Journal of Engineering Sciences
Vol 11, Issue 4 ,April/2020
ISSN NO:0377-9254

Article Clustering Using A Normalised Semantic Data Representation

Meghana Santoshi Janapareddy[1], Nirmala Paul Nirujogi[2] , Nandini Prasada[3], R.Sree Meghana[4]
[1234]Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam.
{meghanajs99, nirmalapaul12}@gmail.com
Dr.J.Hyma[5]
Associate Professor
[5]Anil Neerukonda Institute of Technology and sciences,Visakhapatnam.
jhyma.cse@anits.edu.in

Abstract. The need for clustering of documents is high in applications like document summarization,information retrievel etc. Huge collections of documents are pilling everyday. It is really challenging as to efficiently clusters can be formed for a given text documents. It is evident that clustering needs to be performed with best preprocessing and analyzing techniques with respect to preserving the order of sequence of words and concept of words in the documents. There are many algorithms and approaches used till date which have their own merits and demerits. The algorithms used for word vectors here is "Word2Vec-Skip grams Model" ,the document is represented by computing a feature vector.A feature vector is calculated by using the word vectors by applying the min max method and for clustering is "k means". It is used for clustering the docments.

Keywords: K-means,PCA , Skip-Grams , Clustering , Word Vectors,Min-Max Vectors,Feature Vectors .

I. INTRODUCTION

Clustering is the task of dividing the data points in groups such that the points that are similar will be in the same group.Clustering is unsupervised and it is important as it determines the intrinsic grouping in the unlabelled data.

There are different methods for clustering:density based,partition based,heirachical, and grid based.What type of clustering method to choose depends on the type of data and the propoerties of data to which the algorithm has to be applied.If the data has varying density regions then density based models are appropriate.For any data k-means is the basic clustering algorithm.

While clustering is unsupervised,classification on the other hand is supervised.In todays world the number of articles published online is increasing rapidly.To use such huge collection of dicuments efficiently the important factor is efficient retrievel of informations from the documents. A given set of articles may belong to different domains like newspapers, it contains a lot of articles each belong to a different category. The list of categories includes sports, economic, editorial, politics, educational etc. Identifying the category of an artilce is crucial in cases like to retrieve articles pertaining to a certain topic or desired search output.

Articles can be categorized by using text clustering. Clustering of articles solves the problem of only fixating to a particular set of lables, new cluster is created when the article does not satisfy any of the clusters already present, while in the case of classification articles are only classified into the given set of classes, no new classes are formed.

The data that is meant to be clustered has to be represented using a suitable data representation technique.The most common representation is TF-IDF.TF-IDF fails to represent the semantic meaning of a document.Word2Vec,Glove,FastText are some of the techniques used to represent the document in semantic way.

Word Vectors are a vectors for words.The numerical representation of a word in terms of row of real values is a word vector.The words which are similar in meaning will have vectors are mapped to proximate or nearer points in the geometric space.Each dimension represents a meaning and the semantics of the word are embedded across the dimension of the vectors.

The words vectors or the word embeddings represent only the words in the document in order to represent the document the features of the document have to extracted the .The feature vector is formed using theese features and then the clustering algorithm is applied.Feature Extraction is a important process in clustering or classification .if the extracted features are not correct the output

generated will be of low accuracy.

To cluster the articles the input document i.e. the article is represented in feature vectors, clustering algorithm is applied to group the articles as efficiently as possible and the dimensions of the vectors are reduced with the help of principal component analysis to helpin visualization of clusters.The different dimensionality reduction techniques are PCA,Random Forest ,ensemble trees and others.Dimesnionality reduction helps in the visual analysis of the data and also speeds up the execution.

II. RELATED WORK

Clustering is an important aspect in efficient information retrieval system, scientific data exploration, text mining and many others. With the increasing amount of articles generated online it becomes a requirement to group the articles that are similar together so that instead of searching the entire articles one can only search the articles similar to it. There are different methods to carry out clustering like hierarchical, partition, fuzzy, model-based and density-based.

