Performance Analysis of K-Means and Bisecting K-Means Algorithms in Weblog Data

K.Abirami
Research Scholar, School of Computing Sciences, Vels University, Chennai, India.

Dr. P.Mayilvahanan
Professor - Dept. of MCA, School of Computing Sciences, Vels University, Chennai, India.

Abstract – Web mining is used to discover interest patterns which can be applied to many real world problems like refining web sites, better understanding the user behavior, product approval etc. Data mining software is one of a number of analytical tools for analyzing data. In this paper we are studying the various clustering algorithms for segmentation model. The basic idea of clustering is to define the similarity between the distance, the distance that represents the data between the data to measure the similarity of the size of the data are classified, until all the data gathering is completed. Cluster analysis or clustering is the task of assigning a set of objects into groups (called clusters) so that the objects in the same cluster are more similar to each other than to those in other clusters. Our main aim to show the comparison of the different- different clustering algorithms of segmentation model and find out which algorithm will be most suitable for the users.

Index Terms – Web Mning, K-means algorithms, Bisecting K-means algorithm, Clustering methods.

1. INTRODUCTION

Data mining is a new kind of data processing technology and efficiently extracts useful information [1]. Data mining it is an Extraction of hidden, analytical information from large databases .It is also called as Knowledge Discovery from Databases .It perform an Identification and assessment of hidden patterns in database [2]. Web mining can be classified into three areas: 1) Web content mining: refers to discovery of useful information from web page contents i.e. text, multimedia data like images, audio, video etc. 2) Web structure mining: it refers to analyzing, discovering and modeling link structure of web pages and/or web site to generate structural. 3) Web usage mining deals with understanding user behavior while interacting with web site, by using various log files to extract knowledge from them.

One of the most important tasks of Web Usage Mining is web user clustering which forms groups of users presenting having common welfares and behavior by analyzing the data collected in the web servers [3]. The K-means is most popular algorithm for clustering and well known for its simplicity and low time complexity [4]. However, it has some major drawbacks like quality of the resulting clusters heavily depends on the selection of initial centroids, clusters produced are of varying sizes, hence unbalanced and may also lead to empty clusters. Bisecting k-means is modification over basic k-means algorithm. As Bisecting k-means is based on k-means, it keeps the merits of k-means and also has some advantages over k-means. Clustering is the task of assigning a set of objects into groups (called clusters) so that the objects in the same cluster are more similar (in some sense or another) to each other than to those in other clusters [1].

2. WEB USAGE MINING PROCESS

The main aim of the innovation system is to find web user clusters from web server log files [5]. These discovered clusters show the characteristics of the underlying data distribution. Clustering is useful in characterizing user groups based on patterns, categorizing web documents that have similar functionalities.

This method allows for the collected works of Web log information for Web pages. This usage data provides the paths leading to accessed Web pages [6]. This information is often gathered automatically into access logs via the Web server

Web Usage Mining is a four-step process. The first step is data collection, the second step is data pre-processing, the third step is pattern discovery and the last step is pattern analysis.

2.1. Preprocessing

The pre-processing stage involves cleaning of the click stream data and the data is partitioned into a set of user transactions with their respective visits to the web site. “Consists of converting the usage, content, and structure information contained in the various available data sources into the data abstractions necessary for pattern discovery”[7].

Performs a series of processing of web log file covering data cleaning, user identification, session identification, path completion and transaction identification.

2.2. Data Cleaning

It is the process of removing irrelevant items such as jpeg, gif, sound files and references due to spider navigation to improve the quality of analysis. User Identification is the process of identifying users by using IPaddress and user agent fields of log entries [8]. A user session is considered to be all of the page accesses that occur during a single visit to a Web site.

