Clustering Blog Information

Mayank Prakash Jaiswal[1] H. Chris Tseng[2]

[1]Computer Science Department, San Jose State University, San Jose, CA- 95192, Phone no.: (408)-207-5842, E-mail: mayankbittu@yahoo.com

[2]Professor, Computer Science Department, San Jose State University, San Jose, CA- 95192, Phone no.: (408)-924-7255, E-mail: tseng@cs.sjsu.edu

Abstract

Blogs form an important source of information in today's internet world. Most of the blog websites have the blogs arranged in chronological order rather than its contents. Such arrangement of blogs makes it difficult for the user searching information about a particular topic from the blog. To resolve this problem, we propose an idea to cluster the blogs. There are several clustering algorithms available. The objective of this paper is to understand various steps involved in clustering blog information and working of clustering algorithms, followed by detailed analysis of FCM and means to improve the clustering using FCM clustering algorithm.

Keywords**: Blog Clustering, TFIDF, FCM, IR**

1 Introduction

In our project, Clustering Blog Information, we divided clustering process in three steps; Data Collection, Data Processing and Clustering Algorithm. Data collection is an elementary process in clustering blog information used to obtain data to be clustered. Data can be obtained online or offline.

We consider blogs data as the source information and ignore any images, additional control buttons within the blogs. Data collection is used to eliminate the images or any factor that is included in blogs other than the text. Online blogs consists of HTML tags which carry no information for clustering. Eliminating HTML tags from the blogs forms an important step in the data collection process.

Data processing follows the data collection process of clustering blog information. Data retrieved from the blog website using data collection, consists of repetitive and less important information. For example, punctuation marks, pronouns, etc carry very little almost null information. Hence, data should be filtered to get rid of repetitive and less important information. Data processing performs the required function and converts the blog data into a format that could be used by clustering algorithms. Data processing uses different weight assigning schemes to assign weight to terms in the blogs. The weight assigned terms are passed as an input data to the clustering algorithms.

Clustering algorithms like K-means, VSM cosine similarity measurement based, LSI and FCM have been used to cluster text documents. We followed the approach of understanding, implementing and comparing, the working of four clustering algorithms on the blog data and selected the optimum clustering algorithm depending on the output. The following section summarizes the results of study for each clustering algorithm.

VSM represents documents and query as vectors. The angle between the document and query defines the similarity between them. In the experiments performed cosine similarity measurement is used to determine the angle between document and query. Thus output of VSM is vector representation of documents and query; incorporation of cosine similarity measurement with VSM gives the similarity measure of document to the query. To retrieve clusters from the similarity measurement an additional threshold is required. Also VSM with cosine similarity measurement fails to cluster documents with different vocabulary but same contents.

LSI overcomes the disadvantages of VSM by introducing clustering based on concepts rather than terms within the documents. However, output of LSI is a score that indicates the similarity of documents to the query based on concept. In order to retrieve the clusters from LSI an additional threshold needs to be implemented on the output score.

VSM cosine similarity measure and LSI have a common disadvantage. The similarity measure in VSM and the score in LSI depend on the query. Hence with change in query, clusters change and eliminate documents that are not related to the query even though they are related to the cluster.

Unlike VSM cosine similarity measure and LSI, output of k-means and FCM is clusters. However, for k-means, clustering depends on the mean and the mean changes with the number of clusters. Dependence of clustering on means, results in clusters with documents that are not correlated to each other. Contrary, the documents in the FCM clusters are correlated to each other. Thus, amongst the discussed clustering algorithms, we select FCM as the clustering algorithm for blog clustering.

1.1 FCM Shortcomings

Documents in the FCM clusters are strongly correlated; however FCM clusters are sensitive to the initialization of membership matrix and center. Sensitivity of algorithm to initialization results in different cluster with single execution. The following table depicts the change in clusters with every execution.

