IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume-3, Issue-10 E-ISSN: 2347-2693

Efficient Clustering of Text Documents for Feature Selection on the use of side Information

Sonal S.Deshmukh[1*] and R.N.Phursule[2]

[1*,2] *Department of Computer Engineering*
JSPM's Imperial College of Engineering & Research, Wagholi, Pune, India

www.ijcseonline.org

Received: Sep/23/2015 Revised: Oct/26/2015 Accepted: Oct /26/2015 Published: Oct /31/ 2015

***Abstract*—** This paper presents efficient clustering with side information using probabilistic latent Semantic indexing. Meta information is available in many texts mining application. It may be useful or sometimes it is a risky approach to add side information. The aim of this work is to resolve clustering problem, for data mining problems, in which auxiliary information is available, to enhance the extraction of text document. The work proposed an approach, Probabilistic Latent Semantic Indexing, which gives more efficiency by considering class labels and also will be applicable for large number of clusters. The goal of this work is to utilize side information available with the documents for clustering, to improve the efficiency of the clusters and also to reduce the time required to form clusters.

***Keywords*—** Text Mining, Side information, Clustering, LSI, Probabilistic Latent Semantic Indexing

I. INTRODUCTION

Nowadays, the usage of World Wide Web has increased enormously. Users are interested for fast browsing of the documents as well as in relevant document. The problem of text clustering arises in many application domains such as the web applications, network applications, and other digital collections. Enormously increasing amounts of text data in the substance of these large online collections has led to a fascination in creating scalable and effective mining algorithms. Clustering is especially useful for organizing documents to enhance retrieval and support browsing.

Many text documents has text data along with other auxiliary attributes, that are also known as side information or Meta information. That side information may be useful for clustering purpose or may be harmful as it has noisy attributes. Web Users access the World Wide Web and with expecting the result in relevant data within less amount of time.

As some Meta information available in text has lots of quality data that is useful for clustering, to make efficient clusters, or users expecting to get that. So for this, we need a principle approach to access this kind of data.

Clustering is the technique, in which clusters are created for specific sort of data.

Text Mining is getting of new, previously unknown information, by obtaining information from various resources.

Corresponding Author: *Sonal S. Deshmukh, sonal.deshmukh4@gmail.com*
Department of Computer Engineering., University of Pune, India

By more conventional means of experimentation, a key element is the associating together of the extracted side information combine to organize new facts or new hypotheses to be explored further. In search, the user is typically looking for something that is previously known and that has been before stated

The problem is pushing aside all the material that currently is not relevant to users' needs in order to gain the relevant information. In text mining, the aim is to discover unknown information, something that is not obtained.

A. Clustering With Side Information

Assume that the corpus S of text documents, total number of documents is N, as T_1, T_2, …..T_N. w is the set of distinct words in the entire corpus S. Set of side attributes $\overline{X}_i$ having d dimensions, which are denoted by x_{i1}…..$x_{id.}$ These attributes are auxiliary attributes. For ease in analysis, assume that each
Attribute is binary. Some example of such attributes is as follows-
For web log analysis, side attribute x_{ir} relates to the 0-1 variable, which shows that, as the i_{th} document has been accessed by the r_{th} user or not. This technique can be used in order to cluster the web pages, so that it will be informative way for user.
For network application, side attribute x_{ir} relates to the 0-1 variable, which shows that, as the i_{th} document T_i has hyperlink to the r_{th} page $T_{r.}$

The paper components have been specified by using following entities: (II) Related Work, (III) Proposed

System, (IV) Experimental Results and Analysis, (V) Data Independency and Architecture and at last conclusion.

II. RELATED WORK

Clustering has been largely studied in the database, in terms of a wide variety of data mining tasks. The clustering problem is stated to be that of determining groups of similar objects in the data. The similarity between the objects is governed with the help of a similarity function. The problem of clustering can be useful in the text domain, where the objects to be grouped, that can be of different formats such as documents, paragraphs, sentences or terms. To enhance retrieval and support browsing clustering is especially useful for organizing documents. Clustering of text documents has already been studied in [2], but it was only for the plain text.
The database community evaluated the problem of text-clustering [2], [3].

Actually this work get motivated from [2], [4].For very large data sets, clustering method BIRCH is proposed in [4], it makes a large clustering problem manageable by focusing on densely occupied portions, and using a compact summary. But for BIRCH's efficiency proper parameter setting is important.

