Journal of Scientometric Research, 2025; 14(1):319-330.
https://www.jscires.org

Research Article

Corpus Characteristics-Based Method to Centroids Number Determination for Clustering Text Documents

Inti Sandino Magallon-Juan-Qui[1,*], Jose Francisco Martinez-Trinidad[2,#], Darnes Vilarino-Ayala[1,#], Jesus Ariel Carrasco-Ochoa[2,#]

[1]FCC Department, Benemerita Universidad Autonoma de Puebla, Avenida San Claudio, Blvrd 14 Sur, Ciudad Universitaria, Puebla, MEXICO.
[2]Department of Computer Science, National Institute for Astrophysics, Optics and Electronics, Luis Enrique Erro 1, Sta. Ma. Tonanzintla, Puebla, MEXICO.

ABSTRACT
Clustering is fundamental for categorizing and exploring information, particularly in written texts. Traditional clustering algorithms such as K-Means generate clusters in which each document is assigned to a cluster based on a centroid or representative of the cluster. Nevertheless, it is common to find situations where a single centroid is not enough to represent a cluster. To solve this problem, some variants of the K-Means algorithm have been introduced by considering more than one centroid per cluster. However, determining the number of centroids per cluster is a challenge; the user must employ trial and error to determine a good value, which is time-consuming. This paper proposes a solution to this problem for Improved-FPAC (Fast Partitional Clustering Algorithm, one of the most recent text document cluster algorithms) by introducing a method to compute the parameter l (number of centroids per cluster) based on the characteristics of the corpus. Based on our experiments on different public standard data sets, the method proposed in this paper, allows Improved-FPAC to obtain better clustering quality than the default value suggested by Improved-FPAC's authors.

Keywords: K-Means, Categorizing, Text Clustering, Centroids.

Correspondence:
Inti Sandino Magallon-JuanQui
Faculty of Computer Science, Benemerita Universidad Autonoma de Puebla, Avenida San Claudio, Blvrd 14 Sur, Ciudad Universitaria, Puebla-72840, MEXICO.
Email: inti.magallon@viep.com.mx

Received: 25-07-2024;
Revised: 02-10-2024;
Accepted: 12-11-2024.

INTRODUCTION

Generating large volumes of textual information by large companies or organizations requires the ability to organize this information for decision-making, research, or knowledge extraction; otherwise, storing this information does not make sense. Finding the necessary information from a large amount of data is a complex but necessary task for the daily activities of most institutions, companies and government entities. One of the widely used techniques to alleviate this problem is text document clustering. Text document clustering allows us to separate similar text documents into different clusters.

We can find several research efforts on text document clustering in the literature. Recently, extensive studies have been carried out on this topic where different methods have been reported in the literature.[1-14] In these studies, K-Means is one of the most commonly used algorithms for document clustering due to its ease of implementation, direct parallelization and relatively low computational cost.[15] However, K-Means presents some difficulties, such as the handling of the high dimensionality of vectors used for representing documents, which considerably increases the algorithm's execution times; this has encouraged the development of new alternatives and variants of the algorithm. One of the recently developed algorithms makes use of the advantages of inverted lists for document retrieval, avoiding the comparison word by word of each document through a similarity function; this solution is presented in the FPAC algorithm,[16] improving the execution time considerably, without decreasing the quality of the clusters. To improve the cluster quality, Improved-FPAC[17] proposes defining multiple centroids for each cluster, which allows to increase the representativeness, without sacrificing too much the execution time.

In Bejos S, *et al*,[17] the authors experimented using different values for the number of centroids (parameter l) ranging from 5 to 30. They suggested using a fixed value l=10 for every corpus due to observing a decline in clustering quality beyond this threshold value in several corpora. However, we hypothesize that the value of this parameter should be defined according to the characteristics of the corpus to be clustered. For this reason, this paper proposes a corpus characteristics-based method for determining the number of centroids per cluster for Improved-FPAC. Our experimental analysis revealed a discernible correlation between the characteristics of a corpus (number of documents, number

DOI: 10.5530/jscires.20251455

of words, number of classes or clusters) and the l value that produced the best clustering quality for this corpus. According to our experiments on public standard corpora, the method proposed in this paper allows Improved-FPAC to enhance its clustering quality results. The contribution of this paper is a method for determining the number of centroids per cluster (a parameter of the Improved-FPAC algorithm denoted as l) based on some of the characteristics of the corpus to cluster, such as the number of documents, number of words and number of classes or clusters. The proposed method facilitates the determination of a good value for l instead of adopting a uniform fixed value for all corpora, as suggested in Improved-FPAC. The proposed method avoids executing the Improved-FPAC algorithm several times to find a good value for the parameter l (the number of centroids) that provides good clustering results.

The rest of the paper is organized as follows. In the Second Section, we present the related work. In Section 3, our proposed method is introduced. The experimental study appears in Section 4. Finally, Section 5 discusses our conclusions and future work.

Related Work

Currently, document clustering is widely studied and we can find several algorithms reported in the literature.[16-23] Among these algorithms, one that has reported good results is Improved-FPAC.[17] As commented in the Introduction Section, this algorithm is based on k-means and follows an information retrieval approach to build the clusters. Since in this paper we introduce a method for determining the number of centroids per cluster to be used in Improved-FPAC, in the rest of the section, we will review the main ideas of this algorithm.

Improved-FPAC consists of 6 steps:

- Selection of the initial centroids (one per cluster).
- Construction of the initial clusters as in FPAC.
- Selection of multiple centroids per cluster.
- Non-Centroids Assignment with multiple centroids per cluster.
- Evaluation of the stop condition.

Selection of initial centroids

The process of selecting the initial centroids (k centroids) in the clustering algorithm involves the following steps: Initially, one centroid (c_1) is randomly chosen from all the documents in the corpus being analyzed. To ensure diversity among the centroids, the next centroid (c_2) is selected randomly from the documents that have not been included in the previously retrieved list of c_1. This process is repeated for each subsequent centroid, where each new centroid (c_3 and beyond) is selected randomly from the remaining documents that have not been part of the retrieved lists of the previously selected centroids (c_1, c_2, ...). The process continues until k initial cluster centroids have been chosen. This approach helps to achieve a diverse and representative set of initial centroids for the subsequent clustering process.

Construction of the initial clusters as in FPAC

To build the initial clusters, a non-centroid document d is assigned to a cluster by utilizing the initial centroids c_1, ..., c_k as queries to retrieve k ranked lists (r) of documents r_1, ..., r_k. Subsequently, if the document d is recovered solely in one ranked list, r_j, it is included in the cluster corresponding to centroid c_j. However, if the document d is retrieved in multiple ranked lists, it is assigned to the cluster where the normalized similarity score is maximized. If no centroid retrieves the document d, it is randomly assigned to any of the k clusters. This procedure facilitates the initial assignment of documents to clusters based on their similarity to the centroids, thus initializing the clustering process.

Selection of multiple centroids

After building the initial k clusters using Improved-FPAC, the algorithm proceeds to compute l centroids per cluster, where l is calculated whit Equation 2. To achieve this, each cluster C_i is divided into l disjoint subsets and these subsets are stored in l lists. Improved-FPAC iterates through the documents assigned to each cluster and distributes them into the disjoint subsets one by one until all documents within the cluster are distributed in the lists.