Consider the documents:

D1: Large Singular Value computations

D2: Software Library for the Space Singular Value Decomposition

D3: Introduction to Modern Information Retrieval

D4: Using Linear Algebra for Intelligent Information Retrieval

D5: Matrix Computations

D6: Singular Value Analysis of Cryptograms

D7: Automatic Information Organization

Run #	Clusters	Document	Objective Function
1	1	D1,D2,D5,D6	
	2	D3,D4,D7	20.3186
2	1	D1,D2,D5,D7	
	2	D3,D4,D6,D7	20.3186
3	1	D1,D5	
	2	D2,D3,D4,D6,D7	20.3186
4	1	D1,D2,D3,D4,D6,D7	
	2	D5	20.3186
5	1	D1,D2,D5,D6,D7	
	2	D3,D4	20.3186

Fig. 1: Result of Classical TFIDF

Referring above table, we see that the documents in the cluster change, documents forming a cluster are different with every run. As long as same documents form a cluster, which cluster they form (1 or 2) does not matter. Thus, we can see that the above cluster is sensitive to the initialization of membership matrix.

FCM gives the cluster depending on the cluster size given by the user. As the cluster size changes the documents belonging to the cluster changes. Thus, clustering of document depends on the optimum number of clusters.

1.2 FCM data and cluster analysis

Large data analysis was performed in understanding the changing behavior of the FCM clusters.

Genetic Algorithm (GA) was performed on the FCM to reduce the sensitivity of the clusters on the random initialization of the membership function. GA succeeded in obtaining the minimum objective function and the number of clusters; however the documents forming clusters changed.

Cluster merging was implemented to reduce the effects of random initialization of membership function, on clusters. Cluster merging assisted in obtaining the optimum number of clusters however, the effect of initialization on the clusters sustained.

Several data sets, consisting of large and small number of documents were implemented to narrow down the factor, which is affected by the initialization of the membership function.

Based on large data analysis, it was concluded:

> *" As long as the input data to the FCM is correct, the random initialization of membership function has null effect on the clusters for membership value 2"*

The conclusion resulted in evolution of a modified keyword weight assigning scheme.

1.3 Modified TFIDF

Part A:

$$\text{Modified TFIDF (t3)} = \frac{\text{maxterm per document}}{[(\text{term occurence in that document}) + (\text{term document length}) + (\text{maxterm occurence})]}$$

Where: Max term per document – Total number of terms in a document
Term occurrence in that document –how many times the term occurs in the document
Term document Length – maximum occurrence of terms in a document in which the term occurs
Max term occurrence – maximum of all the term occurrences in the document

Part B:

$$\text{Modified TFIDF} = \frac{(\text{t3}) * (\text{total document in which term occurs})}{(\text{sum of TF along row})}$$

Modified TFIDF performs cumulation of terms. The weight is assigned such that the value indicates total number of terms in the document in which term occurs and the number of documents in which the term occurs. The modified TFIDF weight of the term determines the contribution of the term in the document and is independent of the terms occurring in other documents.

1.4 Cluster Size

FCM clusters depend on Cluster size. As the cluster size changes the documents in the cluster change. This problem could be solved using cluster merging. In cluster merging, FCM starts with large number of clusters and stops when the cluster could no longer be merged. However, considering the application, clustering blog information, the size of the cluster should meet user's requirement. User should have the option, if he/she wants a general overview of what types of documents are present under the topic or he/she is looking for specific information within the blogs. The approach to meet this requirement is, provide two clusters for the dataset and have a certain depth within each cluster. The peculiarity of clustered information increases with the depth within each cluster. As a result the user has both general overview and specific information at its display and has the choice as per user's requirement.

1.5 Comparison of Classical TFIDF and Modified TFIDF

In classical TFIDF weight of a term in a document affects the weight of other terms in other documents. Classical TFIDF gives the closeness of the documents however; it does not include information about how different the documents are from each other.

Run #	Clusters	Document	Objective Function
1	1 2	D3,D4,D7 D1,D2,D5,D6	0.5176
2	1 2	D3,D4,D7 D1,D2,D5,D6	0.5176
3	1 2	D1,D2,D5,D6 D3,D4,D7	0.5176
4	1 2	D3,D4,D7 D1,D2,D5,D6	0.5176
5	1 2	D3,D4,D7 D1,D2,D5,D6	0.5176

Fig. 2: Result of Modified TFIDF

1.6 Results

Results of FCM Clustering on the blogs mentioned in above section are summarized in the graphs below. Figure 3 shows data center plot for the seven documents introduced in above sections. Blue data points are the documents and green data points indicate the center of the cluster. Figure 4 is the graph of objective function vs. Iteration. The graph proves that the objective function decreases and eventually reaches a steady value. Figure 5 is the final graph, indicating the data points and clusters in the right half, objective function variation with each iteration and sub clusters objective function.