In [6] proposed a Word2Vec model to output word vectors , these vectors can be used to represent the text (large) or even the whole article.The first step is to train the data using the model and then evaluate the similarity between the words. The words which are similar are clustered together,the clusters obtained are used to fit the data inorder to decrease the data dimension.The word vectors are clustered to k clusters ,similar words are grouped under the same cluster.By doing so the aimed to reduce the data dimension and also speed up the classification .

[8] designed the CNN(Convolutional Neural Network) having one fully connected layer and three convolutional layers ,the input is the word vectors of a paragraph formed using word2vec (CBOW) and the output layer is avector as a paragraph vector. Stanford Sentiment Treebank dataset was used. The dataset contains sentences ,each sentence has a label in the range 00 to 1.0 with closer to 0.0 indicationg very negative and closer to 1.0 indicating very negative.

In[2] proposes a method to find topics in the text with the help of a self organizing map(SOM) .The SOM is used to cluster the word vectors (generated using Word2Vec) . K-means is applied to separate the output of SOM and obtain k clusters.The centroid of the cluster is the word vector and the word is the topic of that cluster.The approach was tested on a dataset of 19997 texts and 20 topics and the results were efficient.

In[5] proposes a new method for determining the number of topics ,it follows two principles: a topic model used to segregate the noise topics by usign ARTM ; dense vector representation using word vector models like Glove fast Text Word2Vec;Cosine metric works better than euclidean metric.It was tested on dataset from oneptero lib and showed good results in assessing the optimal number of topics built using a small collection of english document.

In[1] The dataset is processed to remove the data that is not useful and deemed as noise. The problems of dimensionality curse will reduce the algorithm efficiency if the terms of the document are in large numbers and the efficiency of the algorithm is reduced.To reduce these terms, some feature selection techniques like TF-IDF should be used.TF-IDF is applied, then heirarchical agglomerative clustering and fuzzy k-means with iterations along with silhouette coeff to determine the number of clusters.

In[7]propose a clustering method for search result using name entity extraction. Name entity extraction extracts the informationwhich is important for recognition and they become label candidates.The first step fetches the documents list ,in second step the terms from each document are extracted, in the third step calculates the score for each term.The score is calculated using a label selection criterion.The label selection criterion is based on two ideas: terms that are uself will not be very frequently occuring or very rarely occuring; terms which are realted to the query are labels.High scored terms are labels.The language independent method can be used to handle other languages and offers better accuracy when compared to other methods.

In[4] support vector machine for supervised learning is presented in this paper.The method uses an item pair similarity measure to increase the efficiency of correlation clustering.The clustering algorithm is constant and the similarity measure is modified to produce required or derired results.The clustering algorithm is trained and the desired clusters are obtained.Inorder to obtain the desired results i.e. clusters the users should provide additional information.Similarity measure is used to calculate if the pair is similar or not based on the value,if it is positive the pair is alike else it is not alike. Each item pair has a feature vector to describe the pair. Correlation clustering is used. The author concludes that this approach provides improved performance

when compared to a naive classification approach.All the words are grouped into K-clusters.

III. PROPOSED METHODOLOGY

In order to cluster the articles the following steps have been followed.

1. Preprocessing
2. Word vector generation
3. Feature vector generation
4. K-means and PCA

A.Preprocessing

Dataset

The dataset is gathered, it is a mis of articles of different areas of news like sports, entertainment, business, politics, technology etc. The dataset has been gathered from the BBC news category dataset [3] . The dataset comprises of 2225 documents in .txt format from the BBC news website.

Then the files from the dataset are read one by one .On each file the contents are read and preprocessing is performed .In preprocessing, the text of the file is first converted into tokens the punctuation, digits and symbols are removed. From the obtained tokens duplicates and stopwords are removed. The list of stopwords is taken from NLTK corpus. To the obtained words stemming is applied (porter stemmer) .In order to remove the verbs and other parts of speech except nouns and adjectives post tagging is performed.

Tokenization

Tokenization can be done by first reading the text document and copying its contents to string. Then applying .split() or by using regular expressions re.findall("[\w']+", text) from the re library. This results in list of strings. One needs to perform Tokenization before removing any stopwords.

