Introduction to KEA-Means Algorithm for Web Document Clustering

Swapnali Ware[#1], N.A.Dhawas*[2]

[#]Department of Computer Engineering, Pune
[*] Department of Information Technology, Pune
SIT, Lonavala, India

***Abstract*-- In most traditional techniques of document clustering, the number of total clusters is not known in advance and the cluster that contains the target information or précised information associated with the cluster cannot be determined. This problem solved by K-means algorithm. By providing the value of no. of cluster k. However, if the value of k is modified, the precision of each result is also changes.**

To solve this problem, this paper introduces a new clustering algorithm known as KEA-Means algorithm which will combines the kea i.e. key phrase extraction algorithm which returns several key phrases from the source documents by using some machine learning language by creating model which will contains some rule for generating the no. of clusters of the web documents from the dataset .this algorithm will automatically generates the number of clusters at the run time here. User need not to specify the no. of clusters. This Kea-means clustering algorithm provides the value of k and will be beneficial to extract test documents from massive quantities of resources.

***General Terms*--Data mining, kea-means algorithm, k-means algorithm, Web document clustering**

***Keywords*-- K-means clustering, Kea key phrase extraction algorithm, KEA-Means algorithm, F-measure.**

I. INTRODUCTION

The increasing size and dynamic content of the World Wide Web has created a need for automated organization of web-pages. Document clusters can provide a structure for organizing large bodies of text for efficien6t browsing and searching. For this purpose, a web-page is typically represented as a vector consisting of the suitably normalized frequency counts of words or terms. Each document contains only a small percentage of all the words ever used in the web. If we consider each document as a multi-dimensional vector and then try to cluster documents based on their word contents, the problem differs from classic clustering scenarios in several ways. Document clustering data is high dimensional, characterized by a highly sparse word-document matrix with positive ordinal attribute values and a significant amount of outliers [1].

The main drawbacks of traditional techniques of the document clustering is that the number of total clusters is not known in priori and the cluster that contain the target information cannot be determined since the semantic nature is not associated with the cluster. To solve this problem, this paper proposed a new clustering algorithm based on the Kea key phrase algorithm that extracts several Key phrases from source documents by using some machine learning techniques. The Kea-means clustering algorithm provides easy and efficient ways to extract test documents from massive quantity of resources.

II. LITERATURE SURVEY

Shen Huang, Zheng Chen, Yong Yu, and Wei-Ying Ma (2006) [7], proposed Multitype Features Co-selection for Clustering (MFCC), a novel algorithm to exploit different types of features to perform Web document clustering. They have use the intermediate clustering result in one feature space as additional information to enhance the feature selection in other spaces. Consequently, the better feature set co-selected by heterogeneous features will produce better clusters in each space. After that, the better intermediate result will further improve co-selection in the next iteration. Finally, feature co-selection is implemented iteratively and can be well integrated into an iterative clustering algorithm.

Michael Steinbach, George Karypis, Vipin Kumar (2000) [2], presented the results of an experimental study of some common document clustering techniques: agglomerative hierarchical clustering and K-means. (They have used both a “standard” K-means algorithm and a “bisecting” K-means algorithm.) These results indicate that the bisecting K-means technique is better than the standard K-means approach and (somewhat surprisingly) as good as or better than the hierarchical approaches that tested by them.

Jiang-chun song, jun-yi shen (2003) [9], based on the Vector Space Model (VSM) of the Web documents, they have improved the nearest neighbor method, put forward a new Web document clustering algorithm, and researched the validity and scalability of the algorithm, the time and space complexity of the algorithm and they have shown that their algorithm better than k- mean algorithm.

Yan Gao, Shiwen Gu, Liming Xia and Yaoping Fei (2006), In this paper, they have proposed a new multi-view information bottleneck algorithm (MVIB) that extends information bottleneck algorithm to multi-view setting to

cluster multi-representative instances. By using two important conditions of multi-view learning, conditional independence and compatibility, the compatible constraint maximizing the agreement between clustering hypotheses on different views is imposed on the individual views to cluster instances. Based on the compatible constraint, they have obtained the set of clustering hypotheses revealing lots of information about the correct one. The final hypothesis can be deduced from these hypotheses. They have applied MVIB to cluster web documents, and analyzed the performance of using different multiple features sets to cluster web documents.

III. METHODOLOGY

A. *Clustering*

Cluster analysis is one of the major data analysis methods widely used for many practical applications in emerging areas. Clustering is the process of finding groups of objects such that the objects in a group will be similar to one another and different from the objects in other groups. A good clustering method will produce high quality clusters with high intra-cluster similarity and low inter-cluster similarity. The quality of a clustering result depends on both the similarity measure used by the method and its implementation and also by its ability to discover some or all of the hidden patterns.

