IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 1 Issue 10, December 2014.
www.ijiset.com
ISSN 2348 – 7968

A Novel Similarity Measure for frequent Term Based Text Clustering on high dimensional data

G. sailaja[1] and B.Prajna[2]

[1]Dept.of CS&SE,Andhra University,

Visakhapatnam-530003,India.

[2] Dept.of CS&SE,Andhra University,

Visakhapatnam-530003,India.

Abstract

Text clustering is one of the main themes in text mining .It refers to the process of grouping document with similar contents or topics into clusters to improve both availability & reliability of the mining, In this research, a frequent item set is a set of words which occur together frequently and are good can-Di Dates for clusters. By considering only the items which occur frequently in the Data, we can also address problems like outlier removal, Dimensionality reduction, etc. The main idea is to apply any existing frequent item finding algorithm such as a priori or Dp-tree to the initial set of text files to reduce the Dimension of the input text files. A Document Feature vector is formed for all the Documents. Then a vector is formed for all the static text input files. The algorithm outputs a set of clusters from the initial input of text files considered.

Keywords—*Commonality measure; frequent item; Clustering; Apriori*

1. Introduction

Text clustering plays an important role in information retrieval, topic tracking and detection, web information mining and other fields. By text clustering, texts with similar characteristics are in the aggregate to the same collection, and texts with dissimilar characteristics are divided into different collections. With the rapid development of Internet technology, the number of text is growing at an alarming rate. This makes it very difficult to browse to the content you're interested in. People need to get the relevant topics fast and accurately from a large number of texts content. In 2002, Google launched its own "news" services. It is different from the practice of traditional media, the news is not edited by human, but by consolidation, classification, and aggregate the computer are. The key technology of it is the automatic clustering of the news.

The relation between complexity of Datasets and learnability may be bank mathematically as follows. ID we can Design an algorithm 'A' which can compress the input Dataset, D to a Feasible or reduced Dataset D' then this means that we have learned some important information. This means that there exists a strong relation between Datasets and learnarability.

Dimensionality reduction is also one of the key issues in text clustering and text classification. Text classification, the Dimensionality of the feature vector is usually huge. For example, 20 Newsgroups and Reuters 21578 top, which are two real world data sets, both have more than 15,000 features. Such high dimensionality can be a severe obstacle for classification algorithms [3], [4]. To alleviate this difficulty, feature reduction approaches are applied before Document classification tasks are performed [5]. Two major approaches, feature selection [6], [7], [8], [9], [10] and feature extraction [11], [12], [13], have been proposed for feature reduction. In general, feature extraction approaches are more effective than feature selection techniques, but are more computationally expensive [11], [12], [14]. Therefore, developing scalable and efficient feature extraction algorithms is highly demanded for dealing with high dimensional document data sets.

The problem of emerging research area addressing using the concept of finding frequent item or item sets is gaining significant importance from Data mining researchers and this Dorms basis for the current work. However the method of Dimensionality reduction by eliminating stop words,

stemming words or using TD-ITD methods has been dealt in the previous works.

In this paper, the process of clustering is carried out by passing the input text files to be clustered to the clustering algorithm. Section II of this paper Deals with some of the related works in the literature. In Section III, we discuss the proposed method of clustering the text files and can be extended to any file types. Section-IV concludes the paper.

2.Related Works

There are several similarity measures Defined in the literature such as Euclidean, Cosine, Jaccard, and Manhattan to name a few. for example, the most well-known and widely used similarity measure is cosine similarity and Euclidean measures which requires input as numerical vectors. ID we are successful in Doing so, then we can make use of the existing machine learning algorithms available in the literature or techniques such as text clustering and text classification in Data mining [15].

Euclidean Distance:

The Distance between Data point X and Data point Y can calculated using a mathematical Formula as Follows.

Euclidean Distance(x, y) = (|x1-y1|2 + |x2-y2|2 + … + |xn-1-yn-1|2 + |xn-yn|2)1/2.

Where |Z| represents the absolute value of Z, X is the first Data point, Y is the second Data point, N is the number of characteristics or attributes in Data mining terminology or fields in Database terminology.

