 are compared for two different dimensional data set mentions in chapter 3.1 which is compared based on the parameters like: the total number of frequent itemset generated, the effect of support percentage on itemset generation and the effect of clustering number on itemset generation and the result is obtained.

4.1.1 Comparison results of total number of frequent item set generated for dataset1

The table 4-1 provides the output for comparison of two algorithms apriori and clustering based apriori for generating frequent itemset over data set iris. For different clustering value and for support percentage 10, the clustering based apriori algorithm produced frequent item set is presented and for support percentage 10, the apriori algorithm produced frequent itemset is listed as apriori algorithm doesn't make use of clustering value.

Table 4-1: Number of frequent itemset generated for support=10% for iris dataset

Clustering Number	Apriori	Clustering based apriori
2	11	17
3	11	19
4	11	19

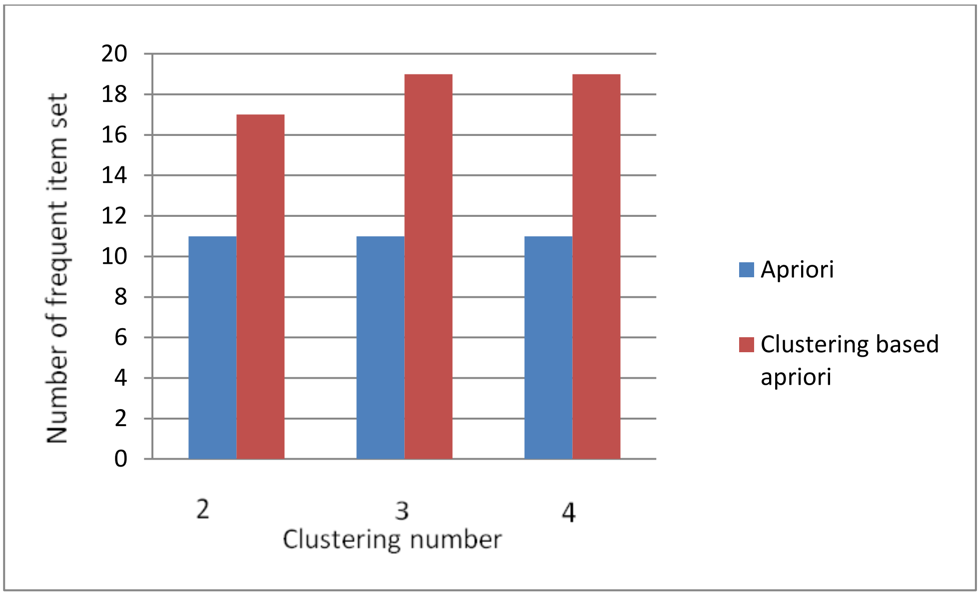

Fig 4-1: Graph of table 4-1.

Based on the fig 4-1, it is clear that the clustering based apriori algorithm generates more frequent item sets than the apriori algorithm. When the support percentage is 10, the clustering based apriori algorithm generates 17, 19 and 19 frequent itemsets in total for clustering number 2, 3 and 4 respectively while the apriori algorithm generates only 11 itemsets.

The table 4-2 provides the output for comparison of two algorithms apriori and clustering based apriori for generating frequent itemset over data set iris. For different clustering value and for support percentage 20, the clustering based apriori algorithm produced frequent item set is presented and for same support percentage, the apriori algorithm produced frequent itemset is listed as apriori algorithm doesn't make use of clustering value.

Table 4-2: Number of frequent itemset generated for support=20% for iris dataset

Clustering Number	Apriori	Clustering based apriori
2	10	11
3	10	19
4	10	13

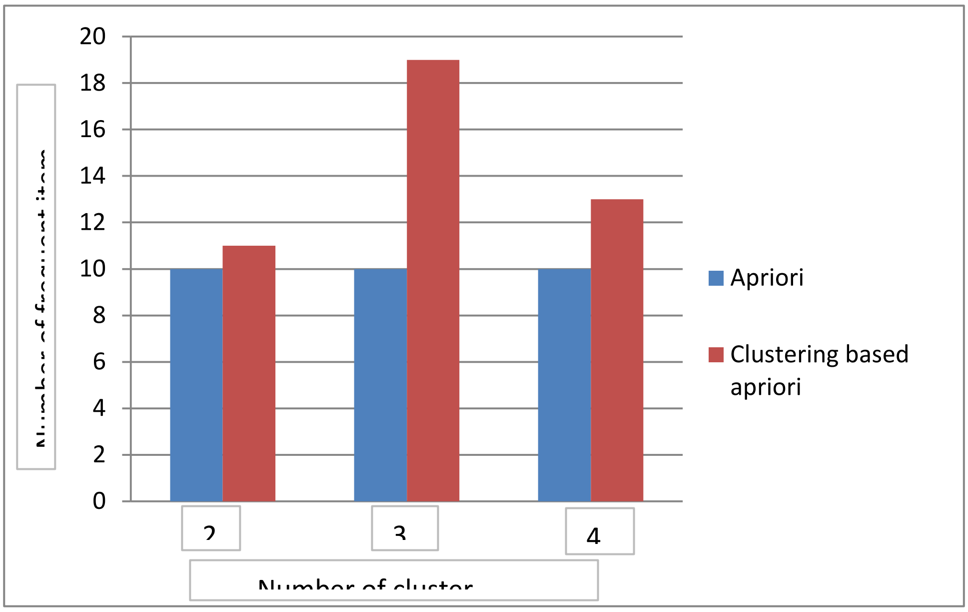

Fig 4-2: Graph of table 4-2.

Based on the fig 4-2, it is clear that the clustering based apriori algorithm generates more frequent item sets than the apriori algorithm. When the support percentage is 20, the clustering based apriori algorithm generates 11, 19 and 13 frequent itemsets in total for clustering number 2, 3 and 4 respectively while the apriori algorithm generates only 10 itemsets.

Similarly, the table 4-3 provides the output for comparison of two algorithms apriori and clustering based apriori for generating frequent itemset over data set iris. For different clustering value and for support percentage 30, the clustering based apriori algorithm produced frequent item set is presented and for same support percentage, the apriori algorithm produced frequent itemset is listed as apriori algorithm doesn't make use of clustering value.

