International Journal of Technical Research and Applications e-ISSN: 2320-8163,
www.ijtra.com Volume 4, Issue 4 (July-Aug, 2016), PP. 139-144

Enhancing Quality of Text Clustering based on Side Information data.

Kajal R. Motwani [1], B. D. Jitkar [2],
Department of Computer Science & Engineering,
[1]D. Y. Patil College of Engineering & Technology,
[2]D. Y. Patil College of Engineering & Technology,
Kolhapur,India.
[1]kajal.motwani06@gmail.com, [2] bjitkar@rediffmail.com

***Abstract*-Many text mining applications contain side-information available along with the text documents. Such side information include: document provenance information, links in the document, user-access behavior from web logs, or other non-textual attributes. Such attributes may contain informative data which can be useful for purposes of text clustering. Traditional text clustering techniques are available that yield good amount of clusters but it is hard to say whether the quality of the clusters is as good as expected. Since side information contains meaningful data, it has an advantage that adding such side information into the clusters can greatly enrich the quality of clusters. However, some of the information is noisy, so it must be added carefully. So, we need a correct way to perform the mining process. In this paper, we propose an effective clustering technique that identifies important side information in the documents by using Shannon information gain and Gaussian distribution and using Bayesian probability, it estimates whether adding such side information enriches the quality of clusters. This approach is further extended for generating classification labels which makes it easier to cluster large number of documents.**

Keywords: **Data mining, text clustering, side information, Gaussian distribution, Bayesian probability.**

I. INTRODUCTION

Text Clustering is a widely studied domain nowadays as most of the data is present in textual format. Text Clustering is needed in many application domains such as documents organization and browsing, digital collections, social networks, etc. The dimensionality of textual representation is high but data present while considering individual document is sparse. Hence to handle such data, effective mining algorithms are needed. Many algorithms are available on the problem of text clustering in the database and information retrieval communities. But, these algorithms are mostly designed for the problem of pure text clustering.

In many text mining applications, side information is associated with text documents which can be useful for text clustering. [1] Such attributes may contain tremendous amount of information for clustering purposes. Examples of such side information include:

- Web logs contain meta-data that corresponds to browsing behavior of users. Such logs can be useful as they can depict associations in content and hence can be used to improve the quality of mining process.
- Different links present in text documents also can be treated as side attributes as they contain useful information for mining purposes which cannot be fetched from raw content alone.
- Many web documents contain different kinds of attributes such as the information about the origin of document, ownership, location, user-tags, etc.

However, some side information may be noisy. In such cases incorporating side information into mining process can be risky because it can either improve the quality of clustering process or degrade it. So a proper way is needed to add such side information.

The motive here is to use both textual attributes and side information for clustering purposes. A possible solution proposed here is, to design an effective clustering technique to cluster the documents by incorporating side information data that will enrich the quality of clusters using Bayesian probability by adding only important and highly distributed side information data which is obtained by applying information gain and Gaussian distribution. In order to make the consequent process more effective this approach is extended to the classification problem.

The paper is organized as follows: Section II presents the related work on the topic. Section III describes the proposed methodology. Section IV presents experimental results. Section V gives the conclusion and future directions.

II. RELATED WORK

Much of the work is done on text clustering [3],[4],[5]. One of the most popular techniques used for text clustering is Scatter/Gather technique [6] which uses partitional clustering and can be used as effective

information access tool. However, this technique and other text clustering algorithms are designed only for pure text clustering and do not consider side information.

Charu C. Aggarwal, Yuchen Zhao, and Philip S. Yu [1],[15] have proposed a model which makes use of partitioning algorithm with Bayesian probability in order to yield an effective clustering technique. They have also given an approach to extend this technique to classification problem. Authors have made use of gini index for eliminating noisy side information and adding only important side information. They have tested the algorithm on 3 real time datasets. The results show that side-information can be useful in enhancing the quality of text clustering and classification.

C. C. Aggarwal and C.-X. Zhai [2] have given an overview of all the methods for text clustering and text classification. They have defined text-specific algorithms for document representation and processing.

