Advances in Computational Sciences and Technology
ISSN 0973-6107 Volume 10, Number 2 (2017) pp. 221-233

http://www.ripublication.com

A study on Topic Identification using K means clustering algorithm: Big vs. Small Documents

Pema Gurung[1] and Rupali Wagh[2]

[1] *Student, Dept. of Computer Science, Christ University, Bengaluru.*

[2] *Associate Professor, Dept. of Computer Science, Christ University, Bengaluru.*

Abstract

Document clustering is a technique which groups similar content documents from the collection. It can further be extended to extract topics of each groups. Document clustering and Topic identification form back bone of information retrieval, but size of documents to be grouped in terms of number of words affects these processes negatively. The sparsity of terms present in big documents impacts weight of individual term and in turn quality of clusters adversely. This paper presents application of cluster analysis for document collection of small documents and document collection of big documents for topic identification from document collection. Results are presented as comparisons to emphasize the concerns with respect to big documents.

Keywords: K-means algorithm, Document Clustering, Topic identification, Topics, Clustering, Text pre-processing.

INTRODUCTION:

Document clustering has played a significant role in many fields like: information retrieval, data mining, understanding etc. Most of the problems of document clustering are high dimension, big volume and complex semantics. Document clustering groups similar documents which helps to organize the documents. There are various clustering algorithm that help to cluster text documents. Some of the clustering algorithms are: K-means clustering, Fuzzy clustering, Hierarchical clustering, Flat clustering etc. The basic idea in clustering is to group documents into different clusters based on

appropriate similarity measure. To make groups, each document is signified by a vector that represents the weight assigned to words in document. Weighting using tf-idf (term frequency-inverse document frequency) scheme is commonly performed for document clustering. A list of cluster is generated with every document at the end result.

A topic concentrates on terms that might be used while discussing a particular subject. Topic also refers to a hidden variable where extraction of the information from the textual data helps to identify topic. In topic modelling, a topic is defined by a cluster of words with each word in the clusters having a probability of occurrence for a given topic. Topic modelling is often used for extracting hidden meaning from the set of text documents. The "topics" generated by topic modelling techniques are clusters of similar words. A topic model captures the meaning which allows analysing a set of documents and learning, based on the statistics of the words. Topic model helps in establishing and offering vision to understand large collections of unstructured data. Initially, topic modelling was developed as a text mining tool, however it is now generally used to detect useful structures in data.

LITERATURE REVIEW:

Document clustering is a widely applied text mining functionality. With the increasing availability of text data, efficient search operation has become the need of the day. Document clustering or text clustering can be considered as back bone of information retrieval systems. Clustering is an activity of grouping similar objects. In text clustering such grouping of documents is done based on similarity of the contents stored in the documents which is very useful for improving efficiency as well as precision and recall of any information retrieval system. Detailed discussion on text clustering basics, various document representation techniques, document similarity measures and pros and cons of various clustering algorithms is described [1]. Topic identification is one of the very important applications of document clustering. Topic identification is identifying labels for clusters obtained by grouping similar documents. These identified keywords then can be used for further analysis tasks. Topic identification has been used for search result clustering [2]. With the popularity of social media in last two decades, topic identification is used for extraction of hot keywords identification [3] from public opinions. Considering semantic space of the document for clustering and topic identification is discussed in [4] where authors discuss other text analysis functionalities through text association rule and text categorization in the framework of topic identification. Application of document clustering for identification of keywords from the clusters and then using these keywords for generating text summaries is discussed in [5]. Document clustering applied to stream of documents to incorporate temporal properties of incoming documents in a repository is discussed in [6], where

such a framework can be used for lifelong learning from continuous inflow of documents. Similar work for clustering news articles and automatically grouping every new article into its appropriate group is discussed in [7]. While vector space model and bag of word representation of documents is used widely for text analysis, capturing semantic structure of documents and representing documents with underlying concepts also is popular in text mining applications. Latent semantic indexing based approaches for topic identification has been discussed in [8] where authors highlight the capacity of semantic based models to overcome the issues of synonyms.

