Ingénierie des Systèmes d'Information
Vol. 29, No. 6, December, 2024, pp. 2503-2513
Journal homepage: http://iieta.org/journals/isi

Unprecedented Rough Sets Model for Arabic Document Clustering

Khitam A. Salman[1,2*], Hussein K. Khafaji[3]

[1] Computer Science Department, University of Technology, Baghdad 10066, Iraq
[2] Informatics Institute for Postgraduate Studies (IIPS), Iraqi Commission for Computers and Informatics (ICCI), Baghdad Ilwiya 10068, Iraq
[3] Computer Communication Engineering Department, Al-Rafidain University College, Baghdad 10014, Iraq

Corresponding Author Email: khitam.a.salman@gmail.com

https://doi.org/10.18280/isi.290635

Received: 28 May 2024
Revised: 1 August 2024
Accepted: 7 August 2024
Available online: 25 December 2024

Keywords:
Arabic document clustering (*ADC*), *document clustering*, *tokens*, *rough set theory*, *feature table* (*FT*), *distance matrix* (*DM*)

ABSTRACT

Numerous features, such as various morphologies, orthography, structural features, unique linguistics, different word meanings, and more uncountable features of Arabic, are considered key challenges in Arabic document clustering. Clustering Arabic documents is a paramount task in the information retrieval and data mining fields. In this paper, we suggest a novel model based on the rough set theory for clustering Arabic documents. Two well-known datasets, CNN and OSAC, are preprocessed and prepared as input for the model. The feature table is created from the preprocessed data. Documents' similarities are calculated by adapting the rough discernibility relation to determine semantically coherent documents. This relation is represented as a weighted distance graph (WDG), from which the similarity matrix was constructed. The resulting similarity values play crucial roles in the suggested clustering algorithm. The model effectiveness was evaluated on CNN and OSAC datasets, achieving an F-score of 0.85 for both.

1. INTRODUCTION

Text mining [1, 2] refers to discovering hidden, significant, unknown patterns in written text or documents. It is a variant of data mining [3] with the difference that data mining deals with structured formats, while text mining can deal with semi-structured and unstructured formats [4]. Summarization, information extraction, clustering, association rule mining, and question answering are some of the text mining technologies used in the mining process to analyze, understand, and even create text.

Compared to languages with simpler patterns, Arabic poses particular difficulty of document grouping tasks. Interestingly, there is ambiguity in Arabic writing since it lacks natural vowel marks. Arabic morphology also includes intricate derivational prefixes and suffixes' that profoundly change the meaning of the words to overcome these obstacles during the document preparation, specific preprocessing techniques like stemming and disambiguation are required. In data mining and information retrieval, document clustering is essential, particularly given the enormous and constantly expanding number of Arabic texts available online. Clustering making it easier to navigate and analyze the information resources efficiently by organizing documents in groups according to themes that they share [5].

Clustering is an unsupervised task [6] that aims to find the most similar objects or structures without predefined knowledge of the target of these objects. Document or text clustering is an interesting research area [7]. It is one of the text mining tools [4] that manages documents into a significant number of clusters by grouping the most similar documents into one coherent cluster by decreasing the intra-distance among documents and making the inter-distance among other dissimilar ones as large as possible.

Clustering algorithms can be categorized as traditional or modern. Traditional algorithms are further divided into several categories, such as partitional, hierarchical, and fuzzy theory-based approaches. Conversely, modern clustering techniques may utilize kernels, ensembles, or swarm intelligence. For detailed information about clustering algorithms, a comprehensive study was presented by Ezugwu et al. [8].

In 1982, Pawlak introduced the concept of rough set theory (RST) [9-11], a framework for handling ambiguous, inaccurate, incompatible, and doubtful knowledge. RST has been successfully applied in various fields of artificial intelligence. For instance, it assists in feature selection [12, 13] by determining relevant features from a corpus and discerning their significance. It is also used in classification and clustering [14], grouping objects according to discernible patterns in the data, and in decision support systems [15], where it helps analyze vague or imprecise data. Additionally, RST is utilized in the medical field to aid in disease diagnosis, classify diseases, and prognosticate patient outcomes by analyzing medical data [16].

The key reasons for using RST for handling imprecise and uncertain data are its simplicity and intuitive model for analyzing complex data. The resulting analyses of the RST model are interpretable and helpful in decision-making, besides its robustness to noise and missing data, hence it is applicable for real-life applications. Moreover, using RST in

data analysis does not require extra information about the knowledge, such as the probabilities in the statistics or the degree of membership in the fuzzy set [17].

RST assumes that we have a finite set of objects called the universe of discourse, which we will denote by the symbol (**Ω**) and *X* is a subset of **Ω**. RST uses the equivalence relation, *the indiscernibility relation R*, $R \subseteq \Omega \times \Omega$. It identifies attributes that can distinguish between objects in a dataset. For a given set of objects, let say *X* which it is a subset of **Ω**, the discernibility relation identifies which attributes can separate them into disjoint equivalence classes. This relation should satisfy the reflective, symmetric and transitive properties and it is important to define the upper and the lower approximations. Two objects *a, b* ∈ **Ω** are said to be indiscernible in *R* if *a R b.* The indiscernibility relation is expressed in Eq. (1).

$$IND(X) = \{(a,b)|(a,b) \in \Omega^2, \forall_{x \in X}(x(a) = x(b))\} \quad (1)$$

Each concept in this universe is associated with some knowledge. So, any concept in this universe of discourse can be represented as a subset *X* of **Ω**. The basic notions of the RST concept are that each set X can be approximated by a pair of disjoint sets: *the lower and upper approximations*. The lower approximation *L* refers to the set of objects that definitely belong to X, while the upper approximation *U* is the set of objects that may belong to *X*. The boundary region *B* is the difference between the upper and lower approximations. Below is the mathematical definition of these approximations [9, 10].

$$L(X) = \{a \in \Omega: [a]_R \subseteq X\} \quad (2)$$

$$U(X) = \{a \in \Omega: [a]_R \cap X \neq \emptyset\} \quad (3)$$

$$B(X) = U(X) - L(X) \quad (4)$$

where, $[a]_R$ means the equivalence class of a.

