

Document Clustering using Linear Partitioning and Reallocation using EM Algorithm

GJCST Classifications:
H.3.3, I.4.6, I.1.2

¹ MS. P.J.Gayathri ² MRS. S.C. Punitha ³ Dr.M. Punithavalli

¹M.Phil scholar, P.S.G.R. Krishnammal College for Women, Coimbatore, India.

² HOD, Department of Computer science and Information Technology, P.S.G.R. Krishnammal College for Women, Coimbatore, India.

³ Director, Department of Computer science and Information Technology, Sri Ramakrishna college of Arts and Science for Women, Coimbatore, India.

¹gaya3jayaram79@gmail.com, ² saipunith@yahoo.co.in

Abstract- Document clustering is a subset of the larger field of data clustering, which borrows concepts from the fields of information retrieval (IR), natural language processing (NLP), and machine learning (ML), there exist a wide variety of unsupervised clustering algorithms. In this paper presents a novel algorithm for document clustering based with an enhancement on the features of the existing algorithms. This paper illustrates the Principal Direction Divisive Partitioning (PDDP) algorithm and describes its drawbacks and introduces a combinatorial framework of the PDDP algorithm and then describes the simplified version of the EM algorithm called the spherical Gaussian EM (sGEM) algorithm. The PDDP algorithm recursively splits the data samples into two sub-clusters using the hyper plane normal to the principal direction derived from the covariance matrix, which is the central logic of the algorithm. However, the PDDP algorithm can yield poor results, especially when clusters are not well separated from one another. To improve the quality of the clustering results problem, it is resolved by reallocating new cluster membership using the sGEM algorithm with different settings. Furthermore, based on the theoretical background of the sGEM algorithm, it can be obvious to extend the framework to cover the problem of estimating the number of clusters using the Bayesian Information Criterion. Experimental results are given to show the effectiveness of the proposed algorithm with comparison to the existing algorithm.

Keywords- Introduction, Document clustering via linear partitioning hyper planes, The proposed Spherical Gaussian EM algorithm, Results and Discussions conclusion and future work.

I INTRODUCTION

Clustering has been applied to various tasks in the field of Information Retrieval. The Document clustering has become one of the most active area of research and the development. One of the challenging problems is document clustering that attempts to discover the set of meaningful groups of documents where those within each group are more closely related to one another than documents assigned to different groups. The resultant document clusters can provide a structure for organizing large bodies of text for efficient browsing [15].

Document clustering referred to as Text clustering is closely related to concept of data clustering. It is a more specific

Technique for unsupervised document organization, automatic topic extraction and fast information retrieval or filtering. The process of clustering aims to discover natural groupings, and thus present an overview of the classes in a collection of documents. Clustering can either produce disjoint or overlapping partitions. In an overlapping partition, it is possible for a document to appear in multiple clusters. The first challenge in a clustering problem is to determine which features of a document are to be considered discriminatory. A majority of existing clustering approaches choose to represent each document as a vector, therefore reducing a document to a representation suitable for traditional data clustering approaches [18].

A wide variety of unsupervised clustering algorithms has been intensively studied in the document-clustering problem. Among the algorithms that remain the most common and effectual, the iterative optimization clustering algorithms have been demonstrated reasonable performance for document clustering, e.g. the Expectation Maximization (EM) algorithm and its variants, and the well-known K-means algorithm. The K-means algorithm can be considered as a special case of the EM algorithm, which has vast vicinity [3] by assuming that each cluster is modeled by a spherical Gaussian, each sample is assigned to a single cluster, and all mixing parameters are equal. The competitive advantage of the EM algorithm is that it is fast, scalable, and easy to implement. Hence, it has been chosen to enhance the algorithm, Expectation Maximization is proposed, Spherical Gaussian EM algorithm.

Principal Direction Divisive partitioning algorithm was developed by Boley [1], which is a hierarchal clustering algorithm that performs by recursively splitting the data samples into two sub clusters. It applies the concept of the Principal Component Analysis for the requirement of the principal eigenvector, which is not computationally expensive. It can also generate a hierarchal binary tree that inherently produces a simple taxonomic ontology. The clustering results produced by the PDDP algorithm compare favorably to other document clustering approaches, such as the agglomerative hierarchal algorithm and associative rule hyper graph clustering. In some cases, the clusters are not well separated from one another, it can yield poor results.

