Available online at sjuoz.uoz.edu.krd

Science Journal of University of Zakho

Vol. 11, No. 2, pp 153.–159, April-June 2023

journals.uoz.edu.krd

p-ISSN: 2663-628X
e-ISSN: 2663-6298

AFFINITY PROPAGATION AND K-MEANS ALGORITHM FOR DOCUMENT CLUSTERING BASED ON SEMANTIC SIMILARITY

Avan Atam Mustafa[1] , Karwan Jacksi[2]
1Computer Science Dept.University of Duhok, Duhok, Iraq avan.mustafa@uod.ac
2Computer Science Dept., University of Zakho, Zakho, Iraq karwan.jacksi@uoz.edu.krd

Received: 2 March., 2023 / Accepted: 25 March., 2023 / Published: 4 April., 2023 https://doi.org/10.25271/sjuoz.2023.11.2.1148

ABSTRACT

Clustering text documents is the process of dividing textual material into groups or clusters. Due to the large volume of text documents in electronic forms that have been made with the development of internet technology, document clustering has gained considerable attention. Data mining methods for grouping these texts into meaningful clusters are becoming a critical method. Clustering is a branch of data mining that is a blind process used to group data by a similarity known as a cluster. However, the clustering should be based on semantic similarity rather than using syntactic notions, which means the documents should be clustered according to their meaning rather than keywords. This article presents a novel strategy for categorizing articles based on semantic similarity. This is achieved by extracting document descriptions from the IMDB and Wikipedia databases. The vector space is then formed using TFIDF, and clustering is accomplished using the Affinity propagation and K-means methods. The findings are computed and presented on an interactive website.

KEYWORDS: Text clustering, semantic similarity; document clustering; Affinity Propagation; K-means; Data mining;

INTRODUCTION

As Internet users continue to rise, the volume of textual material on the Internet has grown exponentially, and the network is now swamped with massive volumes of textual data [1], Without proper categorization and summarizing of document content, retrieving valuable information is very difficult [2][3] [4].

Text clustering is the process of looking at a group of texts and putting ones that are similar together. Text clustering is a type of unsupervised data mining that doesn't require to train the model ahead of time or markup texts by hand [5]. The clustering method is more efficient and needs less human interaction than other Natural Language Processing (NLP) techniques, such as classification. Text clustering has therefore become a crucial method in NLP and has been extensively used in several disciplines, including information retrieval, organization, and processing [6].

Semantic document clustering is the process of determining the similarity of texts based on semantic rather than statistical criteria [7]. In contrast, the traditional clustering approach clustered words based on their grammatical format, which failed to group words with identical meanings [8]. This is occurring because of synonymy and polysemy difficulties; a term with several meanings it is difficult to be classified with their related meaning, or non-semantically similar words are put in the same cluster. A semantic document clustering is important to address this issue [4][9].

This article employs document clustering based on semantic similarity by grouping 100 movie synopses taken from the IMDB and Wikipedia databases. The approach's major phases begin with obtaining movies and synopses from various databases, then combining them and applying preprocessing to make them more convenient to use. Following that, to transform these synopses into integer form Term Frequency-Inverse Document Frequency (TFIDF) method is applied making them suitable for use by clustering methods. TFIDF is a technique for determining the importance of a word as a numerical statistic, where TF calculates the frequency of the term in a document and IDF calculates the frequency of the word in all corpuses, and then TF and IDF are multiplied to produce the numerical weight of a single word [10].

Lastly, two techniques were used for clustering: affinity propagation and the K-means algorithm. Several internal and external assessment measures were used to diverse datasets to compare the performance of the two methods.

RELATED WORKD

Several comparable papers are analyzed to determine the current condition of the research area.