The knowledge that describes the universe of discourse is represented as an information table, IT [18]. It consists of a group of objects characterized by a combination of features. The IT is defined by the quadruple, which is defined in Eq. (5).

$$IT = \{\Omega, T, \{V_t | t \in T\}, \{I_t | t \in T\}\} \quad (5)$$

where, Ω is a finite, none empty set of objects, T is none empty set of attributes, V_t a non-empty set of values which is subset of T and I_t is a function, I_t: $\Omega \times V_t$.

Let's explore a dataset of cars that have three different features: buying price (low, high), luggage bot size (big, small), and safety (very high, high, low) as presented in Table 1. Our aim is to decide whether the care is acceptable (yes) or unacceptable (no) based on these features.

Table 1. Cars' dataset

C	Price	Luggage	Safety	Acceptable
c1	Low	Big	High	Yes
c2	High	Small	High	Yes
c3	High	Big	Very high	Yes
c4	Low	Big	Low	No
c5	High	Small	High	No
c6	Low	Big	Very high	yes

From car dataset c1, c2 and c5 are indiscernible with respect to Price feature, c3 and c3 are indiscernible with respect to Luggage and Acceptable features and c2 and c5 are indiscernible with respect Price, Luggage and Safety features. So, Luggage feature creates two equivalence classes {c1, c3, c4, c6} and {c2, c5} while, features Price and Safety generates five equivalence classes {c1}, {c2, c5}, {c3}, {c4] and {c6 and so on the equivalence classes can be generated from any subset of features.

Car cannot be scribed in term of features Price, Luggage, and Safety since c2 is acceptable while, c5 is not and they are indistinguishable with regard to these features. Therefore, c2 and c5 are boundary line and in such a situation they cannot be appropriately classified based on information at hand. The remaining cars c1, c2, and c3 can be certainly categorized as acceptable cars in term of Price, Luggage and Safety and c4 is certainly unacceptable car. So, the lower approximation of the acceptable cars is, $L(X) = \{c1,\ c3,\ c6\}$, while the upper approximation, the care that may be classified as acceptable is, $U(X) = \{c1, c2, c3, c5, c6\}$ meanwhile the boundary region is, $B(X) = \{c2,\ c5\}$.

A discernibility matrix $D(IT)$, is an essential tool of RST used to represent the discernibility relation between features. The matrix element indicates whether two features are discernible or indiscernible to each other based on the given dataset. The discernibility relation plays crucial role in constructing this matrix. The discernibility matrix's main goal is to determine an object's discernibility relations according to its properties.

Arabic language is increasingly spoken last few years. According to the Statista statistics website (www.statista.com\statistics), more than 270 million people speak Arabic, and as a consequence, the number of web contents written in Arabic is potentially increasing. Diverse features, such as various morphologies, orthography, structural features, unique linguistics, different word meanings, and more uncountable features of Arabic, are considered key challenges in Arabic document clustering (ADC) [19-21].

The contribution of this paper is to cluster Arabic documents using an unprecedented adapted rough set model. The model is based on creating the feature table for the manipulated data, measuring the similarities among documents depending on the properties of rough sets, and then using these similarities in the clustering process. The results achieved by this model of 85% for CNN and 85% for OSAC offer a promising contribution in the field of ANLP.

The rest of this paper is managed as follows: the related works covers works that made on Arabic document clustering. The proposed model is described in detail in Methodology. Meanwhile, the results and discussion examine the results that we obtain. The main contribution and the future works demonstrate in conclusion.

2. RELATED WORKS

The rapid growth of Arabic content on the internet demands effective automated Arabic document (ADC) techniques. Few studies have been conducted in this area, and the most of them used the k-means method for the clustering process, according to a recent review [22] of ADC methodologies and techniques. In this section, we will discuss previous studies on ADC. To the best of our knowledge, no previous studies have been done for ADC using RST.

In 2017, Daoud et al. [23] enhanced Arabic document

clustering using a hybrid technique of K-means and particle swarm optimization (PSO) to overcome the problem of the initial seeding of centroids of clusters by employing PSO for this purpose. The results showed that using this combined technique improved the clustering results. This method exhibits the same drawbacks as its forerunners, PSO and K-means, including its vulnerability starting initialization, significant computing overhead, and tendency to converge too soon to unsatisfactory solutions.

In 2018, k-means and its variants were employed by Sangaiah et al. [24] to enhance the clustering of Arabic documents. The developed models achieved dimension reduction (DR) with excellent clustering outcomes when compared with the results of other techniques.

Alhawarat and Hegazi [25] addressed clustering problems with high data dimensionality. They used the LDA modeling technique for dimensionality reduction before implementing the k-means algorithm for text documents clustering. The study showed that normalizing the data enhanced the clustering results. The computation complexity of the model increases with the number of subjects and documents, which can limit the scalability of LDA when applied to massive datasets. Furthermore, hyperparameter tweaking pertaining to the number of topics, sampling technique, and Dirichlet priors affects LDA's performance. Inappropriate selection of hyperparameters can have a negative impact on clustering results.

Al-Sarrayrih and Al-Shalabi [26] implement hierarchical clustering based on frequent itemsets (FI) with N-gram (FIHC) for document clustering. This approach is implemented to manipulate European languages. Implementing this technique on Arabic document achieved better clustering results than those achieved for European languages. the lack of transparency concerning the dataset used in the experiment limits the research finding' capacity to broadly applied. Moreover, utilizing only one dataset hiders a thorough assessment of algorithm's effectiveness and predictive power.

Mohamed [27] evaluates the effect of different dimension reduction techniques on text clustering. He used principal component analysis (PCA), nonnegative matrix factorization (NMF), and singular value decomposition (SVD) reduction techniques prior to text clustering. Each of these techniques is applied to two linguistic corpora; Arabic and English, and the results reveal that PCA outperforms the two others in terms of clustering quality, interpretability, and computational efficiency.

