International Conference on "Emerging Trends in Computer Engineering, Science and Information Technology "-2015
Special Issue of International Journal of Electronics, Communication & Soft Computing Science and Engineering, ISSN: 2277-9477

Clustering of Documents Based on Partitioning the Features

Madhuri V. Malode **Prof. J.V. Shinde**

Abstract **-To find the appropriate number of clusters and to partitioned the documents is crucial in document clustering. In this paper we will focus on various clustering techniques and our proposed system is to discover the cluster structure without giving the total number of clusters as input. Document features or even we can say that the various attributes will be with no human interference separated into two groups, in particular, discriminative words and nondiscriminative words, and contribute differently to document clustering. There is variational inference algorithm in which we infer the document collection structure and words at the same time partition of document. our proposed approach for the semisupervised document clustering. Semi-supervised clustering lies between both automatic categorization and auto-organization. Here the supervisor need not specifies a set of classes, but only to provide a set of texts grouped by the criteria to be used to generate Clusters.**

Keywords**— Database applications, text mining, pattern recognition, clustering document clustering, feature partition**

I. INTRODUCTION

1.1 Clustering

A *cluster* is so a collection of objects which are "similar" between them and are "dissimilar" to the objects belonging to other clusters.

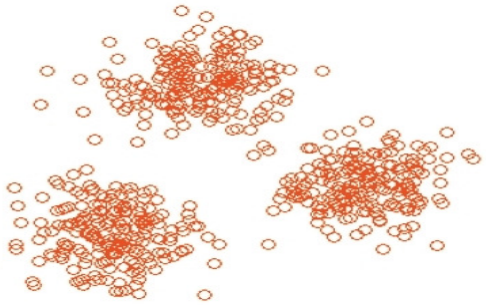


Fig : 1.1 Clustering Overview

Search engines or any information retrieval application are an invaluable tool for retrieving information from the Web.

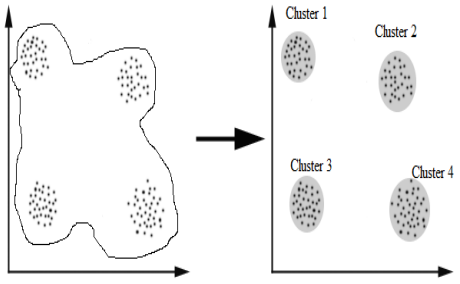


Fig : 1.2 Cluster Formation

The purpose of clustering is to decrease the amount of data by categorizing or grouping similar data items and present them collectively. Such grouping is persistent in the way humans process information, and one of the inspirations for using clustering algorithms is to provide automated tools to help in constructing categories or taxonomies. The user starts at the top of the list and chase it down examining one result at a time, until the sought information has been found. Last method is searching results clustering, which consists of grouping the results returned by a search engine into a hierarchy of labeled clusters (also called categories).

1.2 Document Clustering

Document clustering has been investigated for use in a number of different areas of text mining and information retrieval. Initially, document clustering was used for improving the precision or recall in information retrieval applications and as an efficient way of finding the nearest neighbors of a document so that system will return the max relevant document in response to user's query. Document clustering has also been used to automatically generate hierarchical clusters of documents.

II. SURVEY ON VARIOUS CLUSTERING TECHNIQUES

There are many clustering techniques which are available in the market, and each of them may give a different grouping of objects. The option of a exacting technique will depend on the kind of output preferred that is it depends on the end user to select one of them as per his requirement and form the desired number of clusters. The recognized performance of method with particular types of data, the hardware and software facilities available and the size of the dataset.

*2.1 Single Pass Clustering Techniques***:** A very simple division method, the single pass method creates a partitioned dataset as follows:

- ❖ In this first object will declare as a cluster representative of that cluster.
- ❖ Then subsequent objects after comparing the threshold value will be compared against the Cluster representative.
- ❖ In this way cluster will be formed of given objects.

2.2 Hierarchical Methods

The hierarchical clustering methods are most commonly used. The construction of this classification can be achieved by the following general steps.

1. Find the 2 nearest objects and merge them to form a new cluster
2. Find and merge the next two nearest objects where a point is either an individual object or a cluster of objects.
3. If more than one cluster remains , return to step 2

2.3 Partition Clustering:

It tries to directly decompose the given data set or objects into a set of disjoint clusters. Naturally the worldwide criteria entail minimizing some measure of dissimilarity in the samples within each cluster, while maximizing the dissimilarity of different clusters.

