International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 1
DOI: https://doi.org/10.17762/ijritcc.v11i1.5982
***Article Received**: 15 November 2022 **Revised**: 16 December 2022 **Accepted**: 28 December 2022*

An Enhanced Expectation Maximization Text Document Clustering Algorithm for E-Content Analysis

Thangaraj M[1], Ponmani K[2]
[1]Professor & Head
Department of Computer Science
Madurai Kamaraj University
Madurai, India
thangarajmku@yahoo.com
[2]Research Scholar
Department of Computer Science
Bharathiar University
Coimbatore, India
ponmanidurai@gmail.com

Abstract—Nowadays, there are many types of digital materials that can be used in the classroom. Students and scholars are migrating from textbooks to digital study materials because textbooks are too large and expensive. Teachers and college students can use and modify the materials that are freely available or with some constraints for their learning and teaching. E-content can be designed, evolved, utilized, re-used, and distributed electronically from anywhere at anytime. Because of the flexibility of time, place, and speed of learning, e-content is becoming extremely popular. It can be readily and instantly shared and communicated with an infinite number of clients all across the globe. Document clustering is most commonly used to group documents that are related to a specific topic. Text document clustering can be used to group a collection of documents regarding the information they include and to deliver search results when a user searches the internet. In this paper mainly focuses on text document clustering to cope with massive collection of E-Content documents. Enhanced Expectation Maximization Text Document Clustering (EEMTDC) clustering algorithm was proposed and compared with Expectation Maximization (EM) clustering, K-Means clustering, and Hierarchical clustering (HC) algorithms. The experiment shows that the performance of proposed EEMTDC algorithm produces greater clustering accuracy than existing clustering algorithms.

Keywords- Text Document Clustering,;K-Means; Hierarchical clustering; EM clustering; EEMTDC.

I. INTRODUCTION

Clustering can help with information retrieval, topic extraction, document structure, and enhance browsing. It is defined as "partitioning a set of data points into a set of groups that are as comparable as feasible." As a result, the purpose is to categorize text documents into groups or clusters based on their similarity, with texts inside a cluster being more similar than texts across clusters. The approaches may be used for various text granularities, including document, paragraph, sentence, and word level [5]. Clustering is the division of a group of data objects into several clusters, with objects in every cluster maintaining a highly significant relationship while being significantly distinct from items in all the other clusters.

A cluster is a group of data components that are equivalent to each other and may indeed be treated as a single entity. Because it involves recognizing a pattern in a set of unlabeled data, clustering is a key unsupervised learning strategy. Text document clustering is a method of separating a group of documents into precisely defined categories based on content similarity. The unstructured format is handled by the set of clusters that include it. It is a popular tool for exploring, organizing, extracting, summarizing, and retrieving vast volumes of text [3]. Initially, document clustering was used to improve the effectiveness of an information retrieval system.

Document clustering is an effective method of locating a document's nearest neighbor within a document collection. Currently, the clustering method is used to explore a number of documents and to normalize the search engine results delivered in response to a search query. To extract significant characteristics and classify them in a meaningful way, text document clustering is performed [2]. Text mining usually denoted in text mining systems as high-dimensional documents with complex semantics. Document clustering is commonly used for automatic topic extraction, document structure, and information retrieval. Despite the fact that much research has been done in the field of text clustering [7], more ideal

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 1
DOI: https://doi.org/10.17762/ijritcc.v11i1.5982
***Article Received**: 15 November 2022 **Revised**: 16 December 2022 **Accepted**: 28 December 2022*

approaches are required to increase the quality of the text document clustering process.

Linked documents tend to be quite related than unrelated texts, according to the text document clustering theory. Without any specified training or taxonomies, this is an automatic method of grouping relevant documents into a single category depending on the content of the document. Topic extraction, quick information retrieval or filtering, and automated document arrangement are some of the uses of text document clustering [1]. Text document clustering is employed for a range of tasks, including grouping similar contents from news, comments, and tweets, consumer analysis, and detecting important insights from papers [5]. There are three types of text document clustering algorithms: hierarchical, agglomerative, and flat. Aside from the categories described above, there are a variety of other techniques available, including distribution models, density models, subspace models, graph and signed graph models, and neural models. The sole difference between any algorithmic models for clustering in the article is the concept of what makes a cluster and the most effective manner of identifying clusters [9].