In the study by Salman and Khafaji [28], a new algorithm for Arabic documents clustering was suggested. The algorithm employed the maximal frequent wordsets (MFWs) as feature representation. These MFWs were extracted employing the Fpmax algorithm, which is a data mining method to determine the most significant word patterns that occur within the dataset. Based on semantic similarity, documents were effectively grouped employing the resulting MFW-based clustering algorithm. The algorithm's performance was assessed on publicly accessible datasets CNN and OSAC using the widely adopted F-score metric, achieving accuracies of 80% and 81%, respectively. Table 2 shows a comparison among related works.

As we mentioned above, no research employed RST Arabic document clustering. But it is used for other purposes, such as a decision support system, determining political Arabic article orientation, detecting extreme Arabic text views in Arabic articles, etc.

In 2019, H. A. Malik used RST with Arabic text mining for a decision support system. The author developed a categorization model that depends on a token's meaning, which organizes the information by recognizing tokens and resolves noisy information issues [15].

Although k-means is a simple, effective clustering algorithm, its performance can be negatively affected by its sensitivity to initial cluster centres. Despite PSO k-means overcoming this constraint, it comes with a significant processing cost. LDA is good at capturing semantic associations, but it can be complicated, especially when handling massive datasets. Clustering efficiency can be increased when combing k-means and dimensionality reduction, but there is a chance of information loss when data is reduced.

Our proposed model will answer the following research questions:

(1) Is the proposed model capable of dimension reduction? Can it maintain the significant semantic meaning of the documents while dimension reduction?

(2) Does the suggested model have the ability to group documents according to semantic similarity without depending on external information or techniques to discover hidden patterns in the dataset?

Table 2. Comparison of related works

Ref. No.	Approach	Dataset	F-score	Purity	Accuracy	Entropy
[23]	k-means + PSO	BBC, CNN, OSAC	47%	50%	-	-
[24]	k-means + DR	Set of documents	70	-	-	45
[25]	k-means + LDA	MSA	70%	45%	-	-
[26]	FIHC	Own dataset	70%	-	-	-
[27]	k-means, SVD, NMF, PCA	Arabic dataset, Reuters	-	-	66%	-
[28]	MFW	CNN, OSAC	81%	-	-	-

3. METHODOLOGY

In this paper, we suggest an unprecedented model for clustering Arabic documents based on rough set theory. As a preliminary step, the dataset is prepared by removing unnecessary and noisy data. The prepared information is then utilized as a key input for the extracted feature table that will be used to describe the universe of discourse. Afterwards, the similarity among documents is measured by calculating the distance matrix. And the last step is utilizing the distance matrix in the clustering process. The model design is shown in Figure 1. The model includes four stages:

1. Dataset preprocessing and Encoding
2. Extracted feature table
3. Construction of weighted distance graph WDD
4. Document clustering

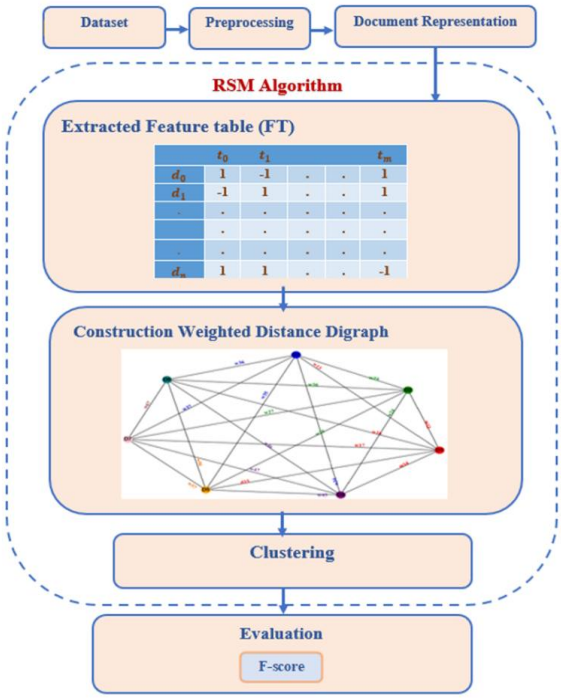

Figure 1. Overall system design

The model's stages are explained in the following sub-sections.

3.1 Dataset preprocessing and document representation

Two public and well-known datasets, CNN and OSAC, were used to assess the suggested model [29]. The CNN dataset was collected from the *CNN* Arabic website and consists of more than 4000 documents divided into six classes (business, history, entertainment, Middle East news, sport, and world news). On the other hand, the OSAC dataset was assembled from diverse Arabic internet sources and has over 22000 articles spread across ten classes (economics, history, entertainment, education and family, religion and fatwa, sport, health, astronomy, law, stories, and cooking recipes).

Preprocessing data is an important step for machine learning techniques since the raw data may contain meaningless, noisy, and non-useful data, and it is also in a non-readable format. So preprocessing is done in a format that is acceptable to the technique used. It includes tokenizing data, refining it, and purifying it from punctuation and stop-words, and then the root of each word is extracted in the stemming step. The datasets are cleaned from unimportant words such as punctuation and stop-words that do not affect the clustering and stemmed using *Tashaphyne stemmer*, which is an open-source project available at Tashaphyne PyPI. After that, it was saved to the CSV file. Algorithm 1 shows the preprocessing phase.

Algorithm 1. Dataset preprocessing algorithm

***Input**: D: Dataset (Document text), specialCharList, normalizedLettersList*
Begin
For each document d in D Do
Read (d)
td=tokenization(d) //perform tokenization
For each token t in td Do
//Check token letter
If letter in normalizedLettersList Do {
normT=Normalization(token) }
If normT in specialCharList Do{
Remove(normT) //perform normalization
Else{
stemmedT=Stemming(normT)//perform stemming
write stemmedT to Preprocessed-d}
Next token
preprocessedDataset ∪= Preprocessed-d
Next document
Return preprocessedDataset
End
***Output**:* preprocessedDataset

The refined data is then encoded using either TF/IDF or counter vectorizer (CV) encoding techniques. TF/IDF is a technique to represent a document, it contributes to dimension reduction. This technique is one of the best encoding methods that are employed to calculate the token's weight in the dataset. Token frequency (token counts in a document) and inverse article frequency (token's count in all dataset's articles) the two crucial elements taken into account while determining TF/IDF values. The significant tokens are considered by this technique by maximizing their weight when these tokens are rare for the whole dataset but significant to understand the document.

$$IDF(t) = log(N/n\_t) \tag{6}$$

$$TF(t,d) - IDF(t,d) = TF(t,d) * IDF(t) \tag{7}$$

where *TF(t,d)* is the frequency of token in document *d* (i.e., the number of times token *t* in a document d) and IDF(*t*) is the inverse document frequency of the token *t*, N is the total number of documents in the dataset, and *n_t* is the total dataset documents that contain the token *t*.