The proposed methodology overcomes the disadvantages of the PDDP algorithm that uses the PCA for analyzing the data and combines it with the EM algorithm as the proposed work. In PDDP splits the data samples into two sub clusters based on the hyper plane normal to the principal direction derived from the covariance matrix of the data. When the principal direction is not representative, the corresponding hyper plane tends to produce individual clusters with wrongly partitioned contents. One practical way to deal with this problem is to run the EM algorithm on the partitioning results. A simplified version of the EM algorithm called the spherical Gaussian EM algorithm is presented for performing such task. Furthermore, based on the theoretical background of the spherical Gaussian EM algorithm, naturally extending this framework to cover the problem of estimating the number of clusters using the Bayesian Information Criterion [9].

The paper is organized as follows. Section 2 briefly reviews some important backgrounds of the PDDP algorithm, and addresses the problem causing the incorrect partitioning. Section 3 presents the proposed algorithm, spherical Gaussian EM algorithm. Section 4 discusses the idea of applying the BIC to our algorithm. Section 5 explains the Artificial Intelligence in EM algorithm. Section 6 explains the data sets and the evaluation method, and shows experimental results. Finally, this paper concludes in Section 7 with some directions of future work.

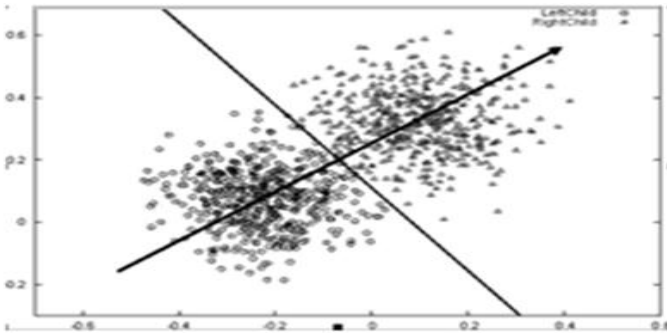


Figure1 The Principal direction and the linear partitioning Hyper plane on the 2d2k dataset.

II DOCUMENT CLUSTERING VIA LINEAR PARTITIONING HYPER PLANES

Considering a one-dimensional data set, e.g. real numbers on a line, the question is how to split this data set into two groups. One simple solution may be the following procedures. The mean value of the data set is first found and then it is compared to each point with the mean value. If the point value is less the mean value, it is assigned to the first group. Otherwise, it is assigned to the second group. The problem arises when it has a dimensional data set. Based on the idea of the PDDP algorithm, this problem can be dealt by projecting all the data points onto the principal direction the principal eigenvector of the covariance matrix of the data set, and then the splitting process can be performed based on this principal direction. In geometric terms, the data points are partitioned into two sub clusters using the hyper plane normal to the principal direction passing through the mean

vector [1]. This hyper plane is referred as the linear partitioning hyper plane. Figure 1 illustrates the principal direction and the linear partitioning hyper plane on the 2d2k data set, containing 1000 points distributed in 2 Gaussians.

The PDDP algorithm begins with all the document vectors in a large single cluster. This procedure continues by recursively splitting the cluster into two sub clusters using the linear partitioning hyperactive plane according to the discriminant functions of the algorithm. This procedure terminates by splitting based on some heuristic, e.g. a pre defined number of clusters. Finally, a binary tree is yielded out as the output, whose leaf nodes form the resulting clusters. To keep this binary tree balanced, it selects an unsplit cluster to split by using the scatter value, measuring the average distance from the data points in the cluster to their centroid.

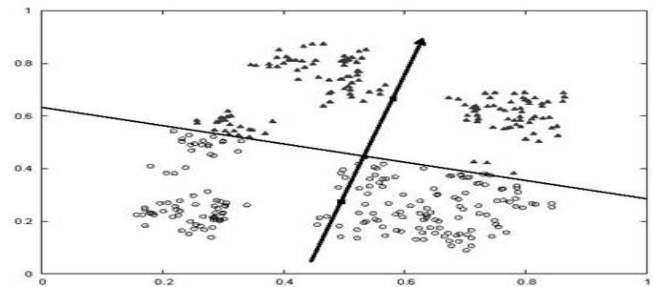


Figure2 Two partitions after the first iteration.