The first stage in the text document clustering process is parsing, which transforms text documents into smaller units (words and phrases) known as tokenization. Bag of words and N-gram are two tokenization methods that are commonly employed [4]. The next phase is stemming and lemmatization, which involves reducing inflected words to a single term. Finally, stop words, punctuation, and term frequencies are removed from the document. Finally, depending on the characteristics, clusters and a variety of papers were created [6]. Various performance criteria, including as cluster purity, recall, accuracy, and so on, are used to assess the efficiency of the cluster models developed. The only difference between text document clustering and classification is that the former is performed unsupervised, while the latter is performed supervised.

The organization of this paper as follows: section 2 includes the existing research work which are related to text document clustering and algorithms. Section 3 contains the proposed methodology for clustering the text documents. Section 4 achieved the results with cluster evaluation metrics for four different datasets. Finally, conclusion is given in section 5

II. LITERATURE REVIEW

Andrea Tagarelli and George Karypis [5] offer a segment-based document clustering method in which documents are clustered using segment-set clustering. This method assisted in discovering each document's many subjects and grouping documents into different clusters based on their topics. Florian Beil et al. [12] proposed the Frequent Term-based Clustering Algorithm and the Hierarchical Frequent Term-based Clustering Algorithm. The FTC is non-overlapping and capable of dealing with high bandwidth dimensionality, huge databases, and cluster descriptions. The descriptive characterization for clusters was built using a frequently used phrase set. HFTC created comprehensible and simple hierarchical clusters. This approach also detected intersecting clusters. Charu. C Aggarwal, et al. [13] investigated text clustering techniques in depth. The similarity between text items was assessed using a similarity measure in the distance-based clustering approach. By expanding the text representation, it performed better in clustering text of short parts. Single Linkage Clustering, Group-Average Linkage Clustering, and Complete Linkage Clustering are the three types of agglomerative and hierarchical clustering algorithms. A natural tree-like structure was created, which was beneficial for the search process. Single, group, and full connections were created from document groups. In a continuous scan, the strategy improved retrieval accuracy.

Distance-based K-medoid clustering algorithm, K-means clustering algorithm, Crisp K-means Algorithm, Fuzzy K-means Algorithm, Online Spherical K-means algorithm (OSKM), and Spherical K-means algorithm (SKPM) were used to classify partitioning algorithms [15]. It proved to be effective in the building of object-based clusters. The ideal selection of relevant papers was selected via K-medoid. To achieve convergence, a high number of iterations were necessary. The convergence of K-means needed a relatively minimal number of iterations. Adaptive text stream clustering was provided by OSKM. In high-dimensional data, SKPM maximized mean cosine similarity. Tao Liu et al. [14] developed an expectation-maximization technique for feature subset selection from distinct clusters, which involves determining the minimal message length and estimating the selected feature relevance. The supervised learning approach is based on a predetermined threshold value, and the possibility of term relevance can be simplified by calculating the score relevance for each term.

Text Clustering with Feature Selection (TCFS) approaches were presented by Yanjun Li in 2003. A supervised feature selection approach, such as CHIR, was integrated into the text clustering process. Even as the TCFS algorithm converged, we achieved an informative feature subset as well as a good clustering result. Wen Zhang and colleagues (2010) investigated the Minimum Spanning Tree Algorithm, which was used to create document clusters using three different kinds of similarity measurements. Each document is seen as a node in the network, connecting all of the papers with the greatest amount of document pair similarity.

Jo Taeho [11] was proposed the Divisive Clustering Algorithm. It begins with a single cluster and divides it into multiple clusters. The number of clusters was used as a criterion for stopping. The division was likewise ended

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 1
DOI: https://doi.org/10.17762/ijritcc.v11i1.5982
***Article Received**: 15 November 2022 **Revised**: 16 December 2022 **Accepted**: 28 December 2022*

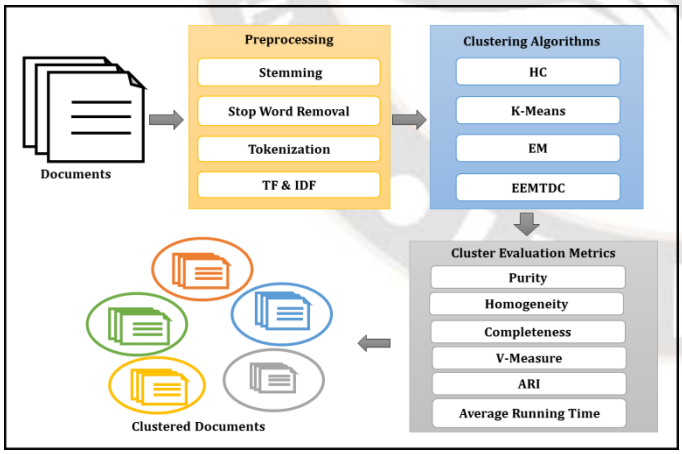
whenever the threshold was attained. The Single Pass Algorithm was both quick and easy to use. It starts with a single cluster and arranges the subsequent data into a new or existing cluster based on the cluster's starting components. The Growing Algorithm started with several skeletons, as there were individual things. From its own example, each cluster steadily extended its diameter from zero. The important elements of this method were the variables, the clustering stage, and the beginning stage.