This partitioning allows each cluster to be covered by multiple subsets and within each subset; a centroid is computed as the average vector of the document vectors it contains. As a result, each cluster C_i obtains l centroids, providing a more granular representation of the cluster's characteristics.

In summary, Improved-FPAC dynamically determines l centroids for each cluster C_i by dividing the cluster's documents into l disjoint subsets and calculating centroids based on the average vectors of each subset. This approach contributes to the algorithm's ability to achieve more refined clustering results.

Non-Centroids Assignment with multiple centroids

For assigning a non-centroid document d to a cluster, Improved-FPAC uses each of the l centroids as a query to retrieve a ranked list r for the cluster C_i. The normalized retrieval score is then computed for each centroid in each cluster C_i. Document d is assigned to the cluster C_i where the sum of the normalized retrieval scores is maximum. In the same way as in the construction of the initial clusters, the documents not retrieved by any centroid are randomly assigned to any of the k clusters.

Evaluation of the stop condition

The selection of multiple centroids and Non-Centroids Assignment with multiple centroids is repeated until the number of documents that change cluster between one iteration and

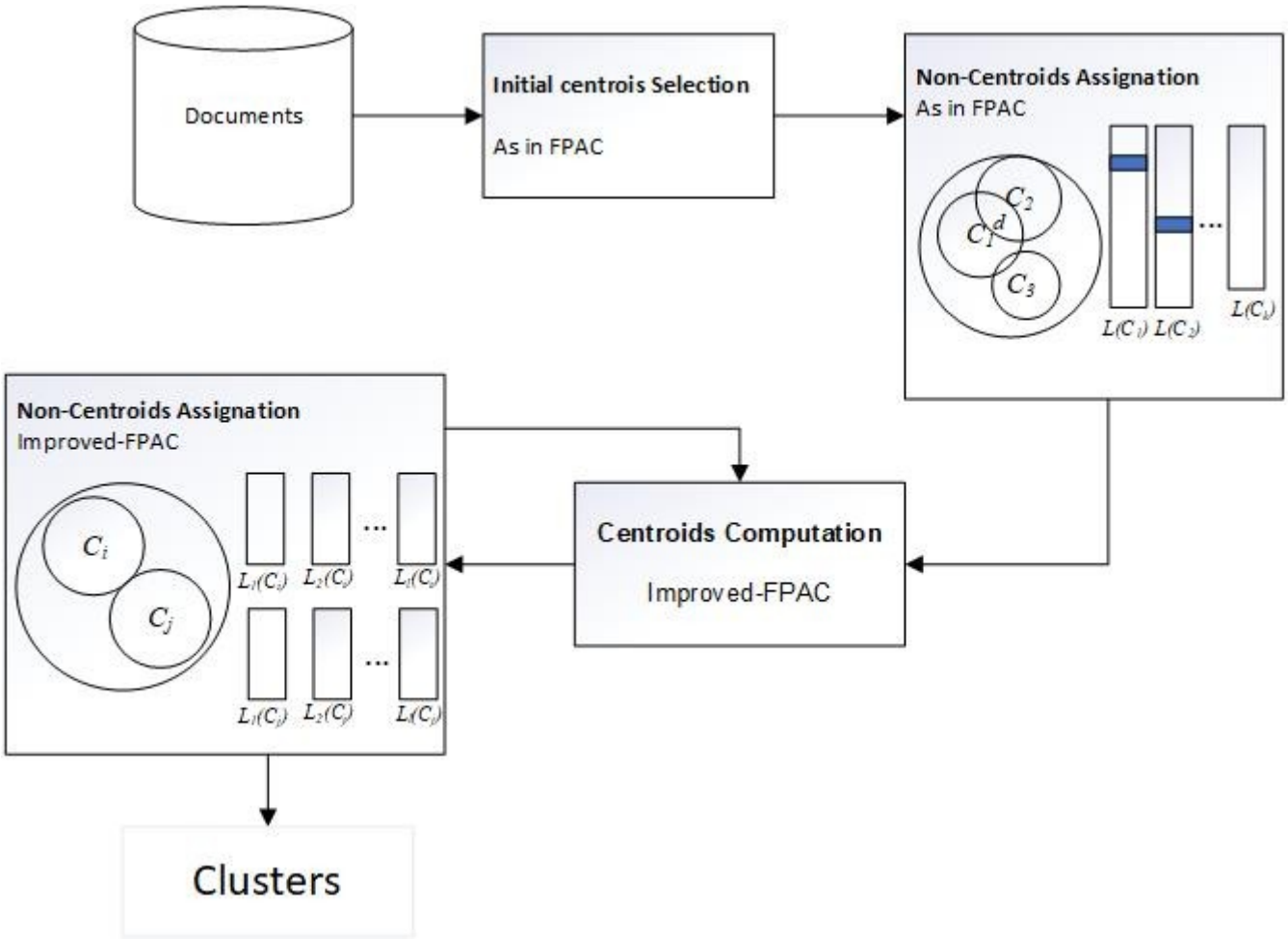

Figure 1: Improved-FPAC whit multiple centroids.

another is not greater than a percentage defined as the stop condition; in Bejos S, *et al.*[17] 10% was used. Below, Figure 1 illustrates the diagram of Improved-FPAC.

Although in Bejos S, *et al*,[17] the authors report that Improved-FPAC with ten centroids for each cluster obtains good results in their experiments. However, determining the value of this parameter is generally not easy; it forces the user to rely on trial and error, which is impractical. For this reason, in the following section, a method to determine the number of centroids for Improved-FPAC from the characteristics of the corpus to be clustered is introduced.

Our method

This section introduces a method, based on the corpus characteristics, to calculate the number of centroids per cluster for the Improved-FPAC algorithm (parameter *l*). This was accomplished by analyzing the results of the Improved-FPAC algorithm using different values of *l* across various corpora utilized in the experiments conducted by Bejos S, *et al.*[17] The algorithm was executed starting with a value of *l*=10 (as recommended in Bejos S, *et al),*[17] Then incrementally increased by 10 in each execution until reaching *l*=200. Higher values were avoided as they did not significantly improve clustering quality. This behavior was consistent across all the used corpora. The F-Score metric was used to evaluate clustering quality for each value of *l* across the various corpora.

Table 1 shows the characteristics of each corpus used, i.e., number of documents (*Docs*), number of words (*Words*) and number of classes (*Class*). Additionally, the last column presents the number of centroids per cluster (*l*) that yielded the best F-Score results for the corpus among 20 executions conducted with values ranging from 10 to 200. We chose the F-Score measure because it is commonly used for evaluating information retrieval and document classification algorithms, which are the basis of Improved-FPAC.

From the experiments performed with the Improved-FPAC algorithm, we observed that the more documents with different vocabulary in a cluster, the more centroids are required to represent the cluster. Note that the larger a cluster is, the more documents it will contain (and the number of terms will likely increase), thus more centroids will be required to represent the cluster.

Table 1: Best value of l and corpus characteristics.

Corpus	*Docs*	*Words*	*Class*	*l*
R52	9,100	18,989	52	40
R8	7,674	17,150	8	90
20 news groups	18,248	69,120	20	130
Webkb	4,168	7,657	4	160
Hitech	2,301	22,119	6	170
BBC News	2,225	20,749	5	180
Reviews	4,069	44,287	5	180
AGNews	127,599	41,337	4	190
Health Tweets	62,718	80,653	16	190
Webace20	3,900	11,606	20	90

On the other hand, the greater the number of clusters in which the documents are clustered, in general, the number of documents in each cluster decreases;[24] therefore, a smaller number of centroids will be required to represent the documents within a cluster.

