TERM BASED SIMILARITY MEASURE FOR TEXT CLASSIFICATION AND CLUSTERING USING FUZZY C-MEANS ALGORITHM

D. Renukadevi , S. Sumathi

Abstract— **The progress of information technology and increasing usability of internet are drastically changing all fields of activity in modern days. As a result, a very large number of people would be required to interact more frequently with computer systems. To make the man–machine interaction more effective in such situations, it is desirable to have systems capable of handling inputs in a variety of forms, such as printed/handwritten paper documents. The computer have to efficiently process the scanned images of printed documents, the techniques need to be more sophisticated. The text documents are pre-processed, Term Frequency and Inverse Document Frequency (TF - IDF) are used to rank the document. Finally the similar information is grouped together using Fuzzy C – Means Clustering algorithm.**

Index Terms— Document clustering, Term Frequency – Inverse Document Frequency, similarity measures, Fuzzy C-Means Clustering Algorithm.

I. INTRODUCTION

Text Mining is automatically extracting information from different textual resources. The goal of text mining is to discover previously unknown Information. The challenges that arise due to unstructured text are large textual database; all publications are also in electronic form, Very high number of possible word and phrase types in the language, Complex and subtle relationships between concepts in text. Information Extraction (IE) is the task of automatically extracting structured information from unstructured and/or semi-structured machine-readable documents. This activity concerns processing human language texts by means of natural language processing (NLP). The overall goal is to create a more easily machine-readable text to process the sentences. Text categorization can be used in applications where there is a flow of dynamic information that needs to be organized. Dynamic information as email, news articles, blogs, patents and legal data. Application includes the automatic routing of customer support requests, tagging medical claims and tracking entities in the flow of information.

Document clustering or Text clustering is an automatic document organization, topic extraction and fast information retrieval or filtering. It is closely related to data clustering. Clustering methods can be used to automatically group the retrieved documents into a list of meaningful categories. Document clustering involves the use of descriptors and descriptor extraction. Descriptors are sets of words that describe the contents within the cluster. Document clustering is generally considered to be a centralized process. Examples of document clustering include web document clustering for search users.

Clustering involves the task of dividing data points into homogeneous classes or clusters so that items in the same class are as similar as possible and items in different classes are as dissimilar as possible. Clustering can also be thought of as a form of data compression, where a large number of samples are converted into a small number of representative prototypes or clusters. Depending on the data and the application, different types of similarity measures may be used to identify classes, where the similarity measure controls how the clusters are formed. Some examples of values that can be used as similarity measures include distance, connectivity, and intensity. Fuzzy c-means algorithm is method of clustering which allows the one piece of data to belong to two or more cluster. This algorithm gives best result for overlapped data set and comparatively better than k-means algorithm.

The rest of the paper is organized as follows. In Section 2 deals with the related work of the project. This chapter describes the previous existing works related to the project. Chap 3 explains the proposed work and deals with the system design of the project. This chapter gives the module description. This includes the step by step procedure of the modules and detailed design of the project. Chapter 4

ISSN: 2278 – 7798

consists of empirical evaluation of the project. Chapter 5 concludes the project.

II. BACKGROUND AND RELATED WORK

Clustering deals with finding a structure in a collection of unlabeled data. A cluster is therefore a collection of objects, which are 'similar' between them and are 'dissimilar' to the objects belonging to other clusters. Clustering of document is an automatic grouping of text documents into clusters such that documents within a cluster have high resemblance in comparison to one another. Fuzzy c means clustering is one of the most popular clustering algorithms. It uses to constraints the membership function. Information that characterize a given knowledge documents are somehow associated with each other. Those information of the documents are related to other documents. Hence, documents may contain information that is relevant to different domains to some degree. Fuzzy clustering methods are used to group the similar information based on their similarity.[9] Proposed to calculate the performance by using entropy and purity. The fuzzy c mean algorithm provide a good efficiency for the document clustering in comparison with the clustering Purity and the entropy accuracy.[11] proposed to given the efficiency of feedback retrieval from the user needing query information. [10] Proposed to give a good probability to the cluster the similar information. [1] Proposed to measure the performance by using precision and recall. This proposed method increasing the performance of document clustering. But this method doesn't give high dimensionality and accuracy. [8] Proposed to identify the optimal conceptual word weight for efficient clustering of text documents. When weighed by the concept, the clustering system can improve the accuracy and performance of text documents. [12] Proposed to increase the similarity measure for collection of the documents. [13] proposed to improve the text clustering quality. The quality of text clustering achieved by this model significantly surpasses the traditional single term-based approaches.