The CV technique depends on token occurrences in the dataset. It transforms a document into a set of numerical values

depending on the tokens' frequencies. This technique deals with a document as distinct tokens, ignoring their order in the documents, so the tokens will be represented as a bag of tokens.

Both methods lessen the effects of high-dimensional feature space while capturing the syntactic and semantic information presented in the dataset. These encoding strategies help provide a compact but beneficial representation of documents by detecting discrimination and meaningful tokens. And this will answer the research's first question.

3.2 Extracted feature table (FT)

The extracted FT is a quadruple model that represents uncertain and vague knowledge. This table is mathematically represented as Eq. (8).

$$FT = \{\Omega, T, \{V_t | t \in T\}, \{I_t | t \in T\}\} \tag{8}$$

where, $\Omega = \{d_1,\ d_2, \dots,\ d_n\}$ defines the finite non-empty set of documents, $T = \{T_1,\ T_2, \dots, T_n\}$ is a non-empty set of discriminative documents' tokens. V_t is a non-empty set of values $t \in T$, $I_t: \Omega \to V_t$ is a function that links a document of Ω to only one value in V_t.

Each row of FT corresponds to a document and each column signifies and significant discriminative document token identified during the encoding process. The cell's value indicates the relation between a specific document and that particular token. This relationship is determined by Eq. (9).

$$FT_{ij} = \begin{cases} 1 & if\ t \in T \wedge t \in d_i \\ -1 & otherwise. \end{cases} \quad \text{for } i = 1, \dots, n \tag{9}$$

where, $n = |\Omega|$, and j=1,2,...,m where m=|T|. Once all the values in the FT are set, each document gets transformed into a vector of 1s and -1s. This transformed table becomes a crucial input for the next step, where it forms the basis for creating the distance matrix. Algorithm 2 presents the process of constructing the extracted feature table.

Algorithm 2. Extracted FT

Input
T – set of discriminative dataset tokens
d_i – document vector
D – number of documents in the dataset
Begin
For d_i in D Do:
For t in T Do:
'If t $\in$ d_i Do:
FT = 1
Else Do:
$FT_t = -1$
Return FT
***Output*:** FT – feature table (the rows denote the documents and the columns denote the tokens)

3.3 Construction of weighted distance graph

The discernibility matrix is an instrumental tool that plays essential role in feature selection that helps in data analytics and discover patterns or similarities for identifying of clusters or groups within documents. It can be viewed as a distance matrix or similarity matrix. It aids in identifying essential tokens that assists in documents differentiation while disregarding superfluous or extraneous ones. It assists to distinguish the documents that are similar to one another and those that can be distinct from one another. It assists in identifying essential symbols that support document differentiation while disregarding unnecessary or extraneous ones. It helps to distinguish between texts that are distinct from one another and those that are similar to one another. Additionally, by identifying discernibility relations, it aids in reducing the complexity of data.

The discernibility matrix of the set of documents is populated according Eq. (10).

$$D_{ij} = \{t \in T : t(d_i) \neq t(d_j)\} \quad \text{for } i, j = 1, \dots, n \text{ and } n = |\Omega| \tag{10}$$

where, Ω is the dataset space, t is a specific token of the whole discriminative token- T, d_i, d_j are i^{th} and j^{th} documents respectively.

The weighted distance graph construction is based on the FT that was extracted in the previous section. The graph nodes refer to the documents and the weighted arcs represent the similarities between documents. The arcs weights are calculated according to Algorithm 3. By constructing the distance graph where weighted edges represent the similarities between documents, this research identifies the model's ability to discover the hidden patterns in the dataset documents and find the semantic relationship among them depending on the available information. This answers the second question of the research.

Algorithm 3. Constructing of weighted distance graph

Input:
FT-feature table
|T|- number of tokens
|D|- number of documents
e_{ij}- edge between document d_i and document d_j
Begin
For i=1 to |D| Do:
For j = i+1 to |D| Do:
For k=1 to |T| Do:
If FT[i,k] = 0 or FT[j,k] = 0 Do:
$e_{i,j} = e_{i,j} + 0.5$
Else if FT[i,k] != FT[j,k] Do:
$e_{i,j} = e_{i,j} + 1$
$e_{i,j} = e_{i,j} / |T|$
Return WDG
***Output: Weighted** distance graph (WDG)*
***Time Complexity**: $O\left(\frac{|D| \times (|D|-1)}{2} \times |T|\right)$*

3.4 Documents clustering

The distance matrix (*DM*) is an adjacency matrix of weighted graph G, where the set of nodes N is the set of documents and the arcs represent the relation "Document D_i bears resemblance to Document D_j to the extent of S."

We can obtain the similarities among documents in the dataset once the matrix has been filled up. For example, the similarity between the i^{th} and j^{th} documents in the dataset is indicated by the value at DM_{ij}.

The distance matrix is used in the next phase of the proposed method to organize the documents into groups according to their degree of similarity and the specified number of the

clusters.

Determining the number of the closest document to each document is an experimental task based on trial and error. At first, we start by selecting it according to the number of desired clusters, i.e., when the desired number of clusters is 6, then the model will select the six closest documents from the similarity matrix. Next, we choose 10% of the maximum features determined by TF/IDF when encoding the documents and gradually increase this value. After multiple implementations, we found that the best number of the chosen closest document depends on the number of clusters and the size of the dataset. After selecting the n closest documents with their similarities, the initial clusters will be formulated and will be equal to the total number of documents in the dataset, and the distance matrix size will be $D \times n$ instead of $D \times D$, where D is the total number of documents in the dataset and n is the number of the closest documents.