From the above, we can infer that the number of documents and the vocabulary size (*Words*) are directly related to the number of centroids needed to represent the clusters. In contrast, the number of clusters (k) is inversely related to the number of centroids required to represent a cluster. Given that a larger k results in more clusters, each with fewer documents, fewer centroids will be needed to represent each cluster. Conversely, if k is small, there will be fewer clusters, but each will contain more documents, requiring more centroids to represent each cluster. Hence, according to the above discussion, we propose the following expression to relate these corpus characteristics:

$$x = \frac{\text{Docs+Words}}{k} \quad (1)$$

By plotting the value of Expression 1 against the value of l that produced the best results from the twenty executions when ranging l from 10 to 200 for each corpus (y-axis), we obtain a set of points on the plane (one for each corpus); using these points, we can find a trend line to calculate the l value for the improved-FPAC algorithm for an unknown corpus from its characteristics. See Figure 2, where x is the normalized value of Expression 1 and y is the number of centroids per cluster (l) that produced the best results for the corpus, among all the executions performed by varying the value of l between 10 and 200. The graph in Figure 2 represents the trend line for all corpora (each corpus corresponds to a point on the graph). Then, a logarithmic trend line was adjusted using regression employing the residual sum of squares.[25]

The following expression represents the trend line computed in Microsoft Excel:

$$l = 48 * ln(x) + 298 \quad (2)$$

This trend line will allow us to calculate the number of centroids per cluster to be used in the Improved-FPAC algorithm from the characteristics of the corpora.

As can be seen, in practice, applying our method requires evaluating the expression 7, 2 where x is as specified in expression 1. In Figure 3, we show the workflow when applying improved-FPAC using the proposed method to compute the number of centroids per cluster (parameter l).

It is necessary to mention that the value obtained from expression 2 must be truncated to an integer to define the value of l for each corpus. Additionally, if the value of l is less than 10, l is set to 10; if greater than 200, l is set to 200. In the next section, we show that applying our proposed method to determine a value for parameter l, which considers the corpus's characteristics, produces good clustering results, as evidenced in Tables 4 and 5.

EXPERIMENTS AND RESULTS

This section will show the experiments to assess the proposed method to calculate the number of centroids per cluster for the Improved-FPAC algorithm.

For the experiments, 20 corpora were used. The characteristics of each corpus are shown in Table 3. The 10 Dataset Classification, German, IMDB Movie Reviews, Medical Dataset, Ohsumed, two corpora built from Reuters-21578 and Text Classification on Email corpora were downloaded from the Kaggle website.[1] In addition, six corpora were generated from the ones used for developing our proposal and six were generated from the News Category corpus, also downloaded from the Kaggle website. Table 2 shows the details for building these corpora. The quality of the clusters was evaluated using Rand Index (RI), Recall, Precision, F-Score, Purity and Normalized Mutual Information (NMI), in the same way as in Bejos S, *et al.*[17] Below, we briefly explain the confusion matrix, which is fundamental for understanding RI, Recall and

[1]https://www.kaggle.com/datasets

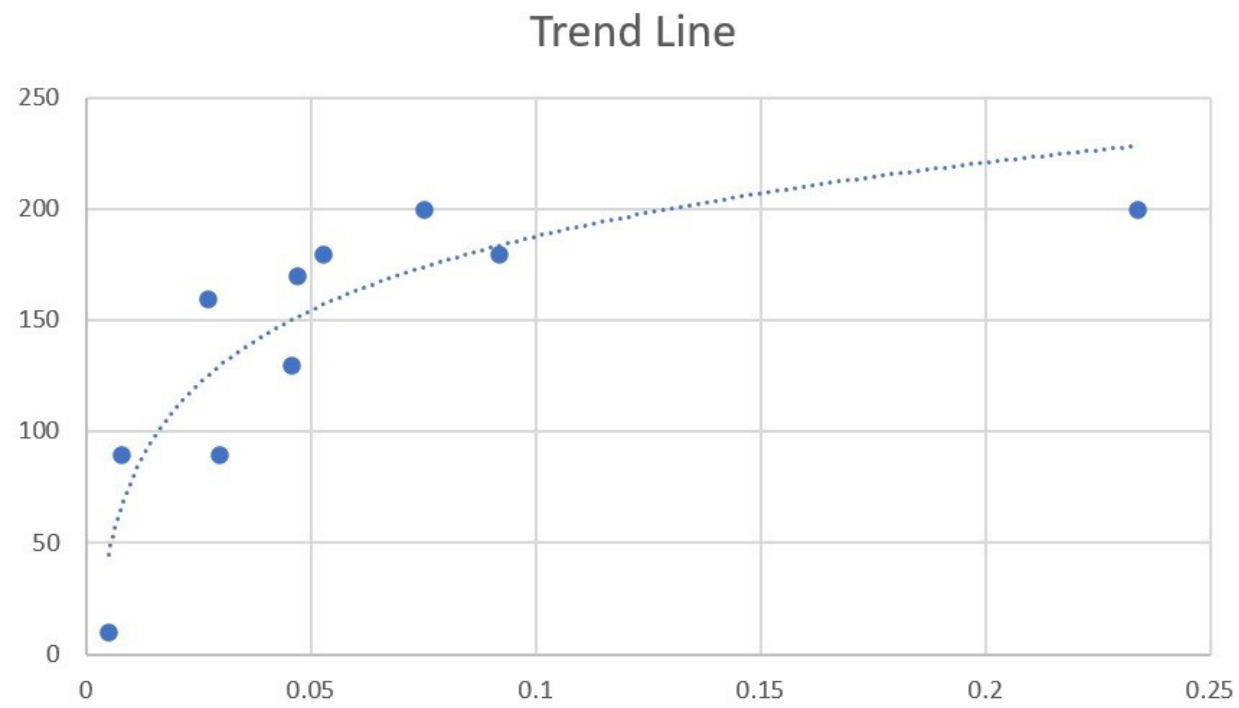

Figure 2: Trend line of the values of expression 1 (x-axis) and the number of centroids per cluster yielding the highest quality resulting when varying the values of *l* (y-axis).

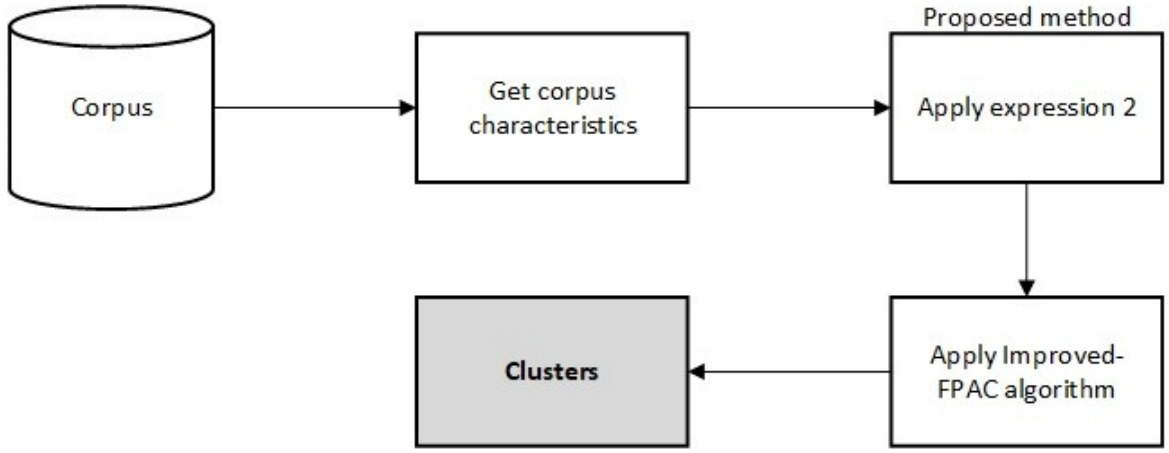

Figure 3: Workflow diagram when applying improved FPAC using the proposed method.