Clustering is functionality with very rich algorithmic support. Hierarchical and partitioning based clustering algorithms are most popular for grouping documents. K means and its many variants like bisecting k means and spherical k means have been used widely on document datasets [9]. K means algorithm's popularity lies in its low computation complexity. But major limitation of k means, sensitivity to initialization and to number of cluster still remains challenging research question. Though hierarchical clustering algorithms are free from these limitations, stopping criteria is the deciding factor in the quality of generated clusters. Hybrid approaches combining partitioning and hierarchical algorithms are used by researchers to get optimum advantages of both methods [10]. Calculation of similarity index of document pairs is central to any document clustering algorithm. In contrast to structured data, similarity calculation is complicated in text due to different writing styles, context sensitivity of natural languages, synonyms etc. Many text specific similarity measures other that cosine and Euclidian similarity are used with document similarity. Finding similarity of terms based on coupled correlation analysis is proposed in [11].

Topic identification from dataset containing big documents is still challenging due to the quality of clusters obtained after grouping the documents. Huge feature set of big documents and inherent complexities of text data are major factors affecting the quality of the results. In this study, document clustering techniques are used for identification of topics from document collection. The dataset used for the study is conference papers and their abstracts. The study presents comparison of results when applied to abstracts (small documents) Vs full papers (big documents). Though a conference paper and its abstract should represent same topic, resulting clusters after applying k means algorithm show differing characteristics.

MATERIALS AND METHODS:

Document clustering is used to group similar documents in clusters which helps to organize and visualize, documents collection. In document clustering, clusters are obtained from the document term matrix. Document term matrix is intermediate representation of document collection where matrix values are calculated based on

importance of each word by considering frequency of the word in the document. These clusters help to identify the topics using various statistical measures.

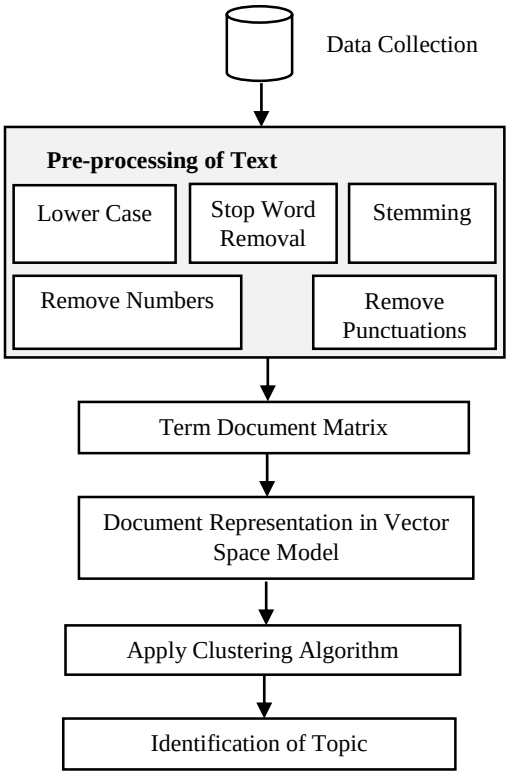

Fig 1. Generic Model of Text Clustering for topic Identification

Generic model diagram shown in Fig. 1, explains the generic process of identifying topics from the cluster of documents. Initially, the documents are collected and are cleaned through the pre-processing techniques as explained below.

1. Converting to lower case: converting all the text to lowercase.
2. Tokenization: splitting up a string into a set of tokens.
3. Remove stop words: E.g. the, a, in, you etc.
4. Remove punctuation: E.g. “,”, “.”, “/”,””, etc.
5. Remove numbers: E.g. 1, 2, 3… N.
6. Stemming: identifying the root word. E.g. flying, flew->fly.

The cleaned corpus is then converted to tf-idf matrix to represent the document in the vector space (VSM) where VSM is just a model for representing text documents as vectors of identifiers. It is mainly used for information filtering, information retrieval, indexing and relevancy rankings. Vector space model first indexes the document where