Table 4-3: Number of frequent itemset generated for support=30% for iris dataset

Clustering Number	Apriori	Clustering based apriori
2	10	10
3	10	17
4	10	10

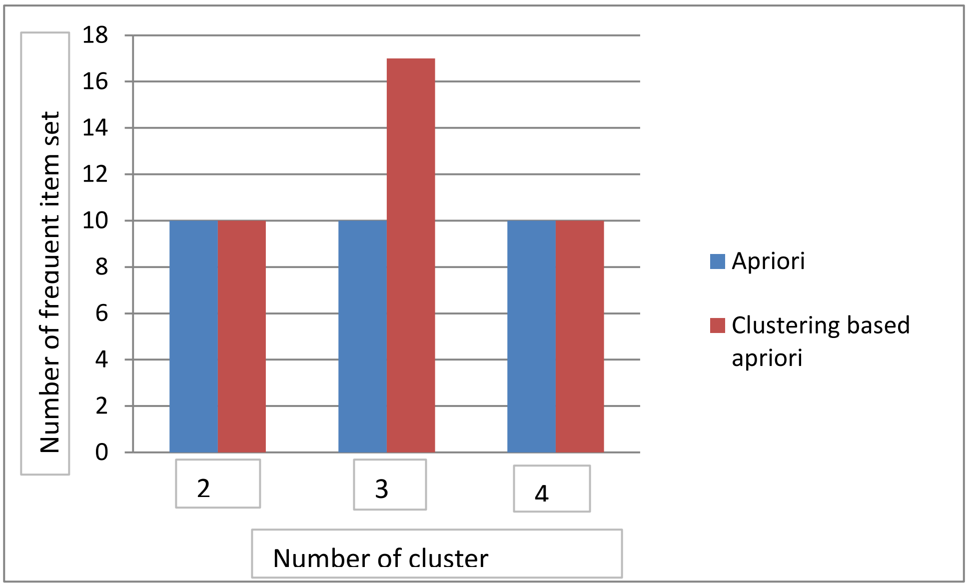

Fig 4-3: Graph of table 4-3.

Based on the fig 4-3, it is clear that the clustering based apriori algorithm generates greater than or equal to the frequent item sets produced by the apriori algorithm. When the support percentage is 30, the clustering based apriori algorithm generates 10 frequent itemsets in total for clustering number 2 and 4 and 17 frequent item set for clustering number 3. The apriori algorithm generates only 10 itemsets.

The table 4-4 provides the output for comparison of two algorithms apriori and clustering based apriori for generating frequent itemset over data set iris. For different clustering value and for support percentage 40, the clustering based apriori algorithm produced frequent item set is presented and for same support percentage, the apriori algorithm produced frequent itemset is listed as apriori algorithm doesn't make use of clustering value.

Table 4-4: Number of frequent itemset generated for support=40% for iris dataset

Clustering Number	Apriori	Clustering based apriori
2	10	10
3	10	11
4	10	10

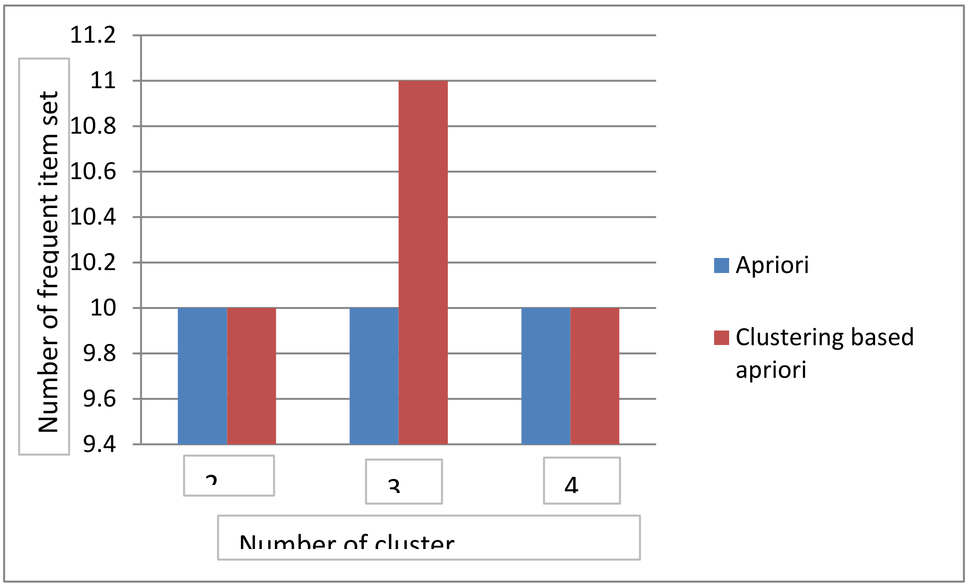

Fig 4-4: Graph of table 4-4.

Based on the fig 4-4, it is clear that the clustering based apriori algorithm generates greater than or equal to the frequent item sets produced by the apriori algorithm. When the support percentage is 40, the clustering based apriori algorithm generates 10 frequent itemsets in total for clustering number 2 and 4 and 11 frequent item set for clustering number 3. The apriori algorithm generates only 10 itemsets.

4.1.2 Comparison results of total number of frequent item set generated for dataset2`

The table 4-5 provides the output for comparison of two algorithms apriori and clustering based apriori for generating frequent itemset over data set pima Indian diabetes. For different clustering value and for support percentage 10, the clustering based apriori algorithm produced frequent item set is presented and for same support percentage, the apriori algorithm produced frequent itemset is listed as apriori algorithm doesn't make use of clustering value.

Table 4-5: Number of frequent itemset generated for support=10% for diabetes dataset

Clustering Number	Apriori	Clustering based apriori
2	107	187
3	107	255
4	107	235

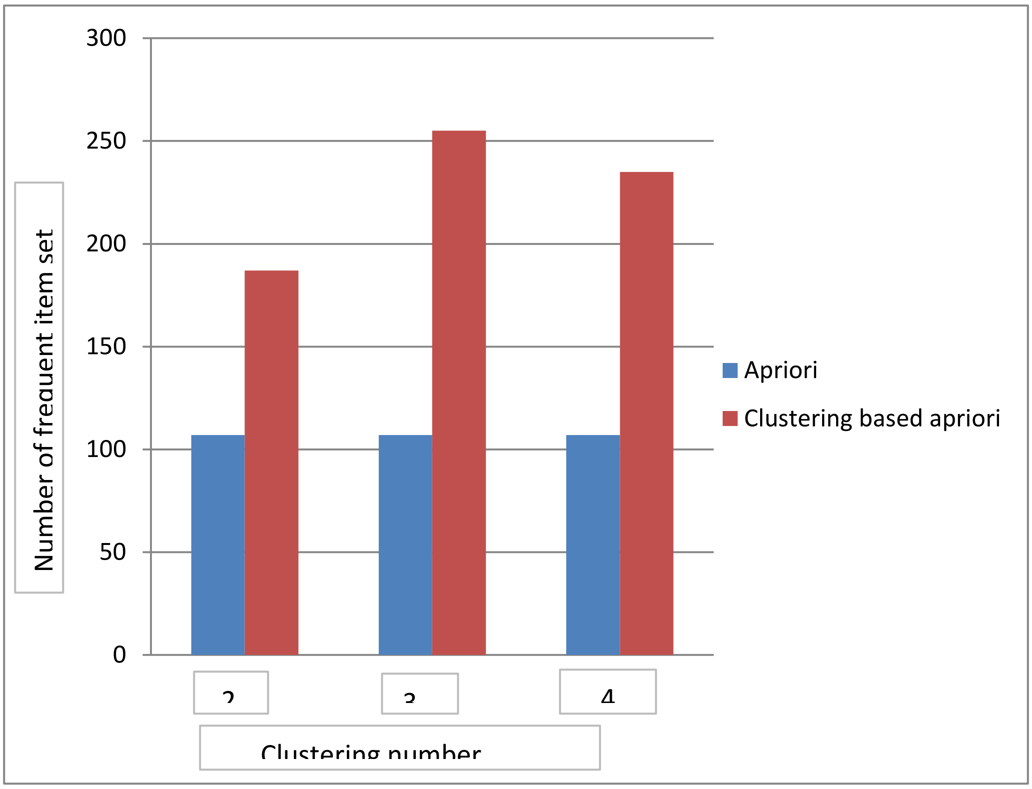

Fig 4-5: Graph of table 4-5.