Tian Xia and Yanmei Chai [7] have used the Term Frequency Inverse Document Frequency (TF-IDF) to find out which words in the collection of documents are important to be used in query based on the weights assigned to words using TF-IDF.

M. Steinbach, G. Karypis, and V. Kumar [8] have compared different classification algorithms. They have come out with the conclusion that *K*-means clustering outperforms agglomerative and hierarchical clustering techniques.

Yiming Yang, Jan O. Perdersen [9] have presented different feature selection methods. They have found out that Information gain(IG) performs quite well by identifying unique features. IG is used with k-nearest neighbor classification algorithm on Reuter-22173 dataset which improved classification accuracy.

Ryan Prescott Adams, George E. Dahl and Iain Murray [10] have proposed a probabilistic matrix factorization model which makes the use of Gaussian process priors for incorporating side information. Authors have applied this method to estimate the scores of basketball games. The side information used here is venue and date of the game.

P. Domingos and M. J. Pazzani [11] have tested the optimality of the Bayesian classifier and verified that it performs quite well in many domains. They showed that Bayesian classifier has advantages in terms of simplicity, classification speed and is better classifier when the data size is small.

III. PROPOSED METHOD

The proposed system is divided into four phases, namely, Preprocessing and formation of initial clusters, Processing of side information data, Enriching Clusters based on both text content and side information, Classification. The first three phases describe the clustering technique. Fourth phase describes the classification technique. Figure 1 gives the architecture diagram of proposed system.

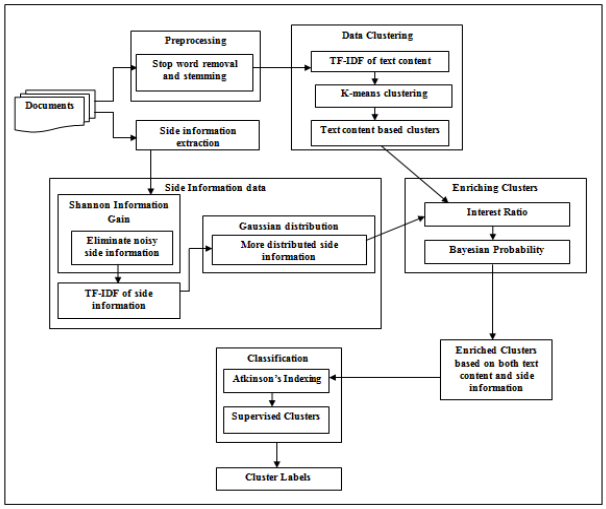

Fig. 1. System Architecture diagram

The overall technique works as follows: The input is set of text documents. Next data mining tasks such as stemming and stopword removal are performed. Using K-means algorithm, initial set of clusters are formed. Side information data is extracted and processed. By computing Shannon information gain, noisy side information data is discarded. This side information is given as input to Gaussian distribution which returns highly distributed side information data. Next using Bayesian probability, side information data gets added to the initial clusters to improve its quality. This approach is extended to classification problem. Classification labels are generated based on which documents get classified.

1) Preprocessing and formation of initial clusters:

In order to ease the data mining task, data representation plays a vital role. Hence preprocessing of data is carried out. Stemming is done wherein the words are reduced to their base forms using Porter Stemming Algorithm [12]. This algorithm is suffix stripping algorithm. Stopwords such as is, an, the, etc are also eliminated using an English stopwords list which contains more than 500 stopwords. Special symbols such as @, #, / etc are also discarded. Once the data is preprocessed, initial set of document clusters are formed based purely on the text content (abstract) of the document and not on the side information. For this, the data is represented in the TF-IDF [7] (Term Frequency-Inverse document frequency) form. For every word present in the abstract of the document, TF-IDF is computed which assigns weights to each word. TF-IDF of each word in the document is computed using (1).

$$w_d = f_{w,d} * log(|D|/f_{w,D}) \quad (1)$$

where,
$f_{w,d} =$ number of times word w appears in document d,
$|D| =$ size of all documents in collection,
$f_{w,D} =$ number of documents in which word w appears in D.

