MapReduce K-Means based Co-Clustering Approach for Web Page Recommendation System

K. Krishnaveni
Research Scholar
Department of Computer Science
Periyar university, Salem-11
krishnavenics1993@gmail.com

R. Rathipriya
Assistant Professor
Department of Computer Science
Periyar University, Salem-11
rathi_priyar@periyaruniversity.ac.in

Abstract- **Co-clustering is one of the data mining techniques used for web usage mining. Co-clustering Web log data is the process of simultaneous categorization of both users and pages. It is used to extract the users' information based on subset of pages. Nowadays, the cyberspace is filled with huge volume of data distributed across the world. The business knowledge acquaintance from such a voluminous data using the conventional systems is challenging. To overcome such complexity the Google invented the Map Reduce, a programming model used to incorporate the parallel processing in the distributed environment. In this paper, MapReduce K-Means based Co-Clustering approach (CC-MR) is proposed for web usage data to identify constant browsing patterns which is very useful for E-Commerce applications like target marketing and recommendation systems. Here, benchmark K-Means clustering algorithm is used to generate constant co-clusters from the web data. Experiments are attempted on real time web dataset to exploit the performance of the proposed MapReduce K-Means based Co-Clustering approach. These experiments are implemented in the MatlabR2016 and this approach yields the promising results.**

Keywords- Co-Cluster, Co-Clustering Algorithm, MapReduce, Web Mining, MR K-means

I. INTRODUCTION

Web mining is to explore interesting information and potential patterns from the contents of web page, the information of accessing the web, page linkages and resources of e-commerce by using techniques of data mining, which can help people extract knowledge, improve Web sites design, and develop e-commerce better [1].

Clustering techniques apply when there is no class to be predicted but rather than the instances are to be divided into natural groups. Clustering Web data can be either user clustering or page clustering [2, 3]. Clustering is a technique to group together a set of items having similar characteristics. In the Web Usage domain, there are two kinds of interesting clusters to be discovered: usage clusters and page clusters.

Clustering of users tends to establish groups of users exhibiting similar browsing patterns. Such knowledge is especially useful for inferring user demographics in order to perform market segmentation in E-commerce applications or provide personalized web content to the users. On the other hand, clustering of pages will discover groups of pages having related content. This information is useful for Internet search engines and web assistance providers. In both applications, permanent or dynamic HTML pages can be created that suggest related hyperlinks to the user according to the user's query or past history of information needs.

A. Clustering

Clustering is one of the useful and active areas of data mining and machine learning techniques. It should help us to cope with the problem of information overload on the Internet. With clustering the goal is to separate a given group of data items (the data set) into groups called clusters. The items in the same cluster are similar to each other than that of the items in other clusters.

B. Limitation of clustering

The group users (or pages) based on global similarities in their expression profiles. However, a set of co-regulated users might only be co-expressed in a subset of experimental pages, and show not related, and almost independent expression patterns in the rest. In the same way, related experiments may be characterized by only a small subset of coordinately expressed users. Indeed, as Wang et al. remarked, there may only be a few

user components that account for most of the response variation across experiments, and thus important relationships among them may be lost in a high dimensional user space [3]. Many users were expressed different interest on the pages based on their need.

Therefore, they might be co-expressed with different groups of users under different pages [4]. Clustering the users into only one group might mask the interrelationships between users that are assigned to different clusters but show local similarities in their expression patterns. To overcome these problems, the concept of co-clustering for click stream data is proposed in this paper.

C. Research contribution

The research contributions of this paper are as follows.

- A new model called K-Means based Co-Clustering algorithm for web data is proposed for constant web pattern identification.
- Implementation of this model with Map/Reduce framework using Matlab
- Experimental results of the proposed work is discussed and analyzed in subsequent sections.

II. REVIEW OF THE LITERATURE

Nowadays, internet is a very fast communication media between business organizations' services and their customers with very low cost. Web Data mining [2] is an intelligent data mining technique to analyze web data. It includes web content data, web structure data and web usage data. Analysis of usage data provides the organizations with the required information to improve their performances.

A clustering process needs to meet a number of challenges to be efficient. These challenges involve the definition of appropriate similarity or distance measures that will adequately capture the relations between data objects and guide properly the clustering process. The application of specific similarity (distance) measures depends on the underlying data nature and the data structures used for their representation.

Web clustering is a well-studied problem and numerous clustering algorithms appeared in literature, which can be broadly categorized into different categories depending on the criteria employed. In a general categorization scheme, clustering algorithms are divided into partition and hierarchical, according to whether they produce flat partitions or a hierarchy of clusters.