Based on the fig 4-5, it is clear that the clustering based apriori algorithm generates more frequent item sets than the apriori algorithm. When the support percentage is 10, the clustering based apriori algorithm generates 187, 255 and 233 frequent itemsets in total for clustering number 2, 3 and 4 respectively while the apriori algorithm generates only 107 itemsets.

Similarly, the table 4-6 provides the output for comparison of two algorithms apriori and clustering based apriori for generating frequent itemset over data set pima Indian diabetes. For different clustering value and for support percentage 20, the clustering based apriori algorithm produced frequent item set is presented and for same support percentage, the apriori algorithm produced frequent itemset is listed.

Table 4-6: Number of frequent itemset generated for support=20% for diabetes dataset

Clustering Number	Apriori	Clustering based apriori
2	36	74
3	36	147
4	36	128

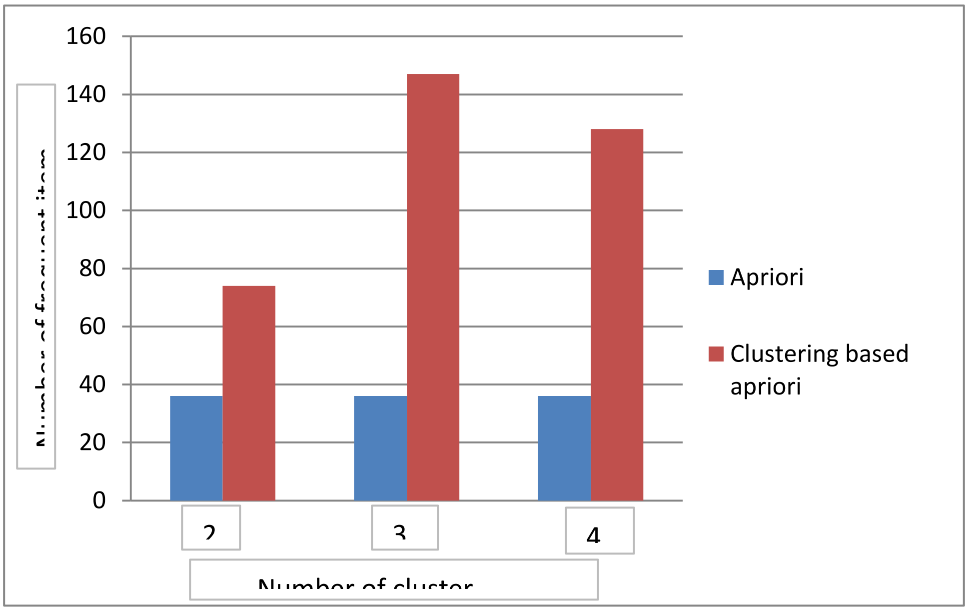

Fig 4-6: Graph of table 4-6.

Based on the fig 4-6, it is clear that the clustering based apriori algorithm generates more frequent item sets than the apriori algorithm. When the support percentage is 20, the clustering based apriori algorithm generates 74, 147 and 128 frequent itemsets in total for clustering number 2, 3 and 4 respectively while the apriori algorithm generates only 36 itemsets.

Similarly, the table 4-7 provides the output for comparison of two algorithms apriori and clustering based apriori for generating frequent itemset over data set Pima Indian diabetes. For different clustering value and for support percentage 30, the clustering based apriori algorithm produced frequent item set is presented and for same support percentage, the apriori algorithm produced frequent itemset is listed.

Table 4-7: Number of frequent itemset generated for support=30% for diabetes dataset

Clustering Number	Apriori	Clustering based apriori
2	17	38
3	17	85
4	17	63

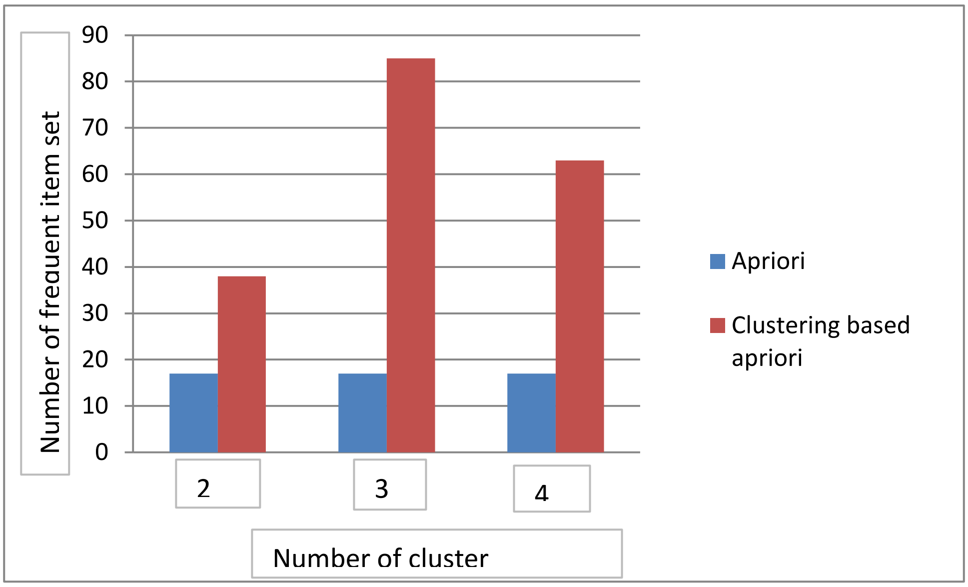

Fig4-7: Graph of table 4-7.

Based on the fig 4-7, it is clear that the clustering based apriori algorithm generates more frequent item sets than the apriori algorithm. When the support percentage is 30, the clustering based apriori algorithm generates 38, 85 and 63 frequent itemsets in total for clustering number 2, 3 and 4 respectively while the apriori algorithm generates only 17 itemsets.

The table 4-8 provides the output for comparison of two algorithms apriori and clustering based apriori for generating frequent itemset over data set Pima Indian diabetes. For different clustering value and for support percentage 40, the clustering based apriori algorithm produced frequent item set is presented and for same support percentage, the apriori algorithm produced frequent itemset is listed.