B. *K-means*

K-means clustering is one of the unsupervised computational methods used to group similar objects in to smaller partitions called clusters so that similar objects are grouped together. The algorithm aims to minimize the within cluster variance and maximize the intra cluster's variance. The K-means clustering algorithm is one of the simplest clustering algorithms in which the number of clusters to be grouped is fixed a priori by the user. The algorithm proceeds by randomly defining *k* centroids and assigning a document to the cluster that has the nearest centroid to the document [2]. Then, for every data point, the minimum distance is determined and that point is assigned to the closest cluster. This step is called cluster assignment, and is repeated until all of the data points have been assigned to one of the clusters. Finally, the mean for each cluster is calculated based on the accumulated values of points in each cluster and the Number of points in that cluster. Those means are then assigned as new cluster Centroids, and the process of finding distances between each point and the new centroids is repeated, where points are re-assigned to the new closest clusters. The process iterates for a fixed number of times, or until points in each cluster stop moving across to different clusters. This is called convergence.

Euclidean metric is considered as it is one of the widely used distance metrics incorporated with K-means clustering and one that is easy to implement. Also it results in a best solution. Euclidean distance is given in below:

$$D(P, C) = \sqrt{\sum_i^n |P - C|^2}$$

Where P is the data point, C is the cluster center, and n is the number of features.

The algorithm is composed of the following steps [3]:

1. Place *k* points into the space represented by the documents that are being clustered. These points represent initial group centroids.

2. Assign each document to the group that has the closest centroid.

3. When all documents 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 documents into groups from which the metric to be minimized can be calculated.

C. *Kea: Automatic Keyphrase Extraction*

Keyphrase extraction is a method of automatically extracting important phrases from text mainly for document summarization [4, 5].*Kea* is a Java-based keyphrase extraction algorithm that generates candidate phrases from a document and selects keyphrases from them by using TF-IDF and naive Bayes classifier [6].

Kea's extraction algorithm has two stages:

1. Training: The training stage uses a set of training documents for which the author's keyphrases are known. For each training document, candidate phrases are identified and their feature values are calculated. Each phrase is then marked as a keyphrase or a no Keyphrase, using the actual keyphrases for that document.

2. Extraction: In the extraction phase, the algorithm chooses keyphrases from a new document using the model. To select keyphrases from a new document, Kea determines candidate phrases and feature values, and then applies the model built during training. The model determines the overall probability that each candidate is a keyphrase, and then a post processing operation selects the best set of keyphrases.

D. *F-measure*

The second external quality measure is the F-measure, a measure that combines the precision and recall ideas from information retrieval. We treat each cluster as if it were the result of a query and each class as if it were the desired set of documents for a query. We then calculate the recall and precision of that cluster for each given class. More specifically, for cluster *j* and class *is given in* Eq. given below,

$$\text{Recall}(i, j) = C_{ij}/C_i$$

$$\text{Precision}(i, j) = C_{ij}/C_i$$

The F measure of cluster *j* and class *i* is then given by Eq. given below,

$$F(i,j) = \frac{2 * Recall(i,j) * Precision(i,j)}{Precision(i,j) + Recall(i,j)}$$

Where C_{ij} is the number of members of topic i in cluster j, C*j is* the number of members of cluster j and C*i is* the number of members of topic i.

For an entire hierarchical clustering the F measure of any class is the maximum value it attains at any node in the tree and an overall value for the F measure is computed by taking the weighted average of all values for the F measure as given by the following.

$$F = \sum_{i} \frac{n_i}{n} \max\{F(i,j)\}$$

Where the max is taken over all clusters at all levels, and *n* is the number of documents.

E. Dataset

This project uses the Reuter dataset. Reuters-21578 is a collection of 21,578 newswire articles. The articles are assigned classes from a set of 119 topic categories. This project has extracted two smaller sample data from the Reuters-21578 for cluster evaluation [10].

IV. THE PROPOSED APPROACH

The following fig. shows the proposed model for the clustering of the web document.