Dhillonet. al. [2] proposed an innovative co-clustering algorithm that monotonically increases the preserved mutual information by intertwining both the row and column clustering's at all stages.

Y. Song et. al [5] presented a constrained co-clustering approach for clustering textual documents. This approach combines the benefits of information-theoretic co-clustering and constrained clustering. This paper used a two-sided hidden Markov random field (HMRF) to model both the document and word constrain and also developed alternating expectation maximization (EM) algorithm to optimize the constrained co-clustering model.

Stefanie Jegelkaet. Al [6] derived the methods for clustering settings using Bregman divergences and also proved an approximation factor for tensor clustering with arbitrary separable metrics.

Fenget.al. proposed a general framework, *CRD*,(CRD stands for Co-clustering based on Column and Row Decomposition) for co-clustering large datasets utilizing recently developed sampling-based matrix decomposition methods.

III. METHODS AND MATERIALS

With the recent explosive growth of the web content on the cyberspace, it has become increasingly difficult for users to find and utilize the information and for content providers to classify the log documents. Traditional web search engines often return hundreds or thousands of results for a search, which is time consuming for users to browse. On-line libraries, search engines, and other large document repositories *(e.g.* customer support databases, product specification databases, press release archives, news story archives, *etc.)* are growing so rapidly that it is difficult and costly to categorize every document manually [6]. In order to deal with these problems, researchers look toward automated methods of working with web documents.

A. Limitations: K-Means based Co-Clustering

The K-Means Co-Clustering has various limitations, they are as follows.

- Difficult to predict K-Value
- Different initial partitions can result in different final clusters.

B. Co-Clustering Approach

Co-clustering process on a data matrix involves the determination of a set of clusters considering both rows and columns simultaneously. Co-clustering in essence is the task of finding these coherent sub-matrices of A. One illustration of co-clustering is shown in the following matrix. The six squares represent the six co-clusters [7].

$$A = \begin{pmatrix} 6 & 6 & 6 & 2 & 5 & 1 \\ 3 & 3 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 8 & 9 & 9 \\ 0 & 0 & 0 & 8 & 9 & 9 \\ 2 & 2 & 2 & 4 & 4 & 4 \\ 2 & 2 & 2 & 4 & 4 & 4 \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \\ A_{31} & A_{32} \end{pmatrix}$$

They are

$A_{11}(\{r_1,r_2\},\{c_1,c_2,c_3\})$, $A_{12}(\{r_1,r_2\},\{c_4,c_5,c_6\})$, $A_{21}(\{r_3,r_4\},\{c_1,c_2,c_3\})$, $A_{22}(\{r_3,r_4\},\{c_4,c_5,c_6\})$, $A_{31}(\{r_5,r_6\},\{c_1,c_2,c_3\})$, $A_{32}(\{r_5,r_6\},\{c_4,c_5,c_6\})$.

C. Co-Cluster

Each co-cluster is defined on a subset of rows and a subset of columns [1]. Moreover, two co-clusters may overlap, which means that several rows or columns of the matrix may participate in multiple co-clusters. Another important characteristic of co-clusters is that each co-cluster should be maximal.

D. Co-Cluster Type

An interesting criterion to evaluate a co-clustering algorithm concerns the identification of the type of co-clusters that the algorithm can find [8].There are four major classes of co-clusters. They are

1. Co-clusters with constant values.
2. Co-clusters with constant values on rows.
3. Co-clusters with constant values on columns.
4. Co-clusters with coherent values.

A co-cluster with constant values in the rows identifies a subset of users with similar expression values across a subset of pages, allowing the different expression levels for different users. Similarly, a co-cluster with constant columns identifies a subset of pages within which a subset of users' present similar expression values assuming that the expression values may differ from page to page.

In the proposed algorithm, seeds are in checkerboard structure which means they are non-overlapping. After seed growing phase, the resultant co-clusters are in the structure of tree structure, non-exclusive, overlapping with hierarchical structure, and arbitrarily positioned overlapping structures as seen in the figure3.1. These types of co-clusters are very useful to know the multiple browsing interest of the user. The interpretations of the results are useful for the E-commerce applications like Recommendation systems and Target Market [5].

E. *Clustering algorithm: K-Means*

K-Means clustering technique is used to create user cluster and page cluster. K-Means is one of the simplest unsupervised learning algorithms for clustering problem. The procedure is simple and easy way to classify a given data set through a certain fixed number of clusters (assume K clusters).