Table 4-8: Number of frequent itemset generated for support=40% for diabetes dataset

Clustering Number	Apriori	Clustering based apriori
2	8	19
3	8	41
4	8	42

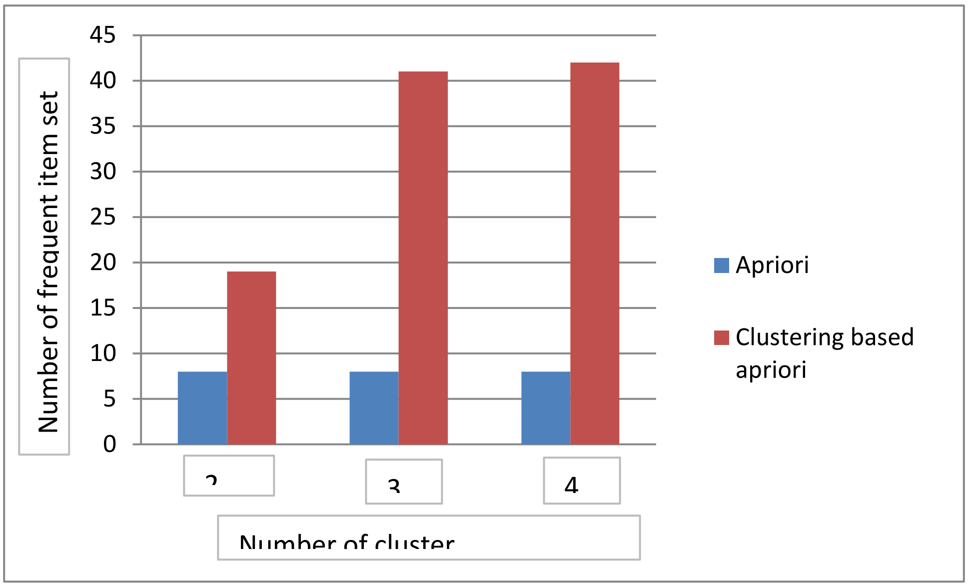

Fig 4-8: Graph of table 4-8.

Based on the fig 4-8, it is clear that the clustering based apriori algorithm generates more frequent item sets than the apriori algorithm. When the support percentage is 40, the clustering based apriori algorithm generates 19, 41 and 42 frequent itemsets in total for clustering number 2, 3 and 4 respectively while the apriori algorithm generates only 8 itemsets.

4.1.3 Comparison result of effect of support percentage on itemset generation for dataset 1

The table 4-9 listed the output for comparison of two algorithms apriori and clustering based apriori for analyzing effect of support percentage on frequent itemset generation over Iris dataset. For clustering value 2 and for different support percentage value ranging from 10 to 40 with difference of 10, the clustering based apriori algorithm produced frequent item set is presented and for same clustering value, the apriori algorithm produced frequent itemset is listed.

Table 4-9: Effect of support percentage on item set generation for k=2 for iris dataset.

min support	Apriori	Clustering based apriori
10	11	17
20	10	11
30	10	10
40	10	10

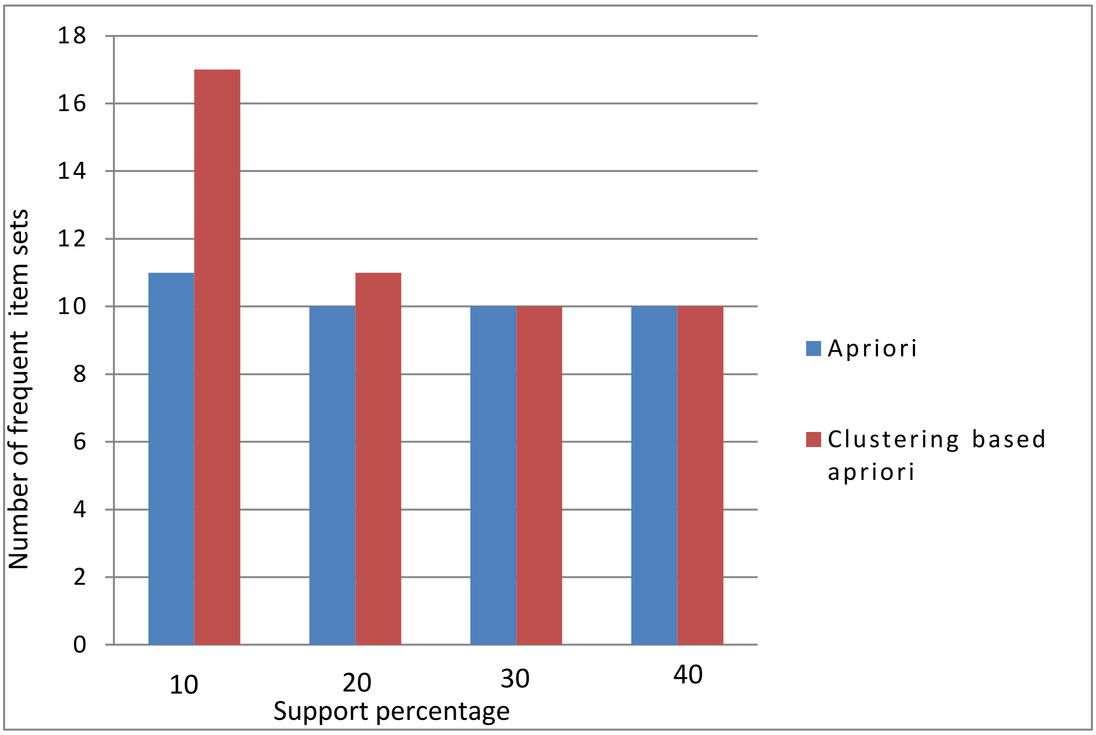

Fig4-9: Graph of table 4-9.

Based on the fig 4-9, it is clear that when the support percentage is increased the number of frequent itemset produced is reduced. When the support percentage is 30 and 40, the clustering based apriori algorithm generates exactly the same number of frequent item set produced by apriori algorithm.

The table 4-10 listed the output for comparison of two algorithms apriori and clustering based apriori for analyzing effect of support percentage on frequent itemset generation over Iris dataset. For clustering value 3 and for different support percentage value ranging from 10 to 40 with difference of 10, the clustering based apriori algorithm produced frequent item set is presented and for same clustering value, the apriori algorithm produced frequent itemset is listed.

Table 4-10: Effect of support percentage on item set generation for k=3 for iris dataset.