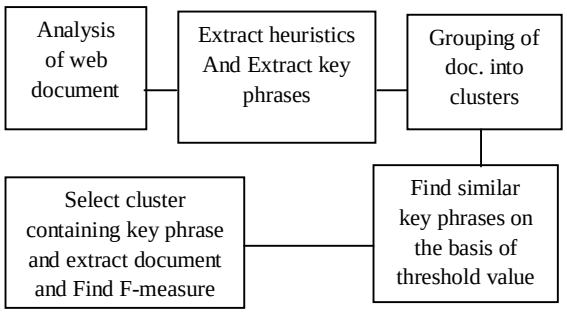

Fig.1: Proposed model of web document clustering

Proposed method involves following Analysis methods-

1. Extract Heuristics and Key Phrases from documents.
2. Determine the number of Clusters
3. Grouping of documents into clusters. (K-means)
4. Find Similar Key Phrases in each Cluster
5. Compare Similar Key Phrase with threshold
6. Select the Cluster where similar Key Phrase exceeds threshold. This is the Resulting Cluster containing the required document.
7. Find the F - measure of the result obtained to obtain the relevancy of the documents in clusters.

A. Extract key phrase

The first step in this project is to perform Key Phrase extraction using Kea algorithm. It includes the training process. which is based on KEA algorithm the application will develop a model for key phrase extraction process where candidate set are selected and features i.e. TF*IDF and First occurrence will be calculated to select the key phrases.

In this step of project, there are 2 stages:

1. Training and Extraction

Here it will take the input file then it will use the KEA-algorithm for extraction of key-phrases from the input document.

The candidate set can be selected by, first taking the document containg words then the stop words i.e. are typically used to filter out non-scientific English words that carry low domain -specific information content(remove irrelevant words). can be removed from that document and then stemming i.e. truncate suffixes and trailing numerals so that words having the same root (e.g., activate, activates, activation, and active). After this these words are stored into hash table those are called as a candidate set from which keyphreases are selected by using following properties.

For each individual word the 2 properties can be checked.

*2. TF*IDF*

TF-IDF combines term frequency (TF), which measures the number of times a word occurs in the document, and inverse document frequency (IDF), which measures the information content of a word - its rarity across all the families in the data set. The inverse document frequency (IDF) is calculated as:

$$idf^a = logN/df^a$$

Where idf^a denotes the inverse document frequency of word a in the data set; df^a denotes the number of documents in which word a is present ; and N is the total number of documents.

TFIDF is defined as:

$$Tfidf_d^a = tf_d^a \times idf^a$$

3. Place of occurrence

In this property it will check whether the word is present in the starting or at the end of the document.

By checking these properties for each individual word it will creates rule (heuristic) for building model which will be used for further KEA-means clustering.

B. Test document

In this step will perform the testing i.e. user will provide a test document, its key file and the model previously created, to the application. The application will extract key phrases and generate the number of clusters.
The number of clusters generated will be used to perform the document clustering.

C. *Document clustering*

This step of the project will measure the similarity based on the cosine similarity and Euclidian distance and display the similarity graphically dynamically while the clusters are being formed. Then measure the accuracy of the KEA-means algorithm after completing each iteration.

Here the Precision and Recall of each cluster formed and based on their values it will find the F - measure of the result obtained to obtain the relevancy of the documents in clusters.

D. F-measure

F – Measure (Maximum F-Measure) of the clusters formed on a graph dynamically for each cluster formed.

V. KEA-MEANS CLUATERING

The Kea-means clustering algorithm is a proposed new clustering method that improves the K-means algorithm by combining it with the Kea keyphrase extraction algorithm. The Kea-means clustering tries to solve the main drawback of K-means that the number of total clusters is pre-specified in advance.

In KEA-algorithm, documents are clustered in several groups like K-means but the number of clusters is determined automatically by the algorithm heuristically by using the extracted keyphreases.

Provides easy and efficient ways to extract test documents from massive quantity of resources. Kea mean clustering algorithm improves K- mean algorithm by combining it with the kea keyphrase extraction algorithm. This provides efficient way to extract test documents from massive quantity of resources. This algorithm develop faster algorithm for clustering.

Kea-means solves this problem by automatically determining this number the value of k is calculated at the run time. The system architecture of the Kea-means clustering is shown in Fig. 2 which shows the combination of k-means and the kea algorithm produces this KEA-Means algorithm.

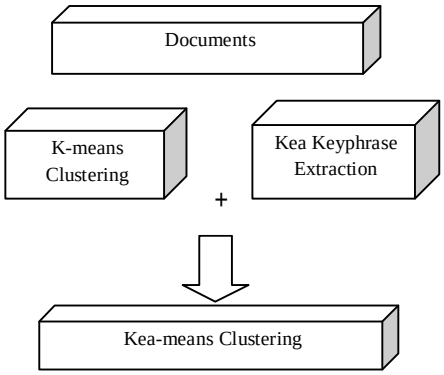

Fig2: Kea-means clustering system architecture

The K-means algorithm uses the cosine measure or the Euclidean distance measure to calculate feature values, the Kea-means clustering algorithm uses these two measures simultaneously. Hence, the similarity of two documents is computed by the following expression:

$$sim(d1, d2) = \frac{cosine(d1, d2)}{euclidean(d1, d2)}$$

VI. CONCLUSION

Keyphrase and K-mean clustering algorithm is important for obtaining the appropriate cluster context and the low quality clustering results will decrease extraction performance. Traditional K-means must specify the number of clusters *k* in advance by the user, which results in the change of clustering results or does not give the précised form in the clusters as the value of *k* changes. Kea-means solves this problem by automatically determining this number k. Kea mean clustering algorithm improves K-mean algorithm by combining it with the kea keyphrase extraction algorithm. This provides efficient way to extract test documents from massive quantity of resources. The proposed algorithm will show the better result as compare to the k-means algorithm and also the similarities and f-measure also calculated for the clusters generated.

ACKNOWLEDGMENTS

Our thanks to the professors and expert who gave guidance throughout implementing this project.

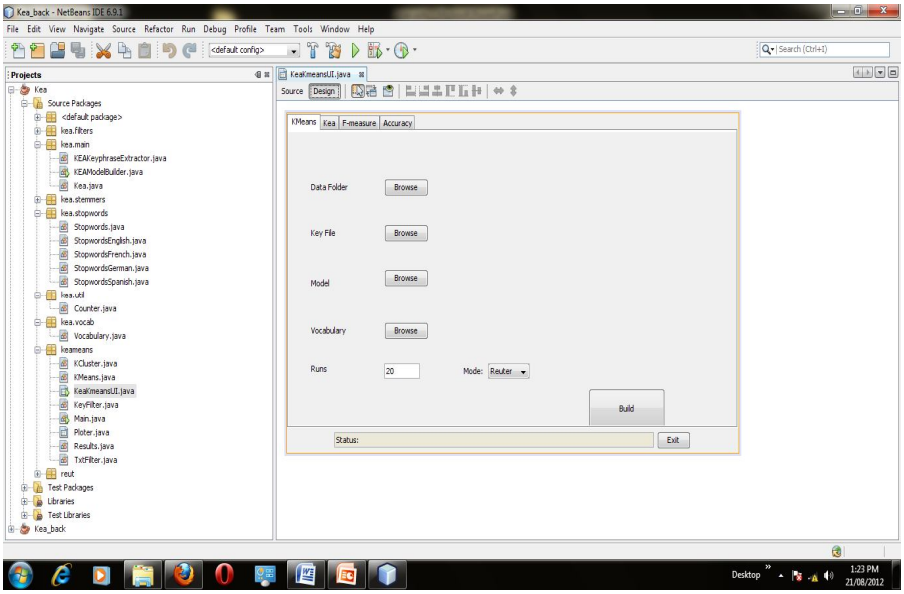

Fig.3 GUI of KEA-Kmeans clustering

REFERENCES

[1] Alexander S., Joydeep G. and Raymond M 2000.Impact of similarity measures on web page clustering. University of Texas at Austin, TX, 78712-1084, USA.

[2] M.Steinbach, G.Karypis, V.Kumar 2000.A comparison of document clustering techniques.proc.KDD Workshop on Text Mining, 1-20.

[3] Teknomo, Kardi. K-Means Clustering Tutorials. http:\\people.revoledu.com\kardi\tutorial\kMean\

[4] P.Turney 1999.Coherent keyphrase extraction via web mining", Technical Report ERB-1057, Institute for Information Technology, National Research Council of Canada.

[5] P.Turney 2003."Learning to extract keyphrases from text", proc.18th International Joint Conference on Artificial Intelligence (IJCAI), 434-439, 2003.

[6] Ian H. Witten, Gordon W. Paynter, Eibe Frank, Carl Gutwin and Craig G. Nevill-Manning 1999.KEA: Practical Automatic Keyphrase Extraction. Dept. of computer science university of Waaikato.

[7] Shen Huang, Zheng Chen, Yong Yu, and Wei Ying Ma. 2006. Multitype Features Coselection for Web Document Clustering. IEEE transactions on knowledge and data engineering, vol. 18, no. 4, April 2006.

[8] Shobha Sanjay Raskar and D.M. Thakore 2010. Kea-mean clustering approach for text mining. International Journal of Power Control Signal and Computation (IJPCSC) Vol. 2 No.

[9] Jiang- Chun Song, Jun-Yi Shen 2003.A web document clustering algorithm based on concept of neighbor. Proceedings of the Second International Conference on Machine Learning and Cybernetics Wan, 2-5 November 2003.

[10] D. Lewis. Reuters-21578 text categorization text collection1.0 .http://www.daviddlewis.com/resources/testcollections/reuters21578/