III. PROPOSED WORK

In the Proposed system the information extracted documents are pre-processed and Term Frequency – Inverse Document Frequency technique is used to calculate frequency weight. Then the documents are ranked. The Fuzzy C- Means (FCM) clustering algorithm is used cluster the similar documents**.**

3.1. DOCUMENT PRE-PROCESSING

a. Stop Words Removal

Many words are not informative and thus irrelevant are remove from the document representation. (e.g.) the, a, an, and, there, their, is, was, were, where, etc. These word typically about 400 to 500. It is used to improve the efficiency and potential problems of removing stop words.

b. Stemming

Reducing words from their root form. A document may contain several occurrences of words like fish, fishes and fishers. An advantage of stemming is to improve the effectiveness to match similar words. Reduce indexing size to combing words with same roots may reduce indexing size as much as 40-50%. Porter algorithm is used to stem the words.

Porter Algorithm:

Step 1: Gets rid of plurals and -ed or -ing suffixes
Step 2: Turns terminal y to i when there is another vowel in the stem
Step 3: Maps double suffixes to single ones:
-ization, -ational, etc.
Step 4: Deals with suffixes, -full, -ness etc.
Step 5: Takes off -ant, -ence, etc.
Step 6: Removes a final -e

3.2. DOCUMENT INDEXING

The Extracted text documents are converted into Boolean weighting by using the indexing technique of Term Frequency – Inverse Document Frequency. TF–IDF is the product of two statistics, term frequency and inverse document frequency. The **term frequency** tf (t, d), the simplest choice is to use the raw frequency of a term in a document, i.e. the number of times that term t occurs in document d. The raw frequency of t by f (t, d), then the simple tf scheme is tf (t, d) = f (t, d). Other possibilities include

- Boolean "frequencies": tf (t, d) = 1 if t occurs in d and 0 otherwise;
- Logarithmically scaled frequency: tf (t, d) = log (f (t, d) + 1);
- Augmented frequency, to prevent a bias towards longer documents

tf (t, d) = 0.5+ 0.5 $\times ft\ (t,d)\max\{f\ w,d\ :w\epsilon d\}$

The **inverse document frequency** is a measure of whether the term is common or rare across all documents. It is obtained by dividing the total number of documents by the number of documents containing the term, and then taking the logarithm of that quotient.

Id (t, D) = log$|D||\ d\epsilon D : t\epsilon d\ |$

|D|: cardinality of D, or the total number of documents in the corpus

$|\ \{\in D:t\in d\}|$: number of documents where the term t appears. If the term is not in the corpus, this will lead to a division-by-zero. It is therefore common to adjust the formula to 1+$|\ \ \{\in D:t\in d\}|$

Mathematically the base of the log function does not matter and constitutes a constant multiplicative factor towards the overall result.
Then TF–IDF is calculated as
tf – idf (t, d, D) = tf (t, d) × idf (t, D)

3.3. Similarity Measure

K-Nearest Neighbor is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure. A similarity measure can represent the similarity between two documents. Similarity measure is a function which computes the degree of similarity between a pair of text objects. The similarity distance is like Euclidean distance, Manhattan, Minkowski etc. The Euclidean distance between two documents are calculated as

$$\text{Dis}(d_i, d_j) = \sqrt{\sum_{i=1}^{k}(d_i - d_j)^2}$$

D is a set, which contains m text documents;
D = {d1, d2,…di..dm} I = 1, 2,…m
There are n words among m text documents.
di = {wi1, wi2,….wij ,….. win }
i = 1, 2, m, j = 1, 2, n

Cosine Similarity measures the cosine of the angle between two documents d1 and d2 as

$$S_{cos}(d_1, d_2) = \frac{d_1.d_2}{(d_1.d_1)^{1/2}(d_2.d_2)^{1/2}}$$

Pairwise-adaptive similarity dynamically selects a number of features out of d1 and d2 is defined as

$$S_{pair}(d_1, d_2) = \frac{d_{1,K}.d_{2,K}}{(d_1.d_1)1/2(d_2.d_2)1/2}$$

Where $d_{i,K}$ is a subset of d_i, i=1,2… containing the values of the features which are the union of K largest features appearing in document d_1 and d_2.
The Extended Jaccard coefficient is an extended version of the jaccard coeficient that is represented in

$$S_{EJ}(d_1, d_2) = \frac{d_1.d_2}{(d_1.d_1)+(d_2.d_2)-(d_1-d_2)}$$

Dice similarity also used to measure the similarity between the two documents. It is represented as

$$S_{Dic}(d_1, d_2) = \frac{2d_1.d_2}{d_1.d_1+d_2.d_2}$$

3.5. FUZZY C – MEANS CLUSTERING

Fuzzy c-mean is a method of clustering which allows one piece of data to belong to two or more cluster. This algorithm works by assigning membership to each data point corresponding to each cluster center on the basis of distance between the cluster center and data point. Here μ_{ij} is represents the membership of $i^{th\ data}$ to j^{th} cluster center. The main objective of fuzzy c-means algorithm is to minimize