min support	Apriori	Clustering based apriori
10	11	19
20	10	19
30	10	17
40	10	11

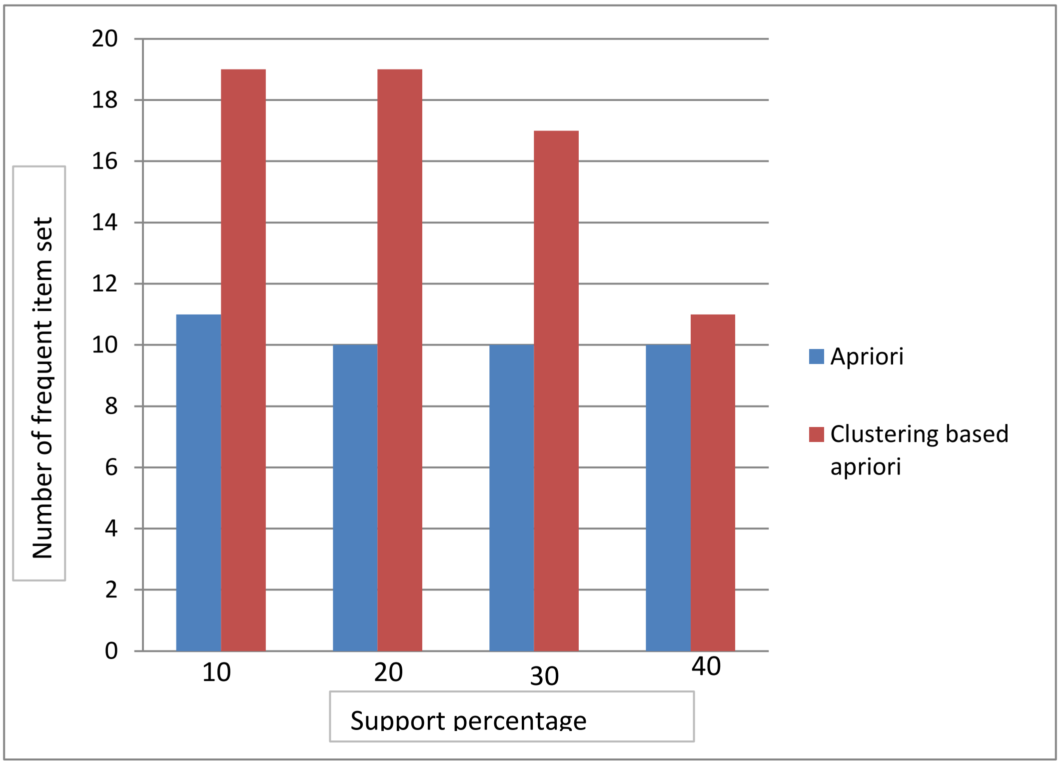

Fig 4-10: Graph of table 4-10.

Based on the fig 4-10, it is clear that when the support percentage is increased the number of frequent itemset produced is reduced. When the support percentage is 10 and 20, the clustering based apriori algorithm generates exactly the same number of frequent item set. Similarly, when the support percentage is 30 and 40 the apriori algorithm also generates the same number of frequent itemsets.

The table 4-11 listed the output for comparison of two algorithms apriori and clustering based apriori for analyzing effect of support percentage on frequent itemset generation over Iris dataset. For clustering value 4 and for different support percentage value ranging from 10 to 40 with difference of 10, the clustering based apriori algorithm produced frequent item set is presented and for same clustering value, the apriori algorithm produced frequent itemset is listed.

Table 4-11: Effect of support percentage on item set generation for k=4 for iris dataset.

min support	Apriori	Clustering based apriori
10	11	19
20	10	13
30	10	10
40	10	10

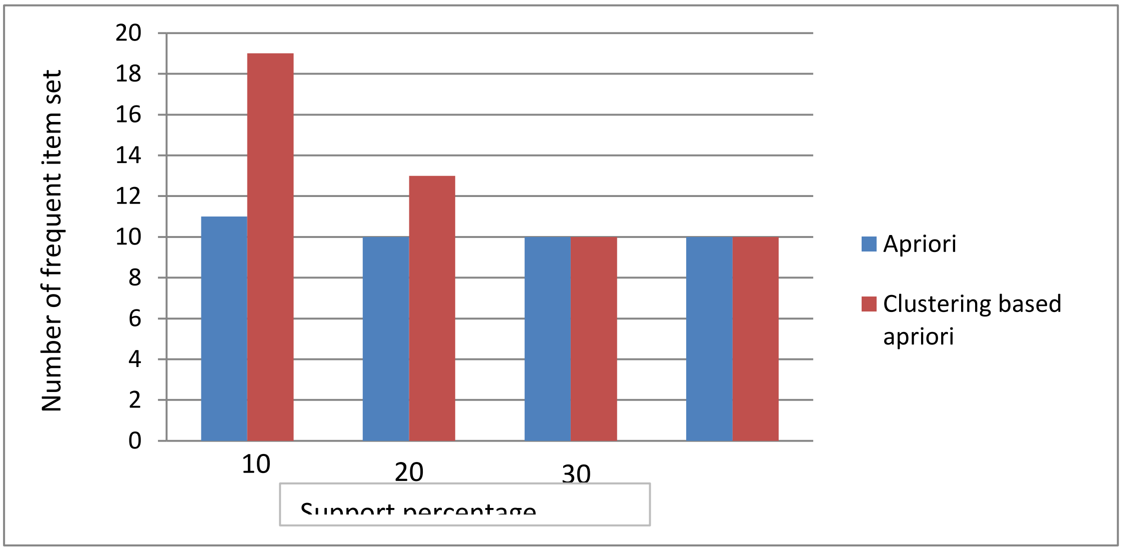

Fig 4-11: Graph of table 4-11.

Based on the fig 4-11, it is clear that in general when the support percentage is increased the number of frequent itemset produced is reduced. When the support percentage is 30 and 40, the clustering based apriori algorithm generates exactly the same number of frequent item set produced by the apriori algorithm.

4.1.4 Comparison result of effect of support percentage on itemset generation for dataset 2

In table 4-12, the output for comparison of two algorithms apriori and clustering based apriori for analyzing effect of support percentage on frequent itemset generation over Pima Indian diabetes dataset is listed. For clustering value 2 and for different support percentage value ranging from 10 to 40 with difference of 10, the clustering based apriori algorithm produced frequent item set is presented and for same clustering value, the apriori algorithm produced frequent itemset is listed.

Table 4-12: Effect of support percentage on item set generation for k=2 for diabetes dataset.

min support	Apriori	Clustering based apriori
10	107	187
20	36	74
30	17	38
40	8	19

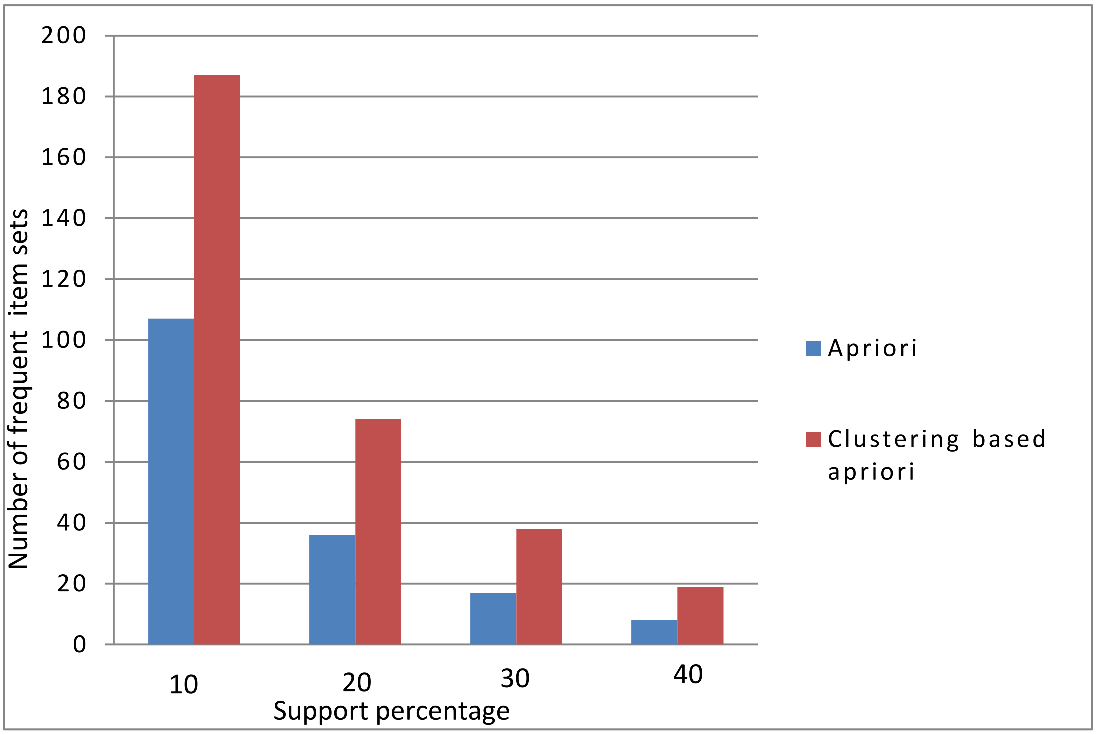

Fig 4-12: Graph of table 4-12.