In [5] gives a survey of text clustering methods. One of the most well -known techniques for text-clustering is the scatter-gather technique [6], agglomerative and partitioned clustering are combined in scatter-gather technique. It demonstrates that document clustering can be an effective information access tool in its own right. It is particularly helpful in situations in which it is critical or undesirable to specify a query formally. To support Scatter/Gather, fast clustering algorithms are necessary. Clustering can be done quickly by working on small groups of documents rather than trying to deal with the complete corpus globally.

An enormous amount of work has been done in recent years on the problem of clustering in text collections [7], [6], [8], [9], [10] in the database and information retrieval communities. However, this work is primarily defined for the solution of pure text clustering, in the absence of other kinds of side attributes.

The major concentration of this work has been on scalable clustering of multidimensional data of different sorts [2], [3], [4], and [11]. A general survey of clustering algorithms may be found in [12]. The problem of clustering has also been studied quite extensively in the context of text data. [4] States a survey of text clustering methods. Other related methods for text-clustering which use similar methods are studied in [8]. This technique selects words from the document based on their relevance, and uses an iterative EM method in order to purify the clusters. [13], [14] stated closely regarded area that is of topic modeling, event tracking, and text-categorization. [15] Discusses methods for text clustering in the context of keyword extraction. A comparative study of different clustering methods may be found in [9]. The text clustering problem has also been studied in context of scalability in [7], [10]. However, these methods are defined for the case of pure text data, and are not useful for cases in which the text-data is merged with other forms of data. In the context of network-based linkage information, some limited work has been done on clustering text data [16], [17] though this work is not appropriate to the case of general side information attributes.

A first approach to incorporating other sorts of attributes in combination with the text clustering. [1] It showed the benefits of using such a method over pure text-based clustering. Such an approach is particularly useful, when the auxiliary information is greatly informative, and gives effective process in creating more coherent clusters. It has extended the method to the problem of text classification, which has been studied extensively in [1].

Side-information is available in many mining applications along with the documents. Side-information like the hyperlinks or non-textual attributes, user-access behavior from web logs. Data mining has attracted a great deal of attention due to the availability of this side-information, in the information industry and in society as a whole in recent years. Because of this, there is a need of using a proposition way to execute the mining process, so that system are able to enhance the advantages from using this meta information which escort to an interest in creating scalable and effective mining algorithms. Data mining mentions to obtaining or "mining" knowledge from large amounts of data means it is the process that finds a small set of valuable things from a great deal of raw material.

Work is strived by research in the field of clustering. Hierarchical method can be disunited into agglomerative and divisive variants. Hierarchical clustering is used to develop a tree of clusters, also known as a dendrogram. Every node consists of child clusters. In hierarchical clustering we allocate each item to a cluster such that if we have N items then we have N clusters. Find nearest pair of clusters and then merge them into single cluster. So, that one cluster get reduce from the whole structure. Then calculate similarities among the new cluster and each of the old clusters. And it work until all k cluster get combined.

Therefore, there are the two different approaches while clustering the data that are agglomerative and divisive. In the first agglomerative method, a bottom-up clustering normally used, it works from bottom to up. This method starts with a single cluster which includes all objects, and then it splits resulting clusters until only clusters of

individual objects left. It finish when individual cluster contain a single object. In the second divisive approach that is top-down clustering method and is rarely used. It works in the reverse direction to that of agglomerative approach. It work from top to bottom and other steps is same as that of agglomerative approach [8].

Dhillon proposed Co-clustering methods for text data. Work is also motivated by research in the field of mining and usually used partitioning method. Partitioning method divide the data into different subset or group such that some criterion that evaluate the clustering quality is optimize. Partitioning algorithm is of different type such as: probabilistic clustering using the Naive Bayes or Gaussian mixture model, EM, K-Means, Aproiri, FP-growth, Fuzzy-C-Means etc.

In a similar manner, Bradley, Fayyad, and Reina (1998) describe a heuristic algorithm for an implementation of the Expectation-Maximization (EM) algorithm applied to Gaussian mixture modeling on massive data sets, which seeks to reduce the number of passes through the data set.