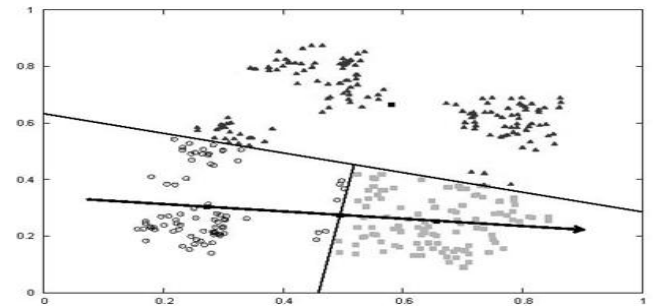


Figure 3 Three partitions after the second iteration

The severe problem of the PDDP algorithm is that it cannot achieve good results when clusters are not well separated from one another. This figure 2 and 3 illustrates this drawback. Figure 2 shows two partitions produced by performing the first iteration of the PDDP algorithm on a dimensional data set. The data set consists of 334 points. The actual class labels are not given, but one can observe that it is composed of five compact clusters [8]. Based on the principal direction and the corresponding linear partitioning hyper plane, it can be seen that the PDDP algorithm starts with significantly wrong partitioning on the middle left hand cluster. Figure 3 shows three partitions after the second iteration. If the partitioning is further performed without making some adjustments, the resulting clusters become worse. This indicates that the basic PDDP algorithm can produce poor solutions in some distributions of the data, which cannot be known in advance. In addition, it may require some information to suggest whether to split the particular cluster or whether to not split on further.

III THE PROPOSED SPHERICAL GAUSSIAN EM ALGORITHM

It is possible to refine the partitioning results by reallocating new cluster membership. The basic idea of the reallocation method [12] is to start from some initial partitioning of the data set, and then proceed by moving objects from one cluster to another cluster to obtain an improved partitioning. Thus, any iterative optimization-clustering algorithm can be applied to do such operation. The problem is formulated as a finite mixture model, and applies a variant of the EM algorithm for learning the model.

The most critical problem is how to estimate the model parameters. The data samples are assumed to be drawn from the multivariate normal density in R^d also assume that features are statistically independent, and a component c_j generates its members from the spherical Gaussian with the same covariance matrix [5]. Figure 4 gives an outline of a simplified version of the EM algorithm. The algorithm tries to maximize $\log L_c$ at very step, and iterates until convergence. For example, the algorithm terminates when $\Delta \log L_c < \delta$, where δ is a pre defined threshold.

begin

Initialization: Set $(z_i)_j^{(0)}$ from a partitioning of the data, and $t \leftarrow 0$.

repeat

E-step: For each $d_i, 1 \leq i \leq n$ and $c_j, 1 \leq j \leq k$, find its new component index as:

$$(z_i)_j^{(t+1)} = \begin{cases} 1, & \text{if } j^* = \operatorname{argmax}_j \log(P^{(t)}(c_j | d_i; \theta_j)) \\ 0, & \text{otherwise.} \end{cases}$$

M-step: Re-estimate the model parameters:

$$P(c_j)^{(t+1)} = \frac{1}{n} \sum_{i=1}^n (z_i)_j^{(t+1)}$$

$$m_j^{(t+1)} = \frac{\sum_{i=1}^n d_i (z_i)_j^{(t+1)}}{\sum_{i=1}^n (z_i)_j^{(t+1)}}$$

$$\sigma^{2(t+1)} = \frac{1}{n \cdot d} \sum_{i=1}^n \sum_{j=1}^k \|d_i - m_j\|^2 (z_i)_j^{(t+1)}.$$

until $\Delta \log L_c(\Theta) < \delta$;

end

Figure 4 A brief SGEM Algorithm.