Kusum Kumari, et al. [17] used the Artificial Bee Colony Algorithm, a population-based algorithm that mimics honeybees' sophisticated foraging behavior and was demonstrated to be successful in handling a variety of search and optimization issues. The ABC method allowed for the magnificent exploration of multiple portions of the search space at the same time in order to find the best answer at a low cost of utilization. Two alternative local search models, chaotic and gradient local search, were merged with the gbest-guided search formula to improve the ABC algorithm's manipulation capacity.

III. PROPOSED METHODOLOGY

A. Document Collection

The documents are collected from online based on the query. The real time dataset i.e. E-Content dataset contains 1500 documents, which is related to the academic field. It has computer science, engineering, medical, statistics, science related documents.



B. Preprocessing

The clustering accuracy is strongly influenced by the preprocessing stage. The process of transforming a number of documents into a machine-readable mathematical data model. [4]. An mxn term-document matrix is the outcome of this preprocessing. The list of frequencies and occurrences of words in each text is used to create this matrix. Here n denotes the quantity of documents, and m denotes the quantity of unique words. In the preprocessing, utilize the following techniques: stemming, stop word removal, tokenization, and TF and IDF term weighting.

Stemming: Stemming is the process of eliminating affixes (prefixes and suffixes) from inflectional words to reduce them to the same root. For instance, section, dissect, and intersect all share the similar essence termed the feature [10]. If a term ends with ly, ed, and ing. These are need to be eliminate by using this process.

Stop word removal: The stop words are for, an, be, and other basic words are still much frequent and simple usable phrases as well find simple weighting [10]. To enhance the effectiveness of text document clustering, these terms should be deleted from the document.

Tokenization: It is a method of separating a sequence of text documents into words or phrases and eliminating meaningless streams. Each term or symbol is extracted between the starting and ending letter, and every word is referred to as a token [8]. However, identifying a "word" might be complicated. Tokenizers frequently use like simple strategies. Blank space characters such as punctuation characters, spaces, line breaks in the output list of tokens that might or might not contain empty space, and punctuation to differentiate tokens.

Term weighting: The frequency of each phrase in the text is used to provide a term weighting for each term or characteristic. In weighting procedures, the Term Frequency-Inverse Document Frequency (TF-IDF) is commonly employed [18]. As indicated in the following equation, each document is denoted as a vector of term weights.

$$d_i = (w_{i,1}, w_{i,2}, \ldots . w_{i,t}) \quad (1)$$

The following Equation is used to compute the term weight for the feature j in document i.

$$w_{i,j} = tf(i,j) \times idf(i,j) = tf(i,j) \times \log\left(\frac{n}{df(j)}\right) \quad (2)$$

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 1
DOI: https://doi.org/10.17762/ijritcc.v11i1.5982
***Article Received**: 15 November 2022 **Revised**: 16 December 2022 **Accepted**: 28 December 2022*

The weight of document i and phrase j is represented by $w_{i,j}$ in Equation (2). The frequencies of term j in document i are denoted by tf(i j), while the inverse document frequency is denoted by idf(i,j). The number n denotes the total number of documents in the data collection, whereas the value df(j) denotes the number of documents that include feature j.

C. Enhanced Expectation Maximization Text Document Clustering Algorithm (EEMTDC)

The standard EM technique belongs to a flat clustering algorithm subclass known as model-based clustering. Model-based clustering considers data produced by a model and then aims to retrieve the original model from the data. The model then goes on to explain clusters and data cluster membership. The EM algorithm is a K-Means method modification in which the model providing the information is a set of K centroids. It changes between an expectation phase, which corresponds to the reassignment, and a maximization phase, which corresponds to the recompilation of the model's parameters. The EM algorithm is an iterative method for calculating maximum likelihood. Furthermore, because it ends at the closest local maximum to the likelihood function's beginning point, this approach is sensitive to initialization. To solve this challenge, the suggested technique employs a novel EM algorithm based on Particle Swarm Optimization (PSO) to [16] determine the best estimate of the likelihood function's global maximum.

