ijergs
Open Access
International Journal of Engineering Research and Generic Science (IJERGS)
Available Online at www.ijergs.in
ISSN: 2455 - 1597
Volume – 2, Issue -2, March - April – 2016, Page No. 57 - 62

Text Mining Based on Content and Side Information

Joshi Shripad S.
Asst. Prof. of CSE
JNTUH College of Engineering Sultanpur,
Pulkal(M), Medak Dist.

Abstract

In various applications of text mining, side information is available along with each text documents. The side information may be of different kinds, like the links in the document, user-access behavior from web logs, document provenance information or other non-textual attributes which are embedded into the text document. Such attributes may contain a lot of information for clustering purposes. In any case, the relative significance of this side-information might be hard to assess, particularly when a percentage of the information is noisy. Existing framework proposes established partitioning algorithms with probabilistic models keeping in mind the end goal to make a successful clustering approach. In proposed system the term frequency cosine angle based similarity is measuring for Content as well as for side information. Implements a modified approach for COATES using cosine angle based similarity measure.

Key Words: Text Mining, Classification, Clustering, Side Information, COATES.

1. Introduction

The issue of text clustering rises in the context of numerous application areas, for example, the web, social networks and other computerized accumulations. The quickly expanding measures of text information in the context of these vast online accumulations have prompted an enthusiasm for making adaptable and powerful mining algorithms. A tremendous amount of work has been done in recent years on the problem of clustering in text collections in the database and information retrieval communities. However, this work is primarily designed for the problem of pure text clustering, in the absence of other kinds of attributes. In many application domains, a tremendous amount of side information is also associated along with the documents. This is because text documents typically occur in the context of a variety of applications in which there may be a large amount of other kinds of database attributes or Meta information which may be useful to the clustering process [1].

The problem of text mining arises in the context of many application domains such as the web, social networks and other computerized accumulations. A large amount of work has been done in recent years on the problem of text collections in the database and information retrieval groups. Then again, this work is primarily designed for the problem of pure text collection, in the absence of other kinds of attributes In many application domains, a large amount of side information is likewise related alongside the document. This is on account of text documents typically occur in the context of a variety of uses in which there might be a lot of other kinds of database attributes or Meta information which may be useful to the mining process. Some examples of such side information are as follows:

A. Web logs

In an application in which track user access behavior of web documents, the user-access behavior may be captured as web logs. For every document, the meta-information may correspond to the browsing behavior of the different users. Such logs can be utilized to improve the nature of the mining process in a way which is more meaningful likewise application-touchy. This is on account of the logs can often pick up subtle correlations in content, which cannot be picked up by the raw text alone.

B. Links present in Text Document

Text documents, which can also be treated as attributes. Such links contain a lot of useful information for mining purposes. Such links contain a lot of valuable information for mining purposes. As in the past case, such attributes may often provide insights about the relationships among documents in a way which may not be effectively accessible from raw content.

C. Meta-data

Text documents, which can also be treated as attributes. Such links contain a lot of useful information for mining purposes. Such links contain a lot of valuable information for mining purposes. As in the past case, such attributes may often provide insights about the relationships among documents in a way which may not be effectively accessible from raw content.

2. Literature Survey

Charu C. Aggarwal, Yuchen Zhao and Philip S. Yu [1] propose an algorithm which consolidates classical partitioning algorithms with probabilistic models to make a powerful clustering approach. At that point they demonstrate to extend the approach to the order issue. Introduced strategies for mining text data with the utilization of side-information. Numerous types of text databases contain a lot of side-information or Meta information, which may be utilized as a part of request to enhance the clustering process. So as to plan the clustering strategy, they consolidated an iterative partitioning system with a probability estimation process which registers the significance of various types of side-information. This general approach is utilized as a part of request to outline both clustering and grouping algorithms. They exhibited results on real data sets showing the effectiveness of their approach.

S. Guha, R. Rastogi, and K. Shim[2] Discovering groups and identifying interesting distributions in the basic data clustering is utilized as a part of data mining. Traditional clustering algorithms either support clusters with spherical shapes and comparative sizes. In this paper a clustering algorithm is exhibited which is called CURE that is more vigorous to outliers, and recognizes clusters having non-spherical shapes and wide variances in size. CURE accomplishes this by speaking to every cluster by a specific fixed number of points that are created by selecting well scattered points from the cluster and then shrinking them toward the focal point of the cluster by a predetermined division. Having more than one representative point for every cluster permits CURE to alter well to the geometry of non-spherical shapes and the shrinking hoses the impacts of outliers. CURE employs a mix of random sampling and partitioning to handle huge databases. A random sample drawn from the data set is initially divided and every segment is in part clustered. The fractional clusters are then clustered in a brief moment go to get the looked for clusters

A drawback is the client determined parameter values, the number of clusters and the shrinking component. A random sample of data objects is drawn from the given datasets. Partial clusters are gotten by partitioning the sample dataset and outliers are distinguished and removed in this stage. Last refined clusters are framed from the halfway cluster set.

