Comparison of Fuzzy C-Means and Hierarchical Agglomerative Clustering Algorithms for Data Mining

Jyoti Patel[1] Om Prakash Yadav[2]

[1]Department of Computer Science & Engineering [2]Department of Electronics & Telecommunications Engineering

[1,2]Chhatrapati Shivaji Institute of Technology Durg, Chhattisgarh

Abstract— Nowadays most of information is available in electronic format and the web pages contain a gigantic amount of information present in unstructured format and semi-structured format like newspaper, stories, email message, books, blogs etc. which can be transformed and extracted to usable information as per our requirements. This paper focuses to extracting and mining the useful or important information from the text corpus. This paper uses the Reuter Data from the Reuter Data set. The main problem in text mining is that the data in text form is written using grammatical rules to make it readable by humans, so to be able to analyze the text; it first needs to be preprocessed. The algorithm may be used to find the similarity between Reuter Data and to create the cluster is Fuzzy C-Mean and Hierarchical agglomerative clustering algorithms.

Key words: Text Mining, Natural Language Processing, Fuzzy C-Mean, Term-Frequency, Inverse Document Frequency, TF-IDF

I. Introduction

Text Mining is a process of analyzing or extracting the knowledge from different perspective and summarizing it into useful or important information that user want [1]. It is challenging issue to find the accurate and important knowledge in text document.

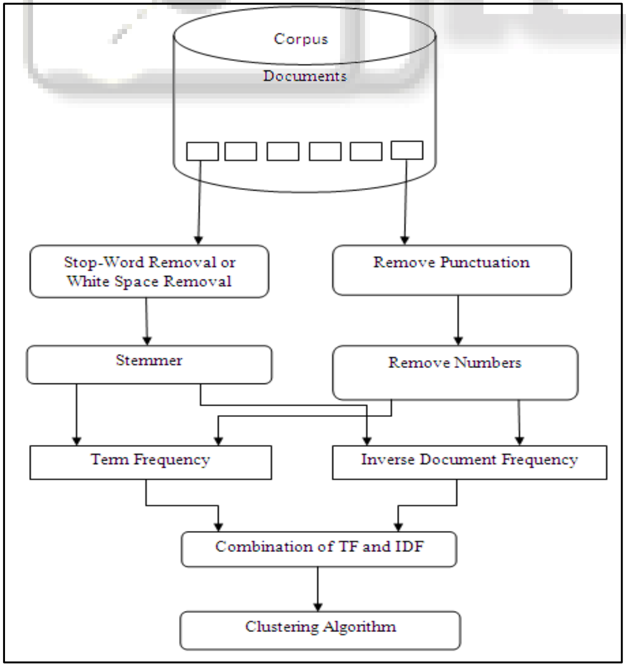

Corpus
Documents
Stop-Word Removal or White Space Removal
Remove Punctuation
Stemmer
Remove Numbers
Term Frequency
Inverse Document Frequency
Combination of TF and IDF
Clustering Algorithm

Fig. 1: Overall process of text mining

The overall process of text mining technique is shown in the Figure 1

A. *Features of Text Mining*

There are many features in Text Mining [2].

- Term Frequency (TF)
- Inverse Document Frequency (IDF)
- Term Frequency-Inverse Document Frequency (TF-IDF)

1) *Term Frequency (TF)*

TF, which measure how frequently a term occurs in the document. TF is calculated by

$$\mathrm{TF}(t) = \frac{\text{Number of times term } t \text{ appers in a document}}{\text{Total Number of terms in the Docunment}} \qquad (1)$$

2) *Inverse Document Frequency (IDF)*

IDF, measure how important a term is. IDF is calculated by

$$\mathrm{IDF}(t) = \log_e \frac{\text{Total Number of document}}{\text{Number of docment with term } t \text{ in it}} \qquad (2)$$

3) *Term Frequency-Inverse Document Frequency (TF-IDF)*

The TF-IDF weight is a product of the TF and IDF. TF-IDF is calculated by

$$\text{TF-IDF} = \text{TF}\times\text{IDF} \qquad (3)$$

B. *Text Preprocessing*

Once we are sure that all text documents loaded are proper, preprocessing of text is required. This step allows us to remove number, capitalization, common words, punctuation, and stem the document called stemming [3][4]. The preprocessing will leave the document with a lot of “white space”. White space is the result of all the left over spaces that were not removed along with the words that were deleted. The white space should be removed. The staging of the data create document term matrix and the exploration of data organize the terms by their frequency.

