International Journal of Engineering Science Invention (IJESI)
ISSN (Online): 2319 – 6734, ISSN (Print): 2319 – 6726
www.ijesi.org || PP. 36-41

Spectral Clustering Based Classification Algorithm for Text Classification

R.Suganthi[1], Dr.S.Manimekalai [2]
[1](*Department of Computer Science, TheivanaiAmmal college for Women, India)*
[2](*Department of Computer Science, TheivanaiAmmal College for Women, India)*

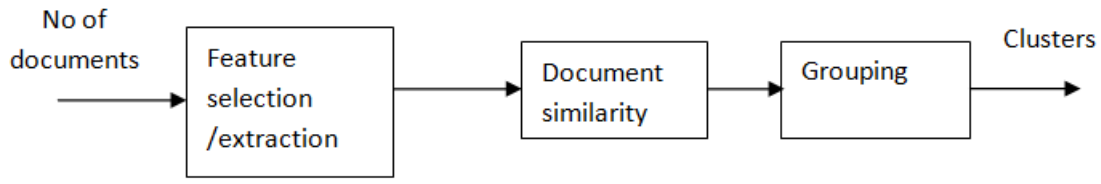
Abstract: *The main aim of text categorization is the classification of documents into a fixed number of predefined categories. In text categorization, the dimensionality of the feature vector is usually high. Various approaches have been proposed to reduce the dimensionality of the feature vector while performing automatic text categorization. We propose a Spectral Clustering-based Classification algorithm that reduces the dimensionality of a feature vector. In order to reduce the calculation cost, an incremental method is added in the present algorithm. The algorithm is applied to several text classification problems. The results show it is more effective and more accurate than the traditional active learning algorithm.*
Keywords: *Computational time,Data Search, Document clustering, spectral clustering model,World Wide Web.*

I. Introduction

Feature clustering is a powerful method to reduce the dimensionality of feature vectors for text classification. The existing system is fuzzy similarity-based self-constructing algorithm for feature clustering. The words in the feature vector of a document set are grouped into clusters, based on similarity test. Words that are similar to each other are grouped into the same cluster. Each cluster is characterized by a membership function with statistical mean and deviation. When all the words have been fed in, a desired number of clusters are formed automatically. We then have one extracted feature for each cluster. The extracted feature, corresponding to a cluster, is a weighted combination of the words contained in the cluster. By this algorithm, the derived membership functions match closely with and describe properly the real distribution of the training data. Besides, the user need not specify the number of extracted features in advance, and trial-and-error for determining the appropriate numbers of extracted features can then be avoided. Experimental results show that our method can run faster and obtain better extracted features than other methods.

This is important that words are necessary elements for any language documents such as natural languages (including Hindi or English) to represent its extract pattern. Each of the words in the content are analyzed by calculating its terms, frequencies and inverse frequencies and then stored in feature array. Once after designing the feature array, the similarity membership value is counted. This value lets us to find out the grouping of the documents. Similarity measures like cosine similarity and jaccard similarity measure are generally used for estimating the similarity score. Clustering is the process of arranging the received data into set of classes for easy retrieval task. The objective of the clustering is to find the similar groups that fit the topics. Document clustering is the sort of textual clustering that clumps the documents. Relied upon the topic, the documents are clumped.

Clustering technology belongs to the class of unsupervised system that assigns and allocates the Documents based on its similarity scores. Since, it's an unsupervised system, the documents are trained and its follows a certain rule for further clarification. Due to its simplicity and reliability, the clustering technologies are adopted by



Clustering is the process of making a group of abstract objects into classes of similar objects. The process of grouping a set of physical or abstract objects into classes of similar objects is called clustering. A cluster is a collection of data objects that are similar to one another within the same cluster and are dissimilar to the objects in other clusters. A cluster of data objects can be treated collectively as one group and so may be considered as a form of data compression. Although classification is an effective means for distinguishing

groups or classes of objects, it requires the often costly collection and labeling of a large set of training tuples or patterns, which the classifier uses to model each group.
Clustering is the process of making a group of abstract objects into classes of similar objects.

- A cluster of data objects can be treated as one group.
- While doing cluster analysis, we first partition the set of data into groups based on data similarity and then assign the labels to the groups.
- The main advantage of clustering over classification is that, it is adaptable to changes and helps single out useful features that distinguish different groups.