S. Guha, R. Rastogi, and K. Shim [3] are targeted to both Boolean data and categorical data. It utilizes the Jaccard coefficient as the measure similarity. The input is a set S of n sampled points to be clustered (that are drawn randomly from the original data set), and the number of desired clusters k. It samples the data set in the same manner as CURE. This agglomerative algorithm assumes a similarity measure between items and defines a link between two questions whose similarity surpasses a limit. Then, clusters are combined repeatedly according to their closeness: the total of the number of links between all pairs of articles between two clusters. ROCK has cubic unpredictability in N, and is unsuitable for large datasets. A pair of things are said to be neighbors if their similarity surpasses some limit. The number of links between two things is defined as the number of common neighbors they have. The goal of the clustering algorithm is to assemble together points that have more links. A recent development in genetics utilized this algorithm called GE-ROCK.

D.Cutting, D. Karger, J. Pedersen, and J. Tukey [4] Explains the Hybrid Technique (Scatter-gather technique is the hybrid clustering technique). An example of the Scatter/Gather strategy, which gives a systematic skimming procedure with the utilization of clustered document collection of the document organization. Initially the framework scatters the collection of document into a small number of several document groups, or clusters, and exhibits short summaries of documents to the clients. The client chooses one or more of the groups for further study based on these summaries. The selected groups are gathered together to form a sub collection documents. Then applies bunching again to scatter the new sub collection into a small number of document groups, which are again introduced to the clients .The scatter-gather approach can be

utilized for organized scanning of gigantic amount of document collections, because it creates a natural hierarchy of similar documents. In any case, these systems are intended for the immaculate content data bunching, and don't work for in which the content data is consolidated with other forms of data. Working so as to bunch can be done rapidly in a local manner on small groups of documents rather than attempting to deal with the whole corpus globally. It is not profoundly accurate grouping and it don't work for in which the content data is consolidated with other forms of data and the quality of the bunching gave by the moderate group subroutine.

T. Liu, S. Liu, Z. Chen, and W.Y. Mama [5] Explains Feature extraction and feature selection strategies are utilized to diminish feature space dimensionality. In feature extraction it removes an arrangement of new features from unique features through some functional mapping. In feature selection it picks a subset from the first feature set by criteria. Report recurrence, data pick up, term quality is a percentage of the feature selection routines. Unsupervised feature selection methods are much more worse than supervised feature selection. With a specific end goal to use the effective supervised technique an iterative feature selection system that iteratively performs clustering and feature selection is proposed in this paper. . Advantage is that feature selection can enhance the content clustering productivity and execution in perfect case, in which features are chosen in view of class data. Disadvantages are the unsupervised feature selection is much more regrettable than supervised feature selection.

3. Existing System

In existing system Cosine similarity is used for document content clustering and Posterior probability is used for side information clustering. In such cases, it can be risky to incorporate side-information into the mining process, because it can either improve the quality of the representation for the mining process, or can add noise to the process.

4. Proposed System

In propose system a modified COATES algorithm is used for efficient clustering approach. The term frequency cosine angel based similarity is calculated for content as well as for side information , to improve the clustering process.

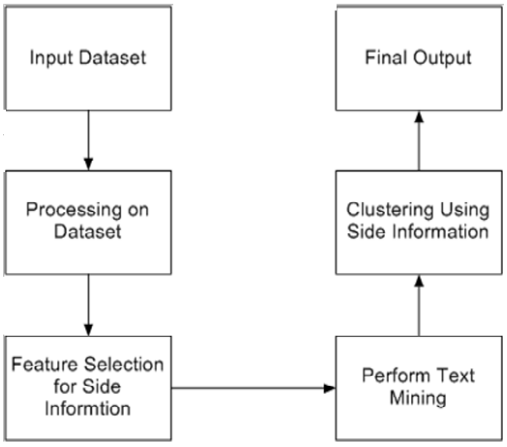


Figure 1: System Architecture.

The system architecture consist of modules Preprocess dataset, Select attributes and mining and Clustering. In Preprocess dataset module take the input dataset and remove null and abnormal data from it. Remove the null value from the dataset which is not useful and the meaningless for the auxillary attributes. In Select Attribute and mining module apply the Stopword removing algorithm for remove the stopwords in the document. Stopword means a, an ,the, are, of etc. Then

apply the Stemming algorithm for convert the varient form of word into the normal form. Stemming means remove the ed, tion ,es etc. Attribute selection is important process in the system. Appropriate attribute must be selected.