Many algorithms are developed and classified into two classes: candidate generation or pattern growth. Apriori algorithm first creates an association rule for frequent pattern matching. Apriori is a representative of the candidate generation approach. It produces length (k+1) candidate item sets based on length (k) frequent item sets. Since Apriori algorithm was first introduced, there have been many attempts to devise more efficient algorithms of frequent item set mining [6]. The existing algorithm has two phases. The initial phase of the algorithm uses the Latent Semantic Indexing (LSI) [8] which is used for the fixed number of cluster and also ignores the class labels of training document. LSI generated representation are not as effective in classification tasks.

III. PROPOSED SYSETM

The proposed method consists of efficient clustering algorithm for clustering the documents. The proposed system consists of the Probabilistic Latent Semantic Indexing which present novel method for automated indexing based on statistical latent class model. The proposed work will enhance the efficiency of text clustering with side information by reducing the time to form clusters.

A. Problem Definition

To utilize side information available with the documents for clustering, to enhance the efficiency of the clustering operation.

B. Proposed Method

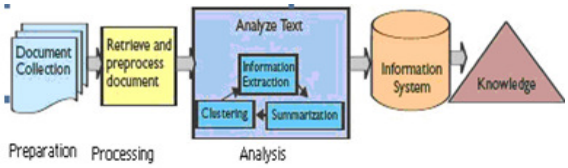

Figure 1. Proposed Work

C. System Architecture

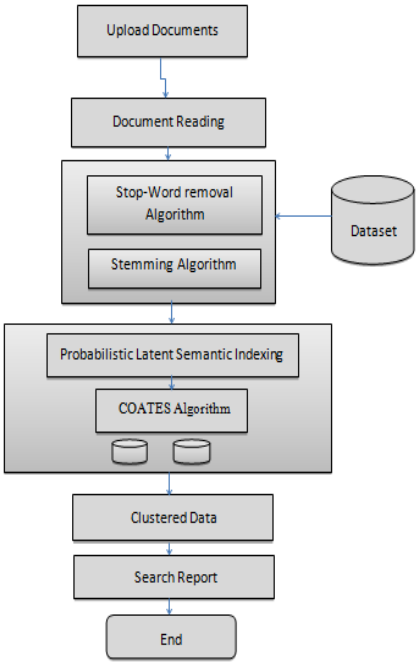

Figure 2. System Architecture

D. Algorithm

COATES (Clusters: k, Corpus: $T1 \ldots T_N$, Auxiliary Attributes: $X_1 \ldots X_N$);

Begin

Use probabilistic latent semantic indexing to create initial set of k clusters

Denote the centroids of k clusters.

t=1; //Number of Iterations

While not (*termination_criterion*) **do**

Begin

{First Iteration}

Use cosine-similarity of each document to centroids, in order to determine the closest clusters;

Update the cluster assignments;

Denote assigned cluster index for document;

Update cluster centroids to the centroids of updated clusters;

{Second Iteration}

Compute gini-index for each auxiliary attribute with respect to cluster;

Mark attributes with gini-index, which is standard deviations below the mean;

For each document, use document assignment method to clusters to determine posterior probability;

t=t+1;

end

end

E. Proposed Technique

Probabilistic Latent Semantic Indexing:

Probabilistic Latent Semantic Analysis (pLSA) is a method from the class of topic models. Its main goal is to model co-occurrence information under a probabilistic structure in order to find the underlying semantic structure of the data. pLSA aims to factorize the sparse co-occurrence matrix in order to reduce its dimensionality.
PLSA considers that our data can be expressed in terms of 3 sets of variables:

1. Documents: $d \in D = \{d_1, \ldots, d_N\}$ ----observed variables. Let N be their number, defined by the size of our given corpus.
2. Words: $w \in W = \{w_1, \ldots, w_M\}$ ----observed variables. Let M be the number of distinct words from the corpus.
3. Topics: $z \in Z = \{z_1, \ldots, z_K\}$----latent (or hidden) variables. Their number, K, has to be specified a priori.

In latent semantic indexing, the actual vector space representation of documents is replaced by a representation in the low dimensional latent space and the similarity measure, based on that representation. Probabilistic Latent Semantic Indexing is a novel method for automated indexing based on a statistical latent class model. This approach has important theoretical benefits over standard LSI, since it is based on the likelihood principle, defines a generative data model, and directly minimizes word perplexity.

F. Multiplexer Logic

A multiplexer of 2^n inputs has *n* select lines, which are used to determine which input line to send to the output. Multiplexers are mostly used to increase the amount of data that can be sent over the network within a certain amount of time and bandwidth. A multiplexer is also known a **data selector**.