Clustering starts by pruning the resulting clusters after determining the n closest document. It is worthy to mention that after eliminating the size of distance matrix, multiple clusters with identical documents will emerge. At this point, these clusters will be combined into a single cluster. Subsequently, the model begins to hard-cluster documents, allocating each document to a particular cluster.

In order to retain the document in the cluster with the highest similarity values, hard clustering involves removing duplicate documents from several clusters based on their respective similarity values. If a document is assigned to many clusters with identical similarity values, it will remain in the first cluster. This process will be repeated until the desired number of clusters is achieved in the clustering.

The proposed model is applied to two datasets (CNN and OSAC) and repeated for different TF/IDF and CV maximum feature values.

3.5 Illustrative example

Consider the following synthetic dataset presented in Table 3.

After data preprocessing stage, the discriminative tokens of the dataset will be presented in the Token list as follows:

Tokens= ['برلم'، 'جرء'، 'جرح'، 'جلس'، 'شفف'، 'علج'، 'عمل'، 'قنن'، "لمز'، 'مثل'، 'مرض'، 'نتخب'، 'نزف"].

According to these words, the feature table will be extracted using algorithm 2. Table 4 depicts the extracted feature table.

Based on Table 3 the similarity among documents will be calculated and the weighted distance graph is calculated according to algorithm 3. The graph is then represented as an adjacency matrix of weighted graph G, where the set of nodes N is the set of documents and the arcs represent the relation "Document D_i bears resemblance to Document D_j to the extent of S." Consider Figure 2 which elucidates the similarity among the documents. Table 5 shows the similarity matrix.

Table 3. Synthetic dataset

D#	Sentence in Arabic	Meaning
D1	فتح الطبيبُ الجرح وعالجه لوقفِ النزيفِ	The doctor opened the wound and treated it to stop the bleeding.
D2	بعد إجراء العملية أصيب المريض بنزف شديد	After the operation the patient suffered severe bleeding.
D3	امتثل المريض للشفاء بعد إجراء عملية جراحية	The patient recovered well after surgery.
D4	شفاء مريض سرطان بعد جلسات علاجية	A cancer patient recovered after treatment sessions.
D5	يصوت البرلمان على قانون الانتخابات التشريعية	Parliament votes on the legislative election law.
D6	اختتم البرلمان أعمال جلسته بعد تصويته على قانون الموازنة	Parliament concluded its session after voting on the budget law.
D7	يعقد البرلمان جلسته لانتخاب رئيسا له	Parliament holds its session to elect its president.

Table 4. Extracted FT

برلم	جرء	جرح	جلس	شفف	علج	عمل	قنن	لمز	مثل	مرض	نتخب	نزف
-1	-1	-1	-1	-1	1	-1	-1	-1	-1	-1	-1	1
-1	1	-1	-1	-1	-1	1	-1	-1	-1	1	-1	1
-1	1	1	-1	1	-1	1	-1	-1	1	1	-1	-1
-1	-1	-1	1	1	1	-1	-1	-1	-1	1	-1	-1
1	-1	-1	-1	-1	-1	-1	1	-1	-1	-1	-1	-1
1	-1	-1	1	-1	-1	1	1	1	-1	-1	-1	-1
1	-1	-1	1	-1	-1	-1	-1	-1	-1	-1	1	-1

Table 5. Similarity matrix

	D1	D2	D3	D4	D5	D6	D7
D1	**1**	0.692307692307 6923	0.384615384615 38464	0.692307692307 6923	0.69230769230 76923	0.46153846153 846156	0.615384615 3846154
D2	0.69230769230 76923	**1**	0.692307692307 6923	0.538461538461 5384	0.53846153846 15384	0.46153846153 846156	0.461538461 53846156
D3	0.38461538461 538464	0.692307692307 6923	**1**	0.538461538461 5384	0.38461538461 538464	0.30769230769 23077	0.307692307 6923077
D4	0.69230769230 76923	0.538461538461 5384	0.538461538461 5384	**1**	0.53846153846 15384	0.46153846153 846156	0.615384615 3846154
D5	0.69230769230 76923	0.538461538461 5384	0.384615384615 38464	0.538461538461 5384	**1**	0.76923076923 07693	0.769230769 2307693
D6	0.46153846153 846156	0.461538461538 46156	0.307692307692 3077	0.461538461538 46156	0.76923076923 07693	**1**	0.692307692 3076923
D7	0.61538461538 46154	0.461538461538 46156	0.307692307692 3077	0.615384615384 6154	0.76923076923 07693	0.69230769230 76923	**1**