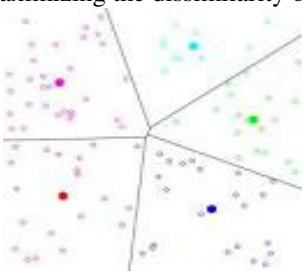


Fig 2.1 Partition Clustering

2.4 Various implementation aspects related with our proposed system

While doing document feature extraction we can any one of the information retrieval such as following:

- Boolean model
- Extended Boolean model
- Vector model
- Vector space model
- Fuzzy model

Following code will describe the comparison of documents based on assignment of the vectors ;

```
public class DocumentVector
{
//Content which are present in the set will represents the document to be clustered
public string Content { get; set; }
//represents the tf*idf of each document
public float[] VectorSpace { get; set; }
}
```

and following code will focus on the document collection which is as follows:

```
class DocumentCollection
{
public List<String> DocumentList { get; set; }
}
```

Tf.idf scheme will work as follows:

```
private static float FindTFIDF(string document, string term)
{
float tf = FindTermFrequency(document, term);
float idf = FindInverseDocumentFrequency(term);
return tf * idf;
}
```

Then whenever we will compare the cosine similarity in between the 'n' documents we will have to implement the same as follows:

```
public static float FindCosineSimilarity(float[] vecA, float[] vecB)
{
var dotProduct = DotProduct(vecA, vecB);
var magnitudeOfA = Magnitude(vecA);
var magnitudeOfB = Magnitude(vecB);
float result = dotProduct / (magnitudeOfA * magnitudeOfB);
//when 0 is divided by 0 it shows result NaN so return 0 in such case.
if (float.IsNaN(result))
return 0;
else
return (float)result;
}
```

III. SURVEY ON DOCUMENT CLUSTERING

3.1 Document Clustering

Document clustering is automatic document collection or grouping, topic extraction, fat and effective information retrieval. It is closely related to data clustering.

Examples:

- Clustering will divides the results of a search for "cell" into groups like "biology," science),"battery," and "prison."

This advance will be very effective if we successfully figure the clusters based on some similarity as we will retrieve the 'n' relevant documents within less steps. Document clustering involves the use of descriptors and descriptor mining. Descriptors are sets of words that explain the contents within the cluster which contains the 'n' objects. The use of document clustering can be categorized to two types, online and offline. Online applications are typically controlled by effectiveness problems when compared offline applications.

3.2 Objective:

When the processing task is to be performed on the documents there is need to partition a given document collection into clusters of similar documents a choice of good features where what requires a good clustering algorithms to give better results.

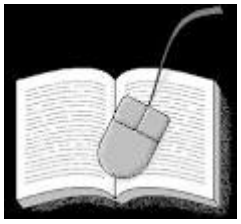


Fig 3.1 Text Processing

An ordinary job of text processing in many information retrieval applications is based on the analysis of word occurrences.

3.3 Existing Dirichlet Process Mixture Model (DPM)

This softness of the DPM form makes it particularly able for document clustering. There is little work investigating this model for document clustering due to the high-dimensional representation of text documents. In the difficulty of document clustering, each document is represented by a large amount of words including discriminative words and nondiscriminative words. Only discriminative words are useful for grouping documents. The participation of nondiscriminative words confuses the clustering procedure and leads to deprived clustering solution in return. When the number of clusters is unidentified, the affect of nondiscriminative words is motivated. Words in documents are partitioned into two groups, in particular, discriminative words and nondiscriminative words. Each document is considered as a mixture of two components. The first component, discriminative words are generated from the specific cluster to which document belongs. The second component, nondiscriminative words are generated from a universal background shared by all documents current in that collection. Scheme is to use only discriminative words to infer the document cluster structure. There are two algorithms to infer DPM model parameters, in particular first one is the variational inference algorithm and second one is the Gibbs sampling algorithm. It is hard to apply the Gibbs sampling algorithm to document clustering since it needs long time to converge.

IV. PROPOSED SYSTEM

We are developing a proposed software system by considering the following application areas such as follows:

We will focus on the following basic flow:

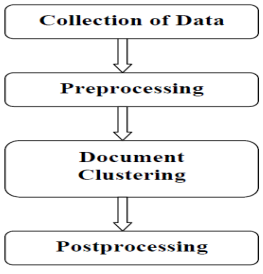


Fig 4.1 Flow of initial processing of system

1. In dataset we have collections of text files which is given as input to our system.
2. In preprocessing step, we perform two twchniques.
   - Stop-words Removal
     In stops words removal unwanted words like "and", "the","there",etc are removed.
   - Stemming
     In stemming words ending with some suffixes like "ing","ed" are processed.
3. In document clustering we apply gibbs sampling algorithm to our processed dataset.
4. Finally we generate clusters of documents.