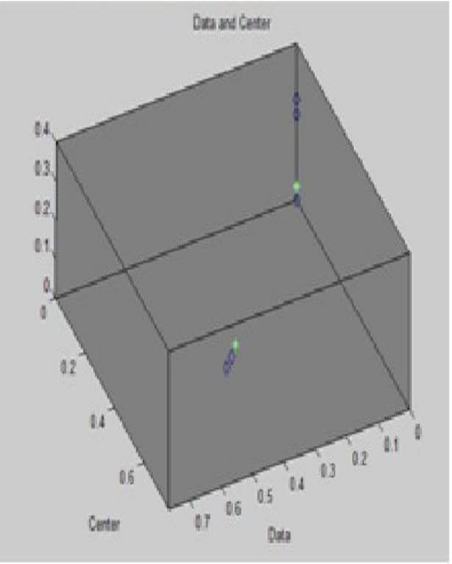

Fig. 3: Data Center Plot

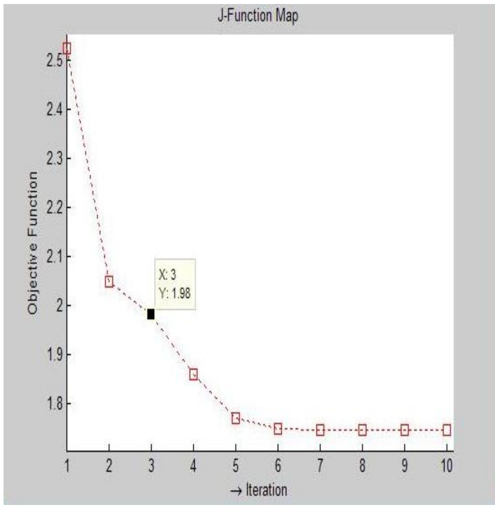

Fig. 4: FCM Cluster Objective Function

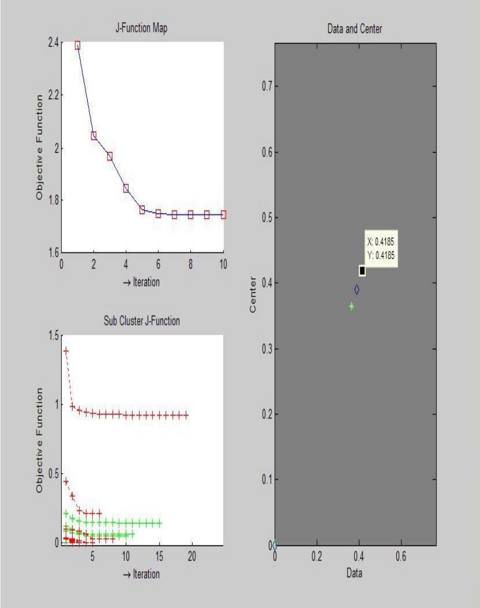

Fig. 5: Cluster & Sub Cluster Objective Function

1.7 Conclusion

Clustering the chronologically arranged blogs as per the contents provides more useful information to the user. Most often the chronological blogs do not have the same information, which in turn makes it difficult for the user to search the information within blogs. Content based clustering blogs, helps the user find the required information.

Modified TFIDF proportionately assigns weight to the term, such that each term knows its contribution in the document. Modified TFIDF gives consistent clusters with the same objective function.

Document clustering using FCM depends on the clustering size. As the cluster size changes the documents in the cluster changes. Increasing the number of clusters increases the granularity within each cluster. Considering the application, clustering blog information, the size of the cluster should be as per user's requirement. This requirement is provided by dividing the dataset into two clusters and specifying depth within each cluster. The number of clusters is determined by the user depending on the granularity of information required by the user.

Thus, with modified TFIDF and sub-clustering the clustering blog information algorithm gives correct result and the user can view the clusters depending on the granularity of information required by the user.