The proposed technique uses the PSO algorithm to find the permanence factor, which is close to the optimal answer when using global search, while also avoiding long calculation times. PSO clustering is disabled in this situation as soon as the maximum number of iterations is obtained. The EM algorithm learns the parameters of a Gaussian Mixed Model (GMM) for each particle in the swarm. The GMM model is important because it will enhance the development of the essential understanding of EM. The mixing weights, the centroid, the covariance matrix of a GMM, and the log-likelihood of this mixture model to describe the input data are all included in its architecture. To minimize the likelihood (or, more commonly, a log-likelihood), attempting to fix the optimal solution:

$$\max_{\Theta} \log P(X|\Theta) = \max_{\Theta} \log \left(\prod_i P(x_i|\Theta)\right) = \max_{\Theta} \sum_i \log \left(P(x_i|\Theta)\right) \quad (3)$$

A mixture of Gaussian is denoted as $(x_i|\Theta)$. The particle with the highest log-likelihood is chosen after learning the EM method, and its GMM is deemed the strongest model for clustering the input data set.

Algorithm 3.1 EEMTDC Algorithm

Input: Number of Documents D with unlabeled data

Output: Clustered Documents

//Initialization the EM algorithm

Step 1: Initialize $\Theta_0^{(2)}, T, t = 0$

Step 2: Each time, to use a swarm of particles to initiate the EM process, with one particle representing a GMM from the swarm.

Step 3: When the EM process converges, the GMM with the largest log-likelihood function is chosen as the best model for the input data set.

// E-Step

Step 4: Re-estimates the expectations based on the prior iteration Θ

$$P(c_i|d_j) = \frac{P(c_i^{old}) P(c_i|d_j)}{\sum_{i=1}^{k} P(c_i^{old}) P(c_i|d_j)}$$

$$P(c_i)^{new} = \frac{1}{N}\sum_{j=1}^{N} P(c_i|d_j)$$

Step 5: Do until the stopping criterion is meet

For $\vartheta = 1,2,\ldots..N$

t=t+1

//M-Step

Step 6: To determine the model parameters, use the formula to maximise the likelihood of the data

$$\mu_i = \frac{\sum_{j=1}^{N} P(c_i|d_j) d_j}{\sum_{j=1}^{N} (c_i|d_j)}$$

$$\sum_i = \frac{\sum_{j=1}^{N} P(c_i|d_j)(d_j - \mu_i)(d_j - \mu_i)^T}{\sum_{j=1}^{N} (c_i|d_j)}$$

Setp 7: To save the final GMM parameters as well as the overall log-likelihood of the selected features in the data set in the relevant particle after convergence

Step 8: For each particle in the swarm, repeat steps 2–7, and choose the optimal GMM stored in the particle with the highest log-likelihood of the data

Step 9: Determine null values using the strongest GMM

Step 10: Evaluate E-Step

Step 11: End for ϑ

Step 12: Merge the value of E- step and M-step

Step 13: Stop the process

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 1
DOI: https://doi.org/10.17762/ijritcc.v11i1.5982
***Article Received**: 15 November 2022 **Revised**: 16 December 2022 **Accepted**: 28 December 2022*

Initialize the EM algorithm with the number of documents in the first stage. Then, in step 2, utilise the PSO particle method to initialise the EM algorithm each time with one particle from the swarm, which indicates a GMM. In step 3, the GMM corresponding to the greatest log-likelihood function is chosen as the best model for the input dataset when the EM method has reached convergence. Step 4 is the expectation step, which re-estimates the expectations based on the prior iteration. Step 5 involves giving conditions until the stop requirement is met. To determine the model parameters, use the formula in the maximisation phase (step 6) to maximise the likelihood of the data. In step 7, to save the final GMM parameters as well as the overall log-likelihood of the selected features in the data set in the relevant particle after convergence. Then, for each particle in the swarm, repeat steps 2–7, and choose the optimal GMM stored in the particle with the highest log-likelihood of the data. In step 9, estimate null values using the strongest GMM. Then, in step 10, evaluate E-Step. Finally, the E-step and M-step values were merged.

