Indian Journal of Science and Technology, *Vol 9(40), DOI: 10.17485/ijst/2016/v9i40/99005, October 2016*

ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Document Clustering using a New Similarity Measure based on Energy of a Bipartite Graph

G. Hannah Grace* and Kalyani Desikan

Department of Mathematics, School of Advanced Sciences, VIT University, Chennai - 600127, Tamil Nadu, India;
hannahgrace.g@vit.ac.in; kalyanidesikan@vit.ac.in

Abstract

Objectives: This paper aims at clustering documents using a new similarity measure based on energy of a bipartite graph. **Methods/Statistical Analysis**: We have made use of bipartite representation of documents and clustered them. The proposed algorithm has been illustrated for a small document set. The documents have been clustered using the new similarity measure based on energy of a bipartite graph introduced by us. **Findings**: Our proposed algorithm gives a better clustering quality comparing with the k means clustering algorithm. **Application/Improvements:** This proposed algorithm can be further extended and applied to cluster large document sets.

Keywords: Bipartite Graph, Cluster Quality, Document Clustering, Energy, Similarity Measure

1. Introduction

Document clustering is one of the text mining techniques which are employed to divide a document corpus into significant clusters by minimizing the intra-cluster distance between documents and maximizing the distance between clusters. This is achieved by adopting a suitable distance or similarity measure.

Since clustering algorithms[1] cannot interpret the documents directly, an indexing procedure that maps a text into a compact representation is applied. Selecting an appropriate representation for text is dependent on the features extracted from the document. The vector space model is the commonly used representation model for text documents. Here each document is represented as a vector of weights for 'm' terms (features) taken from the document, given by $d_j=(w_{1j},w_{2j},.....w_{mj})$, where d_j represents the jth document, m the number of terms and w_{ij} is the weight of the ith term in the jth document. When the features are identified with the words in the text documents, we call it the "bag of words" document representation. If there are n documents in the document corpus, combining all these n document vectors, we can get a n x m matrix usually called as the term document matrix $tf_{i,j}$ which gives the frequency count of the terms in the document. The terms occurring frequently within a document reflect the key concepts presented in the document more strongly than terms occurring less frequently and hence, have higher weights. But, words that occur more frequently across documents may not have significant value in the corpus. To tackle this situation we use term frequency inverse document frequency (tf-idf)[2] weighting scheme. Inverse Document Frequency (IDF) is given by $idf_i=\log(n/n_i)$ where n is the total number of documents in the document set and n_j is the number of documents in which the term j exists. Under the tf-idf weighting scheme, the elements of the term document matrix would be given by $w_{i,j}= tf_{i,j} \times idf_i$. This increases the weight of terms that frequently occur in a smaller set of documents and decreases the weight of terms that frequently occur across the entire corpus.

A graph based model is one of the ways to represent the document corpus. A graph is a set of vertices and edges denoted by G={V,E}, where V is the set of vertices and E is the set of edges. Graph representation takes the structure of the graph into consideration and is useful in

*Author for correspondence

cluster analysis to group vertices of the graph into clusters.

In this paper we present a new graph based document clustering technique motivated by the recent advances in the area of graph based document clustering. In this work we have considered a document corpus, and collected the unique terms in the corpus after pre-processing. We have given a graph representation of the Documents and unique terms/words. Depending on the most frequently occurring terms, we have reduced the sparsity of the term document matrix and obtained a reduced term document matrix. For this reduced term document matrix, we have given a bipartite graph representation based on the reduced set of words. We have then computed the energy matrix for the bipartite graph. We have introduced a novel method to find the similarity between documents using energy of a bipartite graph.

This paper is organized as follows. In the second section we have given the preliminaries related to energy of a graph and energy between pairs of documents; a new similarity measure is also introduced in this section. The next section contains an experimental analysis of a set of six documents represented in terms of a reduced term document matrix along with its bipartite representation. Using the reduced term document matrix, we calculate the energy matrix and the new similarity matrix which we have proposed. We then perform the clustering of the documents considering the similarity matrix in its normalised form. We finally present the discussions and conclusion of our clustering result.