Based on the fig 4-12, it is clear that when the support percentage is increased the number of frequent item set produced is reduced. Both algorithms reduce the number of frequent item set vastly when support percentage is increased by 10.

In table 4-13, the output for comparison of two algorithms apriori and clustering based apriori for analyzing effect of support percentage on frequent itemset generation over Pima Indian diabetes dataset is listed. For clustering value 3 and for different support percentage value ranging from 10 to 40 with difference of 10, the clustering based apriori algorithm produced frequent item set is presented and for same clustering value, the apriori algorithm produced frequent itemset is listed.

Table 4-13: Effect of support percentage on item set generation for k=3 for diabetes dataset.

min support	Apriori	Clustering based apriori
10	107	255
20	36	147
30	17	85
40	8	41

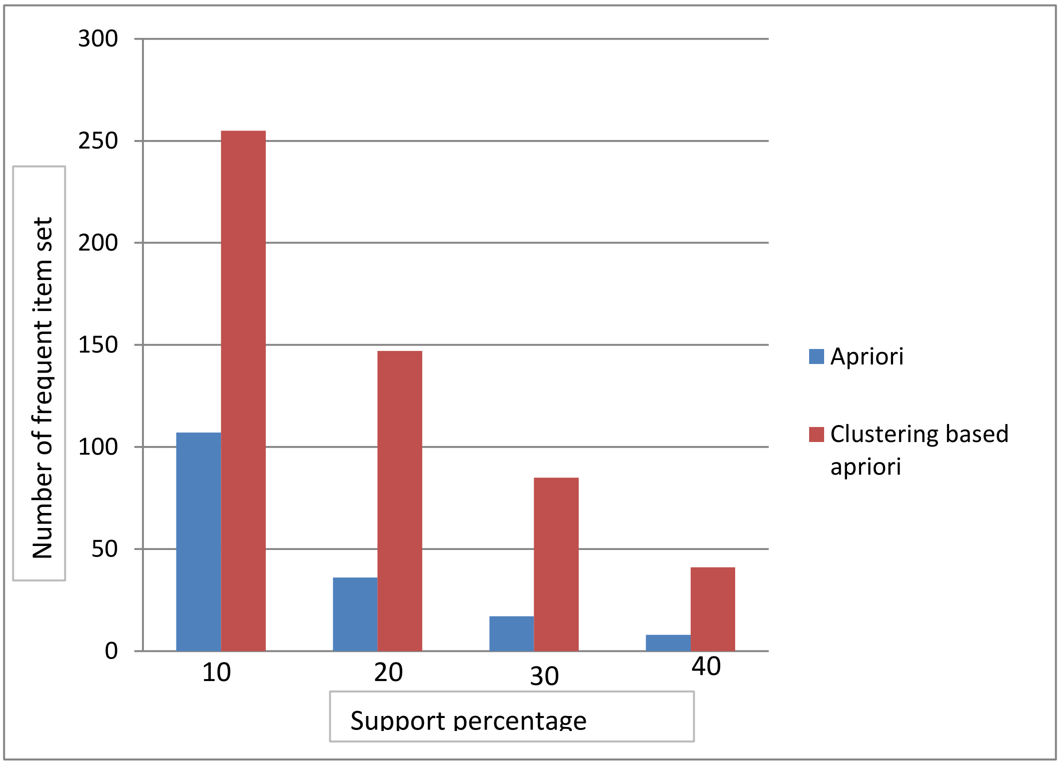

Fig 4-13: Graph of table 4-13.

Based on the fig 4-13, it is clear that when the support percentage is increased the number of frequent item set produced is reduced. Both algorithms reduce the number of frequent item set vastly when support percentage is increased by 10.

In table 4-14, the output for comparison of two algorithms apriori and clustering based apriori for analyzing effect of support percentage on frequent itemset generation over Pima Indian diabetes dataset is listed. For clustering value 4 and for different support percentage value ranging from 10 to 40 with difference of 10, the clustering based apriori algorithm produced frequent item set is presented and for same clustering value, the apriori algorithm produced frequent itemset is listed.

Table 4-14: Effect of support percentage on item set generation for k=4 for diabetes dataset.

min support	Apriori	Clustering based apriori
10	107	235
20	36	128
30	17	63
40	8	42

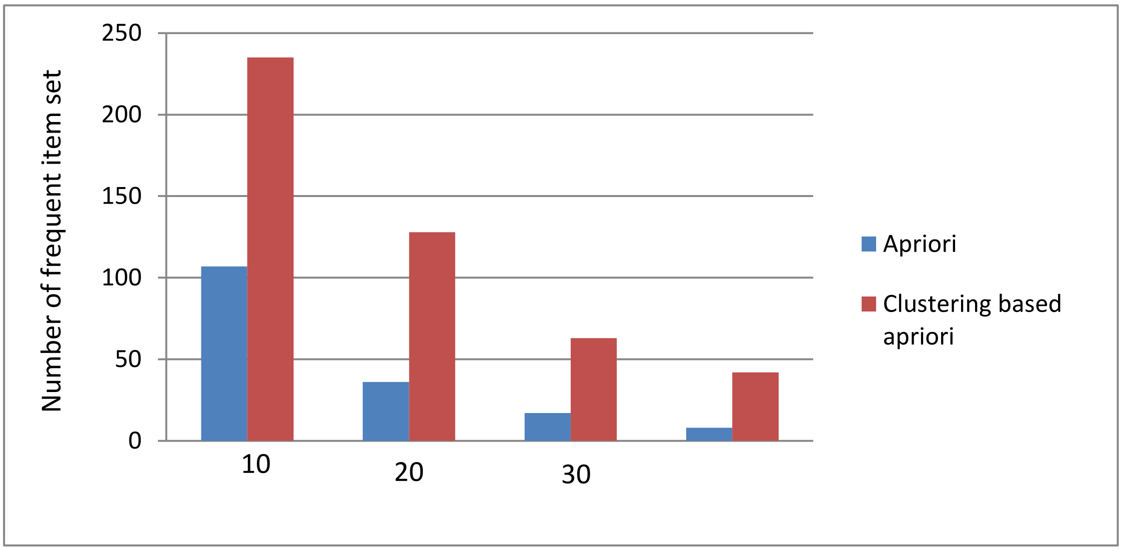

Fig 4-14: Graph of table 4-14.

Based on the fig 4-14, it is clear that when the support percentage is increased the number of frequent item set produced is reduced.