Guan et al., [11], proposed a similarity assessment based on Unilateral Feature Set, Cofeature Set, and Significant Cofeature Set is expanded from the Cosine coefficient utilizing structural information. These three sets represent various textual elements at various places. Their structural details enhance the clustering outcomes. The new measure of similarity may be used to directly compute asymmetric similarity, which is not restricted to the symmetric space. In addition, the Seeds Affinity Propagation clustering approach has been developed, which merges Affinity Propagation with semi-supervised learning. The use of SAP for text clustering expands the applicability of Affinity Propagation. In compared to the traditional k-means clustering method, SAP not only decreases the processing cost of text clustering and increases accuracy, but it also successfully avoids being random initialized and caught in local minimum. Furthermore, SAP is more robust and less susceptible to data distribution than k-means, traditional AP, SAP (CC), and AP (Tri-Set).

Authors of [12] developed and implemented an efficient semantic clustering strategy for movie datasets obtained from Wikipedia and IMDB. The objective was to cluster these movies based on their synopses. After generating the synopses using the NLTK dictionary, the (TFIDF) technique is utilized to transform

* Corresponding author

them into an integer form that can be used by clustering algorithms. K-means and HAC clustering techniques are applied, and the findings are explained in a way that can be compared. Furthermore, the running consumption time for both algorithms has been reported, with the HAC method yielding the best results in all situations. As established by internal and external validation the result demonstrates that the k-means method performed best in all cases, while the HAC algorithm performed better in terms of time rating in all situations. So, the output illustrates that the K-means algorithm has better performance, while HAC has less computing time.

Another work that demonstrates a semantic similarity approach to document clustering using the NLTK dictionary is revealed by [4]. The approach involves generating synopses from IMDB and Wikipedia datasets, then the defined data is tokenized and stemmed. After that, a strategy for text vectorization is developed with TFIDF, and for clustering, the ward's approach and K-means algorithm are used. WordNet is another technique that is used to cluster documents based on their semantic approach. During the implementation stage, each method was tested using three different scenarios: 1) without preprocessing; 2) preprocessing without stemming; and 3) preprocessing with stemming. For measuring similarity, the Silhouette metric and many other metrics are used with the five different datasets. The Silhouette measure using the (nltk-Reuters) dataset delivers the best similarity ratio for all clusters when utilizing the K-means method, and k = 10 provides the highest ratio. Similarly, using Ward's approach, the maximum similarity range of the Silhouette metric is produced for all clusters by combining the (IMDB and Wiki top 100 movies, and Nltk-brown) datasets, and the best similarity ratio is obtained for the (Wiki & IMDB top 100 movies) dataset when k = 5. Compared to the existing literature, the results demonstrate that Ward's technique performs better than K-means for small datasets. Finally, they used an interactive webpage to display the result and explain the link between all the clusters.

Another paper reveals a semantic similarity method to document clustering utilizing the Glove word embedding and DBSCAN clustering algorithms [13]. The two-word embeddings that are most often used in document clustering are Word2vect and Glove. This involves analyzing and quantifying the frequency with which a given word occurs in its context. The work is made up of four key parts: gathering the datasets; preprocessing the data (tokenizing, stemming, and eliminating stop-words); using the Glove word embedding technique with PPwS and PPwoS of the data; and finally using the DBSCAN clustering algorithm on the word vectors from the two types of preprocessing. The results of their experiments show that the suggested system does better than a system that uses TFIDF and k-means clustering when the dataset is large and complex. The TFIDF and K-means methods perform better than their proposed approach when the dataset size is small, nevertheless. The findings were evaluated using the most commonly used evaluation metrics in document clustering.

The findings of an investigation into the clustering technique of 83 scientific document files consisting of three topics (Convolutional Neural Network, Hypertension Retinopathy, and Deep Learning) are reported by Triwijoyo and Kartarina [14]. After the information has been turned into plain text, the retrieval process includes tokenization, an English filter for "stop words," porter stemming, and changing all characters to lower case. Cosine similarity was then utilized to determine document similarity. Finally, for document clustering, the researcher uses the K-means algorithm. Their results illustrate that the applied method provides an accuracy of 84.3%.

Table I shows the results for the above-mentioned approaches that used the same dataset.