1.1 Applications of Cluster Analysis:

- Clustering analysis is broadly used in many applications such as market research, pattern recognition, data analysis, and image processing.
- Clustering can also help marketers discover distinct groups in their customer base. And they can characterize their customer groups based on the purchasing patterns.
- In the field of biology, it can be used to derive plant and animal taxonomies, categorize genes with similar functionalities and gain insight into structures inherent to populations.
- Clustering also helps in identification of areas of similar land use in an earth observation database. It also helps in the identification of groups of houses in a city according to house type, value, and geographic location.
- Clustering also helps in classifying documents on the web for information discovery.
- Clustering is also used in outlier detection applications such as detection of credit card fraud.
- As a data mining function, cluster analysis serves as a tool to gain insight into the distribution of data to observe characteristics of each cluster.

II. Problem Formation

Let us presume a set of documents D= {d1, d2, d3. dn} where N is the aggregated value of documents that splits into two semantic process of k sub-groups. The clustering process presumes as C= {c1, c2, ck} clusters, with each c_i being non -empty. Since, the document clustering is in its infancy stage, our research study paves a way for its revolution. It started off on the popular vector based approach where documents were treated as a bag of words and clustering criteria was the presence of common words in the documents. Several modifications were applied on this method to improve this method as the result set would only provide us information on what words were present in a group of documents, not the actual content or context of the documents. There was a need of more intuitive ways of clustering that would provide us sound knowledge of the content present inside the documents.

III. Related Work

The work of clustering is mainly affected by the better quality of the cluster formation. It affected by three factors, namely, data representation, similarity function and clustering model [3, 4, 5, and 6]. In recent times, the vector space model is widely used for the data representation of the document clustering systems. The words in the documents are stored in the vector space model. Generally, each word has TF and IDF values. Similarity between those values is studied for preparing the cluster index number. Every word has TF and IDF values. Similarity between any words is measured by the jaccard functions [7, 8]. In this method, only word represents whole document hence not finds many valuable information from word proximity [4]. Other methods based on Vector space model also not consider incremental processing [9]. Survey of web clustering engines that incremental processing increases the effectiveness while useful in the clustering schemes [10]. The most related work that takes into account the information about proximity of words and phrase based analysis in an incremental way is Suffix Tree clustering (STC) [11].Frequent item sets finding problem is explain in detailed paper [12]. Normally, frequent items are derived by association rules mining. As the technology grows, the method is able to estimate the frequent item sets under different forms. Paper [13] mention method for clustering documents by calculating neighbor function value from given document set. Another method for given in paper [14] for maximum frequent item set length finding. Document classification is also done by Gaussian membership value between documents. This value is helpful for clustering. In [15], the categorization of documents is complete by allowing for Gaussian relationship and making use of it to achieve clusters by discovery term. Every cluster is defined by its term behavior find by Fuzzy Gaussian relationship if cluster created. A novel technique known as Maximum Capturing is projected for text file clustering in [16]. Maximum Capturing involved two actions as decision document clusters and giving cluster members. In [17], algorithm to search for a pattern in a text is proposed which can be used to search for component of interest in the component repository. The hierarchical clustering makes use of non-hierarchical clustering model that estimates a value

known as centroid. It generally groups the documents that are in single cluster which contain lesser distance values [18]. This centroid estimation poses a research oriented challenges with different qualities [19]. Florian Beil et al introduced two clustering algorithms FTC (non-hierarchical) and HFTC (hierarchical) in [20] based on the concept of frequent Term Set and analysed their behaviour. They have explored about the Agglomerative Hierarchical Clustering (AHC), bottom-up clustering model that falls under the class of association rule mining. Many researchers have efficiently used this method in information retrieval and Web Clustering [20]. Most of the clustering technologies employs single linkage, complete linkage and cluster average linkage that seeks the similarity estimation to arrange the data into pair of clusters. A semantic based distance metrics are used for estimating the terms and words of the documents. Rudi L. Cilibrasiet. al. introduced a replacement similarity, called Normalized Google Distance(NGD) [20], to effectively capture the semantic similarity between words and phrases based on information distance and Kolmogorove Complexity. Later, Alberto J Evangelista et. al. reviewed the work of Rudi L. Cilibrasi to boost their distance operate through elimination of random data. We tend to adopt this technique to estimate the similarity between among terms clusters rather than just two words.

IV. Spectral Clustering

We described clustering a data set by creating a Markov chain based on the similarities of the data items with one another, and analyzing the dominant eigenvectors of the resulting Markov matrix. In this section, we show how to classify a data set by making two changes. First, we modify the Markov chain itself by using class labels, when known, to override the underlying similarities. Second, we use a classification algorithm in the spectral space rather than a clustering algorithm.