Using these TF-IDF values of the words, documents are clustered. For clustering documents *K*-means algorithm [8] is used. It is the simplest and most popular algorithm used for clustering. The algorithm partitions the words into *k* clusters wherein each word belongs to cluster with nearest mean.

2) Processing of Side information data:

Side information data present in the document is extracted and preprocessed. The side information data that is extracted is URL, Title, Author, Keyword, Address, Affiliation, Phone, web and date. Stemming is done and stopwords are eliminated. Not all side information is important for mining purposes. Hence we need to eliminate the unwanted side information. For eliminating noisy side information, we use Shannon Information gain [9]. This is an attribute selection measure. Shannon information gain reflects impurity and is calculated using (2).

$$Info(D) = -\sum_{i=1}^{m} p_i \, log_2(p_i) \quad (2)$$

where,
p_i is the is the nonzero probability that an arbitrary tuple in *D* belongs to class C_i and is estimated by, $|C_{i,D}|/|D|$.

A threshold is set to 0.6 and only the side information above this threshold are considered as the important side information and hence are retained and side information below 0.6 indicates noisy side information and hence are discarded[14]. The TF-IDF (Term Frequency-Inverse document frequency) of this side information obtained is calculated using (1). This TF-IDF of side information is given as an input to the Gaussian distribution which returns highly distributed side information. Gaussian distribution [15] is known as normal distribution. It is a popular probabilistic model used for distribution. In statistics real valued random variables are more often represented by using Gaussian distribution. The main reason behind the usefulness of Gaussian distribution function is central limit theorem. The curve formed due to normal distribution is also synonym as a bell curve. The Gaussian distribution is effectively calculated using (3):

$$P(a < y < b) = \frac{1}{\sigma\sqrt{2\pi}} \int_a^b e^{-\frac{(y-\mu)^2}{2\sigma^2}} \quad (3)$$

where,
$y =$ Continues variable value of TF-IDF of side information,
$a =$ minimum TF-IDF value of side information,
$b =$ maximum TF-IDF value of side information,
$\mu =$ mean of distribution,
$\sigma =$ variance of distribution.

Using the value returned by Gaussian function, a range is obtained as follows:

min_range = (mean) – (gaussian function)
max_ range = (mean) + (gaussian function) (4)

All the side information whose TF-IDF values lie in the min to max range are considered. These are highly distributed side information.

3) Enriching Clusters based on both text content and side information:

The initial clusters formed using K-means algorithm and the side information obtained from the Gaussian function is considered. This is the main phase wherein the coherence of side information is determined with respect to text based clusters and eventually clusters get enriched based on the side information. Using Interest Ratio, the importance of side information with respect to the cluster is determined. For finding the interest ratio, firstly, for each text based cluster, the minimum and maximum range of TF-IDF (Term Frequency- Inverse Document Frequency) is calculated for each cluster. Then, the side information which lie in between this range of text based clusters are found out. Before this comparison is done, the values of TF-IDF of side information are normalized because TF-IDF values obtained from Gaussian distribution are too low. Next using Bayesian probability, probability is estimated as to which side information gets added into the clusters and eventually considering this probability clusters get enriched. Once interest ratio is computed, probability is estimated as to which side information gets fit into which clusters for enriching cluster quality. For this, Bayesian probability [10] is used. It is estimated as follows:

$$P(H/E) = \frac{P(E/H)*P(H)}{P(E)} \quad (5)$$

where,
$P(H/E)$ = posterior probability, i.e. probability of *H* given *E*.
$P(H)$ = prior probability, i.e. probability of *H* before *E is* observed.
$P(E/H)$ =Probability of observing *E* given *H.*
P(E)= 1(same for all classes)

Here, 0.3 is considered as threshold and side information having probability above 0.3 are considered, rest are discarded. Based on the dataset used, a threshold of 0.3 suits

best for this scenario. If this particular side information improves the cluster quality, side information will be added into the clusters.