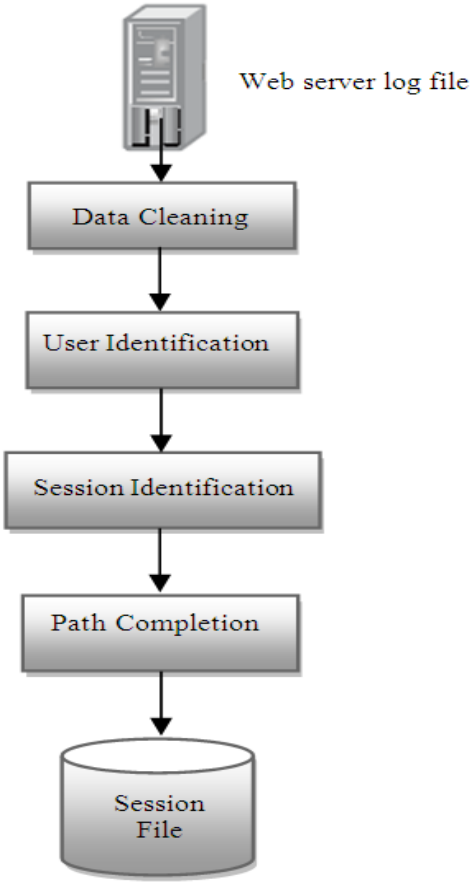

Fig.1. Weblog data preprocessing Phase

2.3. Pattern Discovery

It is the process of removing irrelevant items such as jpeg, gif, sound files and references due to spider navigation to improve the quality of analysis. User Identification is the process of identifying users by using IPaddress and user agent fields of log entries. A user session is considered to be all of the page accesses that occur during a single visit to a Web site.

2.4. Pattern Analysis

Pattern Analysis is the final stage of WUM (Web Usage Mining), which involves the validation and interpretation of the mined pattern.

- Validation: to eliminate the irrelevant rules or patterns and to extract the interesting rules or patterns from the output of the pattern discovery process.
- Interpretation: the output of mining algorithms is mainly in mathematic form and not suitable for direct human interpretations.

2.5. User Identification

User identification phase will be processed after preprocessing. This step should be to identify unique users. If you use firewalls and proxy servers will be complex to record this information [9]. In EPA web log, each user has individual IP address. So, each IP address represents different user.

2.6. User Session Identification

The purpose of user session identification is to determine the division of access each user has a separate session. The simplest method is to use an expiration time, i.e. the time spent in a page passes a certain threshold, and it is assumed that the user has started a new session. The default time for user session identification is thirty minutes [3]. In this paper for user session identification is considered 30 min expiration time. This default value is used in various studies [4]. Long and convoluted user access paths along with low use of a web page indicate that the web site is not laid out in an intuitive manner. With the help of this analysis, one can re-structure the web site with the navigation results.

3. CLUSTERING ALGORITHMS

Cluster analysis groups objects based on the information found in the data describing the objects or their relationships. The goal is that the objects in a group will be similar related to one other and different from the objects in other groups. Clustering algorithm is classified in to two categories: 1) Decomposition (top-down) 2) Agglomerative (bottom-up). If K-Means and Bisecting K-Means algorithms clusters are decomposed. But Hierarchical clusters are bottom-up approach. It is deterministic [8]. The clusters will be create a complete binary tree. The various clustering algorithms are compared and find which one is the best.

3.1. K-Means Algorithm

K-mean is one of the simplest unsupervised learning algorithms that solve the well-known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters.

Basic K-means Algorithm for finding K clusters

1. Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.

2. Assign each object to the group that has the closest centroid.
3. When all objects have been assigned, recalculate the positions of the K centroids.
4. Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

K- Means algorithms follows.

K-means-cluster (in S : set of vectors : k : integer)

```
{  let C[1] ... C[k] be a random partition of S into k parts;
  repeat {
        for i := 1 to k {
          X[i] := centroid of C[i];
          C[i] := empty
         }
        for j := 1 to N {
          X[q] := the closest to S[j] of X[1] ... X[k]
          add S[j] to C[q]
         } }
   until the change to C (or the change to X) is small enough
}
```