D. Cluster Evaluation Metrics

It is a technique for validating the efficiency of the outcome of clustering algorithm after they have been clustered. External and internal validating criteria are two types of validating criteria that can be used. To evaluate the quality of text document clustering algorithms, we employed external criteria like purity, homogeneity, completeness, V-measure, and adjusted rand index (ARI), are used to assess clustering performance.

Purity. The purity of the clusters is determined by whether they include documents from a specific category. Purity values vary from 0 to 1, with 1 being the purity value of a perfect cluster.

The following equation shows calculates the purity score to measure the count of properly assigned documents by N.

$$\text{Purity}(P, C) = \frac{1}{N} \sum_k \max_j |\rho_k \cap c_n| \qquad (4)$$

Where $\{ P = \{\rho_1, \rho_2, \dots \dots \rho_n\}$ is the group of clusters and $C = c_1, c_2, \dots \dots, c_n$is the group of classes.

Homogeneity and Completeness. The both have a range of 0 to 1. Homogeneity equals 1 if all of the variables in a cluster are in the same class. Completeness equals 1 if all of the variables in a given class belong to the same cluster. Equation 5 and Equation 6 denote homogeneity and completeness.

$$h = 1 - \frac{H(C|D)}{H(C)} \qquad (5)$$

$$c = 1 - \frac{H(D|C)}{H(C)} \qquad (6)$$

The following equation denote the conditional entropy of the classes H(C|D):

$$(C|D) = \sum_{c=1}^{|C|} \sum_{D=1}^{|D|} \frac{n_{c,d}}{n} \times \log\left(\frac{n_{c,d}}{n_d}\right) \qquad (7)$$

To define the entropy of the classes H(C) using the following equation :

$$H(C) = \sum_{c=1}^{|C|} \frac{n_c}{n} \times \log \frac{n_c}{n} \qquad (.8)$$

The total number of variables is given by n. The number of variables in class c is represents as n_c and cluster d is represents as n_d. The number of variables from class c allocated to cluster d is given by $n_{c,d}$.

V-Measure. The symmetrical mean of homogeneity and completeness can be defined as follows:

E. $$V = 2 \times \frac{h \times c}{h + c} \qquad (9)$$

ARI. The rand index (RI) assesses the similarities between document data clusters by examining all sets of observations and identifying groupings that are assigned to the same or distinct clusters in the predicted and actual clusters. The ARI score properly accounts for the potential of unprocessed RI and is stated as follows:

$$\text{ARI} = \frac{(\text{RI} - \text{Expected RI})}{(\max(\text{RI}) - \text{Expected RI})} (10)$$

The ARI is a number that varies from 0 to 1. If the clusters are extremely similar, the ARI value is 1.

IV. Result and Discussion

A. Dataset Description

In this study, we examined the real time dataset (E-Content) compared with benchmark datasets like 20 newsgroup, Reuters, BBC Sport with varied numbers of documents, and clusters. The real time dataset i.e. E-Content dataset contains 1500 documents, which is related to the academic field. It has computer science, engineering, medical, statistics, science related documents.

B. Results and Discussions

The performance of the clustering algorithms was investigated using several metrics, including purity, homogeneity, completeness, V-measure, ARI, and average running time in this work, which used traditional HC, K-Means, and EM for text document clustering algorithms, and one proposed EEMTDC algorithm using four datasets was compared to these results.

Table 4.1 displays the mean score values of the evaluation criteria for all clustering algorithms using four datasets,

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 1
DOI: https://doi.org/10.17762/ijritcc.v11i1.5982
***Article Received**: 15 November 2022 **Revised**: 16 December 2022 **Accepted**: 28 December 2022*

including 20 newsgroups, Reuters, BBC Sport, and E-content. Figure 4.1 shows the measures used to monitor the effectiveness of each algorithm (HC, K-means, EM, and EEMTDC). The proposed EEMTDC method has the greatest purity mean scores (0.724, 0.839, 0.873, and 0.898), while the HC algorithm has the fewest purity mean scores (0.636, 0.692, 0.734, and 0.833).