2 References

[1] Agarwal.G., Goswami.A., and Jin. R. (2004) *Fast and Exact Out of Core K-means Clustering*. Proceedings of the Fourth IEEE International Conference on Data Mining (ICDM'04)

[2] Berry. M.W., Drmac.Z., znd Jessup.R.E. *Matrices, Vector Spaces and Information Retrieval.* SIAM Rev. 41, pp. 335-362

[3] Carven, M. *Introduction to Information Retrieval,* Retrieved on March 6, 2007 from University of Wisconsin, Department of CS Website: http://www.cs.wisc.edu/~shavlik/cs540/introToIR.ppt

[4] Church, K.W and Gale, W.A. *Inverse Document Frequency (IDF): A Measure of Deviations from Poisson.* AT&T Bell Laboratories

[5] Djouani.K & Nefti.S (2003) *Extended Fuzzy Clustering Algorithm Based on Inclusion Concept* The 12th IEEE International Conference on Fuzzy Systems, 2003. Fuzz'03 (869-874) vol.2

[6] El-Sharkawi, M.A. *Fuzzy System and Control,* Retrieved on March 10, 2007 from University of Washington, Department of EE, Computational Intelligence Applications Laboratory (CIA) lab website:

http://cialab.ee.washington.edu/index_files/tutorial/fuzzy.pdf

[7] Fan.W, Gordon.M.D, and Pathak.P *A generic ranking function discovery framework by genetic programming for information retrieval*. Retrieved on September 7, 2007 from http://filebox.vt.edu/users/wfan/pper/ARRANGER/ip&m2003.pdf

[8] Fielong. X. *Latent Semantic Indexing,* Retrieved on March 20, 2007 from http://www.coli.unisaarland.de/~schulte/Teaching/Klassifikation-04/feilong.pdf

[9] Gen.M, Tsujimura.Y, and Zhao.L *Genetic Algorithm for Fuzzy Clustering*. Dept. of Ind. & Syst. Eng., Ashikaga Inst. of Technol., Japan; Proceedings of IEEE International Conference on Evolutionary Computation, 1996. 20-22 May 1996 716 – 719

[10] Jea.Y.T *Basic Concepts of Data Mining, Clustering and Genetic Algorithms* Department of CSE SUNY at Buffalo

[11] Jing.L. Ng. M., and Huang.M.Z (2006) *Text Clustering: Algorithms, Semantics and Systems.* The University of Hong Kong and Hong Kong Baptist University.

[12] Joachims,T(2004). *Representing and Accessing Digital Information: Information Retrieval: Indexing*. Retrieved March 10, 2007, from Cornell University Department of CS website: http://www.cs.cornell.edu/courses/cs630/2004fa/lectures/tclust_6up.pdf

[13] Karayiannis.N.B, & Randolph-Gips.M.M (2002) *Non-Euclidean c-means clustering algorithms.* Intelligent Data Analysis 7 *(2003)* (405-425)

[14] Keller. M.J, Krishnapuram.R., Kuncheva.I.L, Bezdek.C.J, & Pal.R.N (1999) *Will the Real Iris Data Please Stand Up?* IEEE Transactions on Fuzzy Systems (368-369) Vol.7

[15] Lazarinis, F. *Porter.Java, IR Linguistic Utilities.* Retrieved on March 7, 2007 from http://www.dcs.gla.ac.uk/idom/ir_resources/linguistic_utils/porter.java

[16] Liu,J. and Xie,W. *A Genetics Based Approach to Fuzzy Clustering.*(1995)Proceedings of 1995 IEEE International Conference on International Joint Conference of the Fourth IEEE International Conference on Fuzzy Systems and The Second International Fuzzy Engineering Symposium., 20-24 March 1995 2233 - 2240 vol.4

[17] Losee.M.R, (1998). *Comparing Boolean and Probabilistic Information Retrieval Systems Across Queries and Disciplines.* J. of the American Society for Information Science

[18] Mendes, M.E.S. and Sacks, M.L. *Knowledge Based Content Navigation in e-Learning Applications.* Retrieved on February 4, 2007 from University College London, Department of EE website http://www.ee.ucl.ac.uk/lcs/papers2002/LCS115.pdf

[19] Mishne.G. & Rijke.D.M (2005) *Vector Space Model.* Informatics Institute University of Amsterdam

[20] Pedrycz.W. *Knowledge Based Clustering- From Data to Information Granules*

Published by Atlantis Press