A. Estimating Number Of Document Clusters

The clustering algorithm is applied to a new data set having little knowledge about its contents, fixing a predefined number of clusters is too strict and inefficient to discover the latent cluster structures. The finite mixture model of EM algorithm covers the problem of estimating the number of clusters in the data set. A model selection technique is applied called the Bayesian Information Criterion (BIC) [9]. Generally, the problem of model selection is to choose the best one among a set of candidate models.

The BIC contains two components, where the first term measures how well the parameterized model predicts the data, and the second term penalizes the complexity of the model [4]. Thus, the model selected has the largest value of the BIC,

$$M^* = \operatorname{argmax}_i \text{BIC}(M_i).$$

As a result, the value is directly obtained of the first term of the BIC from running the sGEM algorithm. However, it can also be compute it from the data according to the partitioning. The number of parameters is the sum of $k - 1$ component probabilities, $k \cdot d$ centroid coordinates, and 1 variance.

Boley's subsequent work [2] also suggests a dynamic threshold called the centroid scatter value (CSV) for estimating the number of clusters. This criterion is based on the distribution of the data. Since the PDDP algorithm is a kind of the divisive hierarchical clustering algorithm, it gradually produces a new cluster by splitting the existing clusters. As the PDDP algorithm proceeds, the clusters get smaller. Thus, the maximum scatter value in any individual cluster also gets smaller. The idea of the CSV is to compute the overall scatter value of the data by treating the collection of centroids as individual data vectors. This stopping test terminates the algorithm when the CSV exceeds the maximum cluster scatter value at any particular point.

The CSV is a value that captures the overall improvement, whereas the BIC can be used to measure the improvement in both the local and global structure. As mentioned earlier, in the splitting process, some information is needed to make the decision whether to split a cluster into two sub clusters or keep its current structure. The BIC is first calculated locally when the algorithm performs the splitting test in the cluster. The BIC is calculated globally to measure the overall structure improvement. If both the local and global BIC scores improve, it is then split the cluster into two children clusters.

IV RESULTS AND DISCUSSIONS

• Data Sets And Setup Information

The 20 Newsgroups data set consists of 20000 articles evenly divided among 20 different discussion groups [10]. This data set is collected from UseNet postings over a period of several months. Many categories fall into confusable clusters. For example, five of them are computer discussion groups, and three of them discuss religion. The Bow toolkit [11] is used to construct the term document matrix (sparse format). The UseNet headers are used, and also eliminated the stop words and low frequency words (occurring less than 2 times). Finally 59965×19950 term document matrix is obtained for this data set.

The well-known tf-idf term weighting technique is also applied. Let $d_i = (w_{i1}, w_{i2}, \dots, w_{im})^T$, where m is the total number of the unique terms. The tf-idf score of each w_{ik} can be computed by the following formula:

$$w_{ik} = \text{tf}_{ik} \cdot \log(n / d_{rk})$$

Where tf_{ik} is the term frequency of w_{ik} in d_i , n is the total number of documents in the corpus, and d_{rk} is the number of documents that w_{ik} occurs. Finally, each document vector is normalized using the L_2 norm. For the purpose of comparison, the basic PDDP algorithm is chosen as the baseline. The number of clusters k is varied in the range [2,

2k], and no stopping criterion was used. Then we applied both the CSV and the BIC to the above settings in order to test the estimation of the number of clusters.

• *Evaluation Method*

Since all the documents are already categorized, comparing clustering results with the true class labels can perform evaluation. In our experiments, the normalized mutual information (NMI) is been used [16]. In the context of document clustering, mutual information can be used as a symmetric measure for quantifying the degree of relatedness between the generated clusters and the actual categories. Particularly, when the number of clusters differs from the actual number of categories, mutual information is very useful without a bias towards smaller clusters, by

Data set	Criterion	Algorithm	k found	NMI	Time (sec.)
20 Newsgroups	CSV	PDDP	34	0.443	15.838
		sGEM	34	0.482	105.39
	BIC	PDDP	25	0.426	14.70
		sGEM	25	0.463	78.45

Table 1: Clustering results by varying stopping criteria on 20 Newsgroups data Sets.