$$J(U, V) = \sum\sum(\mu_{ij})^m \|x_i - v_j\|^2$$

Where $\|x_i - v_j\|$ is the Euclidean distance between i^{th} and j^{th} cluster center.
Algorithm steps for Fuzzy c-means clustering
Let X = {x_1, x_2, x_3....,x_n} be the set of data points and V = {v_1, v_2, v_3….,v_c}be the set of centers.
1. Randomly select 'c' cluster centers.
2. Calculate the fuzzy membership 'μ_{ij}' using

$$\mu_{ij} = 1/\sum_{k=1}^{c}\left(d_{ij}/d_{ik}\right)^{\left(\frac{2}{m}-1\right)}$$

3. Compute the fuzzy membership 'μ_{ij}' using

$$V_j = \frac{\left(\sum_{i=1}^{n}(\mu_{ij})^m x_i\right)}{\left(\sum_{i=1}^{n}(\mu_{ij})^m\right)}, \forall_j = 1, 2,\ldots..c$$

4. Repeat step 2 and 3 until the minimum 'J' value is achieved or $\|U^{(k+1)} - U^{(k)}\| < \beta$.

Where,
'k' is the iteration step.
'β' is the termination criterion between [0,1].
'U = $(\mu_{ij}')_{n*c}$ is the fuzzy membership matrix
'J' is the objective function.

Figure 3.5.1 Fuzzy c-means algorithm

IV. EXPERIMENTAL RESULT

Distribution of documents per class in reuters-21578.

Class	Training documents	Testing documents	Subtotal of documents
acq	1596	696	2292
crude	253	121	374
grain	41	10	51
interest	190	81	271
ship	108	36	144
Total	2188	944	3132

Reuters – 21578 dataset is used in this experiment. The documents in the Reuters -21578 collection appeared on the Reuters newswire articles in 1987. It contains 22 files and each of the first 21 file contains 1000 documents and last one contains 578 documents. The documents are read from the dataset and then text pre-processing is performed.

4.1 Pre-processing

ISSN: 2278 – 7798

Pre-processing, remove the stop words from the document and porter algorithm is used to reduce the word from root.

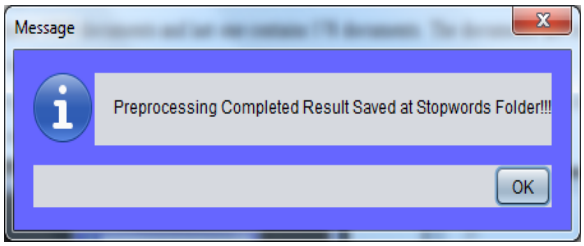

Figure 4.1a Pre-processing

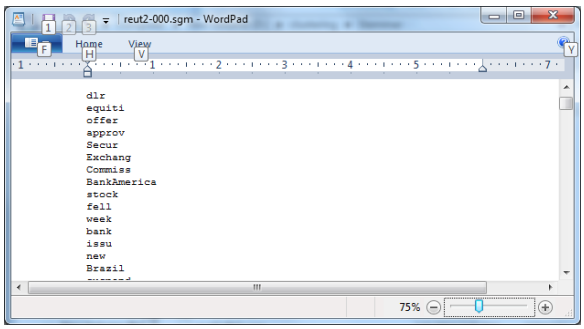

Figure 4.1b pre-processing result stored in folder

4.2 Document Indexing

Term Frequency and inverse document frequency technique is used to calculate the word of the document. These can be used in future process.