4.1.5 Effect of clustering number on item set generation for both data sets

The clustering number is useful only on clustering based apriori algorithm for frequent item set generation. So clustering number has no impact on general apriori algorithm. Based on the table and graph presented in chapter 4, it is clear that when the clustering number is even, the clustering based apriori algorithm generates less number of frequent itemset. On the other hand, when the clustering number is odd the clustering based apriori algorithm generates more number of frequent itemset.

CHAPTER 5

5. CONCLUSION AND FUTURE RECOMMENDATION

5.1 Conclusion

In this research, the comparative analysis of frequent itemset generation algorithms (apriori algorithm and clustering based apriori algorithm) using various measure parameters like: number of frequent itemset generated, the effect of support percentage on itemset generation and the effect of clustering on itemset generations over the two different dataset with different dimension and size are evaluated. From the result analysis, the total number of frequent itemset generated by the clustering based apriori alogorithm is greater than or equal to the frequent itemset generated by the apriori algorithm. In general, when the support percentage is increased the number of frequent itemset generated by both algorithm is reduced. When the clustering value is of balanced partitioning this will produce the less number of frequent itemset. The clustering based apriori algorithm partitions the large dataset into a group of small clusters in which each clusters are given as an input to clustering based apriori algorithm so it is highly scalable.

5.2 Future Recommendation

The importance of generating frequent item set will continue to grow along with large volume of datasets. The efficiency of the clustering based apriori algorithm for generating frequent itemset can be increased by reducing the multiple scans of the same data in apriori algorithm. The effectiveness of the algorithm can be evaluated based on other different performance parameters.This approach can also be used in different recommender system to recommend items.

REERENCES

1. Deekshatulu B.L.,Chandra Priti and et.al, Cluster Based Association Rule Mining for Heart Attack Prediction,Journal of Theoretical and Applied Information Technology,Vol. 32 No.2, October 2011.

2. Deshmukh P.R and JaswantePrajakta,Review on Text Clustering Based on Frequent Itemset, Journal of Computer Science and Information Technology, Vol. 3, Issue. 4, April 2014.

3. Ghoshsounmadip ,SushantaBiswas and et.al , Mining frequent itemsets using Genetic Algorithm, International journal of Artificial Intelligence and Applications,Vol.1,No:4, October 2014.

4. Hadian Ali, Nasiri Mahdi and et.al.,Clustering Based Multi-Objective Rule Mining using Genetic Algorithm, International Journal of Digital Content Technology and its Applications ,Volume 4, Number 1, February 2010.

5. Hamed Ali Mohammed, Nafie Ali Mohammed, Usage Apriori and Clustering Algorithms in WEKA tools to Mining Dataset of Traffic Accidents, Journal of Information and Communication, Volume:02, No:03, 2018.

6. JaiswalAshish and JanweNitin, Fuzzy Association Rule Mining Algorithm to Generate Candidate Cluster: An Approach to Hierarchical Document Clustering, International Journal of Computer Science Issues, Vol. 9, Issue 2, No 3, March 2012.

7. KamberMicheline, Han Jiawei and et.al, Data Mining Concepts and Techniques,Third Edition.

8. Kosters Walter A., Marchiori Elena and et.al, Mining Clusters with Association Rules, Leiden Institute of Advanced Computer Science, Leiden Univerisity, Netherlands.

9. Lad Abhimanyu, Saggar Manish, Kumar AgrawalAshis, Optimization of Association Rule Mining using Improved Genetic Algorithms,IEEE International Conference on Systems, Man and Cybernatics,2004.

10. LawranceR.andSerin J.,Clustering based Association Rule Mining to discover user behavioural pattern in Web Log Mining, International Journal of Pure and Applied Mathematics, Volume 119 No. 17, 2018.

11. M.Ramesh Kumar and Dr. K. lyakutti, Genetic Algorithms for the prioritization of Association Rules ,IJCA Special Issue on Artificial Intelligence Techniques- Novel Apporaches and Practical Applications,AIT,PP:35-38, 2011.

12. Mittal Kavita and AggarwalGaurav, A Comparative Study of Association Rule Mining Techniques and Predictive Mining Approaches for Association Classification , International Journal of AdvancedResearch in Computer Science,Volume 8, No. 9, November-December 2017.`

13. Negm Noha, Elkafrawy Passent and et.al, Investigate the Performance of Document Clustering Approach Based on Association Rule Mining, International Journal of Advanced Computer Science and Application, Volume:04, No:8, 2013.

14. P. Prithiviraj and R. Porkodi,A Comparative Analysis of Association Rule Mining Algorithms in Data Mining: A Study,American Journal of Computer Science and Engineering Survey, ISSN:2349 – 7238.

15. Rajurkar Archana M., Deshpande Deepa S. and et.al, Mammogram Classification using Assocition Rule Mining, Department of Computer Science and Engineering, India.

16. Reddy Sunitha, Swathi V. and et.al, A Novel Association Rule Mining and Clustering Based Hybrid Method for Music Recommendation System, International Journal of Research in Engineering and Technology , Volume: 03 Special Issue: 05 ,May 2014.

17. Swathi V. , Reddy M. Sunita and et.al, A Novel Associaiton Rule Mining and Clustering Based Hybrid Method for Music Recommendation System, International Journal of Research in Engineering and Technology, Volume: 03, Issue:05, May 2014.

APPENDIX

1. Sample dataset1 some portion (iris dataset)

5.1, 3.5, 1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
4.7,3.2,1.3,0.2,Iris-setosa
4.6,3.1,1.5,0.2,Iris-setosa
5.0,3.6,1.4,0.2,Iris-setosa
5.4,3.9,1.7,0.4,Iris-setosa
4.6,3.4,1.4,0.3,Iris-setosa
5.0,3.4,1.5,0.2,Iris-setosa
4.4,2.9,1.4,0.2,Iris-setosa
4.9,3.1,1.5,0.1,Iris-setosa
5.4,3.7,1.5,0.2,Iris-setosa
4.8,3.4,1.6,0.2,Iris-setosa
4.8,3.0,1.4,0.1,Iris-setosa
4.3,3.0,1.1,0.1,Iris-setosa
5.8,4.0,1.2,0.2,Iris-setosa

2. Normalized dataset1 some portion

1,0,1,1,0
1,1,1,1,0
1,0,1,1,0
1,0,1,1,0
1,0,1,1,0
1,0,1,1,0
1,0,1,1,0
1,0,1,1,0
1,0,1,1,0
1,0,1,1,0
1,0,1,1,0
1,0,1,1,0
1,0,1,1,0
1,0,1,1,0
1,0,1,1,0

3. **Sample dataset2 some portion (Diabetes dataset)**

6	148	72	35	0	33.6	0.627	50
8	183	64	0	0	23.3	0.672	32
0	137	40	35	168	43.1	2.288	33
3	78	50	32	88	31	0.248	26
2	197	70	45	543	30.5	0.158	53
8	125	96	0	0	0	0.232	54
10	168	74	0	0	38	0.537	34
1	189	60	23	846	30.1	0.398	59
5	166	72	19	175	25.8	0.587	51
7	100	0	0	0	30	0.484	32
0	118	84	47	230	45.8	0.551	31
7	107	74	0	0	29.6	0.254	31