the terms are extracted from the text document and it then finds the weight of the indexed terms. The terms obtained are represented as vectors of the corpus in the vector space model, to compute the distance and similarities between the documents. For document clustering, there are two commonly used approaches: k means and hierarchical clustering. Hierarchical clustering gives a better quality clusters but suffers from quadratic time complexity while K-means have comparatively efficient with a linear time complexity. Michael Steinbach et al. [12] states that K-means technique is better than hierarchical approach as per their evaluation. In this paper, we have used K-means clustering algorithm which uses the cosine similarity between vectors. Cosine similarity of two vectors is computed by dividing the dot product of two vectors by their product of their magnitude. Cosine of the angle between the vectors indicates similarity because at the closest the two vector are zero degree apart. All the closest vectors from the dataset are clustered into groups by the k-means algorithm. The clusters formed are analysed and topics were identified from the set of the vectors in the clusters that belong to the same cluster. The topics are keywords formed from words that are most frequently used in the document which gives a baseline in identifying the topics. There are various algorithms to identify the topics such as Differential cluster labelling, cluster internal labelling. Differential cluster labelling, labels cluster by comparing term distributions across clusters. Terms with low frequency can be ignored in labelling a cluster, instead differential test can be used to achieve the best results with differential cluster labelling. Cluster internal labelling, selects labels which are content dependent of the cluster. Cluster-internal labelling, use various methods such as finding terms that occur frequently in the centroid. For a particular cluster of documents, centroid can be calculated by finding the arithmetic mean of all the document vectors. When a new vector appears in the centroid that has more value then it implies that the term has more number of frequency in the cluster. The words with more frequency can be termed as label of the clusters.

A document and its summary or abstract portray same topic through the text written in them. But when the documents are big like research papers, this intuitive notion of conceptual similarity between these mentioned documents is affected by large dimensionality of documents. Big documents result in sparse term document matrix and further impact the quality of any statistical method applied on them. This study focuses on document clustering used for topic identification from document collection. Two different but related datasets are used – Conference research papers which represent document collection of big documents and collection of their respective abstracts which represents document collection of small documents. The study further presents a comparison of results obtained in the process of applying document clustering on these datasets. This study focuses on impact of large dimensionality on document clustering.

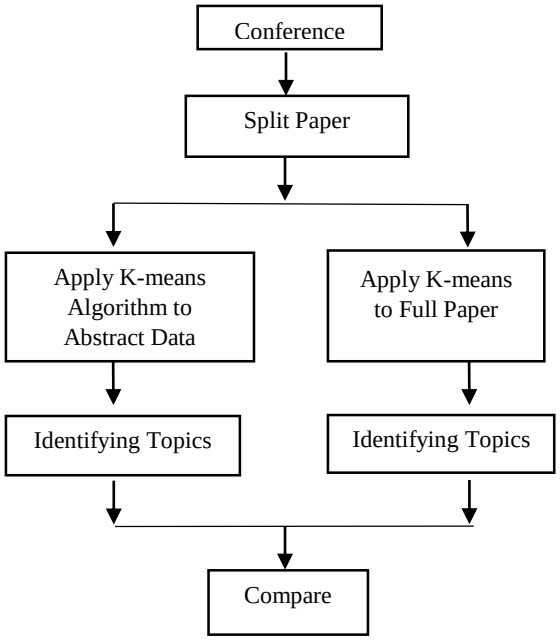

Fig 2. Methodology

Fig. 2, explains methodology followed during this study to compare the big and small documents where the Methodology follows generic model as explained in previous section. The first process of comparing the corpus requires collection of documents. For the analysis, papers were collected from the online source. Documents were further split into small document (only abstract was extracted) and big document (text from introduction to conclusion). Linguistic pre-processing as mentioned in the previous section was applied to both the big and small document.

Table 1: Data Description

Document	Number of Document	Min Terms	Max Terms
Abstract	33	82	274
Full Text	33	2252	6182

Table 1 shows the number of documents for abstract corpus and full text corpus. In abstract corpus, the minimum number of terms present is 82 and maximum number of terms present is 274. While in full text corpus, minimum number of terms present is 2252 and maximum number of terms present is 6182.

K-means Algorithm: In this study, K-means is used for clustering algorithm to group similar documents through cosine similarities as explained in the "Generic model" section. The K-means algorithm follows the following steps [13].

1. Choose k number of clusters to be determined.
2. Choose k objects randomly as the initial cluster centroid.
3. Repeat
   - Assign each object to their closest cluster.
   - Compute new clusters, i.e. Calculate mean points.
4. Until
   - No changes on cluster centers (i.e. Centroids do not change location any more) OR
   - No object changes its cluster

RESULTS AND DISCUSSION:

This section present results obtained during the course of study. Following prominent statistical measures are used during the course of analysis.

1. Frequent words and their weights in document collection
2. Assignments of Document to Cluster
3. Frequent Words and weight within cluster

Table 2: Most frequent terms in Abstract dataset.

Terms	X
Hastags	0.438643
Articles	0.433192
Twitter	0.403837
Network	0.395254
Sources	0.392346
Search	0.384568
Users	0.361975
Ads	0.337033
Path	0.307585
Questions	0.299681

Table 2 contains top 10 most frequently term used in the Abstract corpus where the value "x" determines the weights of the terms used in all the document from Tf-idf.