The K-Means algorithm is significantly sensitive to the initial randomly selected cluster centers. Run K-Means algorithm repeatedly with different random cluster centers (called centroids) approximately for ten times. Choose the best centroids with least intra cluster distance

F. A Background Study: Map-Reduce

Map-Reduce, originally described in [9] are a core component in an emerging ecosystem of distributed, scalable, fault-tolerant data storage, management, and processing tool. Map-Reduce are essentially a distributed grep-sort-aggregate or, in database terminology, a distributed execution engine for select-project via sequential scan, followed by hash partitioning and sort-merge group-by. It is ideally suited for data already stored on a distributed file system which offers data replication as well as the ability to execute computations locally on each data node. The overview of the MapReduce model is shown in the figure 1.

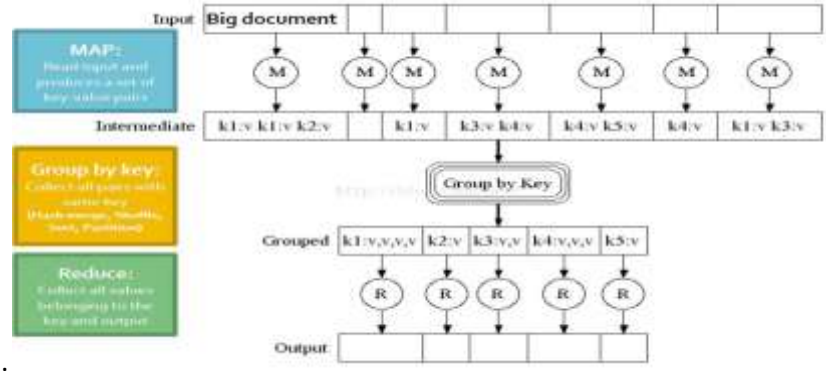
1) Programming Model and Data Flow

The Map-Reduce draws from a well-established abstraction in functional programming. The previous sections illustrate most of the programming model aspects. Formally, a computation is decomposed into a map operation followed by a reduce operation. These are represented by two functions,

- MAPPER:$\langle \mathrm{k_{in}}, \mathrm{v_{in}} \rangle \rightarrow \langle k_{int,v_{int}} \rangle$
- REDUCER:$\langle \mathrm{k_{int}}, \mathrm{V} \equiv \{\mathrm{v_{int}}\} \rangle \rightarrow \langle \mathrm{k_{out}}, \mathrm{v_{out}} \rangle$

Both the functions operate on key-value pairs, which we denote using angle brackets $\langle k, v \rangle$. The key is used primarily in the reduction step, to determine which values are grouped together. Values may carry arbitrary information.

This computation needs to be eventually executed on a large cluster. This paper focus is on the data flow model (see Figure 1) Overview of the Map-Reduce execution framework. The map input is be partitioned into a number of input splits. Processing each split is assigned to one map task. Subsequently, all map outputs are partitioned among a number of reduce tasks, by hashing on the intermediate keyk_{int}. Each reducer receives one part of the intermediate key space. Subsequently, it merges all in- puts received from all mappers, sorts them based on kint to group equal keys together, and applies the reducer function to obtain the final results.



.

Figure 1. Overview of the Map-Reduce execution framework

2) Mean Squared Residue (MSR) Score

Cheng and Church *(2000)* [10] defined a biclusters as a subset of rows and subset of columns, which has low Mean Squared Residue (MSR) Score as in equation (1). This criterion have widely used in this field.

$$\mathrm{MSR}\ (\mathrm{B}(\mathrm{U}', \mathrm{P}')) = \frac{\sum_{\mathrm{i}\in\mathrm{I},\mathrm{j}\in\mathrm{J}} \mathrm{R}(\mathrm{b_{ij}})^2}{|\mathrm{U}'|*|\mathrm{P}'|} \quad (1)$$

where $R(b_{ij}) = b_{ij\_}b_{iP'\_}b_{U'j+}b_{U'P'}$ is the residue score of each expression value, $b_{iP'} = \frac{\sum_{j\in J} b_{ij}}{|P'|}$, $b_{U'j} = \frac{\sum_{i\in I} b_{ij}}{|U'|}$ and $b_{U'P'} = \frac{\sum_{\substack{i\in U' \\ j\in P'}} b_{ij}}{|U'|.|P'|}$.