Table 1: Compared result of related work

100 Movies IMDB & WIKI			
Ref.	Method	Silhouette Score	No. of Clusters
[13]	Glove word embedding with DBSCAN	0.005	17
[4]	K-means and Ward’s Method	0.0258	5
[12]	K-means	0.0298	5
[12]	HAC	0.0252	5

PROPOSED APPROACH

The process consists of five main stages, The first step involves the process of collecting 100 synopses of the top 100 movies from both the IMDB and Wikipedia databases, document preprocessing, representation of documents, clustering the documents, and presenting the result. The objective is to group these descriptions using the Affinity propagation algorithm. In addition, the Affinity propagation findings are compared to the K-means clustering technique. The framework of the proposed strategy is shown in Fig. 1.

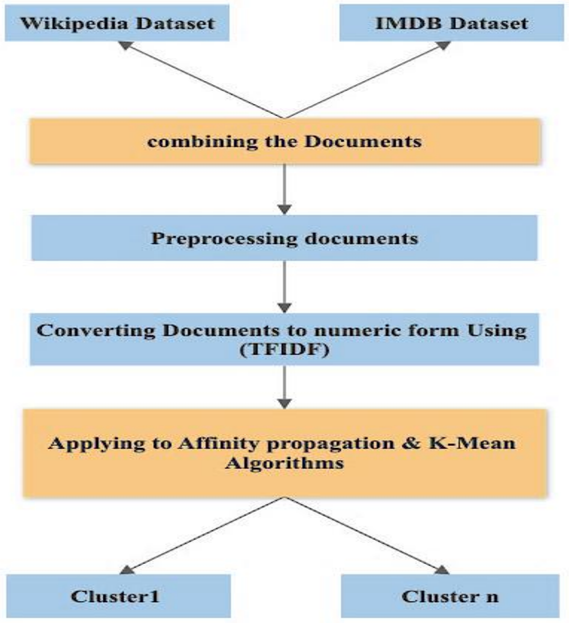

Wikipedia Dataset
IMDB Dataset
combining the Documents
Preprocessing documents
Converting Documents to numeric form Using (TFIDF)
Applying to Affinity propagation & K-Mean Algorithms
Cluster1
Cluster n

Fig. 1. Outline of the proposed method

STEPS OF PROPOSED APPROACH

A. step1:Combining the Documents

After downloading movie synopses from IMDB and Wikipedia, the documents are placed in two distinct arrays. The summary of each movie contains 100 words to describe the movie. After that, to make the array stronger, the Wiki and IMDB were combined.

B. step2: Preprocessing Document

The objective of document preprocessing is to define the documents such that their retrieval and storage are very productive. In the stage of preprocessing, the natural language toolkit (NLTK) English dictionary is loaded. There are three kinds of preprocessing: The steps are as follows: 1) Filtering: transforming each synopsis into an array of tokens and then filtering every token by alphabet range (A to Z or a to z) to exclude any punctuation, digits, and symbols. 2) tokenize and stemming: applying the snowball stemmer included in the (NLTK), every token returns back to its root word. For instance, the term "going" will be back to (go, with both words being treated as one feature. 3) Stop word removal: words such as "are, is, an, you, etc." are considered insignificant and eliminated.

C. Step3: Generating TFIDF Matrix for Document *Representation*

Representing the document is a critical stage in document processing and information retrieval systems. The full-text versions of documents must be turned into form vectors in order to find related documents among a large number of documents. This kind of translated document displays information from the original text depending on the words in the term index and used in indexing, related keyword rankings for improved search results, information filtering, and retrieval. The vector model, is a typical algebraic model that uses vectors to represent text documents. Documents are displayed using the vector model, which employs the Term Frequency (TF), the Inverse Document Frequency (IDF), or the TF-IDF weighting method [15]. In the suggested technique, the TFIDF term is used to make each word from the synopses a feature on the matrix so that there are no repeats. Then, calculate the frequency with which each component occurs in all of the synopses and use that number as the weight for each feature.