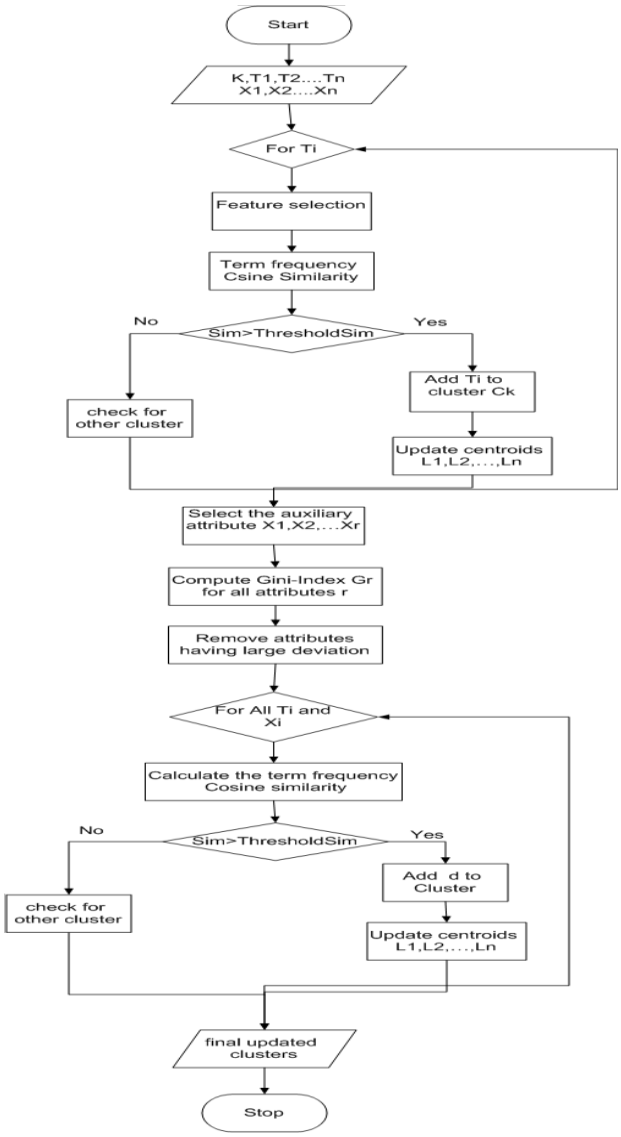
A. Similarity based on term Frequency

Tf idf, short for term frequency inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. It is often used as a weighting factor in information retrieval and text mining. The tf-idf value increases proportionally to the number of times a word appears in the document, yet is offset by the frequency of the word in the corpus, which adjusts for the fact that a few words appear all the more as often as possible in general. One of the most straightforward summing so as to rank functions is processed the tfidf for each term; many more sophisticated ranking functions are variants of this basic model.

B. Flowchart

Figure 2: shows the proposed systems flow chart. This flow chart contains Add cluster, Update Centroid, compute

Gini index, Term frequency Cosine Similarity functions. These functions are used for clustering the document.



C. Gini Index

Need to dispose of the noisy attributes. This is particularly vital, when the quantity of auxiliary attributes is very huge. In this manner, toward the start of every auxiliary iteration, figure the gini-index of every trait based on the clusters made by the last content based iteration. This gini-index gives an evaluation of the biased force of every credit as for the clustering process.

5. Data Sets

There are three real data sets in order to test approach. The data sets used as follows:

(1) DBLP-Four-Area Data Set

The DBLP-Four-Area data set [31] is a subset extracted from DBLP that contains four data mining related research areas, which are database, data mining, information retrieval and machine learning. This data set contains 28,702 authors, and the texts are the important terms associated with the papers that were published by these authors. In addition, the data set contained information about the conferences in which each author published. There are 20 conferences in these four areas and 44,748 author-conference pairs. Besides the author conference attribute, we also used co-authorship as another type of side information, and there were 66,832 coauthor pairs in total. [1]

(2) IMDB Data Set

The Internet Movie Database (IMDB) is an online collection2 of movie information. We obtained ten-year movie data from 1996 to 2005 from IMDB in order to perform text clustering. We used the plots of each movie as text to perform pure text clustering. The genre of each movie is regarded as its class label. We extracted movies from the top four genres in IMDB which were labeled by Short, Drama, Comedy, and Documentary. We removed the movies which contain more than two above genres. There were 9,793 movies in total, which contain 1,718 movies from the Short genre, 3,359 movies from the Drama genre, 2,324 movies from the Comedy genre and 2,392 movies from the Documentary genre. The names of the directors, actors, actresses, and producers were used as categorical attributed corresponding to side information. The IMDB data set contained 14,374 movie-director pairs, 154,340 movie-actor pairs, 86,465 movie-actress pairs and 36,925 movie-producer pairs. [1]