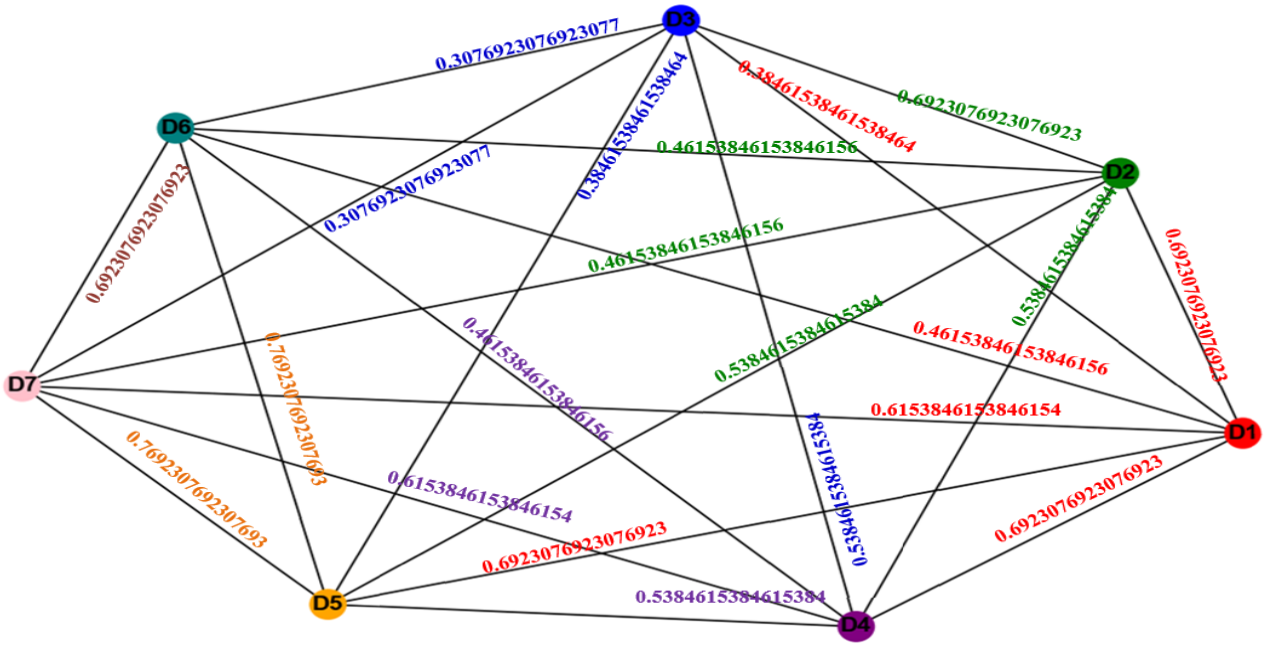

Figure 2. Weighted distance graph

Table 6. Closest documents with similarities

Doc.	Similarities
[0,4,3,1]	[1,0.6923076923,0.6923076923,0.692307692]
[1,2,0,4]	[1,0.6923076923,0.6923076923,0.538461538]
[2,1,3,4]	[1,0.6923076923,0.5384615385,0.384615384]
[3,0,6,4]	[1,0.6923076923,0.6153846154,0.538461538]
[4,6,5,0]	[1,0.7692307692,0.7692307692,0.692307692]
[5,4,6,3]	[1,0.7692307692,0.6923076923,0.461538461]
[6,4,5,3]	[1,0.7692307692,0.6923076923,0.615384615]

Table 7. Pruning clusters

Doc.	Similarities
[0,3,1]	[1,0.6923076923, 0.6923076923]
[1,2,0]	[1,0.6923076923, 0.6923076923]
[2,1]	[1,0.6923076923]
[3,0]	[1,0.6923076923]
[4,6,5]	[1,0.7692307692, 0.7692307692]
[5,4,6]	[1,0.7692307692, 0.6923076923]
[6,4,5]	[1,0.7692307692, 0.6923076923]

Table 8. Merging identical clusters

Doc.	Similarities
[0,3,1]	[1,0.6923076923, 0.6923076923]
[1,2,0]	[1,0.6923076923, 0.6923076923]
[3,0]	[1,0.6923076923]
[2,1]	[1,0.6923076923]
[4,6,5]	[1, 0.7692307692, 0.7692307692]

Table 9. Final clusters

Doc.	Similarities
[0,3,1,2]	[1,0.6923076923, 0.6923076923, 0.6923076923]
[4,6,5]	[1,0.7692307692, 0.7692307692]

After the distance matrix is constructed, the clustering process starts: sorting the distance matrix and determine the number of clusters (clus_no) and the number of the closest documents (close_doc). Let us choose clus_no=2 and close_doc=3. Table 6 represents the closest documents with their similarities.

By applying the proposed clustering process, the clusters are pruned, the identical clusters will be merged, and finally, the sub-clusters will be merged with the closest one. The clustering process is shown in Tables 7-9.

4. RESULTS AND DISCUSSION

Precision, Recall, and F-measure are often employed as evaluation metrics for clustering algorithms. The proportion of data points in a cluster that actually belong to that cluster is measured by precision (P), which is a measure of cluster purity. It is determined by dividing the total number of data points given to that cluster by the ratio of true positives, or accurately assigned data points. A high precision indicates that only a small number of documents are incorrectly assigned to a cluster, while a low precision indicates a significant number of misassigned documents.

Recall is a statistical method that gauges the accuracy of assigning documents to a cluster to assess its completeness. A high recall value signifies that all documents have been correctly assigned to a cluster, while a low recall value indicates that many documents are not correctly assigned.

The F-score balances precision and recall, indicating the accuracy and comprehensiveness of document clustering, with a high F-score indicating appropriate document assignment. The metrics are determined by utilizing Eqs. (11)-(13).

$$P = \frac{\#TP}{\#TP + \#FP} \tag{11}$$

$$R = \frac{\#TP}{\#TP + \#FN} \tag{12}$$

$$F - score = \frac{2(P \times R)}{(P + R)} \tag{13}$$

The experiments are repeated for the datasets using the *TF/IDF and CV* encoding techniques using 200, 300, and 400 features, as well as employing all available features as maximum features for each technique. It is important to mention that we used the same datasets for evaluation, omitting the actual labels during model construction and using them for assessment during the evaluation stage. This guarantees that the algorithms rely only on the informative content of the documents when clustering them, resulting in more unbiased evaluation procedure. Table 10 displays the F-score results.

Table 10. F-score for CNN and OSAC datasets

#	Dataset	Encoding Tech.	#Features	F-score
1	CNN	TF/IDF	All	0.30
			200	0.83
			300	**0.85**
			400	0.60
		TF	All	0.20
			200	**0.75**
			300	0.66
			400	0.54
2	OSAC	TF/IDF	All	0.30
			200	**0.85**
			300	0.73
			400	0.74
		TF	All	0.25
			200	**0.55**
			300	**0.53**
			400	**0.49**

The results of experiments, which are presented in Table 10, reveal that the suggested RSM for document clustering performs differently on CNN and OSAC datasets depending on the selected document representation technique and dimensionality. For instance, the effective clustering outcome is achieved employing TF/IDF with 300 features for CNN (0.85) and 200 features for OSAC (0.85). The results achieved employing TF/IDF outperformed those obtained utilizing CV across all features for both datasets.