Precision. Subsequently, we provide detailed explanations of each of these measures.

The confusion matrix provides a tabular representation of clustering outcomes regarding an already known clustering (ground truth) in terms of True Positives (TP), False Positives (FP), True Negatives (TN) and False Negatives (FN).

True Positives (TP)

This is the number of pairs of documents that the clustering algorithm puts together in the same cluster and they are together in a cluster in the already known clustering.

False Positives (FP)

This is the number of pairs of documents that the clustering algorithms puts together in the same cluster and they are in different clusters in the already known clustering.

Table 2: Description of how to build the generated corpora.

Corpus	Modification
News Category 4	The classes of News Category were sorted alphabetically in ascending order and the last 37 classes in the corpus were removed.
News Category 5	The classes of News Category were sorted alphabetically in ascending order and classes 20, 21, 22, and 23 were used.
News Category 8	The classes of News Category were sorted alphabetically in ascending order and the last 33 classes in the corpus were removed.
News Category 12	The classes of News Category were sorted alphabetically in ascending order and the last 19 classes in the corpus were removed.
News Category 16	The classes of News Category were sorted alphabetically in ascending order and the last 15 classes in the corpus were removed.
News Category 20	The classes of News Category were sorted alphabetically in ascending order and the last 11 classes in the corpus were removed.
03 AG News	The classes of AG News were sorted alphabetically in ascending order and the last class in the corpus was removed.
10 news groups	The classes of 20 News Group were sorted alphabetically in ascending order and the last 10 classes in the corpus were removed.
BBC News Reviews 10	All classes of the corpora BBCNews and Reviews were included in one corpus.
Health Tweets 8	The classes of Health Tweets were sorted alphabetically in ascending order and the last 8 classes in the corpus were removed.
Reuters 8	The classes money-fx, grain, crude, trade, interest, ship, and wheat were used from Reuters-21578.
Reuters 10	The classes wheat, corn, dlr, money-supply, oilseed, sugar, coffee, gnp, gold, and veg-oil were used from Reuters-21578.
R26	Classes of R52 were sorted alphabetically in ascending order and the last 26 classes in the corpus were removed.
R30	Classes of R52 were sorted alphabetically in ascending order and the last 22 classes in the corpus were removed.

True Negatives (TN)

This is the number of pairs of documents that the clustering algorithm puts in different clusters and they are in distinct clusters in the already known clustering.

False Negatives (FN)

This is the number of pairs of documents that the clustering algorithm puts in different clusters and they are in the same cluster in the already known clustering.

Next, we explain each measure used to evaluate the experiments:

RI

Rand Index measures the similarity between two clusterings by considering all pairs of documents and calculating the number that are either in the same or in different clusters in both clusterings. It provides a score between 0 and 1, where higher values indicate better agreement between the clusterings, considering the confusion matrix is defined as follows:

$$RI = (TP + TN)/(TP + FP + FN + TN)$$

Recall

Recall measures how well a clustering algorithm identifies objects that should be together in a cluster. It is the ratio of true positives to the sum of true positives and false negatives. It is defined as follows:

$$Recall = TP/TP + FN$$

Precision

Precision measures how accurate a clustering algorithm is when it puts instances together in a cluster. It is the ratio of true positives to the sum of true and false positives. It is defined as follows:

$$Precision = TP/TP + FP$$

Table 3: **Test corpora and corpus characteristics.**

Corpus	Docs	Words	Class
03 AG News	89,059	33,824	3
10 Dataset Classification	1,000	21,187	10
10 News groups	9,595	41,490	10
BBC News Reviews 10	6,294	47,319	10
Document Classification	5485	14,690	8
German	10,273	140,624	9
Health Tweets8	29,141	41,057	8
IMDB Movie Reviews	50,000	70,577	2
Medical dataset	14,438	24,511	5
News Category4	13,307	21,202	4
News Category8	24,045	29,102	8
News Category12	45,846	40,094	12
News Category16	57,488	45,480	16
News Category20	72,924	51,420	20
Ohsumed	23,166	28,063	23
r26	8,577	18,487	26
r30	8,679	18,558	30
Reuters8	3,410	30,632	8
Reuters10	1,723	18,097	10
Text Classification on Email	9,119	80,304	5

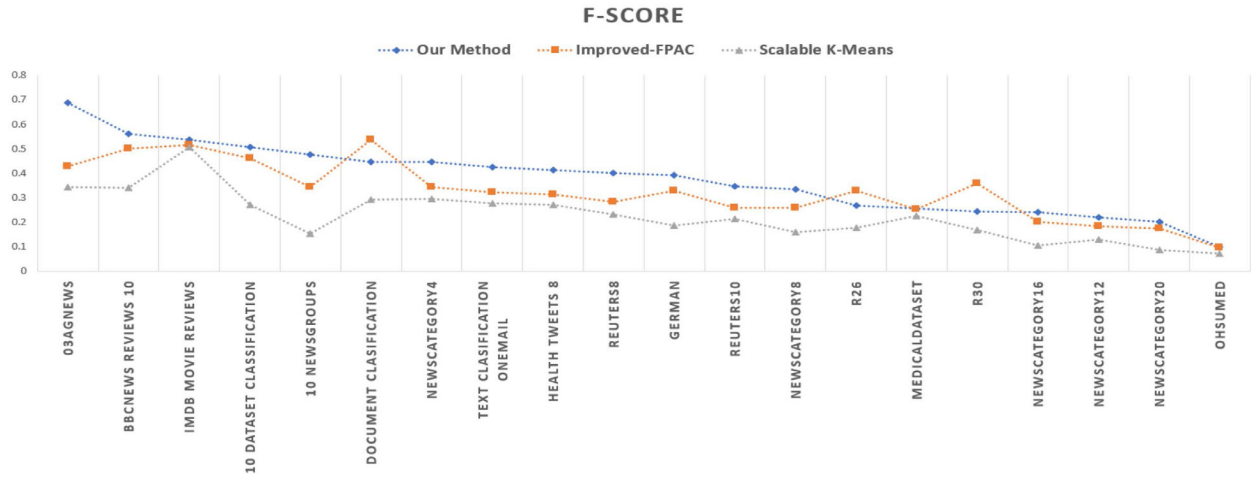

Figure 4: F-Score results for each corpus for the compared methods.