2. Preliminaries

2.1 Graph

A Graph G is denoted as G (V, E) where V is the set of n nodes, and E is the set of m edges between them.

2.2 Directed Graph and Undirected Graph

For a directed graph G(V,E), each edge (i, j) ϵ E represents a directed edge that starts at node i and terminates at node j. If edges point in both directions i.e., $(i, j) \in E \Rightarrow (j,i) \in E$ then we get an undirected Graph[3].

2.3 Bipartite Graph

A Graph G(V,E) is said to be bipartite if its vertex set V can be partitioned into two disjoint subsets X and Y where $V = X \cup Y$ with $X \cap Y = \phi \ni$ every edge e ϵ E joins some vertex in X to some vertex in Y[5].

2.4 Weighted Graph

A weighted Graph is denoted as G (V, E, W) where V is the vertex set, E the edge set and $w \in W, w > 0$ represents the weight assigned to the edges[3].

2.5 Adjacency Matrix of a Bipartite Graph

The adjacency matrix A of a weighted bipartite graph G(V,E,W) with $V = X \cup Y$ is an N_1 x N_2 matrix , such that[3]

$$A_{ij} = \begin{cases} w_{ij}, & if\ (i,j) \in E, \qquad i = 1,2, \dots N_1 1, j = 1,2, \dots N_2 \\ 0, & otherwise \end{cases}$$

Here N_1 represents the number of rows and N_2 represents the number of columns of the adjacency matrix. In the context of document clustering, X represents the set of terms in the document set, Y represents the document set and ω_{ij} gives the weight assigned to the edge (i,j). W=| ω_{ij} | is a matrix that represents the frequency of term i in document j.

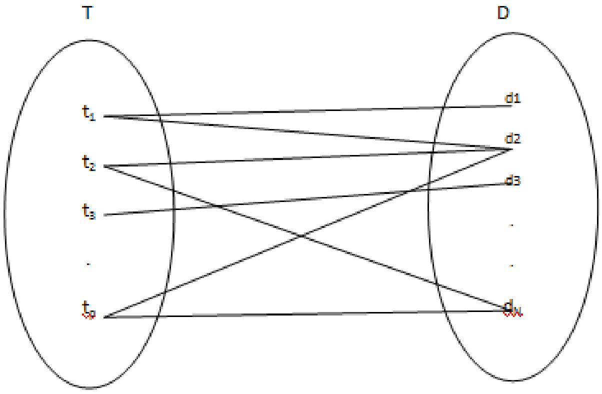

Figure 1. Bipartite representation of documents in G.

The adjacency matrix of a bipartite weighted graph G (V,E,W) is represented in a block matrix format as

$\begin{pmatrix} 0 & W \\ W^T & 0 \end{pmatrix}$ where W^T is the transpose of matrix W.

2.6 Energy of a Graph

If $A = [a_{ij}]$ is the adjacency matrix of a graph G with n vertices and m edges and $\lambda_1, \lambda_2, ...\lambda_n$ are the eigen values of A, then the energy of the graph G is defined to be the sum of the absolute values of its eigen values[4] i.e.,

$E(G) = \sum_{i=1}^{n} |\lambda_i|$. The set $\{\lambda_1, \lambda_2, ...\lambda_n\}$ is known as the spectrum of G[6,7].

2.7 Energy of a Bipartite Graph

Consider the bipartite representation[3,5,8] of the document corpus as a graph G where $t_1,t_2,t_3,...t_p$ є T is the term set, p being the number of terms and $d_1,d_2,d_3,..d_N$ є D is the document set, N being the number of documents.