II. Previous Work

Swamy et al in 2014 discussed about the Indian Language Text Representation and Categorization using Supervised Learning Algorithm. The availability of constantly increasing amount of text data of many Indian regional languages in electronic form has accelerated. Text mining techniques such as k-Nearest-Neighbor classifier, decision tree for text categorization and naive Bayes classifier have been used for categorization and representation of Indian language. The author used k-Nearest-Neighbor algorithm to find the nearest neighbors of document means, to take a new unlabeled document and predict its label. Decision tree C4.5 is an algorithm used to generate a decision tree and the naïve Bayes classifier are based on applying Bayes theorems with naïve independence assumptions. The decision tree C4.5 gives 97.33% accuracy, naïve Bayes classifier gives 97.66% accuracy and the k-Nearest-Neighbor gives the 93% accuracy. These result shows that the text mining algorithm can also applicable for Indian language for categorization and representation [5].

Charjan & Pund in 2013 discussed about Pattern Discovery for Text Mining using Pattern Taxonomy. This paper develops an efficient mining algorithm for discovering pattern from a gigantic amount of data and search the useful, valuable and interesting pattern. An innovation and effective pattern discovery techniques includes the pattern deploying and pattern evolving process to improve the effectiveness of updating and using patterns. The Pattern Taxonomy Model (PTM) is used to find the pattern. There are two main stages to be considered in PTM models. The First is "how to extract useful phases from text documents" and the second "how to use these discovered patterns to improve effectiveness of a Knowledge Discovery System (KDS)". The patterns of repetitions, other features can be extracted from the given text and explored in a similar fashion [6].

T. Ahmad & Doja, M. N. in 2013 discussed about the Opinion Mining using Frequent Pattern Growth Method from Unstructured Text. This paper the area of opinion mining has been experienced. This paper the authors presented the FP-growth (Frequent Pattern) method for frequent pattern mining from review document. The Candidate Identification and Frequent Pattern Generation (CI-FPG) provides an integrated view of the extracted features using NLP techniques. The CI-FPG uses two steps to apply this concept. The first step uses Stanford parser and generates the dependency tree. And the second step uses FP-Growth Algorithms to generate the frequent pattern from the extracted features. Every feature found by the system applying the rules of the extraction are filtered out and then system processes these feature and extract frequent pattern using FP-Growth Algorithm [7].

III. Implementation

A. *Natural Language Processing (NLP)*

NLP defines field of computer science and Artificial Intelligence. NLP deals with automatic processing and analysis of the unstructured and semi structured textual Information. NLP is a technology that concerns with two systems: - the Natural Language Understanding (NLU) and Natural Language Generation (NLG). The NLU is a system that computes the meaning representation, essentially restricting the discussion to the domain of computational linguistic. The application of NLG is machine translation system. The system analyzes texts from a source language into conceptual or grammatical representations and then generates corresponding texts in the target language [8].

B. *Fuzzy C-Means (FCM)*

Clustering is a process of partitioning or grouping a given set of unlabeled pattern into a number of clusters such that similar pattern are assigned to one cluster. It works by assigning membership to each data point corresponding to each cluster center on the basis of distance between the data point and the cluster center [9][10].

1) *FCM Algorithm*

The FCM algorithm consists of the following steps:

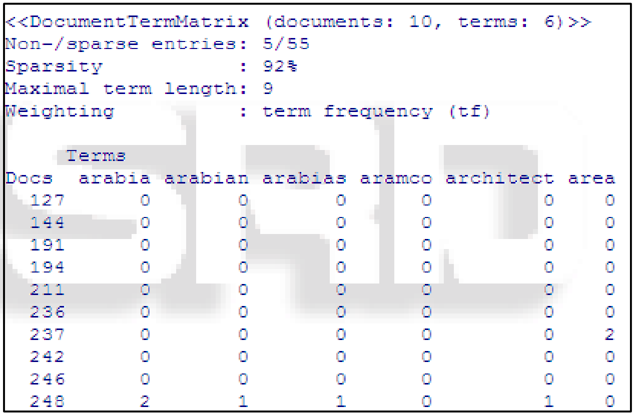
1) Initilize U = {U_{ij}} Matrix U^0
2) At *k* step calculate the center vector
   C^k = {C_j} with U^k
   $$C_j = \frac{\sum_{i=1}^{n}(u_{ij})^m x_i}{\sum_{j=1}^{n}(u_{ij})^m}$$
3) Update U_k, U^{k+1}
   $$U_{ij} = \frac{1}{\sum_{k=1}^{c}\left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|}\right)^{\left(\frac{2}{m}-1\right)}}$$
4) If $\| U^{(k+1)} - U^{(k)} \| < \delta$ then stop,
   Otherwise return to the step 2

Where U is the Membership Matrix.
m is any real number which is greater than 1.
X_i is the i^{th} of d-dimensional measured data.
C_i is the d-dimensional center of the cluster.
|| * || is any norm expressing the similarity between any measured data and the center.
And δ is termination criterion between 0 & 1 or k is an iteration steps.