(3) Cora Data Set

The Cora data set1 contains 19,396 scientific publications in the computer science domain. Each research paper in the Cora data set is classified into a topic hierarchy. On the leaf level, there are 73 classes in total. We used the second level labels in the topic hierarchy, and there are 10 class labels, which are Information Retrieval, Databases, Artificial Intelligence, Encryption and Compression, Operating Systems, Networking, Hardware and Architecture, Data Structures Algorithms and Theory, Programming and Human Computer Interaction. We further obtained two types of side information from the data set: citation and authorship. These were used as separate attributes in order to assist in the clustering process. There are 75,021 citations and 24,961 authors. One paper has 2.58 authors in average, and there are 50,080 paper-author pairs in total.[1]

6. Experiment Result

Propose framework use the text preprocessing methods, for example, stops words removal, stemming and term frequency computation and the time required for side-information preprocessing is insignificant. The running times just for the clustering portions keeping in mind the end goal to sharpen the comparisons and make them more meaningful. The modified COATES algorithm is required to incorporate side-information into the clustering process in an a great deal more significant way than other baseline, its running times are correspondingly anticipated that would be somewhat higher. The goal is to demonstrate that the overheads associated with the better qualitative results of the modified

COATES algorithm are somewhat low, also tried the effectiveness of the strategy with increasing data size. This is finished by sampling a portion of the data, and reporting the results for various sample sizes.

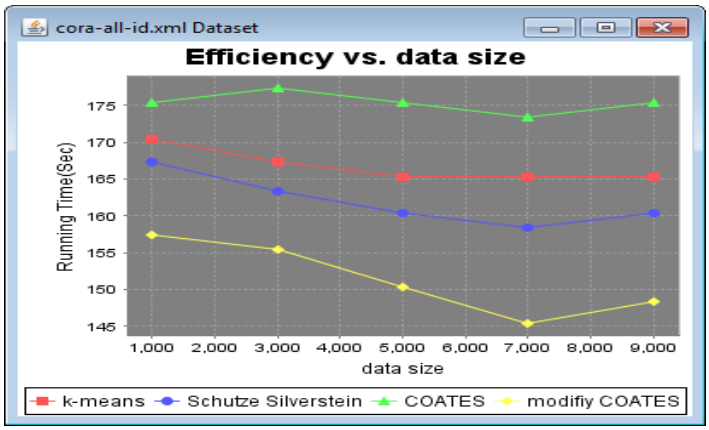


Figure 3: Efficiency comparison of clustering Algorithm on CORA Dataset.

7. Conclusion

In this paper, we presented methods for mining text data with the use of side-information. Many forms of text databases contain a large amount of side-information or meta-information, which may be used in order to improve the clustering process. The term frequency cosine angel based similarity is calculated for content as well as for side information to improve the clustering process. The use of side-information can greatly enhance the quality of text clustering while maintaining a high level of efficiency.

8. References

[1]. Aggarwal, C.C., Yuchen Zhao, Yu, P.S."On the Use of Side Information for Mining Text Data", Knowledge and Data Engineering, IEEE Transactions on, Vol. 26, No. 6, pp. 1415-1429, June 2014.

[2]. S. Guha, R. Rastogi, and K. Shim, "CURE: An efficient clustering algorithm for large Databases," in Proc. ACM SIGMOD Conf., New York, NY, USA, 1998, pp. 73–84.

[3]. D. Cutting, D. Karger, J. Pedersen, and J. Tukey, "Scatter/Gather: A cluster-based approach to browsing large document collections," in Proc. ACM SIGIR Conf., New York, NY, USA, 1992, pp. 318–329.

[4]. T. Liu, S. Liu, Z. Chen and W.Y. Ma, "An evaluation of feature selection for text clustering, In Proc. ICML Conf., Washington, DC, USA, 2003, pp. 488 – 495.

[5]. S. Zhong, "Efficient streaming text clustering," Neural Netw., vol. 18, no. 5–6, pp. 790–798, 2005.

[6]. S. Guha, R. Rastogi, and K. Shim, "ROCK: A robust clustering algorithm for categorical attributes," Inf. Syst., vol. 25, no. 5, pp. 345–366, 2000.

[7]. H. Frigui and O. Nasraoui, "Simultaneous clustering and dynamic keyword weighting for text documents," in Survey of Text Mining, M. Berry, Ed. New York, NY, USA: Springer, 2004, pp. 45–70.

[8]. S. Zhong, "Efficient streaming text clustering," Neural Netw., vol. 18, no. 5–6, pp. 790–798, 2005.

[9]. F. Sebastiani, "Machine learning for automated text categorization,"ACM CSUR, vol. 34, no. 1, pp. 1–47, 2002