F-Score (F1-Score)

It is defined as the harmonic mean of the clustering algorithm's precision and recall and is defined as follows:

$$\mathrm{F-Score}=\frac{2*\mathrm{Precision}*\mathrm{Recall}}{\mathrm{Precision}+\mathrm{Recall}}$$

Purity

Purity assesses how well the clusters built by a clustering algorithm correspond to known cluster in the ground truth. It calculates the proportion of documents in a cluster that belong to the most common class in that cluster. It Is defined as follows:

$$\mathrm{Purity}=\frac{1}{N}\sum_{i}\max_{j}\left(\mathrm{Intersection}(C_i,T_j)\right)$$

Where:

- N is the total number of documents in the dataset.

-C_i is the predicted cluster to which the i-th document belongs.

Table 4: Comparison of results obtained using our method, the proposed value in Bejos S, *et al.*[17] and Scalable K-Means. The statistically better values were marked with an asterisk ($p<0.01$) and the best averages were highlighted in bold.

	l	RI		Recall		Precision		F-Score		Purity		NMI		Max Iter
Corpus		**AVG**	**STD**	**AVG**	**STD**	**AVG**	**STD**	**AVG**	**STD**	**AVG**	**STD**	**AVG**	**STD**	
03 AG News														
Our Method	200	0.7896*	0.0786	0.6936*	0.1107	0.6827*	0.1178	0.688*	0.1143	0.7809*	0.1186	0.5006*	0.1317	5
Improved-FPAC	10	0.6165	0.0233	0.433	0.0357	0.426	0.0346	0.4295	0.035	0.5371	0.0418	0.1315	0.0484	5
Scalable K-Means	-	0.5607	0.0057	0.3452	0.0105	0.3424	0.009	0.3438	0.0097	0.3884	0.0347	0.0124	0.012	10
Varying *l*	200	0.7937	0.0399	0.699	0.0576	0.6888	0.059	0.6938	0.0582	0.782	0.0637	0.5083	0.0819	5
10 Dataset Classification														
Our Method	119	0.8936*	0.0161	0.5519*	0.0756	0.4701*	0.0711	0.5072*	0.071	0.604*	0.0821	0.5712*	0.0623	5
Improved-FPAC	10	0.8882	0.015	0.482	0.0843	0.4406	0.0712	0.4603	0.0772	0.5827	0.0702	0.5127	0.0775	5
Scalable K-Means	-	0.8472	0.0093	0.2856	0.0432	0.2568	0.0399	0.2703	0.0412	0.4198	0.0507	0.2767	0.0546	2
Varying *l*	60	0.9167	0.0178	0.6285	0.0999	0.5724	0.0826	0.5989	0.0902	0.6884	0.085	0.6618	0.087	5
10newsgroups														
Our Method	160	0.8932*	0.0055	0.4868*	0.0302	0.4683*	0.0265	0.4773*	0.0277	0.5985*	0.0282	0.5106*	0.0222	6
Improved-FPAC	10	0.8646	0.0082	0.3516	0.0385	0.3338	0.038	0.3425	0.0382	0.4615	0.0304	0.364	0.0343	6
Scalable K-Means	-	0.8206	0.0056	0.1619	0.0166	0.1456	0.0158	0.1533	0.016	0.2514	0.0269	0.0904	0.0224	2
Varying *l*	200	0.8933	0.0071	0.4872	0.0342	0.4689	0.0328	0.4778	0.0333	0.5952	0.0229	0.5099	0.0249	6
BBC News Reviews 10														
Our Method	164	0.8892*	0.0131	0.5202*	0.043	0.6156*	0.0635	0.5616*	0.0363	0.7382*	0.0412	0.6788*	0.0282	4
Improved-FPAC	10	0.8776	0.009	0.4511	0.0304	0.5647	0.042	0.5011	0.0318	0.6851	0.0298	0.614	0.0252	7
Scalable K-Means	-	0.8353	0.0153	0.314	0.0394	0.3784	0.0499	0.342	0.0395	0.528	0.0554	0.4104	0.0408	2
Varying *l*	180	0.889	0.0107	0.5204	0.0266	0.6148	0.0508	0.5612	0.0314	0.735	0.0357	0.6783	0.0218	4
Document Clasification														
Our Method	121	0.7227	0.0166	0.3115	0.0346	0.7901	0.0419	0.4465	0.0419	0.8035*	0.0172	0.4703*	0.028	6
Improved-FPAC	10	0.7463	0.036	0.4152*	0.0907	0.77	0.0594	0.5366*	0.0891	0.7455	0.0259	0.3663	0.0453	5
Scalable K-Means	-	0.6606	0.0152	0.1947	0.021	0.587	0.0641	0.2923	0.0308	0.6913	0.0129	0.2321	0.0298	2
Varying *l*	10	0.7463	0.0314	0.4152	0.0786	0.77	0.0487	0.5366	0.0796	0.7455	0.0255	0.3663	0.033	7
German														
Our Method	200	0.85*	0.0089	0.3793*	0.0322	0.405*	0.0364	0.3916*	0.0337	0.5333*	0.0467	0.3646*	0.0282	5
Improved-FPAC	10	0.8345	0.0072	0.3206	0.0268	0.3406	0.0274	0.3301	0.0259	0.4667	0.0252	0.2796	0.0243	5
Scalable K-Means	-	0.7991	0.0054	0.181	0.0146	0.1926	0.0164	0.1866	0.0151	0.3134	0.0274	0.1079	0.0221	2
Varying *l*	200	0.8506	0.0077	0.3795	0.0298	0.407	0.03	0.3927	0.0284	0.5364	0.0303	0.3668	0.0221	5
Health Tweets 8														
Our Method	183	0.831*	0.0106	0.3874*	0.0384	0.4455*	0.0403	0.4143*	0.0388	0.5851*	0.0338	0.3831*	0.0389	5
Improved-FPAC	10	0.8039	0.0064	0.2922	0.0244	0.3419	0.025	0.3151	0.0247	0.4769	0.0228	0.2345	0.0224	5
Scalable K-Means	-	0.7869	0.018	0.2552	0.0431	0.2883	0.0557	0.2705	0.0482	0.4359	0.0584	0.1866	0.0649	2
Varying *l*	200	0.8333	0.0079	0.3941	0.0304	0.454	0.0307	0.4219	0.0305	0.596	0.0184	0.3949	0.0227	5

-T_j is the true or actual cluster to which the j-th document belongs in the ground truth.

-Intersection (C_i, T_j) represents the size of the intersection between C_i and T_j.