TABLE 4.1 PERFORMANCE EVALUATION OF PURITY MEASURE

Dataset	HC	K-Means	EM	EEMTDC
20 Newsgroup	0.636	0.665	0.695	**0.724**
Reuters	0.692	0.74	0.775	**0.829**
BBC sport	0.734	0.786	0.839	**0.873**
E-Content	0.833	0.848	0.876	**0.898**

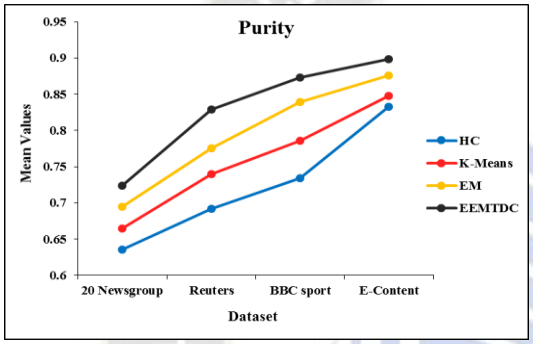


Figure 4.1 Performance Evaluation of Purity Measure

Table 4.2 and figure 4.2 provides the mean score values of the evaluation criteria for all clustering algorithms using four datasets. Here, the proposed EEMTDC algorithm obtain the greatest homogeneity mean scores (0.249, 0.36, 0.541, and 0.793) and HC has the minimum homogeneity mean scores (0.152, 0.251, 0.259, and 0.591) for four datasets respectively.

TABLE 4.2 PERFORMANCE EVALUATION OF HOMOGENEITY

Dataset	HC	K-Means	EM	EEMTDC
20 Newsgroup	0.152	0.181	0.208	**0.249**
Reuters	0.251	0.284	0.322	**0.36**
BBC sport	0.259	0.314	0.427	**0.541**
E-Content	0.591	0.637	0.733	**0.793**

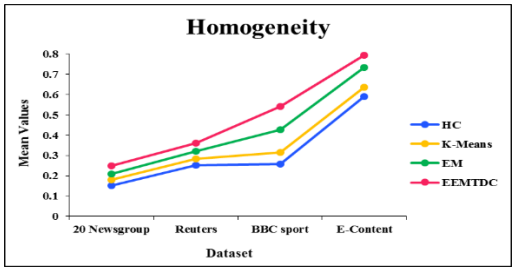


Figure 4.2 Performance Evaluation of Homogeneity

Table 4.3 and figure 4.3 depicts the mean score values of the evaluation criteria for all clustering algorithms using four datasets. Here, the proposed EEMTDC algorithm attain greatest completeness mean scores (0.253, 0.39, 0.556, and 0.797) and HC attain minimum score (0.137, 0.224, 0.297, and 0.605) for four datasets respectively.

TABLE 4.3. PERFORMANCE EVALUATION OF COMPLETENESS

Dataset	HC	K-Means	EM	EEMTDC
20 Newsgroup	0.137	0.185	0.212	**0.253**
Reuters	0.224	0.291	0.346	**0.39**
BBC sport	0.297	0.347	0.448	**0.556**
E-Content	0.605	0.65	0.735	**0.797**

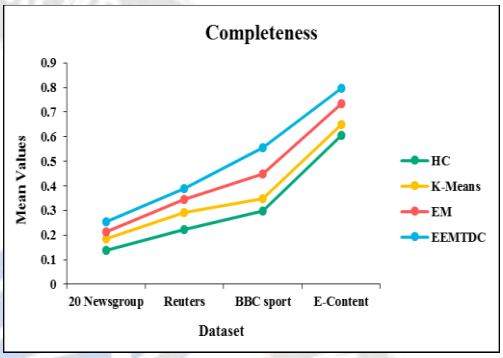


Figure 4.3 Performance Evaluation of Completeness

Table 4.4 and figure 4.4 illustrates the mean score values of the evaluation criteria for all clustering algorithms using four datasets. Here, the proposed EEMTDC algorithm achieved greatest v-measure mean score (0.151, 0.365, 0.548, and 0.643) and HC achieved the lowest v-measure mean score (0.054, 0.248, 0.305, and 0.41) for four datasets respectively.

TABLE 4.4 PERFORMANCE EVALUATION OF V-MEASURE

Dataset	HC	K-Means	EM	EEMTDC
20 Newsgroup	0.054	0.093	0.11	**0.151**
Reuters	0.248	0.292	0.329	**0.365**
BBC sport	0.305	0.349	0.437	**0.548**
E-Content	0.41	0.47	0.525	**0.643**