In text based clusters obtained using k-means, TF-IDF is associated with each file which is obtained by finding the mean of all the TF-IDF of words present in that particular file. Once the probabilities of side information are computed, probability of each of the side information is compared with the TF-IDF of the file and the file having closest TF-IDF value as compared to probability is found. Once the file is found, the probability of the word (side information) is added to the TF-IDF of file to obtain new TF-IDF values. This is done for every word of side information obtained using Bayesian probability. This generates new TF-IDF values for each of the file if it is closer to any side information. So considering these new TF-IDF values, *K*-means algorithm is implemented to form new clusters. Thus we get enriched clusters which are based on both text content and side information as probability of side information is used to get the new TF-IDF values. These clusters are enriched clusters and of better quality.

4) Classification:

The clustering approach can be extended to classification problem to ease the process of clustering large number of documents. Classification labels are generated. For generating labels we use the side information obtained from Bayesian probability. Using Atkinson index supervised clusters are formed. Further, cosine similarity is computed between supervised clusters and initial text based clusters. Next labels are generated based on rule and saved in the database. Once the labels get saved in the database, the system matches the TF-IDF values of side information and classification labels saved, based on that it clusters the documents. If there is no match, then the document will be clustered according to the clustering technique. Classification technique is described next.

Side information is considered. Supervised clusters are formed. These clusters are formed based on the centroid and Atkinson index values. Atkinson index is measure of inequality. It is useful in finding out as to which end of distribution contributed most to the observed inequality. Here we determine how much TF-IDF value of each side information data deviate from mean TF-IDF value. Side information gets added into respective clusters based on the conditions below:

Cluster 1: Tf_Idf < Centroid
Cluster 2: Tf_Idf >= Centroid
Cluster 3: Atkinson Index Difference < Atkinson Index
Cluster 4: Atkinson Index Difference >= Atkinson Index

Next similarity is found between supervised clusters and the initial content based clusters. For computing similarity, cosine similarity measure [13] is used. Cosine similarity for two documents DOC_i and DOC_j is computed as follows:

$$SIM(DOC_i, DOC_j) = \frac{\sum_{k=1}^{t}(TERM_{ik}.TERM_{jk})}{\sqrt{\sum_{k=1}^{t}(TERM_{ik})^2.\sum_{k=1}^{t}(TERM_{jk})^2}} \quad (6)$$

where,
$\sum_{k=1}^{t}(TERM_{ik}.TERM_{jk})$ = component-by-component vector product.
$\sqrt{\sum_{k=1}^{t}(TERM_{ik})^2}$ = length of the term vector for DOC_i.
$\sqrt{\sum_{k=1}^{t}(TERM_{jk})^2}$ = length of the term vector for DOC_j.

Once cosine similarity is obtained, i.e., clusters having cosine similarity above 0.7 are considered. This is because more the value of cosine similarity gets closer to 1, more the documents in respective clusters are similar to each other. Only these clusters are considered for label generation.

Classification labels are generated for each cluster based on the mean TF-IDF values of each cluster. Labels are in the form of ranges. These labels get saved in the database. Next label checking is performed and if the TF-IDF values of side information fits into any of the labels ranges, the documents get clustered based on these labels.

IV. EXPERIMENTAL RESULTS

In order to evaluate the performance of clustering system, the side information due to which the documents get clustered are evaluated. Here, system is tested on two datasets, Cora dataset and Reuters21578 dataset is used. Cora dataset is collected from "http://www.cs.umass.edu/~ mccallum/code-data.html". The Cora data set contains 19,396 scientific publications in the computer science domain. Side information extracted from the Cora data set are url, author, title, keyword, address, affiliation, web. Reuters dataset is collected from "http://www.daviddlewis.com/resources/testcollections/reuters21578/". It contains 21578 Reuters news documents from 1987. Side information extracted from Reuters21578 dataset are places, author, title, dateline. These are used as separate attributes in order to assist in the clustering process. http://www.daviddlewis.com/resources/testcollections/reuters21578/

A) For evaluating the side information, three parameters namely, Precision, Recall, F-measure are used. Results indicate that side information greatly enriches the quality of clusters.