NMI

NMI quantifies the amount of shared information between two clusterings, considering both agreement and randomness. NMI provides a normalized score ranging from 0 to 1, where higher

[2]https://lucene.apache.org/

Table 5: Comparison of results obtained using our method, the proposed value in Bejos S, *et al.*[17] and Scalable K-Means. The statistically better values were marked with an asterisk ($p<0.01$) and the best averages were highlighted in bold.

	l	RI		Recall		Precision		F-Score		Purity		NMI		Max Iter
Corpus		**AVG**	**STD**	**AVG**	**STD**	**AVG**	**STD**	**AVG**	**STD**	**AVG**	**STD**	**AVG**	**STD**	
IMDB Movie Reviews														
Our Method	200	0.5276*	0.0169	0.5452*	0.0143	0.5267*	0.0166	0.5358*	0.0152	0.6117*	0.0382	0.0422*	0.0257	4
Improved-FPAC	10	0.505	0.0033	0.5277	0.0143	0.5048	0.0031	0.5159	0.0075	0.5463	0.02	0.0078	0.0051	4
Scalable K-Means	-	0.5059	0.0056	0.5061	0.0056	0.5059	0.0056	0.506	0.0056	0.5474	0.0279	0.0085	0.0081	2
Varying *l*	200	0.5279	0.0057	0.5451	0.0138	0.5271	0.0055	0.5359	0.0077	0.6124	0.0252	0.0426	0.0086	4
Medical Dataset														
Our Method	178	0.6762*	0.0064	0.2404*	0.01	0.2741*	0.0135	0.2562*	0.0115	0.3824*	0.019	0.0686*	0.0213	4
Improved-FPAC	10	0.6745	0.0075	0.2366	0.0156	0.2697	0.0179	0.2521	0.0166	0.3759	0.0246	0.0598	0.0248	4
Scalable K-Means	-	0.6636	0.002	0.2119	0.0044	0.2422	0.0046	0.2261	0.0044	0.3418	0.0095	0.016	0.0062	2
Varying *l*	60	0.6778	0.0076	0.2423	0.0144	0.2772	0.0176	0.2586	0.0158	0.3842	0.0254	0.0745	0.0247	5
News Category 4														
Our Method	183	0.6716*	0.0198	0.3906	0.0304	0.5178*	0.0392	0.4453*	0.0342	0.6347*	0.0427	0.2073*	0.0392	6
Improved-FPAC	10	0.6144	0.0171	0.3009	0.0268	0.404	0.0344	0.3449	0.0301	0.5387	0.0361	0.0685	0.0313	5
Scalable K-Means	-	0.5827	0.0026	0.2579	0.0056	0.3428	0.0033	0.2943	0.0042	0.4517	0.0097	0.0075	0.0044	2
Varying *l*	140	0.674	0.0168	0.3933	0.0261	0.5226	0.0333	0.4488	0.0292	0.6483	0.0348	0.2207	0.0311	7
News Category 8														
Our Method	168	0.797*	0.0086	0.2953*	0.0303	0.3881*	0.0343	0.3354*	0.0323	0.5157*	0.0263	0.2538*	0.0246	6
Improved-FPAC	10	0.776	0.0068	0.2247	0.0243	0.3033	0.0289	0.2582	0.0265	0.4458	0.0327	0.1541	0.0237	6
Scalable K-Means	-	0.7429	0.0026	0.14	0.0051	0.1839	0.0045	0.1589	0.0046	0.2729	0.0138	0.0175	0.0077	2
Varying *l*	60	0.7986	0.0072	0.2979	0.0263	0.3935	0.0296	0.339	0.028	0.5258	0.0331	0.2612	0.0244	6
News Category 12														
Our Method	170	0.7938*	0.0022	0.1646*	0.006	0.334*	0.012	0.2206*	0.0078	0.4868*	0.0204	0.2169*	0.0091	6
Improved-FPAC	10	0.7858	0.0022	0.1355	0.0074	0.2824	0.0138	0.1832	0.0097	0.4425	0.018	0.1463	0.0174	6
Scalable K-Means	-	0.7648	0.0017	0.0994	0.0036	0.189	0.004	0.1303	0.0037	0.3513	0.0027	0.0191	0.0048	2
Varying *l*	120	0.7945	0.0031	0.1661	0.0102	0.338	0.019	0.2228	0.0132	0.4937	0.0237	0.2229	0.0191	6
News Category 16														
Our Method	163	0.8521*	0.0022	0.1837*	0.0103	0.3494*	0.0171	0.2408*	0.0128	0.4851*	0.0128	0.2648*	0.0117	6
Improved-FPAC	10	0.8462	0.0021	0.1537	0.01	0.3001	0.0172	0.2033	0.0127	0.4406	0.0179	0.2047	0.0148	6
Scalable K-Means	-	0.8183	0.0026	0.0829	0.0041	0.141	0.0031	0.1044	0.0037	0.287	0.0034	0.0235	0.003	2
Varying *l*	120	0.8529	0.0021	0.1865	0.01	0.355	0.017	0.2445	0.0126	0.4913	0.0175	0.2714	0.0142	7
News Category 20														
Our Method	161	0.8822*	0.0025	0.1599*	0.0138	0.276*	0.0233	0.2025*	0.0173	0.4152*	0.0221	0.2358*	0.0154	6
Improved-FPAC	10	0.88	0.0015	0.1359	0.0097	0.2447	0.016	0.1747	0.0121	0.3852	0.012	0.2033	0.0132	7
Scalable K-Means	-	0.8527	0.0046	0.0757	0.0044	0.1045	0.0043	0.0877	0.0029	0.2264	0.0048	0.0257	0.0054	2
Varying *l*	50	0.8846	0.0021	0.1665	0.0123	0.2936	0.0207	0.2125	0.0154	0.4346	0.0165	0.2577	0.0128	8

values indicate better agreement between the clusterings and it is defined as follows:

$$NMI=2\times MI/H(C)+H(T)$$

Where:

- MI stands for Mutual Information between the predicted clusters C and the true clusters T.

- H (C) represents the Entropy of the predicted clusters C.

- H (T) represents the Entropy of the true clusters T.

Table 6: Comparison of results obtained using our method, the proposed value in Bejos S, *et al.*[17] and Scalable K-Means. The statistically better values were marked with an asterisk ($p<0.01$) and the best averages were highlighted in bold.