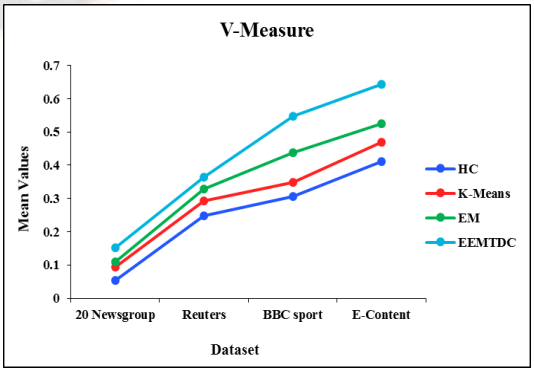


Figure 4.4 Performance Evaluation of V-Measure

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 1
DOI: https://doi.org/10.17762/ijritcc.v11i1.5982
***Article Received**: 15 November 2022 **Revised**: 16 December 2022 **Accepted**: 28 December 2022*

Table 4.5 and figure 4.5 illustrates the mean score values of the evaluation criteria for all clustering algorithms using four datasets. Here, the proposed EEMTDC algorithm produced the best ARI mean scores (0. 363, 0.598, 0.627, and 0.632) and HC produced the least mean scores (0.248, 0.321, 0.378, and 0.453) for four datasets respectively.

TABLE 4.5 PERFORMANCE EVALUATION OF ARI

Dataset	HC	K-Means	EM	EEMTDC
20 Newsgroup	0.248	0.263	0.318	**0.363**
Reuters	0.321	0.344	0.473	**0.598**
BBC sport	0.378	0.452	0.559	**0.627**
E-Content	0.453	0.492	0.536	**0.632**

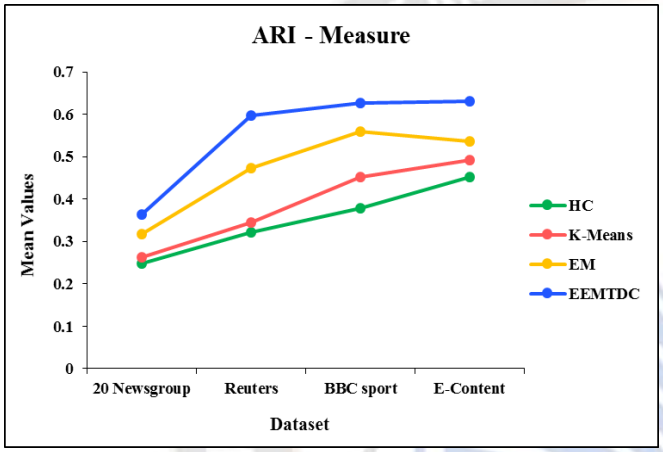


Figure 4.5 Performance Evaluation of ARI

Figure 4.6 shows the average of overall execution time of the existing clustering algorithms and proposed EEMTDC algorithm for each dataset. Here, proposed EEMTDC achieve less execution time than HC, K-Means, and EM algorithms. The proposed EEMTDC algorithm takes minimum running time comparing to the other algorithms for each dataset.

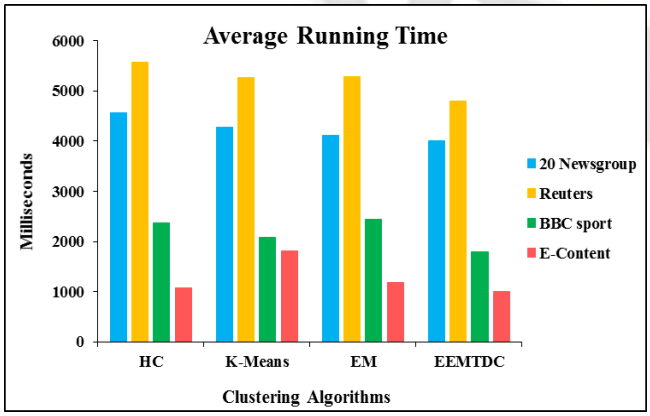


Figure 4.6 Average Running Time in Milliseconds

Finally, experiment results are evaluated the clustering algorithms for text document clustering with significant evaluation metrics clearly. Comparing all results, the proposed EEMTDC performs best for all the dataset and specifically for E-Content dataset. Based on the results, the proposed EEMTDC achieved the best performance than EM, K-means, and HC.

V. CONCLUSION

The text document clustering problem is now a hot topic among text mining and E-content analysis experts. The primary objective of the research is to evaluate efficient algorithms and find innovative solutions to problems in order to get the best results. In this research, examined the traditional clustering algorithms (HC, K-Means, EM) and proposed EEMTDC algorithm for text document clustering problem. To analyze the performance of the clustering algorithms using significant clustering evaluation metrics such as purity, homogeneity, completeness, V-measure, ARI, and average running time. From the comparison, the proposed EEMTDC outperformance well than other algorithms with greater accuracy and executes with minimum time. The documents are clustered in five groups like computer science, engineering, medical, statistics, and science using the proposed EEMTDC with best result. In future, the optimization algorithms will be applied to obtain optimal results for massive collection of text documents.