This paper describes the co-cluster, co-clustering approach, co-clustering type, co-clustering structure, and K-Means clustering algorithm, Map-Reduce, programming model and data flow. The next section discusses about K-Means based Co-clustering using MapReduce

IV. Introduction Proposed Method K-Means Co-Clustering Alorithm using Map Reduce Model

Map-Reduce adopt an extremely simple but powerful abstraction from functional programming. Many data processing tasks can be easily formulated as Map-Reduce jobs. Inspired by that, many higher-level programming abstractions have been implemented for large-scale data processing compared to them; our focus is to illustrate a complete data mining process involving multiple interconnected steps that all require large-scale data processing. The Co-Clustering approach is incorporated in the Map phase, whereas the highly coherent co-clusters are highlighted and emitted in the Reduce phase. The entire architecture and working principles are discussed in the upcoming section.

A. Map Phase

MapReduce works based on both Map and Reduce functions, the Map function receives chunks of data and outputs intermediate results, that are fed to the Reduce function and produces a final result. Thus, it is normal to break up a calculation into two related pieces for the Map and Reduce functions to fulfill separately.

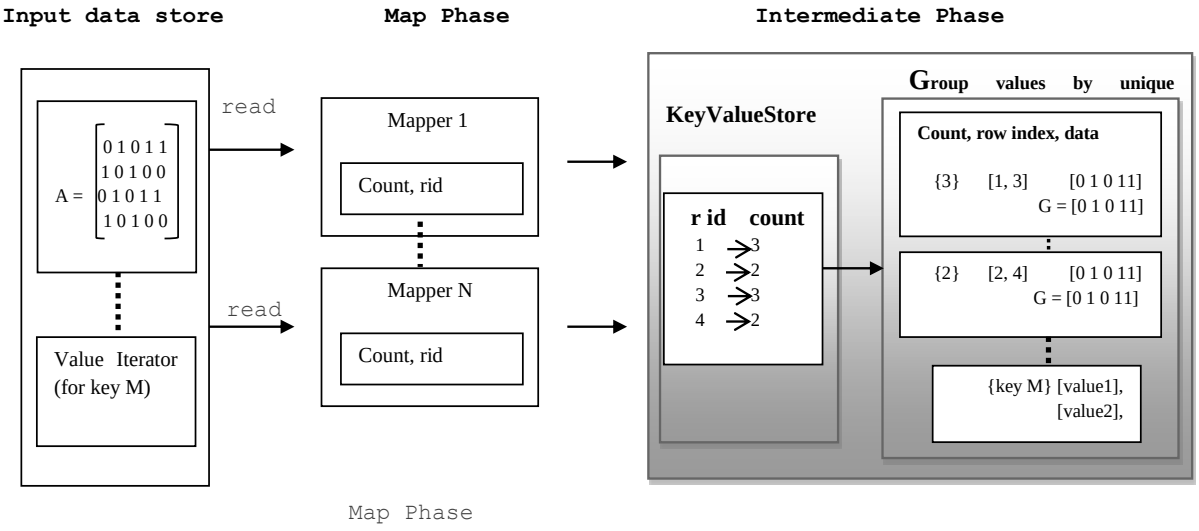


Figure 2. Map Phase: An Overview

The working of Map phase in Matlab based MapReduce model (figure 2) has the following steps:

1. MapReduce reads chunks of web data using the read function on the input datastore, and then calls the map function to work on the chunk.
2. The map function then works on the individual chunk of web data as given in algorithm-1 and adds one key-value pairs(i.e., Count, Rid) to the intermediate KeyValueStore object using the add functions.
3. MapReduce repeats this process for each of the chunks of web data (i.e., number of rows in the chunks) in the input datastore, so that the total number of calls to the map function is equal to the number of chunks of data.

Algroithm 1: Map Function

```
Function Coclust_Map (A,k,v)
for each row in A do
        r_count(i) = find number of ones in A(i , :)
end
    emit(row_key, r_count )
end
```

The Map phase of the MapReduce algorithm is complete when the map function processes each of the chunks of data in the input datastore. The result of this phase of the MapReduce algorithm is

a KeyValueStore object that contains all of the key-value pairs added by the map function. After the Map phase, MapReduce prepares for the Reduce phase by grouping all the values in the KeyValueStore object by unique key (i.e., count).

In addition to these basic requirements for the map function, the key-value pairs added by the map function must also meet these conditions:

1. Keys must be numeric scalars or character vectors. Numeric keys cannot be `NaN`, complex, logical, or sparse.
2. All keys added by the map function must have the same class.
3. Values can be any MATLAB object, including all valid MATLAB data types.