IV. Results

First we have to remove all stop words, white spaces, numbers, punctuation from the text document and then stemming of words is done. Then TF, IDF and TF-IDF weight is calculated. On the basis of this weight centriod and degree of membership matrix is generated. To create cluster and to find the association among words in a text document FCM and Hierarchical Agglomerative Clustering Algorithms are used [11][12].

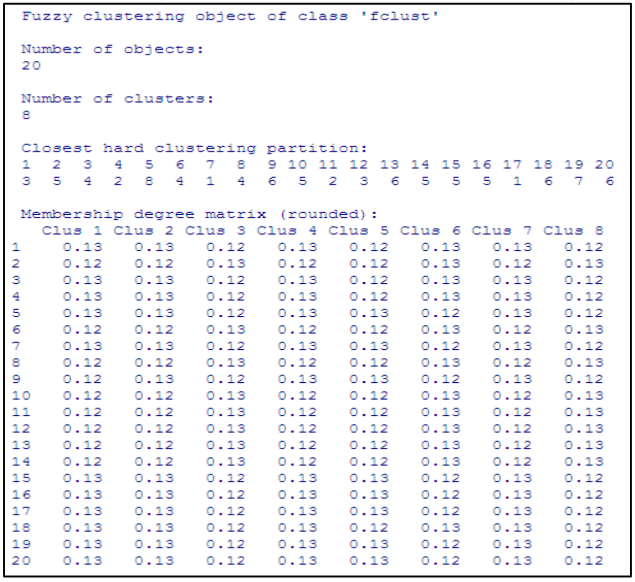
A term document matrix is shown in the Figure 2



```
<<DocumentTermMatrix (documents: 10, terms: 6)>>
Non-/sparse entries: 5/55
Sparsity           : 92%
Maximal term length: 9
Weighting          : term frequency (tf)

    Terms
Docs  arabia arabian arabias aramco architect area
  127      0       0       0      0         0    0
  144      0       0       0      0         0    0
  191      0       0       0      0         0    0
  194      0       0       0      0         0    0
  211      0       0       0      0         0    0
  236      0       0       0      0         0    0
  237      0       0       0      0         0    2
  242      0       0       0      0         0    0
  246      0       0       0      0         0    0
  248      2       1       1      0         1    0
```

Fig. 2: Illustrates TDM for TF-IDF measure

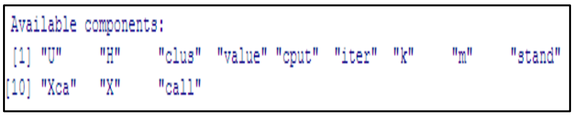
We use 20 documents and 8 cluster size to calculate membership degree matrix of FCM. The membership degree matrix of FCM is shown in the Figure 3



```
 Fuzzy clustering object of class 'fclust'

 Number of objects:
 20

 Number of clusters:
 8

 Closest hard clustering partition:
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
 3  5  4  2  8  4  1  4  6  5  2  3  6  5  5  5  1  6  7  6

 Membership degree matrix (rounded):
   Clus 1 Clus 2 Clus 3 Clus 4 Clus 5 Clus 6 Clus 7 Clus 8
1    0.13   0.13   0.12   0.13   0.12   0.13   0.13   0.12
2    0.12   0.12   0.13   0.12   0.12   0.13   0.12   0.13
3    0.13   0.13   0.12   0.13   0.12   0.13   0.13   0.12
4    0.13   0.13   0.12   0.13   0.12   0.13   0.13   0.12
5    0.13   0.13   0.12   0.13   0.13   0.12   0.13   0.12
6    0.12   0.12   0.13   0.12   0.12   0.13   0.12   0.13
7    0.13   0.13   0.12   0.13   0.13   0.12   0.13   0.12
8    0.12   0.12   0.13   0.12   0.12   0.13   0.12   0.13
9    0.12   0.13   0.12   0.13   0.13   0.12   0.13   0.12
10   0.12   0.12   0.13   0.12   0.12   0.13   0.12   0.13
11   0.12   0.12   0.13   0.12   0.12   0.13   0.12   0.13
12   0.12   0.12   0.13   0.12   0.12   0.13   0.12   0.13
13   0.12   0.12   0.13   0.12   0.12   0.13   0.12   0.13
14   0.12   0.12   0.13   0.12   0.12   0.13   0.12   0.13
15   0.13   0.13   0.12   0.13   0.13   0.12   0.13   0.12
16   0.13   0.13   0.12   0.13   0.13   0.12   0.13   0.12
17   0.13   0.13   0.12   0.13   0.13   0.12   0.13   0.12
18   0.13   0.13   0.12   0.13   0.12   0.13   0.13   0.12
19   0.13   0.13   0.12   0.13   0.13   0.12   0.13   0.12
20   0.13   0.13   0.12   0.13   0.13   0.12   0.13   0.12
```