It[9] has been proved that for a bipartite graph G with n vertices and m edges the energy of G is given by the inequality

$$E(G) \le \frac{4m}{n} + \sqrt{(n-2)(2m - \frac{8m^2}{n^2})} \qquad (1)$$

Since we are working with document clustering, we calculate the energy between pairs of documents considering only the upper bound of the above equation.

2.8 Energy between Pairs of Documents

Consider the sub graph of the bipartite graph, which is again a bipartite graph, which contains p terms and we consider pairs of documents from the document corpus to obtain the energy matrix of this sub graph (bipartite graph). Let $t_1,t_2,t_3,...t_p$ єT be the p number of terms. Considering two documents d_i,d_j for all i=1,2,3,...N and j= 1,2,3,...,N the bipartite subgraph representation is as given in Figure 2.

We have modified the formula[9] to find the distance (energy) between these two documents and it is as given below

$$E(d_i, d_j) = \frac{4m'}{n'} + \sqrt{(n'-2)(2m' - \frac{8m'^2}{n'^2})} \qquad (2)$$

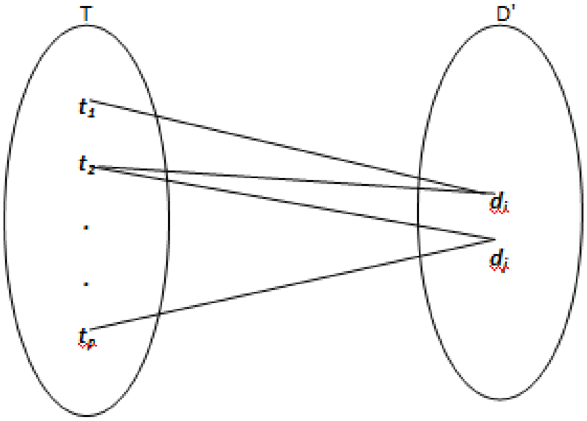

Figure 2. Bipartite representation of subgraph of G.

Since we consider only two documents at a time, we have n'=p+2 where p is the number of terms and 2 is the number of documents taken at a time. Also m' is the number of edges (multiple) incident from d_i, d_j to vertices in T.

Using n' = p+2 in (2) we get

$$E(d_i, d_j) = \frac{1}{p+2}(4m' + \sqrt{2pm'(p+2)^2 - 8pm'^2}) \qquad (3)$$

where p =|T| is the number of terms in the document corpus , m' is the number of edges incident from documents d_i ,d_j to vertices in T. We have calculated the energy between documents i and j using the formula (3).

3. New Similarity Measure

Distance measures play a vital role in clustering data points. Choosing the correct distance measure for a given dataset is important. The similarity between various objects is defined by a distance measure. The various distance/similarity measures are Euclidean distance, squared Euclidean distance measure, Minkowski distance measure, Chebychev distance, power distance, Manhattan distance measure, Bit-vector distance measure, comparative-clustering distance measure, Huffman-code distance measure and Dominance-based distance measure.

Based on the laws of physics for energy and coulombs law, a new similarity measure introduced by us is given by

$$S(d_i, d_j) = \frac{E(d_i, d_j) r^2}{q_i q_j} \qquad [4]$$

where $S(d_i, d_j)$ is used to find the similarity between pairs of documents. Here $E(d_i, d_j)$ is the energy between pairs of documents, q_i is the number of terms in document i, q_j is the number of terms in document j, r is the number of words common to both the documents d_i, d_j.

4. Experimental Analysis

For illustration purpose we have restricted our analysis to a document set comprising of 6 documents: D1, D2, D3, D4, D5, D6 given below.

4.1 Document Corpus

D1: CLUTO is software for clustering low and high dimensional datasets.

D2: CLUTO is a tool for analyzing the discovered clusters.

D3: Given a set of documents, the clusters formed have a high degree of association between them.

D4: Document clustering minimizes intra-cluster distances between documents.

D5: Internal quality measure and external quality measure are used to evaluate document clustering.