B. Reduce Phase

The working of Reduce phase in Matlab based MapReduce in figure 3 are as follows

1. The result of the Map phase of the MapReduce algorithm is an intermediate key value store object that contains all of the key-value pairs (i.e., Count, Rid) added by the map function. Before calling the reduce function, MapReduce groups the values in the intermediate KeyValueStore object by unique key. Each unique key in the intermediate KeyValueStore object results in a single call to the reduce function.

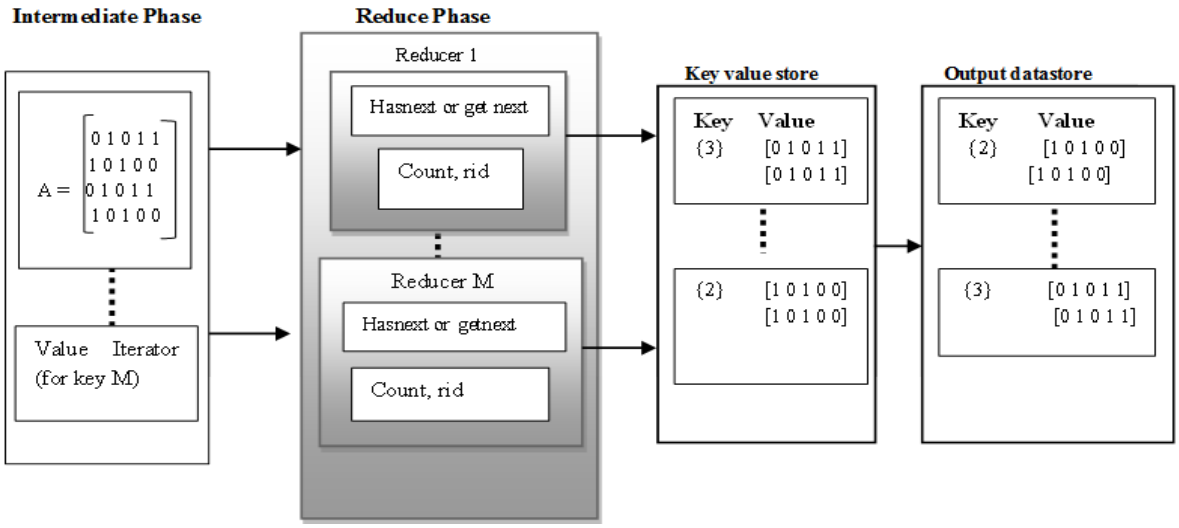


Figure 3. Reduce Phase

2. For each key, MapReduce creates a ValueIterator object that contains all of the values (i.e.) row id & count associated with that key.
3. The reduce function scrolls through the values from the ValueIterator object using the hasnext and getnext functions, which are typically used in a while loop.
4. After performing a summary calculation, the reduce function adds one or more key-value pairs to the final KeyValueStore object using as given in algorithm 2 the add functions.

Algroithm 2: Reduce Function

```
Function Coclust_Reduce (A,key,value)
    1. Intialize 'k' clusters, distance measure for column cluster
    2. For each key (key_i) do
           rowcluster (key_i )= [Combine the rows of key_i]
       end
    3. For each rowcluster (key_i ) do
           Kmeans (rowcluster (key_i ), k)
       end
    emit(cocluster, cocluster_value )
end
```

The figure 3 shows the Reduce phase of the Co-clustering using Map Reduce. The Reduce phase of the MapReduce algorithm is complete when the reduce function processes all of the unique intermediate keys and their associated values. The result of this phase of the MapReduce algorithm (similar to the Map phase) is a KeyValueStore object containing all of the final key-value pairs (i.e. Row_id,r_count) added by the reduce function.

C. Co-clustering Using Map-Reduce Phase (CC-MR)

MapReduce moves each chunk of data in the input datastore through several phases before reaching the final output. The following figure outlines process in each phases of the Co-clustering algorithm using MapReduce as in the figure 4.