Stop words Removal

From NLTK corpus load the stopwords and use stopwords.words (list_name) to remove the stopwords. The list of stopwords are in appendix 1

Stemming

For stemming implement the Porter stemmer. This stemmer though time consuming is simple to implement and is known to produce the best possible output when compared to other stemmers. NLTK provides list of commonly agrred upon stopwords.The list of stopwords used is in Appendix.

B. Word vector generation

The mathematical way of representing words is a word vector.A word vector represents the meaning of the word.One word vector is a row of real valued numbers where semantically similar words have similar vectors.The words which are similar will be grouped in the same area in the vector space.There are different ways of generating word vectors the most common models are CBOW,Skip-Grams,Word2Vec and Glove. Skip-grams is the method chosen to generate the vector for each word.

Skip Gram Model

A skip gram model helps in predicting the context of a word. If two words have a lot of context words in common, similar vectors are given to those two words,a NN(neural network) is used which predicts the target word form the context word. As the neural network learns and updates the weights of the hidden layer with a large enough corpus, these weights will magically become our word vectors that represent the context word.

After preprocessing the hyper parameters are defined, window size, learning rate, epochs and dimension of the word vector. Then the training data is generated which is a vector in one hot representation. The generated training data is used to train the model which is a neural network two matrices of the dimension NX100 and 100XN called as the weight matrices where N is the number of unique words in the corpus.

In the forward pass the dot product of weight matrix w2 and target word in one hot representation w_t produces hidden layer variable h , the dot product of h and w2 produces output layer variable u and finally softmax of u to force each element in the range 0 to 1 .The prediction y_pred, hidden layer variable h and output layer variable u are used to calculate error for a target word, context word pair. Calculate the sum of y_pred with each of context words in w_c.

Next the amount of adjustment or delta of weights is calculated using error EI, h and w_t. The weights are updated.

.Now after training we get the word vectors.

Journal of Engineering Sciences

Vol 11, Issue 4 ,April/2020
ISSN NO:0377-9254

C. Feature Vector Generation

The generated word vectors represent the vector for each unique word in the corpus, for the BBC dataset 2240 files contained 186059 words and 16776 unique words. Each unique word has a vector of dimension 100. The words which are similar are represented with vectors that occur nearer in the vector space. But these vectors represent the words in a document or article and not the document on whole. In order to group the documents it is essential to represent the document in a numeric way to apply the algorithm , to tackle this problem feature vector has to be calculated, which will represent the information in the document in a numerical way. The feature vector is generated by finding the minimum vector and maximum vector of all the word vectors for the document and taking the average of the two. The result now becomes the feature vector of the document.

Minimum Vector is of dimension equal to the dimension of the word vectors that is 100 and is the vector that is obtained by comparing the values at index i for all vectors and finding the minimum value , this becomes the value at the i^{th} position in the minimum vector.

$$\boldsymbol{MinVec} = (\boldsymbol{min_0}, \boldsymbol{min_1}, \ldots \boldsymbol{min_i}, \ldots \boldsymbol{min_n}) \quad \textbf{(1)}$$

Maximum Vector is of dimension equal to the dimension of the word vectors that is 100 and is the vector that is obtained by comparing the values at index i for all vectors and finding the maximum value , this becomes the value at the i^{th} position in the maximum vector.

$$\boldsymbol{MaxVec} = (\boldsymbol{max_0}, \boldsymbol{max_1}, \ldots \boldsymbol{max_i}, \ldots \boldsymbol{max_n}) \quad \textbf{(2)}$$

Feature Vector (FV) is calculated using by the flowering Eq. (3).

$$\boldsymbol{FV} = (\boldsymbol{MinVec} + \boldsymbol{MaxVec})/2 \quad \textbf{(3)}$$