Cosine similarity: $SIM_C(\vec{t_a}, \vec{t_b}) = \frac{\vec{t_a} \cdot \vec{t_b}}{|\vec{t_a}| \times |\vec{t_b}|}$

Where $\vec{t_a}$ and $\vec{t_b}$ are m-Dimensional vectors over the term set T = {t1. . . tm}. Each Dimension represents a term with its weight in the Document, which is non-negative. As a result, the cosine similarity is non-negative and bounded between [0, 1]. An important property of the cosine similarity is its independence of document length. For example, combining two identical copies of a Document D to get a new pseudo Document D0, the cosine similarity between D and D0 is 1, which means that these two Documents are regarded to be identical. Meanwhile, given another Document l, D and D0 will have the same similarity value to l, that is, sim ($\vec{t_d}, \vec{t_l}$) = sim ($\vec{t_d}, \vec{t_l}$). In other words, Documents with the same composition. But different totals will be treated identically. Strictly speaking, this does not satisfy the second condition of a metric, because after all the combination of two copies is a different object from the original document. However, in practice, when the term vectors are normalized to a unit length such as 1, and in this case the representation of D and D0 is the same.

Jaccard Coefficient:

The Jaccard coefficient, which is sometimes referred to as the Tanimoto coefficient, measures similarity as the intersection divided by the union of the objects. For text document, the Jaccard coefficient compares the sum weight of shared terms to the sum weight of terms that are present in either of the two Documents but are not the shared terms.

The Jaccard measure considers the commonality among two pair of documents or text files. The function is defined as Jaccard

(Textfile1, Textfile2) =

$$\frac{\text{Textfile1} \cap \text{Textfile2}}{\text{Textfile1 U Textfile2}}$$

3. Proposed Work

3.1. Problem Definition

To Cluster a given set of text files so as to make text files of similar nature f all into one cluster and those of dissimilar nature into other set of clusters.

A.1 Algorithm Text Clustering and Topic Identification (N, Frequent items, Text files)// N – number of text files

A.1.1 Pre-processing Phase:

Begin

Check if the input file is in .txt or .doc or .docx format. If not, convert it in to proper format

Step1: For each text file of the form .txt or .doc do

Begin

Step a. Eliminate Stop words

Step b. Eliminate Stemming words

Step c. Apply any frequent item finding algorithm

Step d. Define feature word size equal to size of all

Frequent item sets obtained in step2

End for

End of Pre-processing Phase

3.2. *Processing Phase*

Step 2: Form a feature set consisting of m Words consisting of each word in frequent items of each document.

Step 3: Form Binary Matrix with row indicating text file and column each frequent item set respectively for each text file in input file set do

Begin
For each frequent item set in feature set do
Begin
If (fk in feature set W is in text file Ti)
Begin
Define Cell value M[Ti, Fk] = 1
// 1 indicates presence of frequent item
Else
Define Cell value M[Ti, Fk] = 0
// 0 indicates absence of frequent item
End if
End for
End for

End of Pre-processing Phase

Step 4: Obtain Similarity matrix for each pair of text files applying new similarity Function

Step 5: Count the number of 0's and place the count in the matrix for each pair of text files.

Step 6: Find the cell with maximum value. Group each pair of each such files into a new cluster.

Step 7: Repeat Step6 until no files exist or we reach the stage of first minimum value leaving zero entry.

Step 8: Display all the clusters formed.

Step 9: Identify topics. Give label to each cluster.

Table 2. Text Files with Frequent Items

	List of items after finding frequent Items
D1	{year, exchange, said, share, price, month, bank, would}
D2	{year, exchange, said, share, month, price, bank, would}
D3	{bank, year}
D4	{year, price, month, exchange, bank, said, share, company, would, rate, wo, wwwwwwi, rate}
D5	{year}
D6	
D7	{rate}{would}
D8	{rate}{would}
D9	
D10	{would}{rate}{bank}

In Table.2, the files denoted by

D1, D2, D4 are traDe1.txt, traDe3.txt and traDe5.txt D7, D8, D10 are earn1.txt, earn2.txt, earn4.txt D3 and D5 are interest4.txt, interest6.txt and D6, D9 are money2.txt and earn3.txt respectively. For simplicity, we show the trace of algorithm by considering 20 example files from Reuter's Dataset -21578. The Documents contain the following frequent items as in Table.2