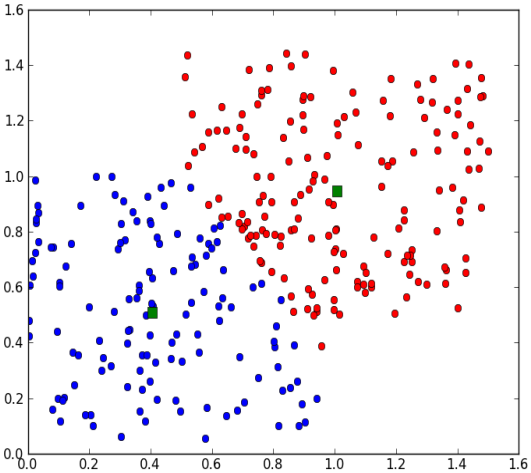

Fig.2. Three clusters are placed to one area

In this case we splitted the data in 2 clusters, the blue points have been assigned to the first and the red ones to the second. The squares are the centers of the clusters.

After using K-Means algorithm three clusters are placed in areas separated in different places in order. The pink square indicate centroids of the clusters.

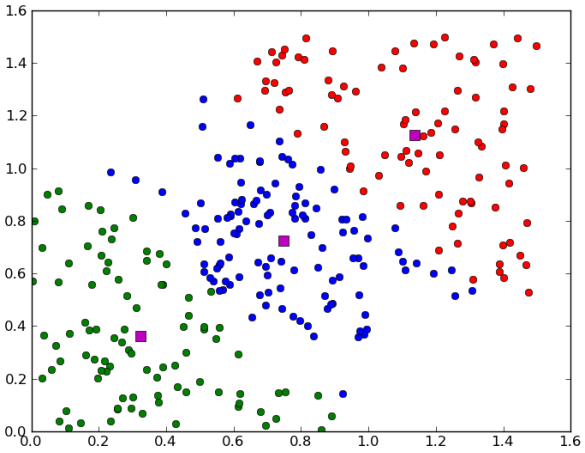

Fig.3. Three clusters are placed to one area

3.2. Bisecting k-means Algorithms

Bisecting k-Means is like a combination of k-Means and hierarchical clustering. Instead of partitioning the data into 'k' clusters in each iteration, Bisecting k-means splits one cluster into two sub clusters at each bisecting step(by using k-means) until k clusters are obtained[3].

Basic Bisecting K-means Algorithm for finding K-Clusters

1. Pick a cluster to split.
2. Find 2 sub-clusters using the basic K-means algorithm.

(Bisecting step)

3. Repeat step 2, the bisecting step, for ITER times and take the split that produces the clustering with the highest overall similarity.
4. Repeat steps 1, 2 and 3 until the desired number of clusters is reached.

Bisecting K-means Algorithm follows

Divisive hierarchical clustering method using K means

```
For i=1 to k-1 do {
Pick a leaf cluster C to split
For j=1 to Iteration do
 {
        Use K-Means split to two sub clusters C1 and C2
        Choose the best of the above splits and make it permanent
        }
        }
```

As Bisecting k-means is based on k-means, it keeps the merits of k-means and also has some advantages over k-means. First, bisecting k-means is more efficient when 'k' is large. For the k-means algorithm, the computation involves every data point of the data set and k centroids [9].

On the other hand, in each Bisecting step of Bisecting k-means, only the data points of one cluster and two centroids are involved in the computation [3]. Thus, the computation time is reduced. Secondly, Bisecting k-means produce clusters of similar sizes, while k-means is known to produce clusters of widely different sizes.

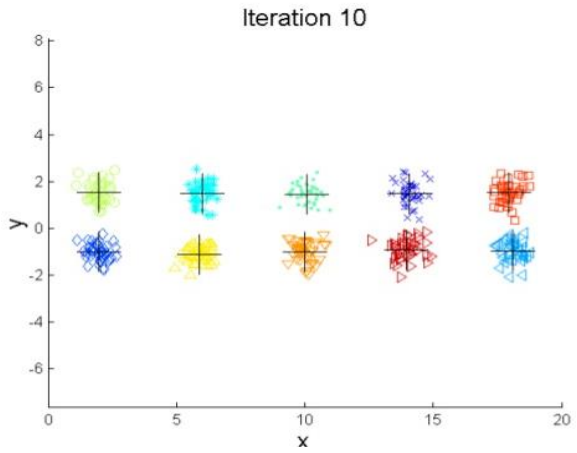

Fig 4.The clusters are split until 10 Iteration

Web log data set, in the form of log file are collected from college web site which consists of various reports and summaries.