4.1 Spectral Clustering Classification Algorithm

Again, if natural classes occur in the data, the Markov chain described above should have cliques. Furthermore, the cliques will become stronger as the number of labeled documents increases. Given this model, we wish to categorize documents by assigning them to the appropriate clique in the Markov chain. The spectral clustering methods given in Section 3 can be adapted to do classification by replacing the final few steps (clustering in spectral space)(which classify in spectral space). The key differences between the spectral classifier and the clustering algorithm are (a) that our transition matrix A incorporates labeling information, and (b) we use a classifier in the spectral space rather than a clustering method. What is novel here is that this algorithm is able to classify documents by the similarity of their transition probabilities to known subsets of B. Because the model incorporates both labeled and unlabeled data, it should improve not only with the addition of labeled data, but also with the addition of unlabeled data.

Form spectral representation:

1. Given data B, form the affinity matrix A 2 Rn×n = f (B).
2. Define D to be the diagonal matrix with Dii = Pj Ai j .
3. Normalize: N = (A + dmax I − D)/dmax .
4. Find x1, . . . ,xk , the k largest eigenvectors of N and form the
matrix X = [x1, . . . , xk] 2 Rn×k−1.
5. Normalize the rows of X to be unit length.

For clustering:

6. Treating each row of X as a point in Rk , cluster into k clusters using k-means or any other sensible clustering algorithm.
7. Assign the original point xi to cluster j if and only if row i of
X was assigned to cluster j .

For classification:

6. Represent each data point i by the row Xi of X.
7. Classify these rows as points in Rk using any reasonable classifier,trained on the labeled points.
8. Assign the data point i the class c that Xi was assigned

4.2Clustering Algorithms

Clustering is thus preformed after the documents matching the query are identified. Consequently, the set of thematic categories is not fixed they are created dynamically depending on the actual documents found in the results. Secondly, as the clustering interface is part of a search engine, the assignment of documents to groups must be done efficiently and on-line. Hence, the documents are accessible from the clustering technologies from variant snippets.

Clustering algorithms can be broadly classified into two categories:

1)Unsupervised linear clustering algorithms

2) Unsupervised non-linear clustering algorithms

V. Unsupervised Linear Clustering Algorithms

5.1Gaussian (Em) Clustering Algorithm

This algorithm assumesapriori that there are 'n' Gaussian and then algorithm try to fits the data into the 'n'Gaussian by expecting the classes of all data point and then maximizing the maximum likelihood of Gaussian centers.

5.2 K-Means Algorithm

K-means is one of the simplest unsupervised learning algorithm. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed Apriori. The main idea is to define k centers, one for each cluster .These centers should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest center. When no point is pending, the first step is completed and an early group age is done. At this point we need to re-calculate k new centroids as barycenter of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new center. A loop has been generated. As a result of this loop we may notice that the k centers change their location step by step until no more changes are done or in other words centers do not move any more.

5.3 Fuzzy C-Means Clustering

This algorithm works by assigning membership to each data point corresponding to each cluster center on the basisof distance between the cluster center and the data point. More the data is near to the cluster center more is its membership towards the particular cluster center. Clearly, summation of membership of each data point should be equal to one.

5.4 Hierarchical Clustering

Hierarchical clustering algorithm is of two types:
i) Agglomerative Hierarchical clustering algorithm or AGNES (agglomerative nesting) and
ii) Divisive Hierarchical clustering algorithm or DIANA (divisive analysis).
Both this algorithm are exactly reverse of each other. So we will be covering Agglomerative Hierarchical clustering algorithm in detail.

Agglomerative Hierarchical clustering -This algorithm works by grouping the data one by one on the basis of the nearest distance measure of all the pair wise distance between the data point. Again distance between the data point is recalculated but which distance to consider when the groups has been formed? For this there are many available methods. Some of them are:
1) single-nearest distance or single linkage.
2) complete-farthest distance or complete linkage.
3) average-average distance or average linkage.
4) Centroids distance.
5) ward's method - sum of squared Euclidean distance is minimized.

This way we go on grouping the data until one cluster is formed. Now on the basis of dendogram graph we can calculate how many numbers of clusters should be actually present.