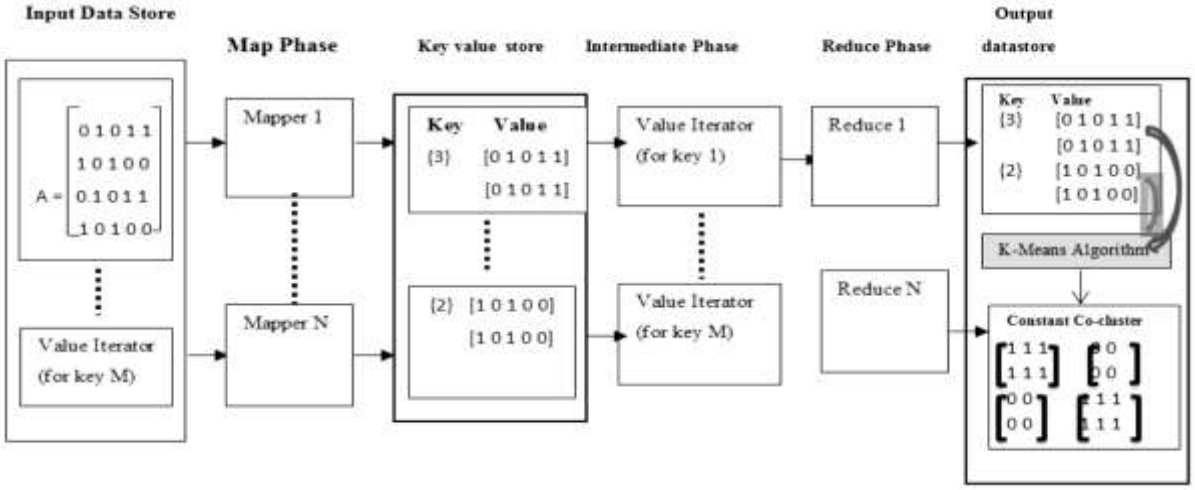


Figure 4. Co-clustering using Map-Reduce Phase (CC-MR)

For example, 4 x 5 input matrixes as given figure 4 the constant sub-matrix is grouped together. The basic steps are:

- **First phase:** For every row, compute the number of ones in it. This count is treated as key value. This process is take place in the Map phase of MapReduce environment. Finally it returns the <count, row id >as the key value pair.
- **Second Phase:** Based on key values, rows of the given data matrix are grouped which forms a cluster. Then K-Means clustering algorithm is applied to these clusters to obtain constant co-cluster. This entire process is take place in the Reduce phase of MapReduce environment.

Therefore, the total number of Co-cluster generated from the given dataset is number of keys 'NK' x 'k' column cluster. Total number of co-cluster= Number of Keys (NK) in the Map phase x 'k' column cluster in the reduce phase. To validate the quality of Co-cluster, a very frequently used co-cluster quality measure called Mean Squared Residue is used for this study

D. Data Representation

The data representation of Web data is in the form of matrix with frequency values. Let A(U, P) be an m x n, user associated matrix where U={U1,U2 ,…….,Um} be a set of users and P={ P1,P2,…….,Pn} be a set of pages of a web site. It is used to describe the relationship between web pages and users who access these web pages. Let 'm' be the number of web user and 'n' be the number of web pages. The element $a_{ij} \in A$ (U, P).

Algroithm 3: Map Reduce function

Function Coclust MapReduce (A, key,value)

1. Input data store (data, info, internkey)
2. Call Mapper / Reducer function using (ds,@coclustmap.@coclustreduce)
3. Call Reducer
4. Return coclusters from ds

End

E. Data Representation

The data representation of Web data is in the form of matrix with frequency values. Let A(U, P) be an m x n, user associated matrix where U={U_1,U_2 ,…….,U_m} be a set of users and P={ P_1,P_2,…….,P_n} be a set of pages of a web site. It is used to describe the relationship between web pages and users who access these web pages. Let 'm' be the number of web user and 'n' be the number of web pages. The element a_{ij} of A (U, P) represents frequency of the user U_i of U visit the page P_j of P during a given period of time.

$$a_{ij} = \begin{cases} \text{Hits}\,(U_i, P_j), & \text{if } P_j \text{ is visited by } U_i \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

Where Hits (U_i , P_j) is the count/frequency of the user (U_i ,) accesses the page (P_j) during a given period of time

F. Data filtering

Data filtering is the task of extracting only those records of weblog files, which are essential for the analysis, thus reducing data significantly necessary for further processing. In this paper, data filtering aims to filter out the users who have visited less than 9 page categories of web site [1].

In this paper, the proposed co-clustering algorithm is tested on real data sets taken from msnbc.com which is available in UCI repository.

V. Result and Discussion

The proposed work is implemented using the MAPREDUCE model of MATLAB environment. The Weblog data is taken from the UCI repository. The dataset description and parameter values are listed in the table I.The results of the proposed work are visualized and their characteristics are discussed further in this section. The table II describes the set of page in each Co-cluster for key 1. In Key1, the total number of ones for each row is one. The graphical representation for the placement of pages in the Co-cluster is shown in figure 5.