4.1 Basic initial flow of the system:

As observe in the above flow chart first of all what we have to do is to prepare a list of words i.e. vocabulary or dictionary. Then selection of document one by one to check whether the term or word is occurred in that particular document or not. Based on this we will try to extract another features of the documents and will prepare a set of features.

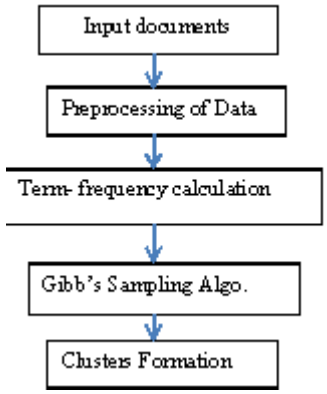


Fig 4.2 Flow of initial processing of system

4.2 Use of Blocked Gibbs Sampling Algorithm

Another effective inference algorithm for our proposed model is the blocked Gibbs sampling algorithm. **Gibbs sampling** or a **Gibbs sampler** is a Markov chain Monte Carlo algorithm for obtaining a sequence of observations which are approximated from a specified multivariate probability distribution as we are using it for identifying the relationship between 'n' documents and trying to form a group of documents based some similar features. For the DMAFP model, the state of the Markov chain is $W=(\gamma, P, n_0, n_1, \ldots n_N, z_1, z_2, \ldots z_D)$, After initializing the latent variable $\{r_1, r_2, r_3, \ldots r_W, z_1, z_2, \ldots z_D\}$ and hyperparameter Θ, the blocked Gibbs sampling procedure iterates between the following steps:

1. Update the latent discriminative words indicator r by repeating the Metropolis step R times: A new candidate r_{new} which adds or deletes a discriminative word is generated by randomly picking one of the W indices in r_{old} and changing its value.The new candidate is accepted with the minimum probability.

2. Conditioned on other latent variables, for i = 1,2,3,…,N if i is not in $\{z_1,z_2,z_3,\ldots,z_M\}$, draw n_i from a Dirichlet distribution with parameter λ.Otherwise,update n_i by sampling a value from a Dirichlet distribution with parameter.
3. Update n_0 by sampling a value from a Dirichlet distribution with parameter:
4. Update P by sampling a value from a Dirichlet distribution with parameter
5. Conditioned on other latent variables, for $d=1,2,\ldots,D$, update z_d by sampling a value fom a dirichlet distribution $\{s_{d1},s_{d2},\ldots s_{dN}\}$.

After the Markov chain has reached its stationary distribution, we collect H samples of $\{z1,\ldots., zD\}$ and $\{r_1,\ldots.r_w\}$.

4.3 Proposed Idea with example

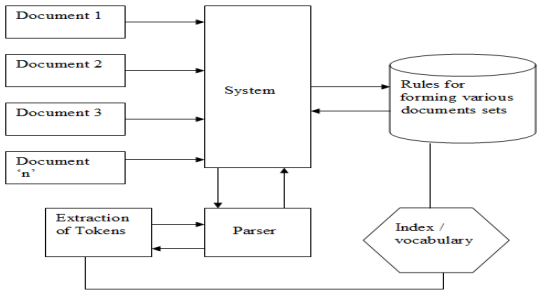


Fig 4.3 Proposed Architecture

Semi-supervised technique.

In proposed system we have applied new technique to generate the clusters which is semi-supervised clustering. Here The supervisor only needs to give a reasonable initialization for the cluster "centers" without the need to define a set of explicit categories. The algorithm is able to remove the noisy terms i.e stop-words stand to improve the separation among the documents (discriminative and non-discriminative) in the different clusters using the regularities available in the large unlabeled collection. In the experiments the algorithm showed very good performance than gibb's sampling theorem.

Here , we have added two more features to semi- supervised technique.

- *Search operation*

In this we can search any particular documents by giving a particular keyword as input file.

- *Time taken*

Here time taken by this technique to generate the clusters are shown in milliseconds of time. From this we can easily prove thar time taken by semi-supervised technique to generate the clusters is much less as compared to gibb's samling theorem.

V. MATHEMATICAL MODEL

1. U= {D, W, T, C}

Where,

$D=\{D_1,D_2,D_3,D_4,\ldots..,D_n/D_n \neq 0\ \}$

D is a set of documents.

Where,

$W=\{W_1,W_2,W_3,W_4,\ldots..,W_n/W_n \neq 0\ \}$

W is a set of words.

