Enhancement in Data Mining Technique for Scattered Document Using Clustering

Sunita
Department of Computer Science and Engineering, Shri Baba Mast Nath Engineering College, Rohtak, India.

Rajiv Sharma
Department of Computer Science and Engineering, Shri Baba Mast Nath Engineering College, Rohtak, India.

Arvind
Department of Computer Science and Engineering, Shri Baba Mast Nath Engineering College, Rohtak, India.

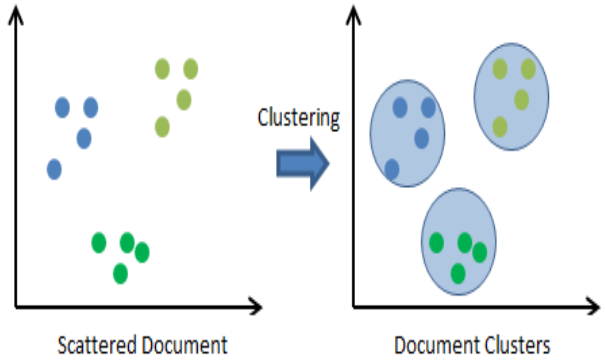
Abstract – Clustering is a widely studied data mining problem in the text documents. The problem finds numerous applications in customer segmentation, classification, collaborative filtering, visualization, document organization, and indexing. In this paper, we will provide a detailed survey of the problem of text clustering. We will study the key challenges of the clustering problem, as it applies to the text domain. We will discuss the key methods used for text clustering, and their relative advantages. We can also create data mining technique for scattered document using clustering.

Index Terms – Clustering, Data Mining, Text Documents, Segmentation, Classification, Filtering.

1. INTRODUCTION

Clustering is especially useful for organizing documents to improve retrieval and support browsing. The study of the clustering problem precedes its applicability to the text domain. With the huge upsurge of information in day-to-day's life, it has become difficult to assemble relevant information in nick of time. But people, always are in dearth of time, they need everything quick. Hence clustering was introduced to gather the relevant information in a cluster. There are several algorithms for clustering information out of which in this paper, we accomplish K-means clustering algorithm and a comparison is carried out to find which algorithm is best for clustering. On the best clusters formed, document summarization is executed based on sentence weight to focus on key point of the whole document, which makes it easier for people to ascertain the information they want and thus read only those documents which is relevant in their point of view. Document clustering has been investigated for use in a number of different areas of text mining and information retrieval. Initially, document clustering was investigated for improving the precision or recall in information retrieval systems and as an efficient way of finding the nearest neighbors of a document. More recently, clustering has been proposed for use in browsing a collection of documents or in organizing the results returned by a search engine in response to a user's query. Document clustering has also been used to automatically generate hierarchical clusters of documents.

For K-means clustering, the cosine measure is used to compute which document centroid is closest to a given document. While a median is sometimes used as the centroid for K-means clustering, we follow the common practice of using the mean. The mean is easier to calculate than the median and has a number of nice mathematical properties. For example, calculating the dot product between a document and a cluster centroid is equivalent to calculating the average similarity between that document and all the documents that comprise the cluster the centroid represents.



Clustering algorithms may be classified as listed below

1. Flat clustering (Creates a set of clusters without any explicit structure that would relate clusters to each other; It's also called exclusive clustering)
2. Hierarchical clustering (Creates a hierarchy of clusters)
3. Hard clustering (Assigns each document/object as a member of exactly one cluster)
4. Soft clustering (Distribute the document/object over all clusters)

2. RELATED WORK

The approach presented in [3] is to cluster multiple documents by using document clustering approach and to produce cluster wise summary based on feature profile oriented sentence extraction strategy. The related documents are grouped into same cluster using threshold-based document clustering algorithm. Feature profile is generated by considering word weight, sentence position, sentence length, sentence centrality, proper nouns and numerical data in the sentence. Based on the feature profile a sentence score is calculated for each sentence. This system adopts Term Synonym Frequency Inverse Sentence Frequency (TSF-ISF) for calculating individual word weight. According to different compression rates sentences are extracted from each cluster and ranked in order of importance based on sentence score. Extracted sentences are arranged in chronological order as in original documents and from this, cluster wise summary is generated. The output is a concise cluster-wise summary providing the condensed information of the input documents.