2	C:\Users\F	visibl	[[-0.00130	kind	[[-0.00189	organ	[[0.00187	move	[[0.0
3	C:\Users\F	cigarett	[[1.95328	dress	[[9.78910	wonki	[[0.00428	gallery'	[[-4.
4	C:\Users\F	greatli	[[0.00047	deal	[[-0.00065	analysts'	[[0.00071	time	[[0.0
5	C:\Users\F	desper	[[0.00058	review	[[0.00060	recruit	[[0.00269	part	[[-0.
6	C:\Users\F	week'	[[-1.10094	place	[[-0.00403	success	[[0.00498	quit	[[-0.
7	C:\Users\F	crackdowr	[[-0.00075	oper	[[0.00086	move	[[0.00434	pc	[[0.0
8	C:\Users\F	royal	[[0.00473	queen	[[0.00444	pele	[[-0.00401	christense	[[0.0
9	C:\Users\F	term	[[0.00382	factor	[[0.00351	look	[[-0.00467	meet	[[0.0
10	C:\Users\F	retent	[[-0.00385	look	[[-0.00467	cover	[[-0.00306	subject	[[0.0

Fig 1: Word Vectors of a Document stored in a row in word in one cell and vector in the cell to its right.

1	C:\Users\F	[-0.00476	[0.004985 [0.0001100888475775718 7, -9.099883027374744e-(
2	C:\Users\F	[-0.00486	[0.004987 [6.342795677483082e-05, -6.847083568572998e-05
3	C:\Users\F	[-0.00477	[0.005044 [0.0001368212979286909, -4.082405939698219e-0
4	C:\Users\F	[-0.00474	[0.005088 [0.0001718723215162754, -0.0001139894593507 05
5	C:\Users\F	[-0.00461	[0.005044 [0.0002145580947399 1394, 0.00016346620395779(
6	C:\Users\F	[-0.00473	[0.005167 [0.0002182289026677 6085, -0.0001783170737326 1
7	C:\Users\F	[-0.00488	[0.004933 [2.5709159672260284e-05, -0.0002742158249020 5
8	C:\Users\F	[-0.00487	[0.005044 [8.723372593522072e-05, 3.2296171411871 91e-05,
9	C:\Users\F	[-0.00478	[0.005177 [0.0001980364322662 3535, -0.0001448080874979 4
10	C:\Users\F	[-0.00461	[0.005205 [0.0002950662747025 49, -3.976537846028805e-05,

Fig 2: Feature Vector of a Document stored in a row

As seen in fig.(2) the vector representation of a document is shown. The vectors in fig.(1) are used to calculate the feature vector.

D. K-Means and PCA (Principal Component Analysis)

A.K-means

The obtained vectors are then used to perform clustering by using k means algorithm. K means is an partition based clustering method that tries to partition the dataset into k predefined partitions where k is the number of estimated clusters clusters. Each point belongs to only one cluster and the points in the cluster are as similar as possible whereas the clusters itself are as distinct as possible. It calculates the sum of the squared distances between data points to the centroids and assigns it to the cluster where the difference is minimum.
The algorithm works as follows:

1. Predefine the number of clusters to be formed- K
2. Shuffle the dataset and initialise centroids by selecting randomly K data points.
3. Iterate until the centroids are constant and there is no allocation of new point to the cluster

4. Compute the sum of squared distance between all centroids and data point.
5. Assign each data point to the closest centroid.
6. The cluster is represented by the centroid which is formed by taking average of all the points in the cluster.

Fig 3: Pseudocode for K-means

B.PCA

The obtained feature vectors are of dimension 100. It is difficult to perform visualization for a 100 feature vectors and therefore dimensionality reduction has to be applied. Principal component analysis (PCA) is a dimensionality reduction technique. PCA maps the data in higher dimensional spaces to lower dimensional space and expects the variance in the lower dimensional space to be maximum.