The aforementioned observation can be attributed to the innate characteristics of the dataset and the selected representation techniques. The effectiveness of the clustering process is significantly impacted by the class imbalance withing the datasets, where classes do not include an equal number of documents, as well as the difference in document lengths (total number of words) across classes.

Documents with a higher word count will naturally possess more weight when employing counter vectorizer due to their larger vocabulary. This may cause a scenario where the longer documents predominate over the shorter ones in the similarity landscape, thus distorting the grouping process. Conversely, TF/IDF utilizes inverse document frequency to minimize the impact of frequently occurring tokens. This helps mitigate the dominance of lengthy documents and may lead to a more balanced encoding of documents within the feature space.

The effect of feature dimensionality on clustering performance is also noteworthy. Higer feature counts have the ability to capture a richer semantic representation of documents, but they can also add noise and complicate processing. Employing 200–300 feature with TF/IDF produced the best results in this experiment, indicating that this range successfully captures the necessary document properties for clustering without succumbing to the drawback of high dimensionality.

These results emphasize how critical it is to choose dimensionality and document representation techniques carefully in document clustering tasks. The choice can significantly affect the clustering process's efficiency, especially for datasets with imbalanced classes and variable document lengths.

The suggested model is compared with earlier research that is referenced in related works. The results demonstrated that our model outperforms the results of these studies. In a comparison with the work of [23], prior to using k-means, the researcher uses PSO to scan the search space in order to tackle the random centroid algorithm problem. Each particle in the swarm is regarded as a cluster centroid, and the local and global optimal positions are obtained at each iteration in order to minimize the fitness function. The clustering results are evaluated using F-score and entropy. Our model results outperformed the results achieved by this study.

In the study [24], the author used unsupervised and semi-supervised clustering methods for clustering Arabic documents, and k-means and incremental k-means were used for these learning methods. Documents collected from online newspapers and magazines are used as datasets in this study. The F-score is used as an evaluation metric for clustering Arabic text. Our model's results are better than the results achieved by this model.

When comparing our model with the study of [25] that suggested using a combined method of LDA for topic modelling and k-means for text document clustering, we found that our model outperformed the results for this model for the same dataset that we used. The study used different datasets for the clustering model and used LDA for dataset normalization, and compared the results of standard k-means with combined suggested method. It was found that normalized data achieves better clustering results, and our model outperforms their clustering method results.

Another comparison of our model with the approach that suggested by Al-Sarrayrih and Al-Shalabi [26]. Using his own dataset, the author applied FIHC clustering on Arabic documents. Due to the large volume of these itemesets, using FIs will expand the search space and lengthen the computation time. Our suggested approach uses rough set notions to get around these issues.

Mohamed [27] uses different dimension reduction techniques to show the most effective one for the clustering process. The SVD, NMF, and PCA are used as reduction techniques, and the experiments revealed that PCA was the best technique, and its use outperformed other techniques' results in terms of clustering quality, interpretability, and computation efficiency.

The last comparison done is with our previous work. Our suggested algorithm leverages the MFWs as document feature representation. These sets are extracted according to specific user defined minimum support value. The algorthom is evaluated on CNN and OSAC datasets achieving an F-score of 81% and 80% respectively. And our model outperforms their clustering results. The comparison is shown in Table 11.

Obtaining an F-score value of over 80% in Arabic document clustering is a significant achievement, demonstrating the effectiveness of the proposed approach in terms of recall and precision. A score like that suggests:

Outstanding accuracy: Since the F-score is a statistic that balances recall and precision, an F-score of greater than 80% is a reliable indicator of the system's accuracy in correctly clustering Arabic documents**.**

Table 11. Comparison of the proposed algorithm with related works

Ref. No.	Approach	Dataset	F-score	Purity	Accuracy	Entropy
[23]	k-means + PSO	BBC, CNN, OSAC	54%	-	-	-
[24]	k-means + DR	Set of documents	70	-	-	45
[25]	k-means + LDA	CNN	59%	40%	-	-
[26]	FIHC	Own dataset	70%	-	-	-
[27]	k-means, SVD, NMF, PCA	Arabic dataset, Reuters	-	-	66%	-
[28]	MFW	CNN, OSAC	81%	-	-	-
Proposed model		CNN, OSAC	85%	-	-	-

Effective clustering: The recommended approach effectively clusters similar Arabic documents while lowering false positives and misclassifications, as indicated by a high F-score.

Robustness: The method consistently achieves an F-score above 80% across a range of datasets and situations, demonstrating its generalizability and robustness.

Comparative advantage: The suggested algorithm outperforms other algorithms that typically provide F-score values

This work advances the field of ADC and explores its potential real-world applications and impact. The following is a presentation of some advancements in the field of ADC and its potential real-world applications and impact:

(1) Advancements in the field of ADC:

Enhanced clustering accuracy: This research leverages rough set principles to address the inherent complexities of Arabic text, such as rich morphology and complex grammar. By implementing rough set-based algorithms, a higher clustering accuracy compared to previous approaches has been achieved, as demonstrated in the experimental results section. This advancement contributes to more precise and meaningful clustering outcomes, which are critical for various applications in text mining and information retrieval.

Novel feature extraction techniques: This work introduces novel methods for feature extraction that are specifically tailored to the Arabic language. By considering unique linguistic features and employing rough set theory to identify key attributes, our research offers a more refined and effective approach to document representation. This leads to improved clustering performance and sets a new benchmark for future research in ADC.

Scalability and efficiency: The proposed algorithms have been optimized for scalability and computational efficiency, making them suitable for large-scale Arabic text corpora. This addresses a significant limitation in existing ADC techniques, which often struggle with processing large datasets. The enhanced scalability ensures that the methods can be applied to real-world scenarios involving vast amounts of data.

(2) Potential real-world applications and impact:

Information retrieval systems: The improved clustering accuracy and efficiency can significantly enhance the performance of information retrieval systems, such as search engines and digital libraries, specifically for Arabic content. This allows for more accurate and relevant search results, benefiting users by providing better access to information.