5.5 Quality Threshold (Qt) Clustering

This algorithm requires the apriori specification of the threshold distance within the cluster and the minimum number of elements in each cluster. Now from each data point we find all its candidate data points. Candidate data points are those which are within the range of the threshold distance from the given data point. This way we find the candidate data points for all data point and choose the one with large number of candidate data points to form cluster. Now data points which belongs to this cluster is removed and the same procedure is repeated with the reduced set of data points until no more cluster can be formed satisfying the minimum size criteria.

VI. Unsupervised Non Linear Clustering Algorithms

6.1 Mst Based Clustering

The basic idea of MST based clustering algorithm is as follows:

First construct MST (minimum spanning tree) using Kruskal algorithm and then set a threshold value and step size. We then remove those edges from the MST,whose lengths are greater than the threshold value. We next calculate the ratio between the intra-cluster distance and inter-cluster distance and record the ratio as well as the threshold. We update the threshold value by incrementing the step size. Every time we obtain the new (updated) threshold value, we repeat the above procedure. We stop repeating, when we encounter a situation such that the threshold value is maximum and as such no MST edges can be removed. In such situation, all the data points belong to a single cluster. Finally we obtain the minimum value of the recorded ratio and form the clusters corresponding to the stored threshold value. The above algorithm has two extreme cases:

1) With the zero threshold value, each point remains within a single cluster.
2) With the maximum threshold value all the points lie within a single cluster.

Therefore, the proposed algorithm searches for that optimum value of the threshold for which the Intra-Inter distance ratio is minimum. It needs not to mention that this optimum value of the threshold mustlie between these two extreme values of the threshold. However, in order to reduce the number of iteration we never set the initial threshold value to zero.

6.2 Kernal K-Means Clustering

This algorithm applies the same trick as k-means but with one difference that here in the calculation of distance, kernel method is used instead of the Euclidean distance.

Advantages:

- Algorithm is able to identify the non-linear structures.
- Algorithm is best suited for real life data set.

Disadvantages:

- Number of cluster centers need to be predefined.
- Algorithm is complex in nature and time complexity is large.

6.3 Density Based Clustering

Density based clustering algorithm has played a vital role in finding non linear shapes structure based on the density. Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is most widely used density based algorithm. It uses the concept of density reach ability and density connectivity. **Density Reach ability** - A point "p" is said to be density reachable from a point "q" if point "p" is within ε distance from point "q" and "q" has sufficient number of points in its neighbors which are within distance ε. **Density Connectivity** - A point "p" and "q" are said to be density connected if there exist a point "r" which has sufficient number of points in its neighbors and both the points "p" and "q" are within the ε distance. This is chaining process. So, if "q" is neighbor of "r", "r" is neighbor of "s", "s" is neighbor of "t" which in turn is neighbor of "p" implies that "q" is neighbor of "p".

6.4 Structural And Textual Feature Extraction For Semi-Structured Document

Author of paper give classification method for XML documents using two different methods one is structural and second is content based feature selection method. This method is making best use of structured present in document for similarity measure between two structured documents as text information present in XML file. Author proposed a new framework for document classification, XML document classification, based on extracting features from different aspects of the document. Extracted a feature vector that represents the document in a compact format; it captures valuable information that can be used later to build an accurate classifier using any of the well-known classification techniques. It also proposed a fast, robust, accurate, and novel approach for content-based document classification. Our proposed model is easy to implement for real world applications.

6.5correlation Similarity Measure

This Paper offers a novel document clustering technique derived from correlation indexing maintain. It at the same time take full advantage of the correlation among the files within the limited patches and reduce the correlation among the files outside these patches. Accordingly, a low down dimensional semantic subspace is resultant wherever the files equivalent to the matching semantics are near to both further. Extensive experiments

on NG20, Reuters, and OHSUMED corpora show that the proposed CPI method outperforms other classical clustering methods. Furthermore, the CPI method has good generalization capability and thus it can effectively deal with data with very large size.

6.6Hybrid Xnor Similarity Function

The paper defines a new similarity function to compute similarity between any two software components or text files. An algorithm to cluster a set of given documents or text files or software components is designed which uses the proposed similarity function called hybrid XNOR to find the degree of similarity among any two entities. The input to algorithm is a similarity matrix and the output is the set of clusters.