Step1: Compute the covariance matrix

Step2: Compute the eigenvectors of covariance matrix

Step3: Eigenvectors having large eigenvalues are used to reconstruct the data

Fig 4: Pseudocode for PCA

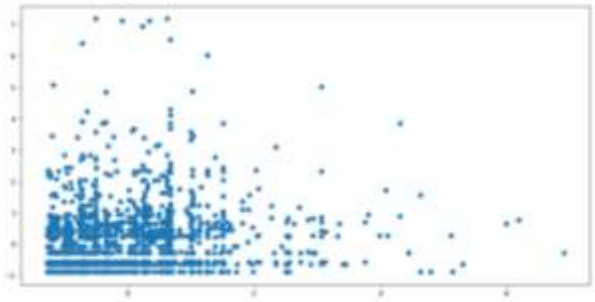

Fig 5: Visualization of raw data before PCA

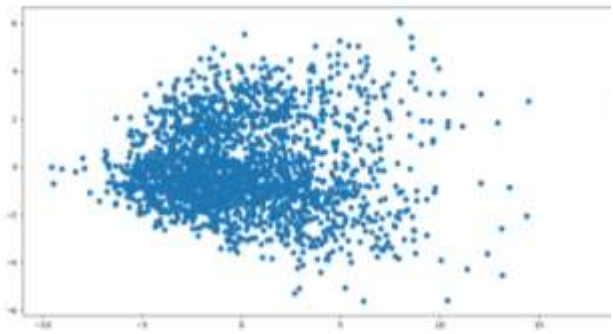

Fig 6: Visualization of raw data after PCA

The dimensionality reduction technique also helps in plotting the high dimensional data. As shown in fig.(4) the before PCA is applied the plotting fails to show the patterns in the data. In fig.(5) after PCA we can better visualize the data.

System Architecture

The proposed system is represented using a flow diagram. Each of the input documents is pre-processed; preprocessing is done in various steps. The first step is tokenization, then the digits and punctuation is removed from the list of tokens, stopwords are removed from the list of tokens, post tagging is applied to remove the verbs and other parts of speech except nouns and adjective, and the lasts step is to perform stemming on the words to get the root words and the unique words in all 2240 files are stored.

The unique words in all words are trained using the word2vector skip-grams model to compute the vectors for each word, each vector is of dimension 100.The computed vectors are used to find the feature vectors of each document.

On these vectors clustering algorithm i.e. K-means is applied to group the documents into clusters based on Euclidean distance and then dimensionality reduction is applied on each document to produced scaled vectors the scaled vectors are used in visualizing the vectors.

Step 1: Preprocessing

Step 1a: Tokenization

Step 1b: Removing digits, numbers and symbols.

Step 1c:Stop word removal

Step 1b:Stemming(Porter Stemmer)

Step 2: Word vectors for each word in a document.

Step 3: Feature vector for each document using the document's words vectors.

Step 4: Clustering algorithm on feature vectors K-means.

Step5:Dimensionality reduction and visualization pf

clusters(PCA)