1	115	70	30	96	34.6	0.529	32
7	196	90	0	0	39.8	0.451	41

4. Normalized dataset2 some portion

1 1 1 1 0 0 1 1

1 1 0 0 0 0 1 0

0 0 0 1 1 1 1 0

0 0 0 1 0 0 0 0

0 1 0 1 1 0 0 1

1 0 1 0 0 0 0 1

1 1 1 0 0 1 0 0

0 1 0 1 1 0 0 1

1 1 1 0 1 0 0 1

1 0 0 0 0 0 0 0

0 0 1 1 1 1 0 0

1 0 1 0 0 0 0 0

0 0 0 1 0 0 0 0

1 1 1 0 0 1 0 1

5. Some Output Screen shoots

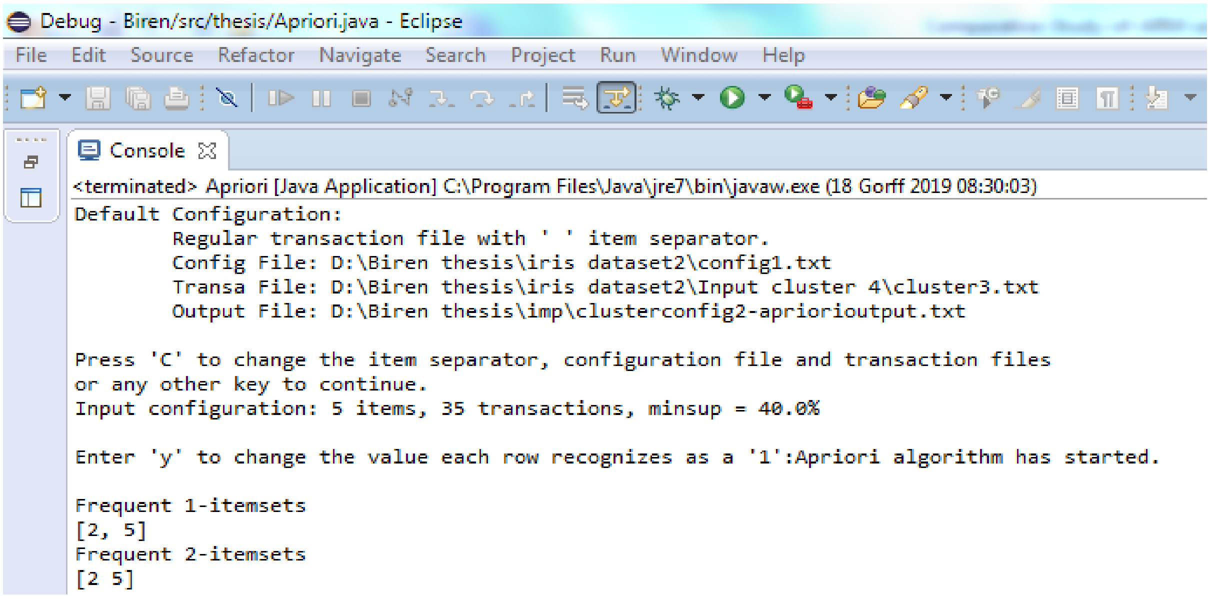

Fig: Screen shot showing frequent item set generation from cluster3.txt input

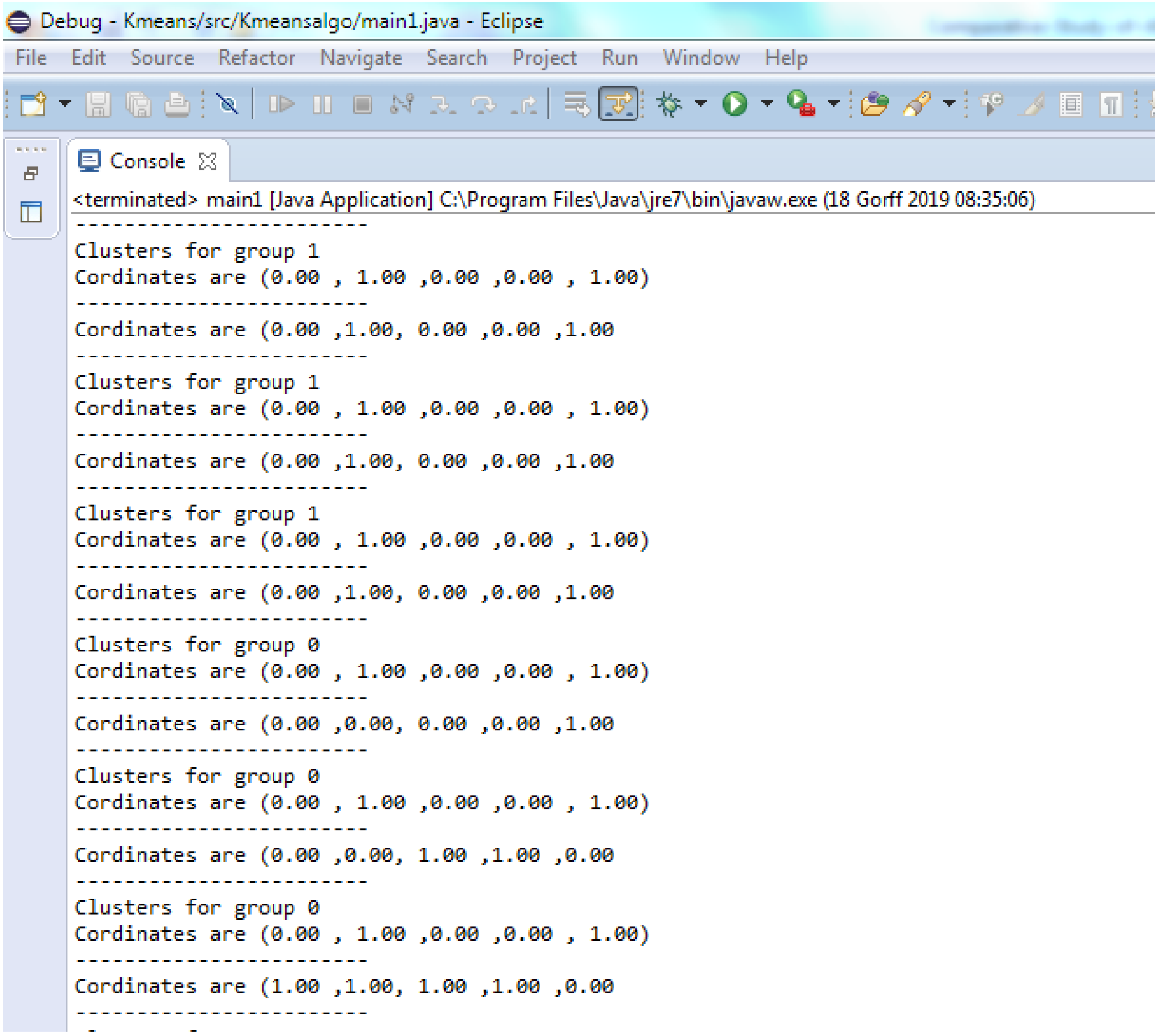

Fig: Screen shot showing cluster result for cluster number 2.