Table 3: Clusters of Abstract corpus

Cluster	Terms	Weight	Document
1	Sources	0.3923464	12, 15
1	Path	0.3075850	20
1	Apps	0.2918245	10
1	Maturity	0.2501352	10
1	Credibility	0.2307893	12
1	Spectral	0.2066479	26
1	Mining	0.2061728	5,15,31
1	Ratings	0.2005485	10
1	Collective	0.1985982	15
1	knowledge	0.1881600	31,33
2	Hashtags	0.4386430	1,9,14,16,18,22,23,30
2	Twitter	0.2879350	9,17
2	Dig	0.2654944	7
2	Duplicate	0.2637592	25
2	Articles	0.2601076	7
2	Chats	0.2501352	9
2	Infrastructure	0.2425189	17
2	Search	0.2185080	3
2	Group	0.2005485	17
2	Detection	0.1978194	9
3	Ads	0.3370328	16,27
3	Mouse	0.2894324	14
3	Eye	0.2480850	14
3	Accounts	0.2193215	8
3	Network	0.2179872	19,26,33

3	Personalization	0.2064957	11
3	Annoying	0.1801569	27
3	Diversity	0.1769963	22
3	Osps	0.1769963	11
3	crowdsourcing	0.1732582	21

Table 3 shows the cluster-wise prominent terms with their weight within the cluster and memberships of individual document when k means algorithm was applied to abstract dataset. Based on the similarity with the centroids, 3 clusters of documents are obtained. Note that the cluster1 contains "knowledge, apps, and search" related papers mainly for "Online Knowledge", cluster2 contains "twitter, dig articles, hashtags" related papers mainly focusing on "Social media sources", and cluster3 contains "personalization learning, modelling, and network" related papers based on "models".

Table 4: Most frequently used words in Full Text dataset.

Terms	**X**
Hashtags	0.18587
Energy	0.172794
Apps	0.170593
Mouse	0.168351
Spectral	0.164245
Maturity	0.15058
Ads	0.148178
Certificate	0.141781
Dig	0.133301
Path	0.130912

Table 4 displays the top 10 most frequently term used in the Full text corpus where the value of each terms determine the sum of weights of the terms used in all the document from tfIdf.

Table 5: Clusters of Full Text corpus

Cluster	Terms	Weight	Documents
1	Energy	0.17279395	2, 29
1	Apps	0.17059315	1, 10
1	Mouse	0.16835128	14
1	Spectral	0.15624436	26
1	Maturity	0.15058039	10, 17
1	Certificate	0.14198068	3
1	Dig	0.13330110	7
1	Btm	0.10811687	1
1	Ils	0.09954967	3
1	Quora	0.09699156	18, 33
2	Ads	0.14641694	11, 14, 16, 27, 29
2	Path	0.10841914	20, 27
2	Annoying	0.10088788	27
2	Contagious	0.08854190	19
2	Rules	0.08279219	5, 9, 11, 13, 21
2	Duplicate	0.07854792	5, 9
2	Personaliza-tion	0.07188426	11, 13, 18
2	Rule	0.07168380	5, 12, 21, 24
2	Suffix	0.06812621	20
2	Amie	0.06702419	5
3	Hashtags	0.17232420	25
3	Dlsh	0.12787363	23
3	Agents	0.11751266	7
3	Stc	0.11318315	24
3	Hashtag	0.11290343	25
3	Agent	0.09058738	6
3	Latency	0.08508165	23, 30
3	Trie	0.08207535	23
3	Diversity	0.07615290	22
3	articles	0.07337923	12, 22, 24

Table 5 shows the cluster of full text dataset with respect to top10 terms of each clusters where, each clusters are derived from the terms in the document. Note that the cluster1

contains “apps, online news, and quora” related papers which is mainly for “online social information”, cluster2 contains “rules, path, and ads” related papers mainly focusing on “learning”, and cluster3 contains “hashtags, agent, and articles” related papers based on “information search and retrieval”.

Table 6: Comparison between “Abstract” and “Full Text” dataset.