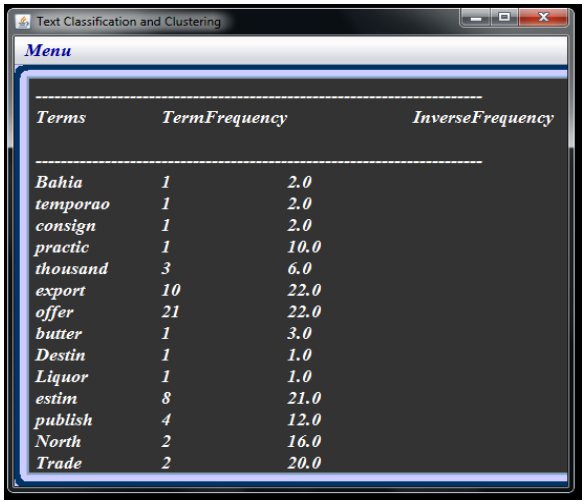

Figure 4.2 TF-IDF

4.3 Document Ranking

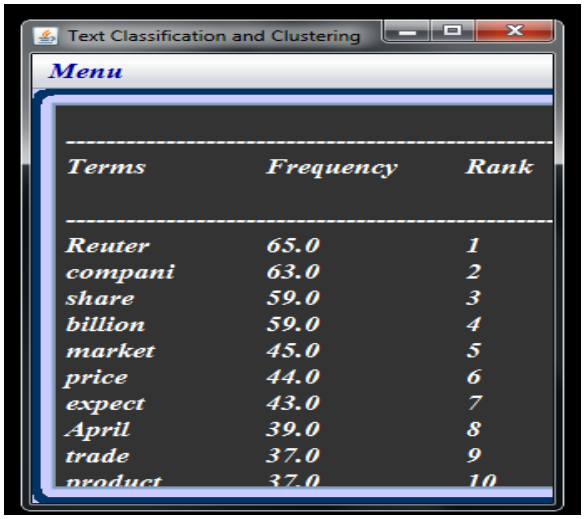

Figure 4.3 Document Ranking

4.4 K-NN Classifier

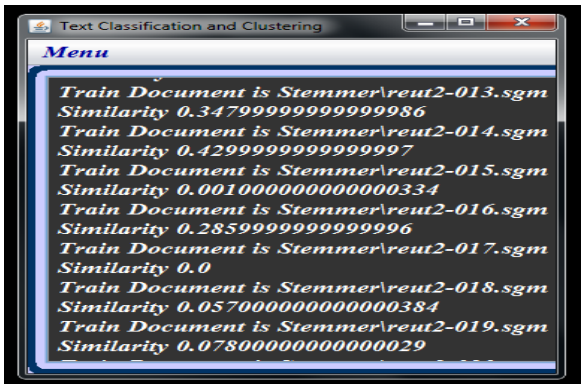

Figure 4.4 K-NN Classifier similarity

V. CONCLUSION

The extracted document is pre-processed. The document is ranked using frequency of each word, that can be calculated by using Term Frequency-Inverse Document Frequency method. Then the similar information is grouped together using fuzzy c-mean clustering which has been experimentally proved and verified by the results. Cluster will give the high efficiency for information retrieval. This algorithm is to improve the clustering accuracy and less classification time.

.

REFERENCES

[1] G.Bharathi, D.Venkatesan, "Improving information retrieval using Document clusters and semantic synonym extraction", Journal of Theoretical and Applied Information Technology, 2012.

[2] Daniela Cruzes, Victor Basili, Forrest Shull, Mario Jino, "Automated Information Extraction from Empirical Software Engineering Literature", Empirical Software Engineering, 2006.

[3] Eugene Agichtein , Silviu Cucerzan ,"Predicting Accuracy of Extracting Information from Unstructured Text Collections‟ , *CIKM'05*, October 31-November 5, 2005.

[4] Eugene Agichtein, "Scaling Information Extraction to Large Document Collections", IEEE Computer Society Technical Committee on Data Engineering, 2005

[5] Jian Zhang, Jianfeng Gao, Ming Zhou, Jiaxing Wang, Improving the Effectiveness of Information Retrieval with Clustering and Fusion, Computational Linguistics and Chinese Language Processing ,Vol. 6, No. 1, February 2001.

[6] Lipika Dey, Muhammad Abulaish, Jahiruddin, Gaurav Sharma, "Text Mining through Entity-Relationship Based Information Extraction", International Conferences on Web Intelligence and Intelligent Agent Technology – Workshops, 2007.

[7] Raymond J. Mooney and Razvan Bunescu, "Mining Knowledge from Text Using Information Extraction", SIGKDD Explorations, 2005.

[8] A.K.Santra, C. Josephine Christy, "An Efficient Document Clustering by Optimization Technique for Cluster Optimality", International Journal of Computer Applications (0975 – 8887) Volume 43– No.16, April 2012.

[9] K.Sathiyakumari, V.Preamsudha, G.Manimekalai, "Unsupervised Approach for Document Clustering Using Modified Fuzzy C means Algorithm", International Journal of Computer& Organization Trends – Volume1Issue3- 2011.

[10] Sumit Goswami, Mayank, Singh Shishodia, "A Fuzzy based approach to text mining and document clustering", International Journal of Computer & Organization Trends, 2005.

[11] L.Suganya M.phil1, Dr.B.Srinivasan, M.C.A, M.Phil, M.B.A, Ph.D 2,"Effcient Semantic Similarity Based Fcm For Inferring User Search Goals With Feedback Sessions", International Journal of Computer Trends and Technology, 2013.

[12] Yung-Shen Lin, Jung-Yi Jiang, and Shie-Jue Lee, "A Similarity Measure for Text Classification and Clustering", IEEE Transactions on Knowledge and Data Engineering, 2013.

[13] Shady Shehata,Fakhri Karray, Mohamed S. Kamel, "An Efficient Concept-Based Mining Model for Enhancing Text Clustering", IEEE Transactions on Knowledge and Data Engineering, , 2010

ISSN: 2278 – 7798