D. Step4: Applying the Clustering Algorithms

The TFIDF matrix is utilized by the Affinity propagation algorithm in this stage to build clusters based on the exemplar number. Exemplar points are those that best describe the other data points and are the most significant in their cluster. Unlike clustering algorithms such as k-means, the affinity propagation method generates clusters by exchanging messages between pairs of instances until convergence.

The dataset is represented as the number of exemplars, which are defined as those most representative of other samples. The proposed approach uses (-3) as an exemplar number because we don’t have a big dataset. The dataset used is limited to 100 synopses. The TFIDF matrix that is used by the K-means cluster technique has a K value of 5.

Affinity propagation and K-means are graph-based cluster algorithms. The general concept of both algorithms is that k-means depends on a distance function and a parameter k that determines the number of clusters. Affinity propagation depends on a similarity function (which can be based on a distance function) and learns the number of clusters without having to be told it in advance [16][17][18].

The Affinity Propagation algorithm consists of the next steps:

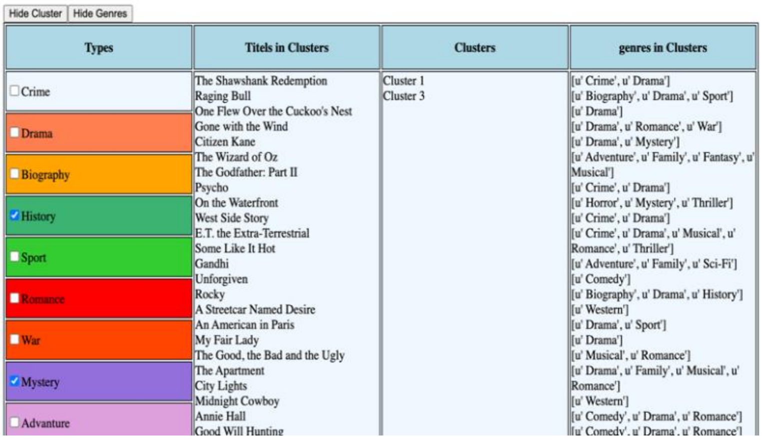
- *Step 1: Initializing the availability matrix to zero, number of exemplars defined as k*
- *Step 2: Update the responsibilities and the availabilities matrix*
- *Step 3: Determine the maximum total for each set of data points, as well as the best exemplar for each set.*
- *Step 4: If No changes occurred on exemplar value, go to step 5; otherwise, continue to step 1.*
- *Step 5: Assign the data points to their respective exemplars based on the highest similarity to identify clusters.*

The essential stages of the K-means method, on the other hand, are [17]:

1. Define number of clusters as K cluster.
2. Determine the centroid position.
3. Using mathematical techniques, such as (Euclidian), determine the distance between datapoints.
4. Compute the Average data points for each cluster, calculate the new midpoint for each group.
5. Repeat step 2 until the position of the centroid datapoint does not change and no other data points move.

E. Step5: presentation of cluster result

6. Fig. 2, illustrates a table view of the outcomes with faceted browsing capability, allowing users to filter information in the table. Fig. 3, shows the procedure's outputs, which are actual clusters of connected movie titles. The results are visualized and presented as a web page exhibiting a graph of movie title clusters with associated genres.