Kamal Sarkar presented an approach to Sentence Clustering-based Summarization of Multiple Text Documents in [4]. Here three important factors considered are:

(1) Clustering sentences

(2) Cluster ordering

(3) Selection of representative sentences from the clusters. For the sentence clustering the similarity histogram based incremental clustering method is used. This clustering approach is fully unsupervised & is an incremental dynamic method of building the sentence clusters. The importance of a cluster is measured based on the number of important words it contains. After ordering the clusters in decreasing order of their importance, top n clusters are selected. One representative sentence is selected from each cluster and included in to the summary. Selection of sentences is continued until a predefined summary size is reached. A query based document summarizer based on similarity of sentences and word frequency is presented in [5].The summarizer uses Vector Space Model for finding similar sentences to the query and Sum Focus to find word frequency. In this paper they propose a query based summarizer which being based on grouping similar sentences and word frequency removes redundancy. In the proposed system first the query is processed and the summarizer collects required documents & finally produces summary. After Pre-processing, producing the summary involves the following steps:

1. Calculating similarity of sentences present in documents with user query.

2. After calculating similarity, group sentences based on their similarity values.

3. Calculating sentence score using word frequency and sentence location feature.

4. Picking the best scored sentences from each group and putting it in summary.

5. Reducing summary length to exact 100 words.

3. CLUSTERING ALGORITHM

For all subsequent experiments, the standard K-means algorithm is chosen as the clustering algorithm. This is an iterative partitional clustering process that aims to minimize the least squares error criterion. As mentioned previously, partitional clustering algorithms have been recognized to be better suited for handling large document datasets than hierarchical ones, due to their relatively low computational requirements. The standard K-means algorithm works as follows. Given a set of data objects D and a pre-specified number of clusters k, k data objects are randomly selected to initialize k clusters, each one being the centroid of a cluster. The remaining objects are then assigned to the cluster represented by the nearest or most similar centroid. Next, new centroids are re-computed for each cluster and in turn all documents are re-assigned based on the new centroids. This step iterates until a converged and fixed solution is reached, where all data objects remain in the same cluster after an update of centroids. The generated clustering solutions are locally optimal for the given data set and the initial seeds. Different choices of initial seed sets can result in very different final partitions. Methods for finding good starting points have been proposed.

The centroids represent the members of their clusters, the squared distance of each vector from its centroid summed over all vectors.

$$\mathrm{RSS}_k = \sum_{\vec{x} \in \omega_k} |\vec{x} - \vec{\mu}(\omega_k)|^2$$

$$\mathrm{RSS} = \sum_{k=1}^{K} \mathrm{RSS}_k$$

Where

- ω_k Document cluster k
- $\vec{\mu}$ Mean or centroid of the documents in cluster ω_k
- $\vec{x}$ Document vector in cluster k

The algorithm then moves the cluster centers around in space in order to minimize RSS.

4. CONCLUSION

Data mining has a wide range of applications that are used for various purposes. One of the most popular clustering algorithm is k-means clustering algorithm, but in this method the quality

of the final clusters rely heavily on the initial centroids,which are selected randomly moreover,the k-means algorithm is computationally very expensive also. As the same enhanced method also choose the initial centroids based upon the random selection.so this method is very sensitive to the initial starting points and it does not promise to produce the unique clustering results. Finally this proposed method mainly focuses on the less similarity based clustering to find the initial cluster centers efficiently. This method also reduces time complexity. In this less similarity based clustering method the initial cluster centers will not be selected randomly so accuracy will be high. The experimental results show that proposed algorithm provides better results for various datasets.The value of k; desired number of clusters is still required to be given as an input to the proposed algorithm.

REFERENCES

[1] Dhillon, I. S., Fan, J. & Guan, Y. (2011). Efficient Clustering of Very Large Document Collections (Chapter 1). doi:10.1145/502512.502550
[2] Ding, C. & He, X. (2009). K-means Clustering via Principal Component Analysis, 225-232.
[3] Satheelaxmi, G., Murty, M. R., Murty, J. V. R. & Reddy, P. (2012). Cluster analysis on complex structured and high dimensional data objects using K-means and EM algorithm. International Journal of Emerging Trends & Technology in Computer Science, 1(1).
[4] Hu, G., Zhou, S., Guan, J. & Hu, X. (2008). Towards effective document clustering: A constrained K-means based approach. Information, Processing and Management, 44(4), 1397-1409.
[5] Jain, S., Aalam, M. A. & Doja, M. N. (2010). K-means Clustering Using Weka Interface. Proceedings of the 4th National Conference; INDIACom-2010. New Delhi: Bharati Vidyapeeth's Institute of Computer Applications and Management.
[6] Barioni, M. C. N., Razente, H. L., Traina, A. J. M. & Traina, C. Jr. (2006). An Efficient Approach to Scale Up K-medoid Based Algorithms in Large Databases.
[7] Wang, D., Zhu, S., Li, T., Chi, Y. & Gong, Y. (2008). Integrating Clustering and MultiDocument Summarization to Improve Document Understanding