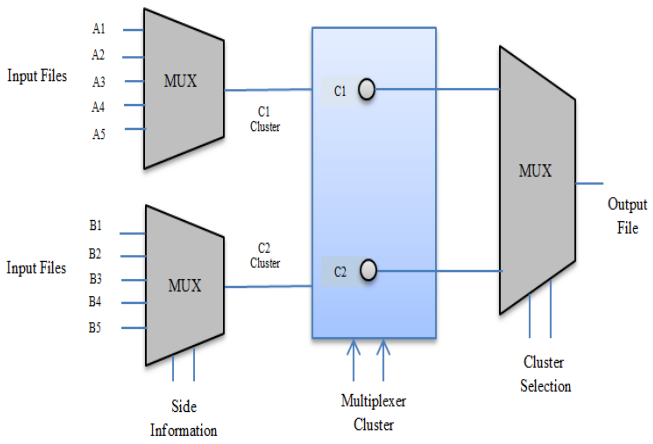

Figure 3. Multiplexer Logic

IV. DATA INDEPENDANCY AND ARCHITECTURE

A. Set Theory

Input: Content with side information
Output: Time required for Clustering process
Function: Input, data preprocessing, clustering with side information, Re-clustering with COATES algorithm, results

B. Mathematical Model

$$M = (Q, \Sigma, \delta, q0, F)$$

Where,
$Q = \{q_0, q_1, q_2, q_3, q_4\}$,
$\Sigma = \{P, Q, R, S, T\}$,
$q_0 = q_0$,
$F = \{q_4\}$,

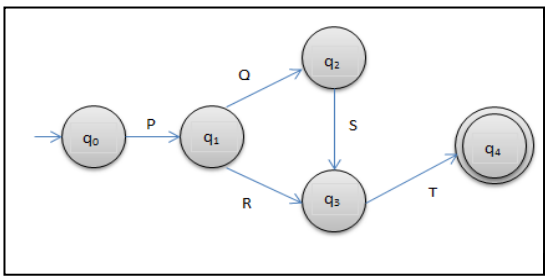

Figure 4. Mathematical Model

Where,
q_0= Document collection
q_1= Preprocessing of documents
q_2= Keyword Extraction
q_3= Clustering
q_4= Re-clustering with COATES algorithm

P= Collected documents
Q= keywords
R= Auxiliary attributes
S= Extracted Keywords
T= Clustered document

δ = State Transition Table:

	P	Q	R	S	T
S1	S2	-	-	-	-
S2	-	S3	S4	-	-
S3	-	-	-	S4	-
S4	-	-	-	-	S5
S5	-	-	-	-	-

V. EXPERIMENTAL RESULTS AND ANALYSIS

A. *Dataset*

For experimentation, this can use any of dataset like Cora Data Set, DBLP-Four Area Data Set and IMDB Data Set- Internet Movie Database. Here, this work uses IMDB Data Set- Internet Movie Database.

B. *Expected Results*

Here, this compares proposed system with existing system on the basis of the time and space required to form clusters. The following table shows the time and space required to form 3 clusters using LSI and PLSI.

Table 1: Time and space required for clustering using LSI and PLSI

No. of Documents	Ti me Required for clustering by COATES (In seconds)		Space Required for clustering by COATES (In Bytes)	
	using LSI	using PLSI	using LSI	using PLSI
21	112.0	75.0	8651192.0	3612008.0