Types	Titels in Clusters	Clusters	genres in Clusters
☐ Crime ☐ Drama ☐ Biography ☑ History ☐ Sport ☐ Romance ☐ War ☑ Mystery ☐ Advanture	The Shawshank Redemption Raging Bull One Flew Over the Cuckoo's Nest Gone with the Wind Citizen Kane The Wizard of Oz The Godfather: Part II Psycho On the Waterfront West Side Story E.T. the Extra-Terrestrial Some Like It Hot Gandhi Unforgiven Rocky A Streetcar Named Desire An American in Paris My Fair Lady The Good, the Bad and the Ugly The Apartment City Lights Midnight Cowboy Annie Hall Good Will Hunting	Cluster 1 Cluster 3	[u' Crime', u' Drama'] [u' Biography', u' Drama', u' Sport'] [u' Drama'] [u' Drama', u' Romance', u' War'] [u' Drama', u' Mystery'] [u' Adventure', u' Family', u' Fantasy', u' Musical'] [u' Crime', u' Drama'] [u' Horror', u' Mystery', u' Thriller'] [u' Crime', u' Drama'] [u' Crime', u' Drama', u' Musical', u' Romance', u' Thriller'] [u' Adventure', u' Family', u' Sci-Fi'] [u' Comedy'] [u' Biography', u' Drama', u' History'] [u' Western'] [u' Drama', u' Sport'] [u' Drama'] [u' Musical', u' Romance'] [u' Drama', u' Family', u' Musical', u' Romance'] [u' Western'] [u' Comedy', u' Drama', u' Romance'] [u' Comedy', u' Drama', u' Romance']

Fig. 2: Cluster faceted browsing

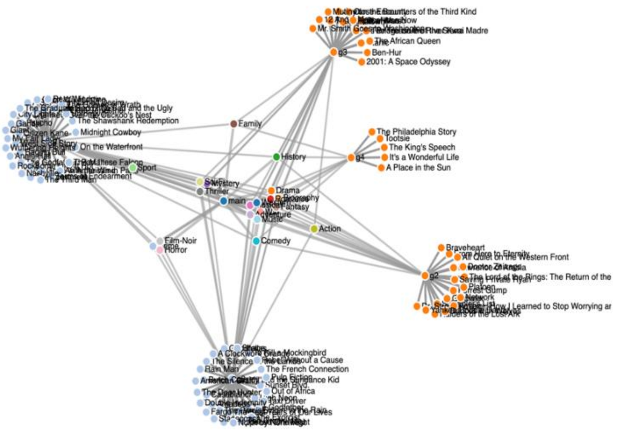


Fig. 3:Semantic Clustering with Affinity Propagation algorithm

EXPERIMENTAL RESULTS

A comparison was made between the suggested algorithmic method and a variety of previously approved document clustering techniques. Affinity Propagation and the K-means approach are used in this paper. Programming in the Python language is used.

F. Datasets

Three datasets are utilized for this work's result implementation: (WIKI & IMDB from 100 Top Movies), which has no target value; (txt_Sentoken), which is a 2000 document of movie reviews; and (Nltk_Brwon) which contains 500 documents of movie reviews. Since the proposed system dataset doesn't have a target, it can't be measured by an external metric. In this case, the method is put into place with the help of both internal and external evaluation metrics.

G. Evaluation Measures

The silhouette metric is used for internal evaluations. It indicates the degree to which things in other groups are similar. The range is from -1 to 1. A negative score indicates that there is a dissimilar point in the cluster, while a positive result indicates that the entities are comparable, with a similarity degree ranging from 0 to 1 [19]. The Silhouette scale relies only on the recorded data, which eliminates the requirement for a target value.

Several metrics, such as purity, V_measure, F1-measure, and accuracy, are used for the external evaluation. These metrics are needed to reach the target value. Wherever Purity is a measurement of the degree to which groupings include just a single class. This indicates that the complete number of elements is accurately assessed, with scores ranging from 0 to 1 [20].

Furthermore, F1-measure is a measure that assesses the aggregation's accuracy with a range from 0 to 1. This scale represents the balance between scale accuracy and recall. However, since recall indicates a number of valid positive outcomes divided by the number of samples which should be classified as positive (true positives and false negatives), it is best used in imbalanced classes. Precision is the number of accurate positive outcomes divided by the number of positive results returned by the classifier [21]. Besides accuracy, which is likewise evaluated between 0 and 1, and it is a measure of all correctly picked cases, it is best used when all instances are of comparable significance and the class distribution is similar [22]. Finding the optimal match between group labels and category labels is critical for grouping accuracy.