Normalizing this criterion to take values between 0 and 1, the NMI can be calculated as follows

Where n_h is the number of documents in the category h , n_l is the number of documents in the cluster l , and $n_{h,t}$ is the

$$NMI = \frac{\sum_{h,l} n_{h,l} \log(n \cdot n_{h,l} / n_h n_l)}{\sqrt{(\sum_h n_h \log(n_h/n))(\sum_l n_l \log(n_l/n))}}$$

Cj	Purity	Entropy	H	Ep	Ec	Em	Ei	Ef	Emu	Et	Ev	Ea	Er	Eo	Emm	Ecu	Es	E	S	P	T	B
9	1.000	0.000	25
10	1.000	0.000	.	30
2	0.998	0.005	488	1
1	0.978	0.036	3	132	.	.	.
3	0.900	0.137	1	1	54	.	4
5	0.878	0.166	5	2	5	86
7	0.865	0.184	.	4	.	.	45	.	.	1	1	1
8	0.719	0.363	.	82	.	4	3	.	12	5	.	1	.	1	.	5	.	1
11	0.718	0.308	.	1	1	28	1	.	1	7
6	0.680	0.351	.	3	6	.	8	1	.	85	21	1
4	0.425	0.372	1	1	48	44	19
0	0.216	0.837	4	128	37	17	54	229	112	67	31	21	157	9	14	44	18	8	9	58	11	32

Table 2 Confusion matrix generated by using sGEM and the BIC

number of documents in the category h as well as in the cluster l . The NMI value is 1 when clustering results exactly match the true class labels, and close to 0 for a random partitioning [17].

• *Experimental Results*

Figure 5 shows the clustering results on the 20 Newsgroups data set. In this data set, it can be seen that the proposed algorithm perform relatively better than the basic PDDP algorithm. However, performing the global refinement after the local refinement as in EM degrades the quality of the clustering results. The global refinement with the sGEM algorithm leads to more decisions to move each document from its cluster to other candidate clusters.

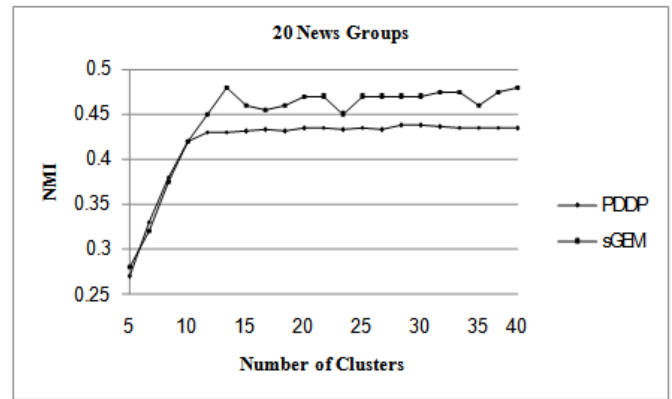


Figure 5: NMI results on the 20 Newsgroups data set.

Cj	Purity	Entropy	H	Ep	Ec	Em	Ei	Ef	Emu	Et	Ev	Ea	Er	Eo	Emm	Ecu	Es	E	S	P	T	B	
4	1.000	0.000	122
7	0.995	0.010	212	1
8	0.994	0.013	155
3	0.992	0.015	1	132
10	0.898	0.139	1	1	53	.	.	4
1	0.564	0.458	.	79	.	3	.	2	8	10	.	1	30	5	1
0	0.517	0.281	.	30	1	26	1
5	0.517	0.587	1	23	3	1	1	.	104	7	2	4	25	2	4	12	2	3	5	1	.	.	.
9	0.507	0.377	3	43	2	53	104
12	0.485	0.383	.	8	1	.	1	79	.	1	2	.	66	.	.	3	1	1
2	0.480	0.312	.	4	.	.	.	45	.	47	1	1
11	0.474	0.536	.	7	14	.	9	100	3	36	35	1	.	1	1	2	2
6	0.387	0.492	.	36	1	.	3	47	.	8	.	.	65	.	.	2	6
14	0.309	0.695	.	4	21	7	42	2	4	25	13	.	.	1	8	2	.	2	.	.	.	2	3
13	0.209	0.796	3	57	3	9	11	3	6	25	.	17	2	17	1	22	2	2	4	58	5	30	