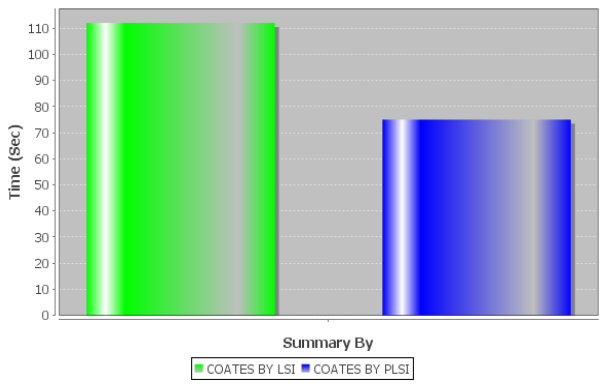

Figure 5. Time Comparison using LSI and PLSI

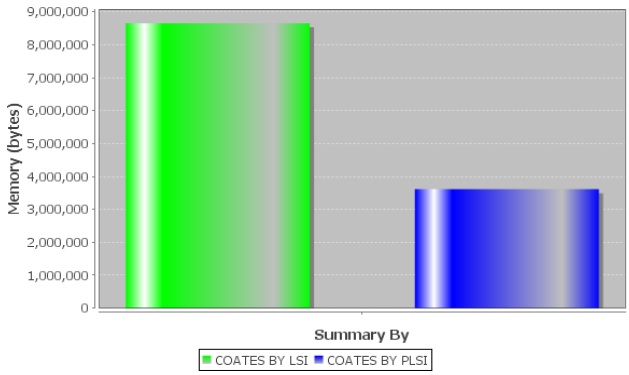

Figure 6. Space Comparison using LSI and PLSI

VI. CONCLUSION

The paper presents probabilistic approach for effective clustering for text documents with side information. Many forms of text-databases contain a large quantity of meta-information or meta-information, which may be used in order to refine the clustering process. The algorithm proposed which utilizes side information available with the documents for clustering, to enhance the efficiency of the clustering operation.

This paper presents the fast clustering algorithm with probabilistic latent semantic indexing that will illustrate the effectiveness of the approach. The result will show that the use of side-information can greatly enhance the quality of text clustering for large number of clusters, while maintaining a high level of efficiency, with minimum amount of time and space required to form clusters. And relevant and most accurate data get generated.

ACKNOWLEDGMENT

I would like to thank my guide Prof. R.N.Phursule for his timely, valuable guidance and support. I would also like to thank to the HOD, Prof. S.R.Todmal for his support and motivation.

REFERENCES

[1] Charu C. Aggarwal and Yuchen Zhao, "On the Use of Side Information for Mining Text Data", in IEEE Transactions on Knowledge and Data Engineering, Vol. 26, No. 6, 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] R. Ng and J. Han, "Efficient and effective clustering methods for spatial data mining," in Proc. VLDB Conf., San Francisco, CA,USA, **1994**, pp. 144–155.

[4] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: An efficient data clustering method for very large

databases," in Proc. ACM SIGMOD Conf., New York, NY, USA, **1996**, pp. 103–114.

[5] Vilas V Pichad and Sachin N Deshmukh, "Role of Document Clustering For Forensic Analysis Investigation System", International Journal of Computer Sciences and Engineering, Volume-03, Issue-03, Page No (116-120), Mar -**2015**, E-ISSN: 2347-2693

[6] 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.

[7] C. C. Aggarwal and P. S. Yu, "A framework for clustering massive text and categorical data streams," in Proc. SIAM Conf. Data Mining, **2006**, pp. 477–481.

[8] H. Schutze and C. Silverstein, "Projections for efficient document clustering," in Proc. ACM SIGIR Conf., New York, NY, USA, **1997**, pp. 74–81.

[9] M. Steinbach, G. Karypis, and V. Kumar, "A comparison of document clustering techniques," in Proc. Text Mining Workshop KDD, **2000**, pp. 109–110.

[10] S. Elakkiya and T. Kavitha, "Detection of Text Using Connected Component Clustering and Nontext Filtering", International Journal of Computer Sciences and Engineering, Volume-03, Issue-04, Page No (53-57), Apr -**2015**, E-ISSN: 2347-2693

[11] 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**.

[12] A. Jain and R. Dubes, Algorithms for Clustering Data. Englewood Cliffs, NJ, USA: Prentice-Hall, Inc., **1988**.

[13] C. C. Aggarwal, S. C. Gates, and P. S. Yu, "On using partial supervision for text categorization," IEEE Trans. Knowl. Data Eng., vol. 16, no. 2, pp. 245–255, Feb. **2004**.

[14] G. P. C. Fung, J. X. Yu, and H. Lu, "Classifying text streams in the presence of concept drifts," in Proc. PAKDD Conf., Sydney, NSW, Australia, **2004**, pp. 373–383.

[15] 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.

[16] C. C. Aggarwal and H. Wang, Managing and Mining Graph Data. New York, NY, USA: Springer, **2010**.

[17] C. C. Aggarwal and P. S. Yu, "A framework for clustering massive text and categorical data streams," in Proc. SIAM Conf. Data Mining, **2006**, pp. 477–481.