The goal of these evaluation techniques is to compare affinity propagation and K-means algorithms. Time of executing for both methods is also calculated, allowing the performance to be computed and compared.

H. Internal and External Evaluation of Proposed Approach

The outcome of the suggested approach was found by applying silhouette as an internal evaluation measure to all datasets and using the affinity propagation and K-means algorithms. The output of both algorithms is illustrated in Table II and Fig. 4.

Table 2: Silhouette Output Score Of Suggested Method

No.	Datasets	Silhouette Score		
		No of cluster	Affinity propagation	K-Means
1.	Wiki & IMDB top 100 movies	4	0.0292	0.0234
2.	Nltk_Brown[1]	5	0.0189	0.0475
3.	Txt_Sentoken[2]	20	0.0124	0.0188

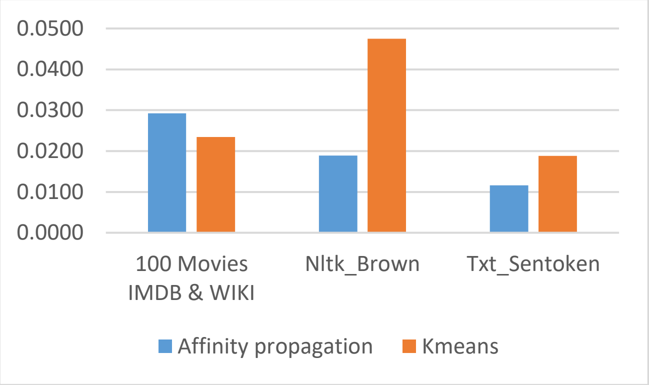


Fig. 4: Silhouette score of the Proposed Method

Using the two datasets (Nltk_Brwon and txt_Sentoken), many metrics are employed for external evaluation. The dataset (100 movies from (IMDB & WIKI) is omitted because it has no target value. Affinity propagation and K-means work results are shown in Table III and Fig. 5.

Table 3: External evaluation results of proposed approach

No.	Metric	Nltk_Brown		txt_Sentoken	
		AP	K-Mean	AP	K-Means
1.	Purity	1	1	0.567	0.624
2.	Accuracy	0.367	0.170	0.0545	0.525
3.	F1-Measur	0.0584	0.107	0.0099	0.0094
4.	V_measure	0	0	0.011	0.028

[1] The dataset is taken from https://www.nltk.org

[2] The dataset is taken from https://www.kaggle.com/datasets/vipulgandhi/movie-review-dataset

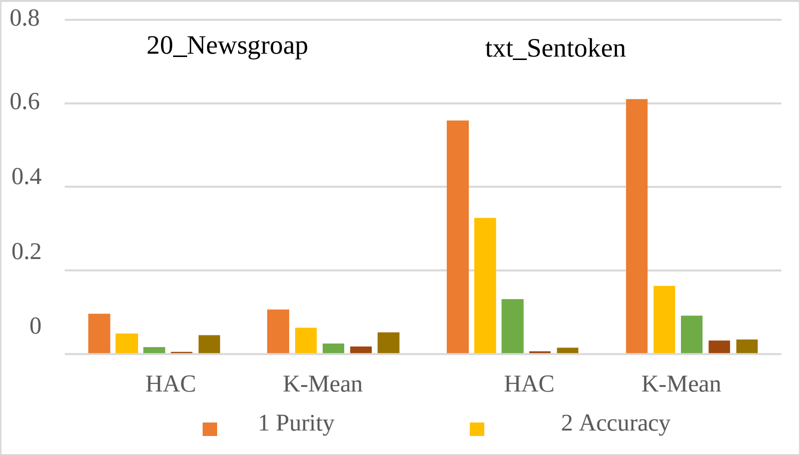


Fig. 5: External Performance of Proposed Approach

PROPOSED APPROACH VS. LITERATURE

As mentioned in related works, some other works have been done using the same dataset (100 movies from IMDB and WIKI). The result of silhouette evaluation shows that our proposed method when using Affinity Propagation with TF-IDF has the best result compared with K-means with TF-IDF in our approach, HAC with TF-IDF [12], and Glove word embedding with the DBSCAN clustering algorithm [13], and the K-means algorithm with Ward's Method [4] as shown in the table IV.