Table 3 Confusion matrix generated by using sGEM and the CSV

V CONCLUSION AND FUTURE WORK

This paper presents several strategies for improving the basic PDDP algorithm. When the principal direction is not representative, the corresponding hyper plane tends to produce individual clusters with wrongly partitioned contents. By formulating the problem with the finite mixture model. This paper describes the sGEM algorithm has tremendous improvement when compared to the PDDP algorithm in several ways for refining the partitioning results. Preliminarily experimental results on two different document sets are very encouraging.

In future work, intends to investigate other model selection techniques for approximating the number of underlying clusters. Recently, work by [7] has demonstrated that estimating the number of clusters in the kmeans algorithm using the Anderson Darling test yields very promising results, and seems to outperform the BIC. The statistical measure can also be applied for this algorithm in further enhancement.

VI REFERENCES

- 1) Boley, D. Principal direction divisive partitioning. *Data Mining and Knowledge Discovery*, 2(4):325–344, 1998.
- 2) Boley, D., and Borst, V. Unsupervised clustering: A fast scalable method for large datasets. CSE Report TR99029, University of Minnesota, 1999.
- 3) Bradley, P. S., and Fayyad, U. M. Refining initial points for kmeans clustering. In *Proceedings of the Fifteenth International Conference on Machine Learning*, pages 91–99, 1998.
- 4) Chickering, D., Heckerman, D., and Meek, C. A bayesian approach to learning bayesian networks with local structure. In *Proceedings of the*

thirteenth Conference on Uncertainty in Artificial Intelligence, pages 80–89. Morgan Kaufmann, 1997.

- 5) Dasgupta, S., and Schulman, L. J. A tworound variant of em for gaussian mixtures. *Sixteenth Conference on Uncertainty in Artificial Intelligence (UAI)*, 2000.
- 6) Golub, G., and Loan, C. V. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, 1989.
- 7) Hamerly, G., and Elkan, C. Learning the k in k - means. In *Proceedings of the seventeenth annual conference on neural information processing systems (NIPS)*, December 2003.
- 8) He, J., Tan, A.H., Tan, C.L., and Sung, S.Y. On Quantitative Evaluation of Clustering Systems. In W.Wu and H. Xiong, editors, *Information Retrieval and Clustering*. Kluwer Academic Publishers, 2003.
- 9) Kass, R. E., and Raftery, A. E. Bayes factors. *Journal of the American Statistical Association*, 90:773–795, 1995.
- 10) Lang, K. Newsweeder: Learning to filter netnews. In *Proceedings of the Twelfth International Conference on Machine Learning*, pages 331–339, 1995.
- 11) McCallum, A. K. Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering. <http://www.cs.cmu.edu/~mccallum/bow>.
- 12) Rasmussen, E. Clustering algorithms. In W. Frakes and R. BaezaYates, editors, *Information retrieval: data structures and algorithms*. Prentice Hall, 1992.
- 13) Salton, G., and Buckley, C. Termweighting approaches in automatic text retrieval. *Information Processing and Management: an International Journal*, 24(5):513–523, 1988.

- 14) Steinbach, M., Karypis, G., and Kumar, V.A comparison of document clustering techniques. KDD Workshop on Text Mining, 1999.
- 15) Strehl, A., Ghosh, J., and Mooney, R. J. Impact of similarity measures on webpage clustering. In Proceedings of AAAI Workshop on AI for Web Search, pages 58–64, 2000.
- 16) Strehl, A., and Ghosh, J. Cluster ensembles a knowledge reuse framework for combining multiple partitions. Journal on Machine Learning Research, 3:583–617, 2002.
- 17) Zhong, S., and Ghosh, J.A comparative study of generative models for document clustering.SDM Workshop on Clustering High Dimensional Data and Its Applications, 2003.
- 18) Nicholas.O.Andrews and Edward.A. Fox. Recent Developments in Document Clustering, Virginia Tech, Blackburg, VA 24060, 2007.