Terms	Abstract Document			Full Text Document		
	Document	**Cluster**	**Weight**	**Document**	**Cluster**	**Weight**
Ads	16,27	3	0.337033	11,14,17,27,29	2	0.146417
Annoying	27	3	0.180157	27	2	0.100888
Apps	10	1	0.291825	1,10	1	0.170593
Articles	7	2	0.260108	12,22,24	3	0.073379
Digg	7	2	0.265494	7	1	0.108117
Diversity	22	3	0.176996	22	3	0.076153
Duplicate	9	2	0.263759	5,9	2	0.078548
Hashtags	25	2	0.438643	25	3	0.172324
Maturity	10	1	0.250135	10,17	1	0.150580
Mouse	14	3	0.289432	14	1	0.168351
path	20	1	0.307585	20,27	2	0.108419
personalization	11	3	0.206496	11,13,18	2	0.071884
spectral	26	1	0.206648	26	1	0.156244

In the above table 6, the weight of terms in datasets of abstracts abstract document are high as compared to the weight terms in dataset of full paper document. It is also observed that the clusters compositions obtained after applying k means to these datasets are also different. Results demonstrate negative impact of sparsity on text clustering results and hence topic identification. In abstract document, the term mouse has weight 0.250135289432 which belongs to cluster 3 and the term is from document 14, while in full text document, the term mouse has weight 0.168351 which belongs to cluster 1 and the term is form document 14. Weight of a term in a document, which is the measure of its importance within the document collection is represented very strongly in collection with small documents. This affects all further results as term document matrix acts as the basis for applying any text analysis techniques.

CONCLUSION

The fundamental contribution of this paper is to highlight the impact of high dimensionality of big documents on clustering results. Application of k means algorithm for document clustering was experimented with two different but related data sets in this paper. Experimental results show that document clustered in abstract corpus and full text corpus differs in many ways. Topic identification and text clustering are two very important tasks in information retrieval domain. Availability of big documents on web impacts the result of these processes as demonstrated in the paper. The results thus highlight the need for more robust methods for documents with more number of terms. The results emphasize on more robust methods like semantic based methods for big document collections.

REFERENCES:

[1] Aggarwal, Charu C., and ChengXiang Zhai, 2012, "A survey of text clustering algorithms", Springer US, 2012, 77-128.

[2] Antoine Naud, and Shiro Usui, 2008, "Exploration of a collection of documents in neuroscience and extraction of topics by clustering", 1205-1211.

[3] Fang, Tingshao Zhu, Yue Ning and Zhiqi, "Hot keyword identification for extracting web public opinion", Pervasive Computing and Applications (ICPCA), 2010 5th International Conference on. IEEE, 2010.

[4] Clifton, Chris, Jason Rennie and Robert Cooley, "Topcat: Data mining for topic identification in a text corpus", 2004, IEEE transactions on knowledge and data engineering 16.8: 949-964.

[5] K. I. M. Kono, K. O. Youngjoong, and S. E. O. Jungyun, 2003, "Topic keyword identification for text summarization using lexical clustering", IEICE transactions on information and systems 86.9: 1695-1701.

[6] Ankur Agarwal, Hurtado, Jose L., and Xingquan Zhu, 2016, "Topic discovery and future trend forecasting for texts", Journal of Big Data 3.1: 7.

[7] Smeaton, Alan F., et al. "An Architecture for Efficient Document Clustering and Retrieval on a Dynamic Collection of Newspaper Texts." BCS-IRSG Annual Colloquium on IR Research. 1998.

[8] Cheng, Xin, et al. "Coupled term-term relation analysis for document clustering." Neural Networks (IJCNN), The 2013 International Joint Conference on. IEEE, 2013.

[9] Steinbach, Michael, George Karypis, and Vipin Kumar. "A comparison of document clustering techniques." KDD workshop on text mining. Vol. 400. No. 1. 2000.

[10] Basu, Tanmay, and C. A. Murthy, 2015, "A similarity assessment technique for effective grouping of documents", Information Sciences 311: 149-162.

[11] Kuhn, Adrian, Stéphane Ducasse, and Tudor Gírba, 2007, "Semantic clustering: Identifying topics in source code", Information and Software Technology 49.3 : 230-243.

[12] Karypis, Michael Steinbach George, Vipin Kumar, and Michael Steinbach, May 2000, "A comparison of document clustering techniques", TextMining Workshop at KDD2000. 2000.

[13] Deokar, Mrs Sanjivani Tushar, 2013, "Text Documents clustering using K Means Algorithm", International Journal of Technology and Engineering Science [IJTES] Vol 1 (4), pp 282 – 286, July 2013.