TABLE I. PRECISION, RECALL, F-MEASURE FOR CORA DATASET.

No. of docs	Precision (%)	Recall (%)	F-measure (%)
20	100%	100%	100%
25	96.77%	100%	98.36%
30	100%	100%	100%
35	98.03%	100%	99%
40	98.07%	100%	99.02%

TABLE II. PRECISION, RECALL, F-MEASURE FOR REUTERS21578 DATASET.

No. of docs	Precision (%)	Recall (%)	F-measure (%)
20	97%	100%	98%
25	100%	100%	100%
30	98%	100%	97.4%
35	95%	100%	96%
`40	100%	100%	100%

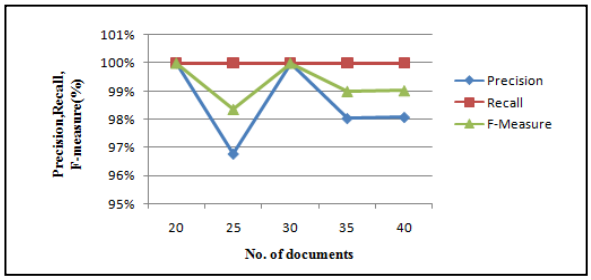

Fig. 2. Graph for Precision, Recall, and F-measure for Cora dataset.

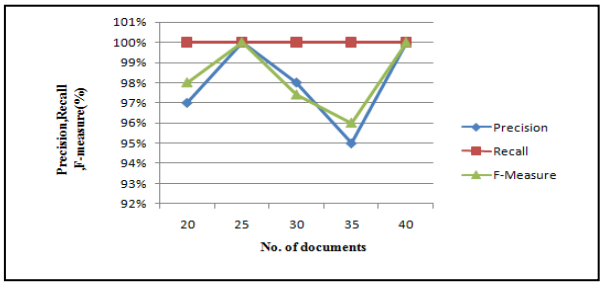

Fig. 3. Graph for Precision, Recall, and F-measure for Reuters21578 dataset.

B) The running time required to cluster documents based on classification labels and without classification labels is calculated. Also the average percentage improvement is calculated. In Table 2 below, percentage improvement reflects that classification labels can be useful in clustering documents in lesser time than the documents clustered without classification labels.

TABLE III. RUNNING TIME AND PERCENTAGE IMPROVEMENT FOR CORA DATASET.

No. of documents	Without classification labels, time required(milliseconds)	With classification labels, time required(milliseconds)	Percentage Improvement (%)
20	15571	11520	26%
25	25850	16472	30%
30	31115	18022	42%
35	28799	20665	28%
40	37642	23861	36%

TABLE IV. RUNNING TIME AND PERCENTAGE IMPROVEMENT FOR REUTERS21578 DATASET.

No. of documents	Without classification labels, time required(milliseconds)	With classification labels, time required(milliseconds)	Percent Improvement (%)
20	16634	13678	17%
25	27980	19400	30%
30	32897	26089	20%
35	33980	25965	23%
40	39034	31890	18%

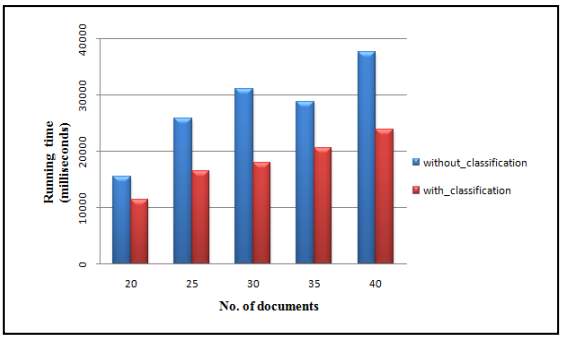

Fig. 4. Graph of running time with classification labels and without classification labels for Cora dataset.