Fig 7: Proposed methodology

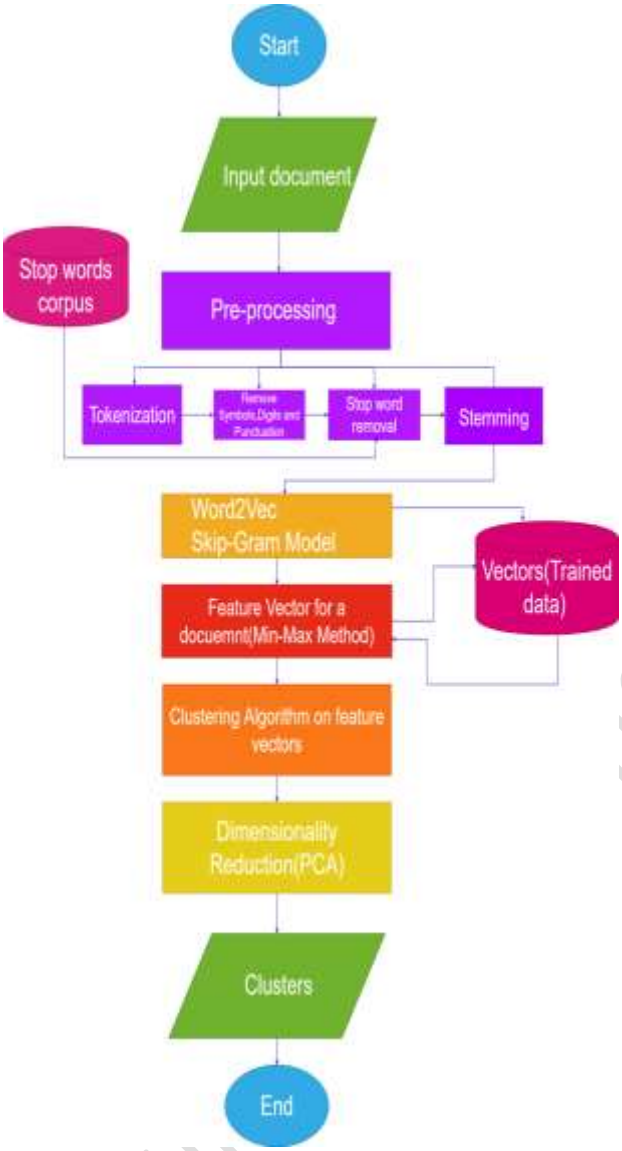

Fig 8: System Architecture

IV. RESULT

The proposed methodology is tested against 50 ,100,551,1251 and 2240 files each set containig a mixture of 5 topics-business,politics,sports,entertainment and technology.The number of words in each set ,number of unique words in each set and the time taken to train these words is represented in the table(1)

No. of articles	50	100	551	1251	2240
Total No. of words	3892	7703	43742	100413	1806059
No. of uique words	2049	3185	8168	12578	16776
Time required to train unique words(epochs: 2) (in seconds)	4224	6381	16412	39657	57905

Table1: Time taken to train words in each set.

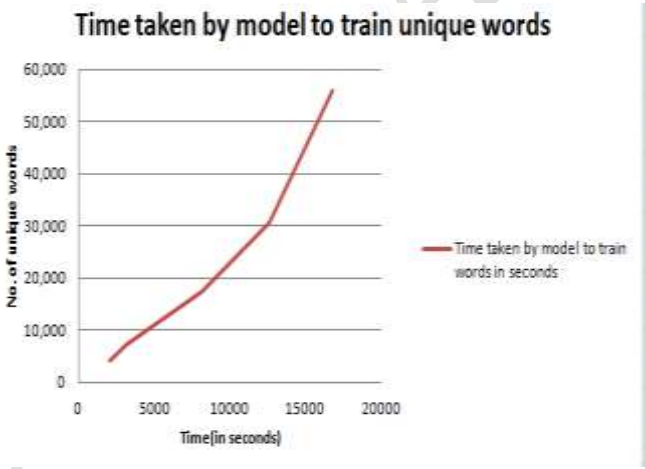

Fig 9: Graph of No.of unique words Vs Time required to train the words

As seen in fig.(7) with the increase in the number of unique words to be trained the amount of time taken to train in seconds increases .