VII. Conclusion

One of the main problems in text classification is a large number of root words after the stemming process that yields high dimensionality of the feature space. We reduce the dimensionality of feature space by using spectral clustering. Spectral clustering groups the root words in to clusters based on the membership function along with its mean and standard deviation. We have presented a spectral clustering based classification algorithm, which is an incremental clustering approach to reduce the dimensionality of the features in text classification. Features that are similar to each other are grouped into the same cluster. Each cluster is characterized by a membership function with statistical mean and deviation. If a word is not similar to any existing cluster, a new cluster is created for this word. Similarity between a word and a cluster is defined by considering both the mean and the variance of the cluster. The proposed scheme has also been extended to measure the similarity between two sets of documents. To improve the efficiency, we have provided an approximation to reduce the complexity involved in the computation.

References

[1]. A.B. Smith, C.D. Jones, and E.F. Roberts, "Article Title", Journal, Publisher, Location, Date, pp. 1-10.

[2]. Jones, C.D., A.B. Smith, and E.F. Roberts, Book Title, Publisher, Location, Date. Crabtree, D., Gao, X., Andreae, P.: "Improving web clustering by cluster selection". In: Proceedings of 2005 IEEE/WIC/ACM International Conference on Web Intelligence, pp. 172–178 (2005).

[3]. Muhammad Rafi, Mehdi Maujood, MurtazaMunawarFazal, Syed Muhammad Ali, "A comparison of two suffix tree-based document clustering algorithms", IEEE, 2010.

[4]. Hammouda, K., Kamel, M.: "Efficient document indexing for web document clustering". IEEE Transactions on Knowledge and Data Engineering 16(10), 1279–1296 (2004)

[5]. Hammouda, K., Kamel, M.: "Phrase-based document similarity based on an index graph model". In: Proceedings of 2002 IEEE International Conference on Data Mining ICDM, pp. 203–210 (2002).

[6]. Chim, H., Deng, X.: "A new suffix tree similarity measure for document clustering". In: Proceedings of the 16th International Conference on World Wide Web, WWW 2007, pp. 121–130. ACM, New York (2007)

[7]. Chim, H., Deng, X.: "Efficient phrase-based document similarity for clustering". IEEE Transactions on Knowledge and Data Engineering 20(9), 1217–1229 (2008).

[8]. Huang, A.: "Similarity measures for text document clustering", pp. 49–56 (2008)

[9]. Joydeep, A.S., Strehl, E., Ghosh, J., Mooney, R.: "Impact of similarity measures on web-page clustering". In: Workshop on Artificial Intelligence for Web Search, AAAI, pp. 58–64 (2000).

[10]. Janruang, J., Guha, S.: Semantic suffix tree clustering. In: First IRAST International Conference on Data Engineering and Internet Technology, DEIT (2011).

[11]. Carpineto, C., Osinski, S., Romano, G., Weiss, D.: A survey of web clustering engines. ACM Computing Surveys 41, 1–38 (2009)

[12]. Zamir, O., Etzioni, O.: Grouper: A dynamic clustering interface to web search results. In: Proceedings of the Eighth International World Wide Web Conference, pp. 283–296. Elsevier, Toronto (1999).

[13]. R. Agrawal, T. Imielinski, A. Swami. Mining association rules between sets of items in very large databases, Proceedingsof the ACM SIGMOD Conference on Management of data, 1993, pp. 207–216.

[14]. CongnanLuo, Yanjun Li, Soon M. Chung. Text document clustering based on neighbors, Data & Knowledge Engineering (68), 2009,1271–1288.

[15]. TianmingHu,Sam Yuan Sung, HuiXiong, Qian Fu. Discovery of maximum length frequent itemsets, Information Sciences (178), 2008,69–87.

[16]. Jung-Yi Jiang et.al A Fuzzy Self-Constructing Feature Clustering Algorithm for Text Classification, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 23, NO. 3, MARCH 2011.

[17]. Wen Zhanga, Taketoshi Yoshida, Xijin Tang, Qing Wang. Text clustering using frequent itemsets, Knowledge-Based Systems 23 (2010) 379–388.

[18]. Radhakrishna.V, C. Srinivas, C.V. Guru rao. High Performance Pattern Search algorithm using three sliding windows, International Journal of Computer Engineering and Technology, Volume 3, issue 2, 2012, pages 543-552. Impact factor 3.85.

[19]. Beil, F., Ester, M., Xu, X.: Frequent term-based text clustering. In: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2002, pp. 436–442. ACM, New York (2002)

[20]. Cilibrasi, R., Vitanyi, P.: The google similarity distance. IEEE Transactions on Knowledge and DataEngineering 19(3), 370–383 (2007).