Where,

$T=\{T_1,T_2,T_3,T_4,\ldots..,T_n/T_n \neq 0\ \}$

T is a set of term frequency.

Where,

$C=\{C_1,C_2,C_3,C_4,\ldots..,C_n/C_n \neq 0\ \}$

C is set of clusters generated.

2. Let $f_W(D)\rightarrow W$

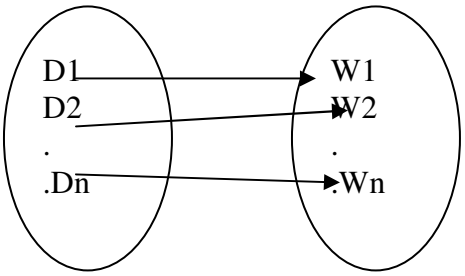
Where f_W is function that take documents and extract words from it.

Let $f_T(W)\rightarrow T$

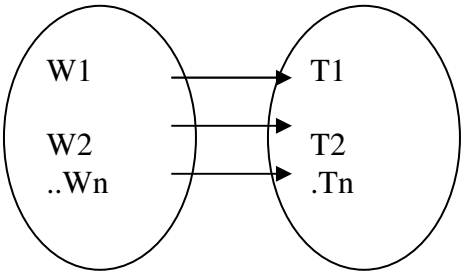
Where f_T is function that calculate the term frequency.

Let $f_C(T)\rightarrow C$

Where f_C is function that generate clusters using DP Model.



This above diagram will depict the association among 'many' to 'one' relationship.



This above diagram will depict the association among 'one' to 'one' documents.

VI. RESULT AND ANALYSIS.

Both data for estimating the model and new data will have the same format as follows:

$[document_1]$ ..$[document_2]$ … ... $[document_N]$

In which the first line is the total number for documents [N]. Each line after that is one document. $[document_i]$ is the i^{th} document of the dataset that consists of a list of M_i words/terms. $[document_i]$ = $[word_{i1}]$ $[word_{i2}]$ ... $[word_{iNi}]$ in which all $[word_{ij}]$ (i=1..N, j=1..m_i) are text strings and they are separated by the blank character.

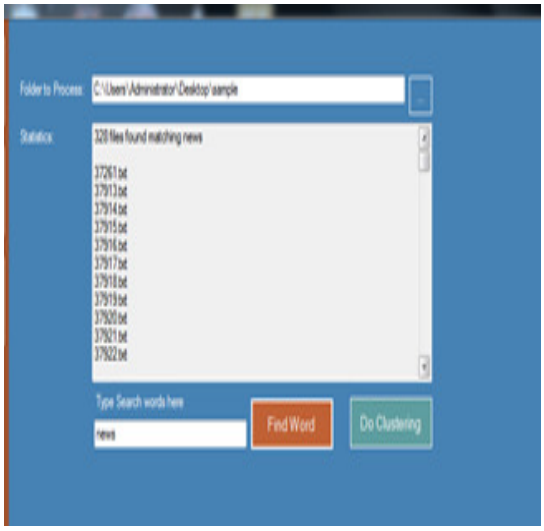


Fig : 6.1 Result of search operation.

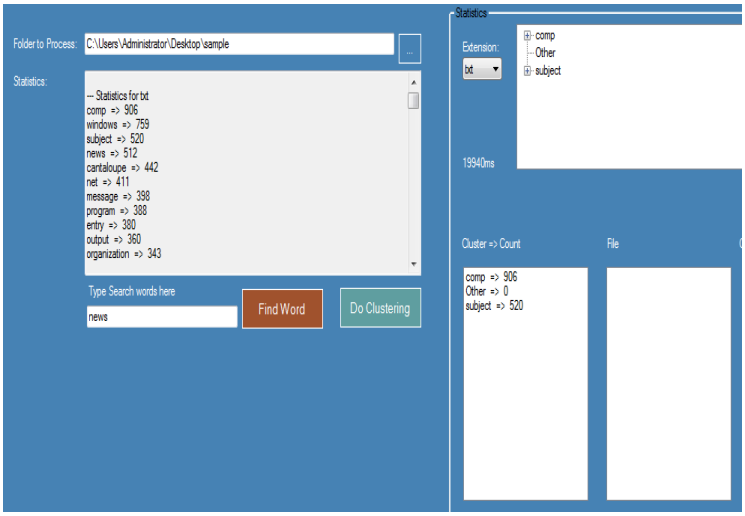


Fig : Cluster formation.