Fig. 3: Membership degree matrix

The Available Component of FCM algorithm is indicated below



```
Available components:
 [1] "U"     "H"     "clus"  "value" "cput"  "iter"  "k"     "m"     "stand"
[10] "Xca"   "X"     "call"
```

The hierarchical agglomerative clustering using average linkage is shown in the Figure 4

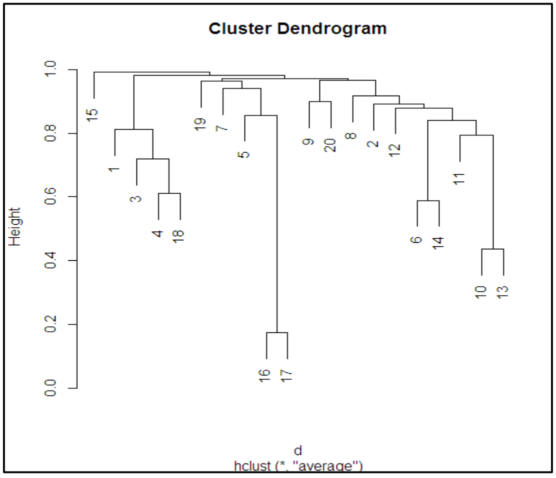


Fig. 4: Illustrates hierarchical agglomerative clustering using average linkage

V. Conclusion

Text Mining is the process of analysis of data contained in Natural Language text. In proposed technique we have taken data from Reuter data. The data is then preprocessed & Stemming is performed. FCM and Hierarchical Agglomerative Clustering algorithms are used to display the output. R statistical analysis tool is used for text mining task. In FCM clustering, each point has a weight associated with a particular cluster, data point is assigned membership to each cluster center as a result of which data point may belong to more than one cluster center and gives best result for overlapped data sets. FCM will reduce the number of iteration and improve the execution performance.

References

[1] Text mining. (2015). Retrieved from Wikipedia, the free encyclopedia website: https:// en.m.wikipedia.org /wiki/Text_mining

[2] Term Frequency-Inverse Document Frequency. (2015). Retrieved from Wikipedia, the free encyclopedia website: https:// en.m.wikipedia.org/wiki/Tf-idf

[3] Gupta, G.K. (2009). Introduction to Data Mining with Case Studies. Clayton, Australia: Prentice- Hall Of India Pvt. Limited.

[4] Pujari, A.K. (2009). Data Mining Techniques. Universities Press Second Edition

[5] Swamy, M. N., Hanumanthappa, M. & Jyothi, N. M. (2014). Indian Language Text Representation and Categorization using Supervised Learning Algorithm. International Conference on Intelligence Computing Applications, 406-410. IEEE DOI:10.1109/ ICICA.2014.89

[6] Charjan D. S. & Pund M. A. (2013). Pattern Discovery for Text Mining using Pattern Taxonomy. International Journal of Engineering Trends and Technology 4(10), 4550-4555. ISSN: 2231-2803.

[7] Ahmad, T. & Doja, M. N. (2013). Opinion Mining using Frequent Pattern Growth Method from Unstructured Text. International Symposium on Conference and Business Intelligence, 92-95. IEEE. DOI:10.1109/ISCBI.2013.26

[8] Kao A. & Poteet S. Text Mining and Natural Language Processing- Introduction for the Special Issue. SIGKDD Explorations. Vol 7, Issue 1. 1-2.

[9] Sahu, S. K. & Jena S. K. (2014). A Study of K-Means and C-Means Clustering Algorithms for Intrusion Detection Product Development. International Journal of Innovation, Management and Technology, Vol 5, No 3. DOI: 10.7763/IJIMT.2014.V5.515

[10] Hung M.C. & Yang D.L. (2001). An Efficient Fuzzy C-Mean Clustering Algorithm. DOI: 0-7695-1119-8/01 $17.00 IEEE

[11] Reuter-21578 Text Categorization Collection Data Set. Retrieved from website: https://archive.ics.uci.edu/ml/datasts/Reuters-21578+Text+Categorization+Collection

[12] R programming Language. Retrieved from the website: https://www.rproject.org