D6: Similarity and distance measures are used to evaluate distance between documents.

These 6 documents contain a total of 67 terms and the maximal term length is 12.

4.2 Text Pre-Processing

Text pre-processing is used to convert the original document set into a structured format that can be readily clustered. Test pre-processing identifies the significant text features that differentiate between text categories. The key purpose of pre-processing is to identify the prime features (terms) in the document set that enhance the relevancy between terms and documents. The primary goal of pre-processing is to divide the text into individual words[10]. Text pre-processing involves tokenization wherein string sequences are broken up into what are known as tokens. Tokens comprise words, keywords, phrases, symbols and other elements. Also, some characters like punctuation marks and white spaces are eliminated[11]. Stop words elimination which involves the removal of prepositions, articles and pronouns that are not important for text mining from text documents, improves the system performance[11]. Stemming is used to remove various suffixes and the words are reduced to exactly matching stems. This saves memory space and time[10,12].

After pre-processing our 6 document set[2,8], we notice that the total number of words in our document set reduces to 43 terms. Also, we see that[8] out of the 43 terms only 26 words (terms) are unique. Hence, it is evident that pre-processing reduces the number of terms that are to be considered for further processing.

4.3 Reduced Term Document Matrix and Bipartite Graph

After applying the pre-processing steps and identifying the unique words, we can represent the document/term collection as an undirected graph.

After pre-processing, our 6 document set is represented as an undirected graph comprising of 32 nodes (26 nodes for the unique words and 6 nodes that represent the documents). The corresponding adjacency matrix A is as follows.

$$A = \begin{pmatrix} 0_{26x26} & W_{26x6} \\ W^T_{6x26} & 0_{6x6} \end{pmatrix}$$

where

$$W^T = \begin{pmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 2 & 0 & 2 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

W is a 26 X 6 matrix whose elements are the frequencies of the most frequent terms (26) in the 6 document set. From the adjacency matrix A, it is evident that the graph is a bipartite graph. We also see that this matrix is highly sparse[8]. For further analysis, we have reduced the sparsity of the matrix W by taking into consideration only those rows whose row sum is greater than 1, that is, only those terms whose frequency is more than 1 across all the documents is considered. We now get the following reduced term document matrix shown in Table 1.

This reduced matrix contains only 8 frequently occurring terms. The bipartite representation of the document corpus and the 8 frequently occurring terms is given in Figure 3. The weights attached to the edges represent the frequency of a particular word in the corresponding document.

We now consider the reduced term document matrix and using Table 1 we construct the energy matrix as in Table 2.

Table 1. Reduced term document matrix

Term/Doc	D1	D2	D3	D4	D5	D6
cluster	1	1	1	1	1	0
Cluto	1	1	0	0	0	0
distanc	0	0	0	1	0	2
document	0	0	1	2	1	1
Evalu	0	0	0	0	1	1
High	1	0	1	0	0	0
measur	0	0	0	0	2	1
Use	0	0	0	0	1	1

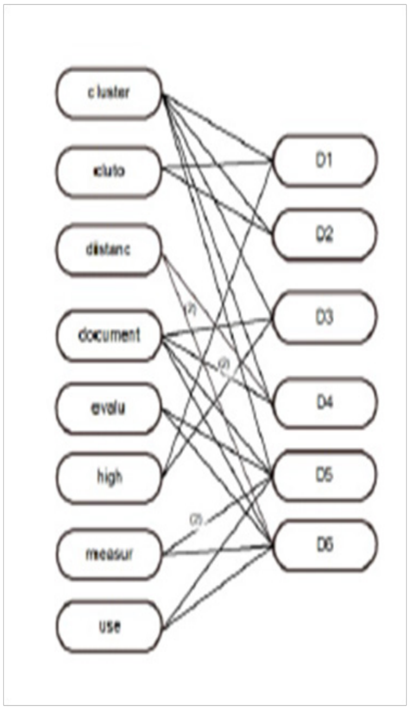

Figure 3. Bipartite graph for reduced set of terms.