CONCLUSION

We have seen that following targets will definitely achieve as follows if we will form a set or clusters of given documents; So it will very useful to have clusters of data based on some similarity. In our proposed system we will use Dirichlet Process Mixture Model, mean variance algorithm and blocked gibbs sampling algorithm. Our proposed system with semi-supervised clustering technique tells us that time taken by semi-supervised technique to generate the clusters is much less as compared to DMAFP algorithm. Also here we have added two more features i.e we can apply searching operation to search a particular document by giving a keyword as input. And also we have shown time taken by different documents to generate the clusters in milliseconds. Hence we can conclude that semi-supervised technique is much faster to form clusters.

REFERENCES

[1] Michael Steinbach George Karypis Vipin Kumar,"A Comparison of Document Clustering Techniques" Department of Computer Science and Engineering, University of Minnesota.

[2] Inderjit Dhillon, Jacob Kogan,Charles Nicholas ,"Feature Selection and Document Clustering"

[3] C. Elkan, "Clustering Documents with an Exponential-Family Approximation of the Dirichlet Compound Multinomial Distribution," Proc. Int'l Conf. Machine Learning, pp. 289-296, 2006.

[4] R. Madsen, D. Kauchak, and C. Elkan, "Modeling Word Burstiness Using the Dirichlet Distribution," Proc. Int'l Conf. Machine Learning, pp. 545-552, 2005.

[5] Ruizhang Huang, Guan Yu, Zhaojun Wang, Jun Zhang, and Liangxing Shi," Dirichlet Process Mixture Model for Document Clustering with Feature Partition" IEEE Transactions on Knowledge and Data Engg, vol 25,no. 8,august 2013.

[6] K. Nigam, A.K. McCallum, S. Thrun, and T.M. Mitchel, "Text Classification from Labeled and Unlabeled Documents Using Em," J. Machine Learning, vol. 39, no. 2, 2000.

[7] C. Antoniak, "Mixtures of Dirichlet Processes with Applications to Bayesian Nonparametric Problems," The Annals of Statistics, vol. 2, no. 6, pp. 1152-1174, 1974.

[8] J. Ishwaran and L. James, "Gibbs Sampling Methods for Stick-Breaking Priors," J. Am. Statistical Assoc., vol. 96, no. 453, pp. 161-174, 2001.

[9] T. Ferguson, "A Bayesian Analysis of Some Nonparametric Problems," The Annals of Statistics, vol. 1, no. 2, pp. 209-230, 1973.

[10] M.H.C. Law, M.A.T. Figueiredo, and A.K. Jain, "Simultaneous Feature Selection and Clustering Using Mixture Models," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 26, no. 9, pp. 1154-1166, Sept. 2004.

[11] H. Bozdogan, "Determining the Number of Component Clusters in the Standard Multivariate Normal Mixture Model Using Model-Selection Criteria," Technical Report UIC/DQM/A83-1,Quantitative Methods Dept., Univ. of Illinois, Chicago, IL, 1983.

[12] G. Yu, R. Huang, and Z. Wang, "Document Clustering via Dirichlet Process Mixture Model with Feature Selection," Proc. ACM Int'l Conf. Knowledge Discovery and Data Mining, pp. 763-772, 2010.

AUTHOR'S PROFILE

Madhuri V. Malode

P.G. student: Department of Computer Engineering, Late G.N. Sapkal College of Engineering, Anjaneri,City: Nasik, Country: India.

University: Savitribai Phule Pune University.

Email id:engg.madhuri19@gmail.com.

1. Paper published in International Journal of Engineering Research and Technology (IJERT)ISSN: 2278- 0181 "Survey Paper on Clustering of Documents Based on Partitioning the Features".

2. Paper Presented in cPGCON 2014 "Practical Aspects of Clustering of documents based on partitioning the features".

Prof J.V.Shinde

Associate Professor: Department of Computer Engineering, Late G.N.Sapkal College of Engineering, Anjaneri, City: Nasik, Country: India. University: Savitribai Phule Pune University.

Email id: jv.shinde@rediffmail.com

She has presented papers at National and International conferences and also published paper in national and international journals on various aspects of the computer engineering.

J.V.Shinde, M.Y.Joshi "*Congestion Control for Reliable Multicast*" International Conference " ITECH 09 "International conference arranged by AVCOE, Sangmner

• J.V.Shinde, M.Y.Joshi "*Performance Analysis of NAK & ACK based Loss Recovery for congestion control in reliable multicast*" National conference on Recent Trends in Instrumentation & control "RTIC-2010" by PDVVP COE Ahmednagar.

• J.V.shinde "*GridCrypt : High Performance Private key cryptography*" a national level conference (ORION) by PREC, Pravaranagar