References

[1] Ramkumar, A.S.; Nethravathy, R. Text Document Clustering using K-means Algorithm. Int. Res. J. Eng. Technol. 2019, 6, 1164–1168.

[2] Jensi, R.; Wiselin , J. A Survey on Optimization Approaches to Text Document Clustering. Int. J. Comput. Sci. Appl. 2013, 3, 31–44.

[3] Selvaraj, S.; Choi, E. Survey of Swarm Intelligence Algorithms. In ICSIM ’20: Proceedings of the 3rd International Conference on

[4] Software Engineering and Information Management; Association for Computing Machinery: New York, NY, USA, 2020; pp. 69–73.

[5] Abualigah, Laith Mohammad, Ahamad Tajudin Khader, Mohammed Azmi Al-Betar, and Osama Ahmad Alomari. "Text feature selection with a robust weight scheme and dynamic dimension reduction to text document clustering." Expert Systems with Applications 84 (2017): 24-36.

[6] Abualigah, Laith Mohammad, Ahamad Tajudin Khader, and Essam Said Hanandeh. "A novel weighting scheme applied to improve the text document clustering techniques." In Innovative Computing, Optimization and Its Applications, pp. 305-320. Springer, Cham, 2018.

[7] Abasi, Ammar Kamal, Ahamad Tajudin Khader, Mohammed Azmi Al-Betar, Syibrah Naim, Sharif Naser Makhadmeh, and Zaid Abdi Alkareem Alyasseri. "A Text Feature Selection Technique based on Binary Multi-Verse Optimizer for Text Clustering." In 2019 IEEE Jordan

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 1
DOI: https://doi.org/10.17762/ijritcc.v11i1.5982
***Article Received**: 15 November 2022 **Revised**: 16 December 2022 **Accepted**: 28 December 2022*

International Joint Conference on Electrical Engineering and Information Technology (JEEIT), pp. 1-6. IEEE, 2019.

[8] Ailem, M., Role, F., & Nadif, M. (2015, October). Co-clustering document-term matrices by direct maximization of graph modularity. In Proceedings of the 24th ACM International on Conference on Information and Knowledge Management (pp. 1807-1810). ACM.

[9] Ailem, M., Role, F., & Nadif, M. (2017). Sparse poisson latent block model for document clustering. IEEE Transactions on Knowledge and Data Engineering, 29(7), 1563-1576

[10] Allahyari, Mehdi, Seyedamin Pouriyeh, Mehdi Assefi, Saied Safaei, Elizabeth D. Trippe, Juan B. Gutierrez, and Krys Kochut. "A brief survey of text mining: Classification, clustering and extraction techniques." arXiv preprint arXiv:1707.02919 (2017).

[11] Beil, Florian, Martin Ester, and Xiaowei Xu. "Frequent term-based text clustering." In Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 436-442. ACM, 2002.

[12] Foong, O. M., & Yong, S. P. (2016). Swarm LSA-PSO clustering model in text summarization. Int. J. Advance Soft Compu. Appl, 8(3).

[13] Hu, Jian, Lujun Fang, Yang Cao, Hua-Jun Zeng, Hua Li, Qiang Yang, and Zheng Chen. "Enhancing text clustering by leveraging Wikipedia semantics."

[14] Jensi, R., and Dr G. Wiselin Jiji. "A survey on optimization approaches to text document clustering." arXiv preprint arXiv:1401.2229 (2014).

[15] Jo, Taeho. "Text Clustering: Approaches." In Text Mining, pp. 203-224. Springer, Cham, 2019.

[16] Karol, Stuti, and Veenu Mangat. "Evaluation of text document clustering approach based on particle swarm optimization." Open Computer Science 3, no. 2 (2013): 69-90.

[17] De Vries, C.M. Document Clustering Algorithms, Representations and Evaluation for Information Retrieval. Ph.D. Thesis, Queensland University of Technology, Brisbane City, Australia, 2014.

[18] Zhou, C.; Gao, H.; Gao, L.; Zhang, W.G. Particle Swarm Optimization (PSO) Algorithm. Appl. Res. Comput.2003, 12, 7–11.