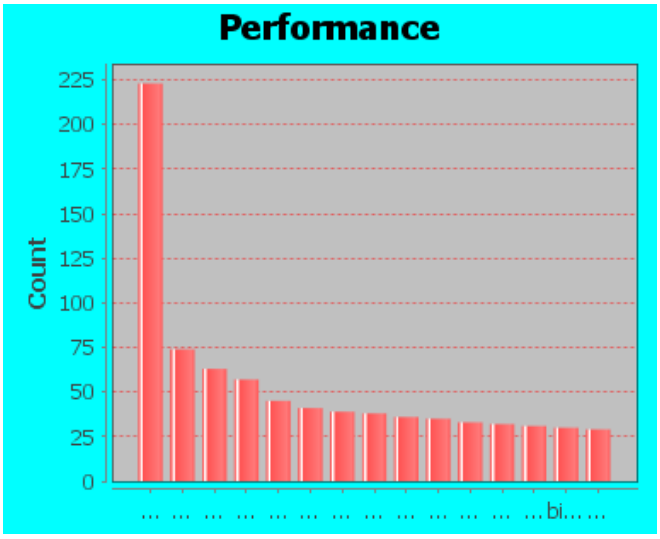
From the above Table II the global frequent items obtained by applying apriori algorithm on all files are given by

For the purpose of finding similarity between two text files, we define the Similarity function S as a function of Documents A and B shown below in the truth table.

Table I: Similarity function over frequent item sets

frequent item In Documents 1	**frequent item In Documents 2**	**Commonality function (D1,D2)**
0	**0**	**U**
0	**1**	**1**
1	**0**	**1**
1	**1**	**0**

Global features = {Year, Bank, Rate, Would, said, Exchange, Share, Price, Month, Compani}.



4. Case Study

Initially we start with the text Documents and reduction the Dimension by eliminating stop words, stemming words. After this step we further reduce the Dimensions by applying a priori algorithm to this Document collection.

A. Din Ding frequent items

The Distribution of each word for a topic shown in the Dig.1 is obtained for the text files considered in table 2 by

D 1 [1, 1, 0, 1, 1, 1, 1, 1, 1, 0]

D 2 [1, 1, 0, 1, 1, 1, 1, 1, 1, 0]

D 3 [1, 1, 0, 0, 0, 0, 0, 0, 0, 0]

D 4 [1, 1, 1, 1, 1, 1, 1, 1, 1, 1,]

D 5 [1, 0, 0, 0, 0, 0, 0, 0, 0, 0,]

D 6 [0, 0, 0, 0, 0, 0, 0, 0, 0, 0,]

D 7 [0,0 , 0, 1, 1, 0, 0, 0, 0, 0,]

D 8 [0, 0, 0, 1, 1, 0, 0, 0, 0, 0,]

D 9 [0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

D 10 [0, 1, 1, 1, 0, 0, 0, 0, 0, 0]

The similarity measure is symmetric, which is the one of the property to be yeartis DieD, hence we can consider only upper Diagonal matrix.

The matrix elements are computed by applying the commonality function S for which each Document pair Dorms the input as shown below

TABLE V. SIMILARITY MATRIX Of TEXT DILE PAIRS

for count = 8

using mat lab version 7.10.0(R2010a).

In Table III we Denote global features as I where i indicates ith feature.

III. Binary representation of table.2 i1

`	D1	D 2	D3	D 4	D5	D6	D7	D8	D9	D1 0
D1	X	8	2	8	1	0	2	2	0	1
D2	X	X	2	8	1	0	0	0	0	1
D3	X	X	X	2	1	0	0	2	0	1
D4	X	X	X	X	1	0	2	2	0	3
D5	X	X	X	X	X	0	0	0	0	0
D6	X	X	X	X	X	X	0	0	0	0
D7	X	X	X	X	X	X	X	2	0	1
D8	X	X	X	X	X	X	X	X	0	1
D9	X	X	X	X	X	X	X	X	X	0
D10	X	X	X	X	X	X	X	X	X	X

The Commonality Value in the cells [D1, D2], [D1, D4], [D2, D4] [D4, D20], [D4, D18], [D18, D20] is 8 .So these files are clustered to Dorm a single cluster [D1, D2, D4,D18,D20] forming Cluster1.