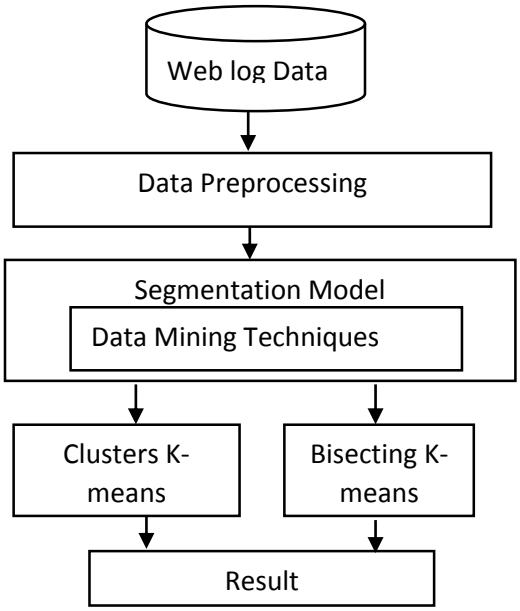

Fig.5. Flow of design Modules

These files are in nonstandard format. Extraction of useful information is done, which includes taking in account bandwidth and web usage reports and summaries. Here the log files are first read and then parsed. Parsing means analyzing a text and converting it into useful form [7]. It consists of displaying of IP address from the Bandwidth reports and the total bytes communicated by it.

4. EXPERIMENTAL RESULTS

Experiment was carried out using a log retrieved. The web log files (Log files) in the form of LOG are collected from Vels University Chennai.

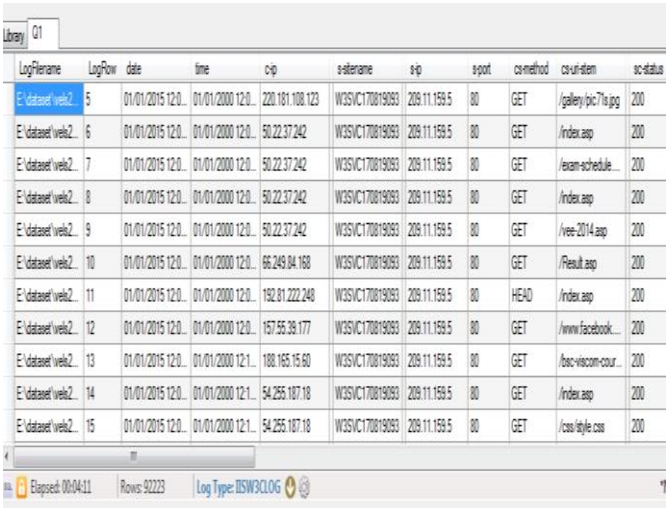

Fig.4. Reading web log file and Parsing the file

The input web log file is the log file, which is first read by the system. The useful text is extracted from a large data and then parsed. Parsing involves displaying of various IP address and total bytes communicated by them

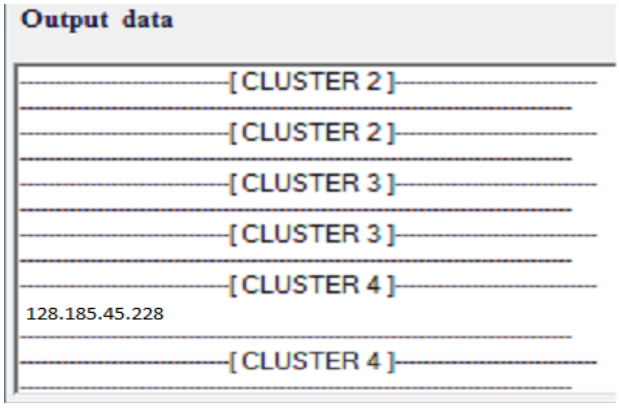

Fig. 5. Formation of clusters using K-Means

Fig .5.shows various empty clusters generated using K-means. Here, we can see a single cluster (cluster1) consisting of large portion of the dataset.