Table 4: Proposed approach compared with other approaches

100 Movies IMDB & WIKI			
Ref.	Approach	Sih.	No of clusters
Proposed approach	Affinity propagation with TFIDF	0.0292	5
	K-means	0.0234	5
[13]	Word embedding with DBSCAN clustering algorithm	0.005	17
[4]	K-means algorithm and Ward's Method	0.0258368	5
[12]	Hac	0.0252	5

DISCUSSION

The findings of three distinct datasets using the silhouette metric for affinity propagation and K-means methods are compared in Table I. The table demonstrates that the Affinity propagation technique produced the best results with the (100 IMDB & WIKI movies) dataset. The K-means approach yielded the best results for the (Nltk_Brown) dataset.

Furthermore, the affinity propagation technique produces the poorest results when used on the txt_Sentoken dataset. In general, affinity propagation produced better results than K-means on (100 IMDB & WIKI) movie datasets.

Table II concludes the output of the two datasets (txt_Sentoken and Nltk_Brwon) using some external metrics (Purity, F1-measure, Accuracy, and V_measure) which have been applied in both Table II is a summary of what was found in the two datasets (txt_Stoken and Nlkt_Brwon) using multiple external measures (purity, accuracy, f1_measure, and V_measure) that were used in both methods (Affinity propagation and K-Mean). The (Nltk_Brwon) dataset gives the best results when the purity metric is used with affinity propagation and the K-means method. Nevertheless, when employing the (V_measure) with the Nltk_Brwon dataset, the affinity propagation technique produces the worst results.

Table III shows which approach runs quicker, and where the time rating using the affinity propagation technique is highest across all datasets. As a result, while utilizing a limited dataset, the Affinity Propagation approach outperforms K-mean.

CONCLUSION

The use of a semantic clustering method that is both effective and efficient has been suggested and applied to movie datasets taken from sites like Wikipedia and IMDB. The main goal was to cluster these movies depending on synopsis. Both the affinity propagation and K-means clustering techniques are utilized, and the results are presented in a way that makes direct comparisons between them easy. More evaluation metrics, both internal and external, are applied to two additional datasets. According to an internal assessment using the silhouette metric, the best results using affinity propagation were achieved when the algorithm was applied to the 100 movies Wiki & IMDB datasets, while the poorest results were obtained when the technique was applied to the txt_Sentoken dataset. Regarding the external measures, the best result was achieved by applying Affinity propagation and K-means with the purity metric to the Nltk_Brown dataset. The lowest result was achieved when applying the Affinity method to the txt_Sentoken dataset with the V_measure metric. In addition, the execution times of both methods have been disclosed. In all states, the affinity propagation algorithm has obtained the best results. In compression with other approached our proposed approach when using Affinity propagation with TFIDF gain highest number of Silhouette Score 0.0292, when using 100 Movies IMDB & WIKI dataset. In conclusion, the proposed approach illustrates a comparison between the two methods. The outcome established by both internal and external evaluations is same. In some circumstances, the affinity propagation technique delivered the best results. This is due to the fact that Affinity propagation performs better with smaller datasets.

REFERENCES

K. Jacksi and S. M. Abass, “Development history of the world wide web,” Int. J. Sci. Technol. Res, vol. 8, no. 9, pp. 75–79, 2019.

D. Q. Zeebaree, H. Haron, A. M. Abdulazeez, and S. R. Zeebaree, “Combination of K-means clustering with Genetic Algorithm: A review,” International Journal of Applied Engineering Research, vol. 12, no. 24, pp. 14238–14245, 2017.

K. Jacksi, S. R. M. Zeebaree, and N. Dimililer, “LOD Explorer: Presenting the Web of Data,” International Journal of Advanced Computer Science and Applications, vol. 9, no. 1, 2018.