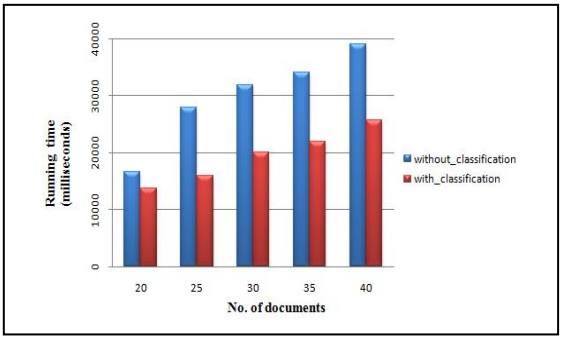

Fig. 5. Graph of running time with classification labels and without classification labels for Reuters21578 dataset.

C) To find the correctness of classification labels, Accuracy measure is used.

TABLE V. ACCURACY

No. of documents	Accuracy (Cora dataset)	Accuracy (Reuters21578 dataset)
20	85%	70%
25	92.78%	82.67%
30	90%	85%
35	89%	78%
40	88.70%	67.78%

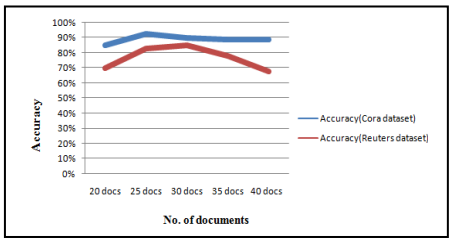

Fig. 6. Graph showing Accuracy.

V. CONCLUSION AND FUTURE DIRECTIONS

Clustering technique for mining text data with the use of side information is presented in this paper. There are many other kinds of attributes known as side information in the document that contain meaningful information that is useful for clustering. Here, system is designed to cluster documents using such side information. Using Gaussian distribution and Bayesian probability only important side information gets added in to the clustering process. This approach is extended to classification where classification labels are generated using both side information and text based clusters. These labels are useful in clustering large number of documents. Results show that use of side information greatly enriches the quality of text clustering and classification.

As a future work, Clustering technique can be useful for optimizing search in web applications. Web application can be designed to fetch documents based on particular side information which is entered as query. For example, when user enters name of one or more authors at a time, the application can be designed in such a way that the application fetches all the documents which contain both author names in one document and also the documents belonging to each author individually.

REFERENCES

[1] Charu C. Aggarwal, Yuchen Zhao, Philip S.Yu, "On the Use of Side Information for Mining Text Data" IEEE,2014.

[2] C. C. Aggarwal and C.-X. Zhai, Mining Text Data. New York, NY,USA: Springer, 2012.

[3] 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

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

[5] 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.

[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] Tian Xia, Yanmei Chai , "An Improvement to TF-IDF: Term Distribution based Term Weight Algorithm" .

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

[9] Yiming Yang, Jan O. Perdersen "A comparative study of feature selection In text categorization"

[10] Ryan Prescott Adams, George E. Dahl, Iain Murray, "Incorporating Side Information in Probabilistic Matrix Factorization with Gaussian Processes", arXiv preprint , 2010.

[11] P. Domingos and M. J. Pazzani, "On the optimality of the simple Bayesian classifier under zero-one loss," Mach. Learn., vol. 29, no. 2–3, pp. 103–130, 1997.

[12] C.Ramasubramanian,R.Ramya"Effective Pre-Processing Activities in Text Mining using Improved Porter's Stemming Algorithm "International Journal of Advanced Research in Computer and Communication EngineeringVol. 2, Issue 12, December 2013.

[13] G. Salton, An Introduction to Modern Information Retrieval. London,U.K.: McGraw Hill, 1983.

[14] Data Mining:Concepts and Techniques,Third Edition,Jiawei Han,University of Illinois at Urbana–Champaign Micheline Kamber,Jian Pei ,Simon Fraser University.

[15] Kevin Lehmann, "Gaussian Distributions" Department of Chemistry Princeton University, Princeton.

[16] C. C. Aggarwal and P. S. Yu, "On text clustering with side information," in Proc. IEEE ICDE Conf., Washington, DC, USA, 2012.

[17] Kajal R. Motwani, B.D. Jitkar "Enriching Quality of text clustering using side information data", 6th National Conference on Emerging Trends in Engineering, Technology & Architecture, NCETETA-2016.