Table I. Parameter Setup

Dataset Name	MSNBC weblog Data
No of Users	3386
No of Pages	17
No of Maps	10
Intermediate (Key, Value)	<count,row id >
(Key, Value)	<Co_clusterid, MSR>
No of Co-Clusters	16
MSR range	0

The table III describes the set of page in the each Co-cluster for key 2. In Key2, the total number of ones for each row is one. The graphical representation for the placement of pages in the Co-cluster is shown in figure 5. The table IV describes the set of page in the each Co-cluster for key 3. In Key 3, the total number of ones for each row is one. The graphical representation for the placement of pages in the Co-cluster is shown in figure 5

Table II. Characteristic of Co-clusters for Key 1

Co-cluster id	Page ID for Key=1	MSR Value
1	1,2,4,6,7	0
2	8,9,13,16	0
3	3,10,11,12	0
4	5,15	0
5	4	0
6	17	0

Table III. Characteristics of Co-clusters for Key2

Co-cluster id	Page ID forKey=2	MSR Value
1	3,10,11,12	0
2	14	0
3	1,2,4,6,7	0
4	5,15	0
5	8,9,13,16	0
6	17	0

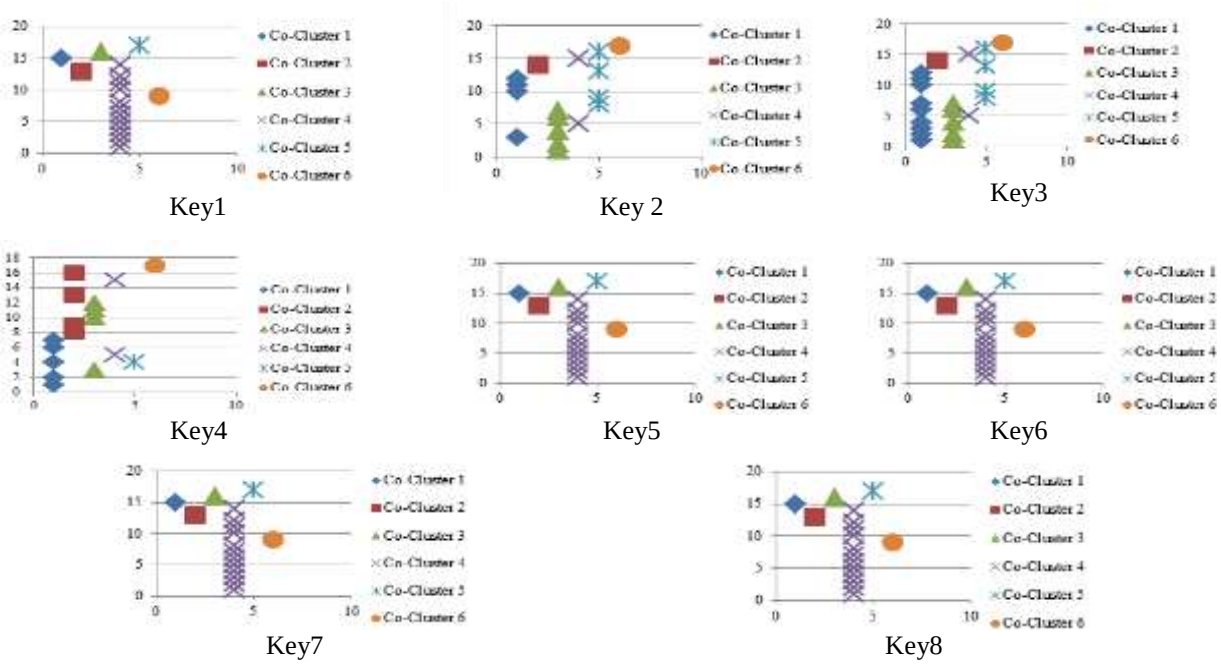


Figure 5. Pages in the Co-clusters for different Key Values

Table IV. Characteristics of Co-clusters for Key 3

Co-cluster id	Page ID forKey=3	MSR Value
1	1,2,3,4,6,7,10,11,12	0
2	14	0
3	1,2,4,6,7	0
4	5,15	0
5	8,9,13,16	0
6	17	0

Table V. Characteristics of Co-clusters for Key 4

Co-cluster id	Page ID forKey=4	MSR Value
1	3,5,10	0
2	14	0
3	1,2,4,6,7,12	0
4	9,13,16,17	0
5	8	0
6	15	0

The table V describes the set of page in the each Co-cluster for key 4. In Key 4, the total number of ones for each row is one. The graphical Representation for the placement of pages in the Co-cluster is shown in figure 5

Table VI. Characteristics of Co-clusters for Key5

Co-cluster id	Page ID forKey=5	MSR Value
1	4,17	0
2	15	0
3	9,13,16	0
4	5	0
5	1,2,3,6,7,10,11,12,14	0
6	8	0

The table VI describes the set of page in the each Co-cluster for key 5. In Key 5, the total number of ones for each row is one. The graphical representation for the placement of pages in the Co-cluster is shown in figure 5.