N. M. Salih and K. Jacksi, “Semantic Document Clustering using K-means algorithm and Ward’s Method,” in 2020 International Conference on Advanced Science and Engineering (ICOASE), 2020, pp. 1–6.

M. B. Revanasiddappa, B. S. Harish, and S. v Aruna Kumar, “Clustering text documents using kernel possibilistic C-means,” in Proceedings of International Conference on Cognition and Recognition, 2018, pp. 127–134.

M. Allahyari et al., “A brief survey of text mining: Classification, clustering and extraction techniques,” arXiv preprint arXiv:1707.02919, 2017.

D. Chandrasekaran and V. Mago, “Evolution of semantic similarity—a survey,” ACM Computing Surveys (CSUR), vol. 54, no. 2, pp. 1–37, 2021.

R. Ibrahim, S. Zeebaree, and K. Jacksi, “Survey on semantic similarity based on document clustering,” Adv. sci. technol. eng. syst. j, vol. 4, no. 5, pp. 115–122, 2019.

N. M. Salih and K. Jacksi, “State of the art document clustering algorithms based on semantic similarity,” J. Inform, vol. 14, no. 2, pp. 58–75, 2020.

P. Bafna, D. Pramod, and A. Vaidya, “Document clustering: TF-IDF approach,” in 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), 2016, pp. 61–66.

R. Guan, X. Shi, M. Marchese, C. Yang, and Y. Liang, “Text clustering with seeds affinity propagation,” IEEE Transactions on Knowledge and Data Engineering, vol. 23, no. 4, pp. 627–637, 2010.

K. Jacksi, R. K. Ibrahim, S. R. M. Zeebaree, R. R. Zebari, and M. A. M. Sadeeq, “Clustering documents based on semantic similarity using HAC and K-mean algorithms,” in 2020 International Conference on Advanced Science and Engineering (ICOASE), 2020, pp. 205–210.

S. M. Mohammed, K. Jacksi, and S. R. M. Zeebaree, “Glove word embedding and DBSCAN algorithms for semantic document clustering,” in 2020 International Conference on Advanced Science and Engineering (ICOASE), 2020, pp. 1–6.

B. K. Triwijoyo and K. Kartarina, “Analysis of Document Clustering based on Cosine Similarity and K-Main Algorithms,” Journal of Information Systems and Informatics, vol. 1, no. 2, pp. 164–177, 2019.

R. N. Rathi and A. Mustafi, “The importance of Term Weighting in semantic understanding of text: A review of techniques,” Multimedia Tools and Applications, pp. 1–23, 2022.

A. M. Serdah and W. M. Ashour, “Clustering large-scale data based on modified affinity propagation algorithm,” Journal of Artificial Intelligence and Soft Computing Research, vol. 6, no. 1, pp. 23–33, 2016.

D. Dueck and B. J. Frey, “Non-metric affinity propagation for unsupervised image categorization,” in 2007 IEEE 11th International Conference on Computer Vision, 2007, pp. 1–8.

K. P. Sinaga and M.-S. Yang, “Unsupervised K-means clustering algorithm,” IEEE access, vol. 8, pp. 80716–80727, 2020.

P. J. Rousseeuw, “Silhouettes: a graphical aid to the interpretation and validation of cluster analysis,” J Comput Appl Math, vol. 20, pp. 53–65, 1987.

C. D. ; R. P. ; S. H. Manning, “Chapter 1: Boolean Retrieval,” in Introduction to information retrieval,

P. Huilgol, “Accuracy vs. F1-score,” medium. com/analytics-vidhya/accuracy-vs-f1-score-6258237beca2, 2019.

S. Morbieu, “Accuracy: from classification to clustering evaluation (2019),” URL https://smorbieu. gitlab. io/accuracy-from-classification-to-clustering-evaluation/#:~: text= Computing% 20accuracy% 20for% 20clustering% 20can, the% 20accuracy% 20for% 20clustering% 20results.