4.4 Energy Matrix

Using the formula for energy between documents given in Equation (3) we get the following energy matrix E for the document corpus.

Table 2. Energy matrix

Energy	D1	D2	D3	D4	D5	D6
D1	7.699231	10	10.94166	11.77998	13.2	13.2
D2	10	6.225864	10	10.94166	12.52952	12.52952
D3	10.94166	10	7.699231	11.77998	13.2	13.2
D4	11.77998	10.94166	11.77998	8.932121	13.79796	13.79796
D5	13.2	12.52952	13.2	13.79796	10.94166	14.792
D6	13.2	12.52952	13.2	13.79796	14.792	10.94166

4.5 Similarity Matrix

Using the energy matrix, we form The normalized similarity matrix is given in Table 4.. This gives the similarity between the documents.

For our analysis, we have normalised the Similarity matrix by dividing each element of the similarity matrix by the largest element, i.e., 7.699. After normalisation, all the values lie between 0 and 1. Given below is the normalised similarity matrix:

We now describe the algorithm for clustering the documents making use of the normalised similarity matrix.

5. Clustering Solution

5.1 Proposed Algorithm for Clustering

We now present the algorithm for clustering the documents.

- From the normalised similarity matrix, identify the document(s) with the highest values along the diagonal as the clustering seeds. The number of seeds identified depends on the number of clusters to be formed. If the number of clusters is two then select two documents with the highest normalised similarity values as seeds s_1,s_2.
- Consider the normalised similarity values between each of the documents and the chosen seeds s_1, s_2.
- Assign the documents to the cluster to which the similarity of the document with the seed is higher.
- Repeat steps 1–3 till all the documents are clustered.

To illustrate the working of our technique, we cluster the documents in our document corpus into two clusters. To form the clusters, we first identify the seeds of the two clusters. From the diagonal of the normalised similarity

Table 3. Similarity matrix

Similarity values	D1	D2	D3	D4	D5	D6
D1	7.699	6.667	4.863	0.982	0.733	0
D2	6.667	6.225	1.667	1.368	1.044	0
D3	4.863	1.667	7.699	3.927	2.933	0.733
D4	0.982	1.368	3.927	5.024	2.299	2.299
D5	0.733	1.044	2.933	2.299	7.598	6.574
D6	0	0	0.733	2.299	6.574	7.598

Table 4. Normalised similarity matrix

Normalised similarity values	D1	D2	D3	D4	D5	D6
D1	1.000	0.866	0.632	0.128	0.095	0.000
D2	0.866	0.809	0.217	0.178	0.136	0.000
D3	0.632	0.217	1.000	0.510	0.381	0.095
D4	0.128	0.178	0.510	0.653	0.299	0.299
D5	0.095	0.136	0.381	0.299	0.987	0.854
D6	0.000	0.000	0.095	0.299	0.854	0.987

matrix given in Table 4, we choose D1 and D6 as seeds to form the two clusters. These two documents are chosen as seeds since they correspond to the two consecutive highest values, 1 and 0.987 respectively, along the diagonal of the normalised similarity matrix. We notice that we could have chosen either D1 or D3 since both have the same normalised similarity value. Here we have chosen D1. Similarly, we could have chosen either D5 or D6 as a seed, since both have the same normalised similarity value. Here we have chosen D6 as the seed.

We find the clustering solution based on the normalised similarity values between each of the remaining documents and the two seeds. We assign a document to a cluster if the normalised similarity value between the document and the corresponding seed is greater. For example, if we consider document D2, its normalised similarity value with respect to D6 is 0, while its normalised similarity value with respect to D1 is 0.866. Hence, D2 is assigned to the cluster with D1 as the cluster center (seed). Proceeding in this way, we get two clusters where D4, D5 and D6 form one cluster and D1, D2 and D3 form the second cluster.

