Web Document Clustering Using KEA-Means Algorithm

Swapnali Ware¹, N.A.Dhawas²

¹Department of Computer Engineering, Pune
²Department of Information Technology, Pune
SIT, Lonavala, India
¹Swapnali25.ware@gmail.com
²dhawasnitin@gmail.com

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 precise 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 proposes a new clustering algorithm known as KEA-Means algorithm which 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 and the k-means algorithm. This algorithm will automatically generate the number of clusters at the run time. This KEA-Means clustering algorithm provides easy and efficient way to extract test documents from massive quantities of resources.

1. 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 efficient 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.

2. Literature review

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. (We 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 or better than the hierarchical approaches that tested by them.
Jiang-chun Song, Jun-yi Shen (2003) [8], 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.

3. Methods

3.1. 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.

3.2. K-means

K-means clustering is one of the unsupervised computational methods used to group similar objects into 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 reassigned 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:

$$D(P, C) = \sqrt{\sum_{i=1}^{n}(P_i - C_i)^2}$$

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.

3.3. 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. 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.
3.4. 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 $i$ is given in Eq. given below,

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

$$\text{Precision}(i, j) = \frac{C_{ij}}{C_j}$$

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

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

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.

$$F = \sum \frac{n_j}{n} \max\{F(i, j)\}$$

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

3.5. Dataset

This project uses the Reuter dataset. Reuters-21578 is a collection of 21,578 newswire articles. This project has extracted two smaller sample data from the Reuters-21578 for cluster evaluation [9].

4. KEA-Means clustering

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. The main idea of Kea-means is the following: Initially, the number of clusters $k$ is set to 1. As in K-means, $k$ clusters are formed by generating $k$ centroids, and the similarities between centroids and documents are measured, and a document is assigned to the cluster that has the nearest centroid. Then, the Kea algorithm is applied to each document that is nearest to the corresponding centroid to extract keyphrases.

At this point, the number of clusters $k$ is determined, and the K-means algorithm is now used for actual clustering. The main characteristics of the Kea-means clustering algorithm can be summarized in two ways. First, compared with previous clustering algorithms in which the task of clustering is done without knowing the semantic nature of each cluster, the Kea-means clustering recognizes this semantic nature of clusters by using the keyphrases extracted from the documents in the cluster. Second, the traditional K-means must specify the number of clusters $k$ in advance by the user, which results in the change of clustering results as the value of $k$ changes.

Kea-means solves this problem by automatically determining this number. The system architecture of the Kea-means clustering is shown in Fig. 1.

![Fig1: Kea-means clustering system architecture](image)

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

$$\text{sim}(d_1, d_2) = \frac{\text{cosine}(d_1, d_2)}{\text{euclidean}(d_1, d_2)}$$
5. The proposed approach

The following fig. 3 shows the proposed model for the clustering of the web document.

The 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.

5.1. 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.

5.2. Testing documents

In this step will perform the testing i.e. we 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.

5.3. 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.

5.4. F-measure

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

6. Result and discussion

Here this project have created the GUI for the application and then 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 as shown in second screen shots given below. The testing i.e. project 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. And also, it will measure the similarity based on the cosine similarity and Euclidian distance and display the similarity graphically.
dynamically while the clusters are being formed. It will measure the accuracy of the KEA-means algorithm after completing each iteration. Here it will calculate 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.

The fig.4 shows the output graph for F-measure and the Similarity measure of the clusters formed per iterations will be displayed dynamically while the clustering process is going on.

- The red line represents the “F-Measure”.
- The blue line represents the “Similarity measure”.

7. 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 as the value of \( k \) changes. Kea-means solves this problem by automatically determining this number. 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.

Acknowledgments

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

References