	l	RI		Recall		Precision		F-Score		Purity		NMI		Max Iter
Corpus		**AVG**	**STD**	**AVG**	**STD**	**AVG**	**STD**	**AVG**	**STD**	**AVG**	**STD**	**AVG**	**STD**	
ohsumed														
Our Method	110	0.8874	0.0006	0.0792*	0.0035	0.134	0.0047	0.0996	0.004	0.2565	0.0054	0.1164*	0.0051	8
Improved-FPAC	10	0.8877	0.0007	0.0757	0.0056	0.1303	0.0082	0.0957	0.0066	0.242	0.0086	0.1051	0.0082	5
Scalable K-Means	-	0.8812	0.0013	0.0591	0.0016	0.094	0.0028	0.0726	0.0018	0.1977	0.0062	0.0391	0.0041	2
Varying *l*	40	0.8877	0.0008	0.0798	0.005	0.1357	0.0075	0.1005	0.006	0.2544	0.0074	0.1191	0.0072	10
r26														
Our Method	73	0.7519	0.0135	0.162	0.0454	0.8308	0.0389	0.2694	0.0623	0.8027*	0.0134	0.486*	0.0151	10
Improved-FPAC	10	0.7606*	0.0178	0.2087*	0.0594	0.8045	0.0573	0.329*	0.0786	0.7556	0.0106	0.4359	0.0189	10
Scalable K-Means	-	0.7215	0.0068	0.1057	0.0136	0.565	0.0614	0.1779	0.0217	0.6679	0.0201	0.2822	0.0172	2
Varying *l*	10	0.7606	0.0112	0.2087	0.0392	0.8045	0.0297	0.329	0.0525	0.7556	0.0115	0.4359	0.0126	10
r30														
Our Method	66	0.7529	0.007	0.1428	0.0233	0.8269	0.0348	0.2432	0.0347	0.7986*	0.0146	0.4853*	0.0145	10
Improved-FPAC	10	0.7741*	0.0181	0.2295*	0.0597	0.8429	0.0631	0.3586*	0.0801	0.7633	0.0182	0.4568	0.0219	10
Scalable K-Means	-	0.7247	0.0061	0.1014	0.0201	0.5348	0.0563	0.1698	0.0284	0.6517	0.0181	0.2766	0.0192	2
Varying *l*	10	0.7741	0.0103	0.2295	0.0356	0.8429	0.032	0.3586	0.0493	0.7633	0.0157	0.4568	0.0113	10
Reuters8														
Our Method	152	0.8269*	0.0153	0.3751*	0.0546	0.4344*	0.0579	0.4026*	0.0562	0.5599*	0.0592	0.4179*	0.0638	5
Improved-FPAC	10	0.7866	0.0137	0.2687	0.0275	0.2973	0.0388	0.2821	0.032	0.4285	0.0403	0.2223	0.0448	4
Scalable K-Means	-	0.775	0.0104	0.2168	0.0223	0.2475	0.0289	0.231	0.0247	0.3781	0.0416	0.146	0.0351	2
Varying *l*	200	0.8331	0.0138	0.3914	0.0337	0.4568	0.0441	0.4215	0.038	0.5791	0.0468	0.4391	0.0511	5
Reuters10														
Our Method	112	0.861*	0.0083	0.3432*	0.0406	0.3501*	0.0399	0.3466*	0.0399	0.475*	0.045	0.4044*	0.0378	5
Improved-FPAC	10	0.8414	0.0076	0.26	0.0335	0.2612	0.034	0.2605	0.0336	0.3855	0.0331	0.2762	0.0413	4
Scalable K-Means	-	0.8257	0.0057	0.2215	0.0374	0.2073	0.0286	0.214	0.0324	0.3259	0.0308	0.1896	0.0374	2
Varying *l*	140	0.8623	0.0068	0.3606	0.0346	0.3589	0.033	0.3596	0.0337	0.4868	0.03	0.4182	0.0401	7
Text Clasification On EMail														
Our Method	200	0.6932*	0.0173	0.35*	0.035	0.5443*	0.0396	0.4259*	0.0373	0.6435*	0.0322	0.2773*	0.0423	5
Improved-FPAC	10	0.641	0.0104	0.2617	0.0185	0.418	0.026	0.3219	0.0217	0.553	0.0415	0.1191	0.0221	4
Scalable K-Means	-	0.6145	0.0046	0.2257	0.0049	0.3555	0.0094	0.2761	0.006	0.4862	0.016	0.0362	0.0112	2
Varying *l*	100	0.6942	0.0156	0.3542	0.0275	0.5465	0.0365	0.4297	0.031	0.646	0.0425	0.2804	0.0339	4

Concerning the parameter *k* and the termination criterion within Improved-FPAC, we established that the count of clusters to be constructed (parameter *k*) matches the count of classes present in each corpus. The maximum number of iterations was set to 10 because, in our experiments, the Improved-FPAC algorithm usually stops before reaching this number of iterations;[17] we can also observe this fact in the last column of Tables 4, 5 and 6, which shows the maximum number of iterations (Max Iter) performed for each corpus and each method. In our experiments, the maximum value of 10 was reached only in two of the corpora (r26 and r30). The stopping criterion (convergence) was defined as having less than 10% of documents reassigned to different clusters between two iterations.

As information retrieval system for Improved-FPAC, we used the Apache Lucene implementation version 7.3.[2] A Red Hat

Enterprise Linux Server Release 7.5 (Maipo) with 160 processors and 128 GB of memory was used for running all our experiments. Our method was applied to each corpus to determine the value of the parameter *l*. Once the value of *l* was determined, the Improved-FPAC algorithm was executed with this value and the quality of the resulting clustering was evaluated. Due to the randomness of Improved-FPAC in steps 1 and 4, the algorithm was executed ten times in each corpus and the average quality of the 10 executions is reported. On the other hand, the Improved-FPAC algorithm was applied ten times in each corpus using l=10 as the number of centroids as suggested in.[17] Additionally, in the same way as in,[17] we included the results obtained by Scalable K-means in our experiments. Finally, to appreciate the effectiveness of our proposal, we also performed a kind of semi-exhaustive trial and error evaluation of the Improved FPAC algorithm, which was executed (also ten times) by varying the value of l from 10 to 200 with increments of 10 and the best result is reported.

The results of the experiments are shown in detail in Tables 4-6. These tables are divided into separate blocks; each block corresponds to a corpus and has five rows; in the first row, the name of the corpus is found and in the remaining four rows, the quality of the clustering results obtained by Improved-FPAC are shown as follows: the second row shows, the results obtained by applying our method to determine the value of the parameter *l*; the third row shows the results obtained by using l=10; the fourth row shows the best result by varying the value of *l* from 10 to 200; and the fifth row shows the quality results obtained for Scalable K-Means. The first column displays the name of the clustering method used and the second column shows the number of centroids per cluster for the respective method. Columns three through fourteen present the average and standard deviation results for RI, Recall, Precision, F-Score, Purity and NMI. The last column of these tables shows the maximum number of iterations in all repetitions. In the results, the best outcome achieved for each method in each of the corpora is highlighted in bold-text as it can be seen in Tables 4-6, the standard deviation is very small in all cases. In Figure 4, we can see the summarized results for the three compared methods using the F-Score metric. The x-axis displays the corpora used in the experiments, while the y-axis shows the F-Score value obtained for each method and corpus.

In Figure 4, we can see the summarized results for the three compared methods using the F-Score metric; the x-axis displays the corpora used in the experiments, while the y-axis shows the F-Score value obtained for each method and corpus. From this figure, we can also see that using our proposal for calculating the number of centroids per cluster allows obtaining the best results in most cases. According to our experiments, in 16 of 20 corpora, the quality of the clustering obtained by the Improved-FPAC algorithm with the value of *l* calculated using our method yields better results than the value of *l* recommended in Bejos S, *et al*.[17] These results are obtained regardless of the evaluation measure used; see the second and third rows in each block in Tables 4, 5 and 6. To validate these results, we applied the statistical Wilcoxon signed-rank test with $p<0.01$, which showed that our proposed method is statistically better (see entries marked with an asterisk in Tables 4-6) than the method proposed by Bejos S, *et al,*[17] on these 16 datasets. Although the quality of the clustering obtained by the Improved-FPAC algorithm using the value of *l* calculated by our method did not coincide in all cases with the best clustering computed by Improved-FPAC varying *l* between 10 and 200, it was close to the best option. Our experiments show that Scalable K-means obtained the worst quality results in all the corpora.

CONCLUSION AND FUTURE WORK

This work introduces a method to compute the number of centroids per cluster (parameter *l*) for the Improved-FPAC algorithm. The proposed method considers the characteristics of the corpus (number of documents, vocabulary and number of classes) to determine the value for parameter *l*. To calculate the value of *l*, some corpora widely used in the literature were used to create an equation from a trend line obtained from the relationship between the best quality value obtained by varying the number of centroids from 10 to 200 and the characteristics of each corpus.

In our experiments, we confirmed a relationship between the characteristics of the corpus and the value of the parameter *l* that should be used in the Improved-FPAC algorithm to obtain better-quality results. In most of the evaluated corpora, the quality obtained with our method exceeds that obtained by using the fixed value suggested.[17] Additionally, our proposal allows obtaining quality values close to the best quality found by varying *l*. However, using our method, it is not necessary to run the algorithm several times changing the number of centroids to find the option with the best quality result, which is not an option in practice.