The following table gives the clustering solution for our proposed algorithm.

The distribution of the documents between the two clusters is given in Table 5. Here 'clu' and 'doc' refer to the class labels.

Table 5. Clustering solution of our proposed algorithm

Cluster/Class	clu	doc
C1	2	1
C2	0	3

5.2 k-Means Clustering Solution

We applied the k-means clustering algorithm[13,14] to the reduced document term matrix with 6 documents. The clustering solution is given below.

Table 6. Clustering solution for k-means algorithm

Cluster/Class	clu	doc
C1	0	2
C2	2	2

5.3 Clustering Solution Validation

We have validated our algorithm by comparing with k-means. We present in Table 7 the values of entropy and purity for our proposed algorithm and k means clustering algorithm. From the entropy and purity values it is evident that our proposed algorithm gives a better clustering solution than the k means algorithm.

Table 7. Clustering solution of our proposed algorithm

Quality/ Algorithm	Proposed Algorithm	K-means Algorithm
Entropy	0.4591	0.6666
Purity	0.8333	0.6666

6. Conclusion

In this paper we have made use of the bipartite representation of documents and clustered the documents based on a new similarity measure that we have introduced. As an extension of this work, we would study the clustering behaviour for a bigger document corpus and also analyse the quality of the clustering result. We would also compare our clustering technique with other clustering algorithms to analyse the efficiency of our technique.

7. References

1. Nagaraj R, Thiagarasu V. Correlation similarity measure based document clustering with directed ridge regression. Indian Journal of Science and Technology. 2014 Jan; 7(5):1-6.
2. Kriegel HP, Kroger P, Sander J, Zimek A. Density-based clustering. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. 2011; 1(3):231–40.
3. Chakrabarti D. Tools for large graph miners [thesis]. Center for Automated Learning and Discovery, School of Computer Science, Carnegie Mellon University, CMU-CALD-05-107; 2005. p. 1-117.
4. Sharief BS, Kartheek E. Laplacian energy of an intuitionistic fuzzy graph. Indian Journal of Science and Technology. 2015; 8(33):1-7.
5. West DB. Introduction to Graph Theory. Prentice Hall; 2001. p. 244.
6. Jack H, Koolen K. Maximal energy graphs. Advances in Applied Mathematics. 2001; 26(1):47-52.
7. Balakrishnan R. The energy of a graph. Linear Algebra and its Applications. 2004; 387:287-95.
8. Grace GH, Desikan K. Reduced term set based document clustering using bipartite graph representation. Proceedings (eBook) of the International Workshop on Graph Algorithms (IWGA2015); USM, Penang, Malaysia. 20015. p. 269-74.
9. *Koolen JH, Moulton* MV. *Maximal energy bipartite graphs. Graphs Combinatorics. 2003; 19(1):131–5.*
10. Pritam C, Gaigole G, Patil LH, Chaudhari PM. Preprocessing techniques in text catagorization. National Conference on Innovative Paradigms in Engineering and Technology (NVIPET-2013); 2013; p. 137-142.
11. Anil A, Kumar S, Chandrasekar C. Text Data preprocessing and dimensionality reduction techniques for document clustering. International Journal of Engineering Research and Technology. 2012; 1(5):1-6.
12. Rama Subramanian C, Ramya R. Effective pre-processing activities in text mining using improved porters stemming algorithm. International Journal of Advanced Research in Computer and Communication Engineering. 2013; 2(12):4536-8.
13. Murilo C, Naldi N, Richardo AFJGB, Campello C. Comparison among methods for k estimation in k-means. IEEE 9th International Conference on Intelligent Systems Design and Application; Brazil. 2009.
14. Rao AS, Ramakrishna S, Babu PC. MODC: Multi-objective distance based optimal document clustering by GA. Indian Journal of Science and Technology. 2016 Jul; 9(28):1-8.