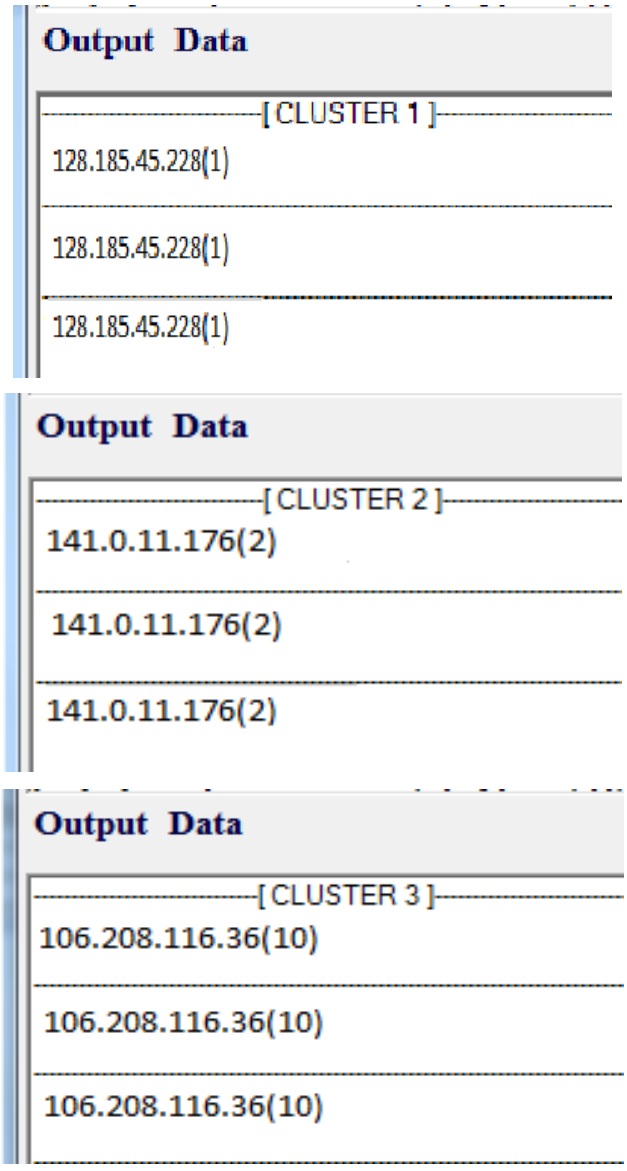

Fig 6.Formation of clusters using Bisecting K-Means

Fig.6. shows outputs of clustering using Bisecting K-means algorithm. The results concludes that BKM does not produce any empty cluster and the clusters are uniform and balanced. Pick one cluster to split two sub clusters.cluster1 are split to two sub clusters. Such as cluster 2 and cluster 3.

5. COMPARISON OF TWO ALGORITHMS

The main objective of this work is to analyze the web usage data by applying machine learning techniques such as K-means and Bisecting K-Means clustering algorithms. To perform the analysis, web access log data has been collected through Internet. The file contains the sequence of user access details. In both clustering algorithms, initially the cluster center randomly chosen based on the cluster centroid and the clustering of data can be evaluated. The comparison of two algorithms are following below. The k-Means and Bisecting K-Means algorithms are compared to above Table 1.but the Bisecting K –Means algorithms Accuracy are Greater than K-Means.