In future work, we will modify Improved-FPAC to allow dynamic values of *l*. Starting from the value defined by our current proposal and dynamically increasing or decreasing it according to the number of documents and vocabulary size of each cluster and the quality results obtained in each iteration of the algorithm.

FPAC uses each cluster center as a query to retrieve a list of documents, assigned to that cluster without requiring distance calculation. However, as the number of centroids increases, in Improved-FPAC, the possibility of overlap and the number of distance calculations also increases. To avoid this, it is necessary to improve the query used in the cluster computation to remove those terms already included in a centroid, reducing overlap and potentially improving the algorithm's performance.

ACKNOWLEDGEMENT

The first author wants to thank the Benemerita Universidad Autonoma de Puebla (BUAP) for all the support to pursue his doctoral studies

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AVAILABILITY OF DATA AND MATERIALS

The data material used in the research is in the public domain and is available at:

https://github.com/intisand/ImprovedFPAC/tree/main/dataset

Code availability: The code is available at:

https://github.com/intisand/ImprovedFPAC/tree/main/

AUTHORS' CONTRIBUTIONS

I.S.M. Wrote the text of the paper, made modifications to the code for the execution of experiments and created the figures for the paper.

J.F.M.T and J.A.C.O. supervised the paper, suggested the figures for the paper and suggested changes to improve the writing.

D.V.A. collaborated in supervising the paper and proposed the list of corpora to be used.

All authors contributed to developing the proposed method and reviewed the manuscript.

[#]These authors contributed equally to this work.

REFERENCES

1. Abualigah L, Gandomi AH, Elaziz MA, Hussien AG, Khasawneh AM, Alshinwan M, *et al.* Nature-inspired optimization algorithms for text document clustering—a comprehensive analysis. Algorithms. 2020;13(12):345. doi: 10.3390/a13120345.
2. Agarwal N, Sikka G, Awasthi LK. A systematic literature review on web service clustering approaches to enhance service discovery, selection and recommendation. Comput Sci Rev. 2022;45. doi: 10.1016/j.cosrev.2022.100498.
3. Alghamdi HM, Selamat A. Arabic web page clustering: a review. J King Saud Univ Comput Inf Sci. 2019;31(1):1-14. doi: 10.1016/j.jksuci.2017.06.002.
4. Cozzolino I, Ferraro MB. Document clustering. WIREs Computational Stats. 2022;14(6):e1588. doi: 10.1002/wics.1588.
5. Dobrakowski AG, Mykowiecka A, Marciniak M, Jaworski W, Biecek P. Interpretable segmentation of medical free-text records based on word embeddings. J Intell Inf Syst. 2021;57(3):447-65. doi: 10.1007/s10844-021-00659-4.
6. Eligüzel N, Çetinkaya Ç, Dereli T. A novel approach for text categorization by applying hybrid genetic bat algorithm through feature extraction and feature selection methods. Expert Syst Appl. 2022;202:117433. doi: 10.1016/j.eswa.2022.117433.
7. Inje, B, Nagwanshi KK, Rambola RK. An efficient document information retrieval using hybrid global search optimization algorithm with density based clustering technique. Clust Comput. 2023:1-17.
8. Kim H, Kim HK, Cho S. Improving spherical k-means for document clustering: fast initialization, sparse centroid projection and efficient cluster labeling. Expert Syst Appl. 2020;150:113288. doi: 10.1016/j.eswa.2020.113288.
9. Malik F, Khan S, Rizwan A, Atteia G, Samee NA. A novel hybrid clustering approach based on black hole algorithm for document clustering. IEEE Access. 2022;10:97310-26. doi: 10.1109/ACCESS.2022.3202017.
10. Pandey KK, Shukla D. Ndpd: an improved initial centroid method of pertitional clustering for big data mining. J Adv Manag Res. 2023;20(1):1-34. doi: 10.1108/JAMR-07-2021-0242.
11. Ponnusamy M, Bedi P, Suresh T, Alagarsamy A, Manikandan R, Yuvaraj N. Design and analysis of text document clustering using salp swarm algorithm. J Supercomput. 2022;78(14):16197-213. doi: 10.1007/s11227-022-04525-0.
12. Song W, Qiao Y, Park SC, Qian X. A hybrid evolutionary computation approach with its application for optimizing text document clustering. Expert Systems with Applications. 2015/science/article/pii/S0957417414006861;42(5): 2517-24. doi: 10.1016/j.eswa.2014.11.003.
13. V(K). & S, S. Developing a conceptual framework for short text categorization using hybrid cnn-LSTM based caledonian crow optimization. Expert Syst Appl. 2023:212, 118517.
14. Yong KS, Liew JS. The more. J Intell Inf Syst. 2023:1-21.
15. Fahad SA, Alam MM. A modified k-means algorithm for big data clustering. Int J Sci Eng Comput Technol. 2016;6:129.
16. Ganguly D. A fast partitional clustering algorithm based on nearest neighbours heuristics. Pattern Recognit Lett. 2018;112:198-204. doi: 10.1016/j.patrec.2018.07.017.
17. Bejos S, Feliciano-Avelino I, Martínez-Trinidad JF, Carrasco-Ochoa JA. Improved fast partitional clustering algorithm for text clustering. J Intell Fuzzy Syst. 2020;39(2):2137-45. doi: 10.3233/JIFS-179879.
18. Abasi AK, *et al.* A hybrid salp swarm algorithm with β-hill climbing algorithm for text documents clustering. Evol Data Clustering Algor Appl. 2021;129.
19. Abualigah LM, Khader AT, Hanandeh ES. A combination of objective functions and hybrid krill herd algorithm for text document clustering analysis. Eng Appl Artif Intell. 2018;73:111-25. doi: 10.1016/j.engappai.2018.05.003.
20. Akter R, Chung Y. An improved genetic algorithm for document clustering on the cloud. Int J Cloud Appl Comput. 2018;8(4):20-8. doi: 10.4018/IJCAC.2018100102.
21. Bezdan T, Stoean C, Naamany AA, Bacanin N, Rashid TA, Zivkovic M, *et al.* Hybrid fruit-fly optimization algorithm with k-means for text document clustering. Mathematics. 2021;9(16):1929. doi: 10.3390/math9161929.
22. Janani R, Vijayarani S. Text document clustering using spectral clustering algorithm with particle swarm optimization. Expert Syst Appl. 2019;134:192-200. doi: 10.1016/j.eswa.2019.05.030.
23. Thirumoorthy K, Muneeswaran K. A hybrid approach for text document clustering using jaya optimization algorithm. Expert Syst Appl. 2021;178:115040. doi: 10.1016/j.eswa.2021.115040.
24. Yuan M, Zobel J, Lin P. Measurement of clustering effectiveness for document collections. Inf Retrieval J. 2022;25(3):239-68. doi: 10.1007/s10791-021-09401-8.
25. Morgan JA, Tatar JF. Calculation of the residual sum of squares for all possible regressions. Technometrics. 1972;14(2):317-25. doi: 10.1080/00401706.1972.10488918.

Cite this article: Magallon-Juanqui IS, Martinez-Trinidad JF, Vilarino-Ayala D, Carrasco-Ochoa JA. Corpus Characteristics-Based Method to Centroids Number Determination for Clustering Text Documents. J Scientometric Res. 2025;14(1):319-30.