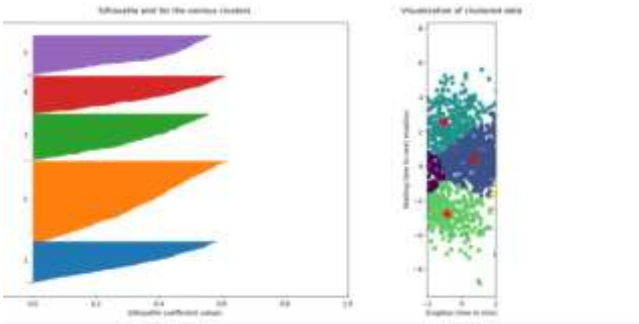

Fig 10: Silhouette Score Analysis of K-means

No.of clusters(k)	Silhouette Score
2	0.4509
3	0.3361
4	0.3375
5	**0.7033**

Table 2: Silhouette score analysis with k=2,3,4,5

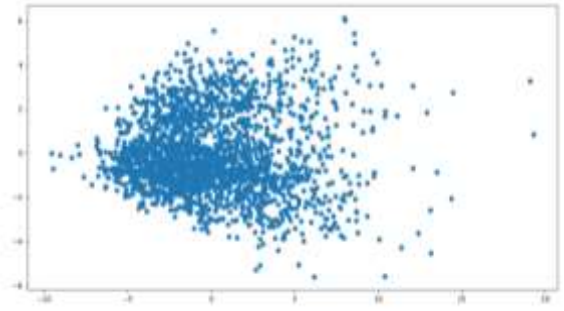

Fig 11: The raw data

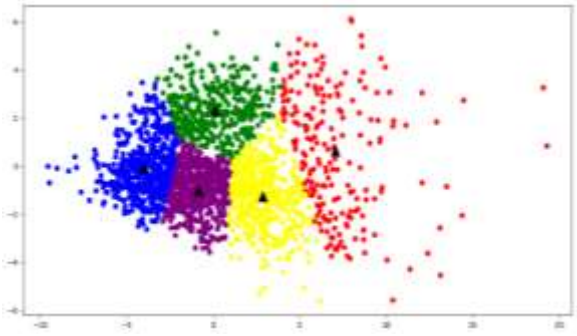

Fig 12: Graphical representation of clusters formed by K-means

The output of the method is shown is fig.(5).Each cluster is given a different colour and the centroid is depicted using the triangle marker,the 5 topics-business,politics,sports,tech and entertainment are formed as five different clusters.The clusters are identified in the raw data visualized in plot in fig.(10)

Metric	Value
Silhouette Score	0.70330435
calinski_harabasz_score	1686.53
davies_bouldin_score	0.3523

Table 3: Cluster Validation Metrics

From the table 3 we can note that the proposed methodology gives 70.33% accuracy and we can also note with the high calinski harabaraz score and davies bouldin score that the clustering performed is good.

V. CONCLUSION AND FUTURE SCOPE

In this paper we provided a methodology for clustering articles. We demonstrated our technique usinf K-means on a dataset of 2240 articles and produces the clusters plotted in a graph. This methodology attains good results in considerably shorter time as compared to standard procedures .The proposed methodology can be extented to other datasets .The methodology can be extended to other applications as well like search engines , information discovery, ,market research,pattern discovery etc.

APPENDIX

['i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're", "you've", "you'll", "you'd", 'your', 'yours', 'yourself', 'yourselves', 'he', 'him', 'his', 'himself', 'she', "she's", 'her', 'hers', 'herself', 'it', "it's", 'its', 'itself', 'they', 'them', 'their', 'theirs', 'themselves', 'what', 'which', 'who', 'whom', 'this', 'that', "that'll", 'these', 'those', 'am', 'is', 'are', 'was', 'were', 'be', 'been', 'being', 'have', 'has', 'had', 'having', 'do', 'does', 'did', 'doing', 'a', 'an', 'the', 'and', 'but', 'if', 'or', 'because', 'as', 'until', 'while', 'of', 'at', 'by', 'for', 'with', 'about', 'against', 'between', 'into', 'through', 'during', 'before', 'after', 'above', 'below', 'to', 'from', 'up', 'down', 'in', 'out', 'on', 'off', 'over', 'under', 'again', 'further', 'then', 'once', 'here', 'there', 'when', 'where', 'why', 'how', 'all', 'any', 'both', 'each', 'few', 'more', 'most', 'other', 'some', 'such', 'no', 'nor', 'not', 'only', 'own', 'same', 'so', 'than', 'too', 'very', 's', 't', 'can', 'will', 'just', 'don', "don't", 'should', "should've", 'now', 'd', 'll', 'm', 'o', 're', 've', 'y', 'ain', 'aren', "aren't", 'couldn', "couldn't", 'didn', "didn't", 'doesn', "doesn't", 'hadn', "hadn't", 'hasn', "hasn't", 'haven', "haven't", 'isn', "isn't", 'ma', 'mightn', "mightn't", 'mustn', "mustn't", 'needn', "needn't", 'shan', "shan't", 'shouldn', "shouldn't", 'wasn', "wasn't", 'weren', "weren't", 'won', "won't", 'wouldn', "wouldn't"]