The Commonality Value in the cells[[D3,D5],[D3,D8],[D3,D11][D7,D8] [D10D11] is 2 .So these files are clustered to Dorm a single cluster [D3,D5,D8,D10,D11] forming Cluster2.

Finally Cluster4 contains D7 and Cluster5 contains D12 Cluster6 contains D13 and Cluster7 containsD15 .By applying the clustering algorithm over Table.5 we get final set of clusters:

1: [traDe1.txt traDe2.txt traDe4.txt traDe18.txt traDe20.txt]

2: [earn3.txt earn5.txt earn8.txt earn10.txt earn11.txt]

3: [null6.txt null9.txt null16.txt null14.txt null17.txt null19.txt]

4: [money2.txt]

5: [earn3.txt]

6: [share7.txt]

7 :[share12.text]

8. [bank13.text]

9 .[bank15.text]

We can label the clusters for the purpose of identification by using candidate item sets approach Followed in [6] or pass it to SVM to classify.

The final set of clusters formed are shown below
CLUSTER1 [D1, D2, D4, D18, D20]
CLUSTER2 [D3, D5, D8, D10, D11]
CLUSTER3 [D6, D9, D14, D16, D17, D19]
CLUSTER 4 [7]
CLUSTER 5 [12]
CLUSTER6 [13]
CLUSTER7 [15]

5. Conclusion

The Proposed algorithm has the input as similarity matrix and output a set of clusters as compared to other clustering algorithms that predefine the count of clusters. In this work, frequent items are generated using APRIORI approach by following a similar method. We can replace a priori algorithm by any frequent item Finding algorithm. The algorithm for clustering considers the set of frequent items generated from all the Documents. This gives the commonality between Document pairs. The count of frequent items serves as the Distance measure.

References

1. Jung-Yi Jiang et.al A fuzzy Seld-Constructing f eature Clustering Algorithm for Text Classification, IEEE TRANYEARCTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 23, NO. 3, MARCH 2011

2. Melita HajDinjak, AnDrej Bauer. Similarity Measures for Relational Databases, InDormatica 33 (2009) 143–149

3. R. Agrawal, T. Imielinski, A. Swami. Mining association rules between sets of items in very large Databases, Proceedings of the ACM SIGMOD Conference on Management of Data, 1993, pp. 207–216

4. D. Beil, M. Ester, X.W. Xu, Frequent term-based text clustering, in: Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2002, pp. 436–442

5. Yung-shen Lin, Jung-Yi Jiang et.al "A similarity measure for text classification and clustering ", IEEE Tranyearctions on Knowledge and Data Engineering, 2013.

6. Yi Peng, Gang Kou. A Descriptive frame work for the field of Data mining and knowledge Discovery , International Journal of Information Technology and Decision making, Volume 7, No.4, 2008,Pages 639-682,Impact factor 3.139

7. Kou,G and Lou,C. Multiple factor Hierarchical Clustering Algorithm for Large Scale Web Page and Search Engine Clickstream Data, Vol 197,Issue 1,Page 123-134, Annals of Operation Search,2012.

8. Niraj Kumar, Kannan Srinathan. Automatic Key phrase Extraction from Scientific Documents Using N-gram filtration Technique. Proceedings of fig. 2. Clusters formed for 20 yearmple files from Reuters-21578 Dataset the eighth ACM symposium on Document engineering, Pages 199-208, 2008

9. G.Suresh Reddy, Dr.T.V.Rajinikanth, Dr.Ananda Rao Text Clustering Using frequent Patterns and Jaccard Dissimilarity function, Proceedings of the second ICACM 2013.

10. Pieter Adriaans and fold Zantinge. Data Mining. Eleventh Pearson education. 2013

11. Daniel Larose. An Introduction to Data Mining. John Wiley Publications

12. Daniel Larose. Data Mining. Models and MethoDs.2012. John & Wiley Publications

13. K. Ravi Shankar, G.V.R. Kcompani, Vikram Pudi. "Evolutionary clustering using frequent item sets". ACM Stream KDD '10. Proceedings of the first International Workshop on Novel Data Stream Pattern Mining Techniques, Pages 25-30, 2010

15.G.suresh ready,Dr.T.V.Rajinikanth, ,Dr.Ananda Rao Text clustering using frequent patterns and jaccard fissimilarity function, proceeding of the second ICACM 2013.
Eleventh Pearson education. 2013.

.