Table VII. Characteristics of Co-clusters for Key 6

Co-cluster id	Page ID forKey= 6	MSR Value
1	1,2,5,6,7,8,10,11,12,17,19	0
2	14	0
3	1,2,4,6,7	0
4	5,15	0
5	8,9,13,16	0
6	17	0

The table VII describes the set of page in the each Co-cluster for key 6. In Key 6, the total number of ones for each row is one. The graphical representation for the placement of pages in the Co-cluster is shown in figure 5. The table VIII describes the set of page in the each Co-cluster for key 7. In Key 7, the total number of ones for each row is one. The graphical representation for the placement of pages in the Co-cluster is shown in figure 5

Table VIII. Characteristics of Co-clusters for Key 7

Co-cluster id	Page ID forKey=7	MSR Value
1	1,2,3,4,6,7,8,10,11,12,14	0
2	13	0
3	17	0
4	9	0
5	5	0
6	16	0

Table IX. Characteristics of Co-clusters for Key 8

Co-cluster id	Page ID forKey=8	MSR Value
1	15	0
2	13	0
3	16	0
4	1,2,3,4,5,6,7,810,11,12,14	0
5	17	0
6	9	0

The table IX describes the set of page in the each Co-cluster for key 8. In Key 8, the total number of ones for each row is one. The graphical representation for the placement of pages in the Co-cluster is shown in figure 5. It is observed from the study that MSR value of all extracted Co-cluster for MSNBC dataset is almost zero. Therefore, it is concluded that the proposed work performs well in extracting constant co-cluster

VI. CONCLUSION

The proposed Co-clustering algorithm for Click stream data is evaluated with real dataset. In this paper, the proposed K-means based co-clustering algorithm using MapReduce is used to find co-clusters with maximum size and less variance in data, and particularly with a very low MSR value. These co-clusters are useful to predict the interests of the users. The results proved its efficiency in grouping the relevant users and web pages of a web site. Thus, interpretation of co-cluster results are used by the company for targeted marketing campaigns to an interesting target user cluster. In future, this work is extended with different clustering Algorithm Rough K-means, Fuzzy C-Mean etc

REFERENCES

[1] Yoon Ho Cho, Jae Kyeong Kim, Soung Hie Kim, A personalized recommender system based on web usage mining and decision tree induction, Expert Systems with Applications, Vol.23, pp.329-342, 2002.

[2] Cho. H, I.S.Dhillon, Y.Guna, and S.Sra, Minimum Sum–Squared Residue Co-clustering of Gene Expression Data, Proceedings Fourth SIAM International Conference of Data Mining, pp.114-125, 2004.

[3] Hartigan J.A., Direct Clustering of a Data Matrix, J. Am. Stat. Assoc. (JASA), Vol.67, No.337, pp.123-129, 1972.

[4] R.Rathipriya , Dr. K.Thangavel , J.Bagyamani, Evolutionary Biclustering of Clickstream Data, International Journal of Computer Science Issues, Vol.8, No.1, pp.341-347, 2011.

[5] Lizhen Liu, Junjie Chen, Hantao Song, The Research of Web Mining, Preprocessing in Data and Web Usage Mining, 2006.

[6] Oren Zamir and Oren Etzioni, Web Document Clustering: A Feasibility Demonstration, Research and Development in Information Retrieval, pp.46-54, 1998.

[7] Qinbao Song, Martin Shepperd, "Mining Web browsing patterns for E-commerce", Science Direct, computers in industry, vol 57, pp. 62-630, 2006.

[8] J. Dean and S. Ghemawat, MapReduce: Simplified data processing on large clusters, CACM, Vol.51, pp.1, 2008.

[9] I. S Dhillon, Mallela, and D.S. Modha, Information theoretical Co-Clustering, Ninth ACM SIGKDD International Conference of Knowledge Discovery and Data Mining (KDD), Vol.03, pp.89-98, 2003.

[10] Y. Cheng and G. Church, Biclustering of expression data, Proceedings ISMB, pp.93-103, 2000.