REFERENCES

1. Alsamurai, Ather Abdulrahem Mohammedsaed. "Text categorization based on semantic similarity with word2vector." Master's thesis, Çankaya Üniversitesi, 2017.
2. Bafna, Prafulla, Dhanya Pramod, and Anagha Vaidya. "Document clustering: TF-IDF approach." In 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), pp. 61-66. IEEE, 2016.
3. Cha, Miriam, Youngjune Gwon, and H. T. Kung. "Language modeling by clustering with word embeddings for text readability assessment." In Proceedings of the 2017 ACM on Conference on Information and

Knowledge Management, pp. 2003-2006. 2017.
4. Dai, Andrew M., Christopher Olah, and Quoc V. Le. "Document embedding with paragraph vectors." arXiv preprint arXiv:1507.07998 (2015).
5. De Miranda, Guilherme Raiol, Rodrigo Pasti, and Leandro Nunes de Castro. "Detecting Topics in Documents by Clustering Word Vectors." In International Symposium on Distributed Computing and Artificial Intelligence, pp. 235-243. Springer, Cham, 2019.
6. D. Greene and P. Cunningham. "Practical Solutions to the Problem of Diagonal Dominance in Kernel Document Clustering", Proc. ICML 2006.
7. Ding, Chris, and Xiaofeng He. "Principal component analysis and effective k-means clustering." In Proceedings of the 2004 SIAM International Conference on Data Mining, pp. 497-501. Society for Industrial and Applied Mathematics, 2004.
8. Finley, Thomas, and Thorsten Joachims. "Supervised clustering with support vector machines." In Proceedings of the 22nd international conference on Machine learning, pp. 217-224. 2005.
9. Hotho, Andreas, Alexander Maedche, and Steffen Staab. "Ontology-based text document clustering." KI 16, no. 4 (2002): 48-54.
10. Krasnov, Fedor, and Anastasiia Sen. "The number of topics optimization: clustering approach." Machine Learning and Knowledge Extraction 1, no. 1 (2019): 416-426.
11. Lewis, David D. "Feature selection and feature extraction for text categorization." In Proceedings of the workshop on Speech and Natural Language, pp. 212-217. Association for Computational Linguistics, 1992.
12. Le, Quoc, and Tomas Mikolov. "Distributed representations of sentences and documents." In International conference on machine learning, pp. 1188-1196. 2014.
13. Maas, Andrew L., Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. "Learning word vectors for sentiment analysis." In Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1, pp. 142-150. Association for Computational Linguistics, 2011.
14. Ma, Long, and Yanqing Zhang. "Using Word2Vec to process big text data." In 2015 IEEE International Conference on Big Data (Big Data), pp. 2895-2897. IEEE, 2015.
15. Miao, Yingbo, Vlado Kešelj, and Evangelos Milios. "Document clustering using character N-grams: a comparative evaluation with term-based and word-based clustering." In Proceedings of the 14th ACM international conference on Information and knowledge management, pp. 357-358. 2005.
16. Pelevina, Maria, Nikolay Arefyev, Chris Biemann, and Alexander Panchenko. "Making sense of word embeddings." arXiv preprint arXiv:1708.03390 (2017).
17. Toda, Hiroyuki, and Ryoji Kataoka. "A search result clustering method using informatively named entities." In Proceedings of the 7th annual ACM international workshop on Web information and data management, pp. 81-86. 2005.
18. Toda, Hiroyuki, and Ryoji Kataoka. "A clustering method for news articles retrieval system." In Special interest tracks and posters of the 14th international conference on World Wide Web, pp. 988-989. 2005.
19. Xiong, Zeyu, Qiangqiang Shen, Yijie Wang, and Chenyang Zhu. "Paragraph vector representation based on word to vector and CNN learning." Computers, Materials & Continua 55, no. 2 (2018): 213-227.
Yoshioka, Koki & Dozono, Hiroshi. (2018). The Classification of the Documents Based on Word2Vec and 2-Layer Self Organizing Maps. International Journal of Machine Learning and Computing. 8. 252-255. 10.18178/ijmlc.20