Table .1.comparison between k-means and bisecting k- means

Parameters	K-means	Bisecting K-means
Functioning	Partitions data into k clusters in each iteration	Splits one cluster into two sub clusters at each Bisecting step (using k-means)
Computational Time	The computation of each iteration involves every data point of data set and k-centroids.	Only data points of one cluster and two centroids are involved at each bisecting step. Hence, computational time is less.
Decadence	May or may not generate empty clusters	Does not generate Empty clusters
Technique	Agglomerative, Bottom- up	Decomposition, Top –Down, Divisive
Nature of clusters	Clusters formed are unbalanced, not uniform is in size.	Uniform clusters are produced having similar sizes
Exactitude	Lesser compared to BKM	Greater compared to K-Means
Overall Performance	Depends on selection of initial centroids which is random, hence not efficient.	No initialization of centroids. Hence, more efficient than K-means

Log data	K-means	Bisecting k-means
Log File 1	68.75%	78.23%
Log File 2	74.78%	81.19%
Log File 3	65.32%	75.24%
Log File 4	72.28%	82.73%
Log File 5	78.75%	84.53%

Table 2. Performance of K-means and BKM

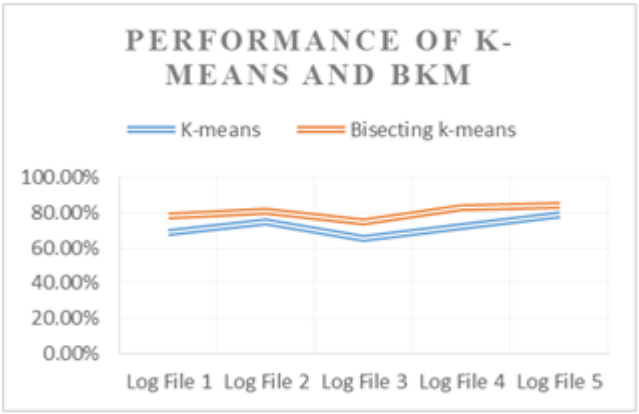

Fig.7. Performance of K-means and BKM algorithm

6. CONCLUSIONS

This article present for clustering algorithm on weblog data. In this work the impotent phases such as preprocessing, pattern discovery and pattern analysis, similarity measurement and clustering technique are used for improve efficiency. The Preprocessing technique applied in log parser and comparing K-means and Bisecting K-means algorithm. The result show the performance, accuracy, computation time identified. Finally verify the performance of Bisecting k-means is relatively efficient than K-Means, but even it is dependent on initial number of cluster 'k' given by the user. The web log data cluster centroid also the experimental result shows the performances of both clustering algorithms.

REFERENCES

[1] Web Log data." International Journal of Computer Applications 116.19 (2015).
[2] Kuang, GuoFang, and MingLi Song. "The Application Analysis of Clustering and Partitioning Algorithm in Web Data Mining." Advances in Computer Science and Information Engineering. Springer Berlin Heidelberg, 2012. 455-460.
[3] Bangoria Bhoomi, M. "Enhanced K-Means Clustering Algorithm to Reduce Time Complexity for Numeric Values." International Journal of Computer Science and Information Technologies 5.1 (2014): 876-879.
[4] Murugesan, Keerthiram, and Jun Zhang. "Hybrid bisect K-means clustering algorithm." 2011 International Conference on Business Computing and Global Informatization. IEEE, 2011.
[5] Patil, Ruchika, and Amreen Khan. "Bisecting K-Means for Clustering Web Log data." International Journal of Computer Applications 116.19 (2015).
[6] Singh, Supinder, and Sukhpreet Kaur. "Web Log File Data Clustering Using K-Means and Decision Tree." International Journal of Advanced Research in Computer Science and Software Engineering 3.8 (2013).
[7] Hirudkar, Arpita M., and S. S. Sherekar. "Comparative analysis of data mining tools and techniques for evaluating performance of database system." Int J Comput Sci Appl 6.2 (2013): 232-237.
[8] Garg, Kanwal, and Deepak Kumar. "Comparing the performance of frequent pattern mining algorithms." International Journal of Computer Applications 69.25 (2013).
[9] Ristoski, Petar, Christian Bizer, and Heiko Paulheim. "Mining the web of linked data with rapidminer." Web Semantics: Science, Services and Agents on the World Wide Web 35 (2015): 142-151.