Content recommendation engines: By accurately clustering Arabic documents, content recommendation systems can offer more personalized and relevant suggestions to users. This is particularly valuable in e-commerce, online news, and multimedia platforms where content customization is key to user engagement.

Text analytics in social media: The methods developed in this research can be applied to social media analytics, where large volumes of Arabic text are generated daily. Improved clustering can help in sentiment analysis, topic modeling, and trend detection, providing valuable insights for businesses, governments, and researchers.

Automated document management: Organizations dealing with large collections of Arabic documents, such as government agencies, educational institutions, and multinational corporations, can benefit from automated document management systems powered by the proposed clustering techniques. This can streamline document organization, retrieval, and archiving processes, enhancing overall efficiency.

5. CONCLUSIONS

In this paper a novel model based on the rough set theory for clustering Arabic documents is suggested. The model relies on creating a feature table for the pre-processed documents, and then calculating the similarities between document according to the discernibility relation which is represented as weighted distance graph.

Most available Arabic document clustering technique employee k-means algorithm or combination of k-means and other technique. Despite that k-means effective and easy to implement algorithm but it suffers from sensitivity to random initializing point which can negatively impact the clustering results. Additionally, the majority of document clustering methods now in use struggle to handle large document dimensionalities. By utilizing the rough set theory concepts along with CV and TF/IDF document encoding techniques, our proposed algorithm achieved promising clustering results. The proposed algorithm was assessed on two well-known benchmark datasets achieving an F-score of 85% on both the CNN and OSAC datasets.

The study emphasizes the importance of careful choice of dimensionality and document representation techniques in document clustering tasks. Moreover, the effectiveness of the clustering process is significantly impacted by class imbalance and document length differences.

The obtained results contribute to the field of ANLP and encourage us to explore more methods for enhancing clustering Arabic documents. We plan to combine various data mining techniques and measure their effects on the clustering results in our future work.

REFERENCES

[1] Qamar, U., Raza, M.S. (2024). Text mining applications. Applied Text Mining, 53-81. https://doi.org/10.1007/978-3-031-51917-8_3

[2] Han, J., Pei, J., Tong, H. (2023). Chapter 12-Data mining trends and research frontiers. In Data Mining (Fourth

edition), pp. 605-654. https://doi.org/10.1016/B978-0-12-811760-6.00022-9
[3] Mladenić, D. (2017). Text mining. In Encyclopedia of Machine Learning and Data Mining, Springer. https://doi.org/10.1007/978-1-4899-7687-1_831
[4] Preeti. (2021). Review on text mining: Techniques, applications and issues. In 2021 10th International Conference on System Modeling & Advancement in Research Trends (SMART), MORADABAD, India, pp. 474-478. https://doi.org/10.1109/SMART52563.2021.9676285
[5] Zitouni, I. (2014). Natural Language Processing of Semitic Languages. Berlin: Springer. https://link.springer.com/book/10.1007/978-3-642-45358-8.
[6] Kubat, M., Kubat, M. (2021). Unsupervised Learning. In An Introduction to Machine Learning, pp. 297-325. https://doi.org/10.1007/978-3-030-81935-4_15
[7] Sarker, I.H. (2021). Machine learning: Algorithms, real-world applications and research directions. SN Computer Science, 2(3): 160. https://doi.org/10.1007/s42979-021-00592-x
[8] Ezugwu, A.E., Ikotun, A.M., Oyelade, O.O., Abualigah, L., Agushaka, J.O., Eke, C.I., Akinyelu, A.A. (2022). A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects. Engineering Applications of Artificial Intelligence, 110: 104743. https://doi.org/10.1016/j.engappai.2022.104743
[9] Kudo, Y., Murai, T. (2023). Rough-Set-Base Data Analysis: Theoretical Basis and Applications. In Advances in Applied Logics: Applications of Logic for Philosophy, Mathematics and Information Technology, pp. 89-111. https://doi.org/10.1007/978-3-031-35759-6_7
[10] Onu, O.P., Muriana, B. (2024). Rough set theory and its applications in data mining. Technology, 7(1): 84-92. https://doi.org/10.52589/BJCNIT-JAK93DUN
[11] Skowron, A., Dutta, S. (2018). Rough sets: Past, present, and future. Natural Computing, 17: 855-876. https://doi.org/10.1007/s11047-018-9700-3
[12] Baroud, M.M.J., Hashim, S.Z.M., Zainal, A., Ahnad, J. (2020). An new algorithm-based rough set for selecting clustering attribute in categorical data. In 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, pp. 1358-1364. https://doi.org/10.1109/ICACCS48705.2020.9074483
[13] Xia, S., Bai, X., Wang, G., Cheng, Y.L., Meng, D.L., Gao, X.B., Zhai, Y.J., Giem, E. (2022). An efficient and accurate rough set for feature selection, classification, and knowledge representation. IEEE Transactions on Knowledge and Data Engineering, 35(8): 7724-7735. https://doi.org/10.1109/TKDE.2022.3220200
[14] Vidhya, K.A., Geetha, T.V. (2017). Rough set theory for document clustering: A review. Journal of Intelligent & Fuzzy Systems, 32(3): 2165-2185. https://doi.org/10.3233/JIFS-162006
[15] Hasanin, A.M., Ahmed, T.S. (2019). Arabic text mining and rough set theory for decision support system. Journal of Advanced Computer Science and Technology Research.
[16] Mathiyazhagan, B., Liyaskar, J., Azar, A.T., Inbarani, H. H., Javed, Y., Kamal, N.A., Fouad, K.M. (2022). Rough set based classification and feature selection using improved harmony search for peptide analysis and prediction of anti-HIV-1 activities. Applied Sciences, 12(4): 2020. https://doi.org/10.3390/app12042020
[17] Zhang, Q., Xie, Q., Wang, G. (2016). A survey on rough set theory and its applications. CAAI Transactions on Intelligence Technology, 1(4): 323-333. https://doi.org/10.1016/j.trit.2016.11.001
[18] Li, X., Wang, J., Wu, C., Tang, J. (2021). A binary relation base reduction in a relation decision system. In 2021 IEEE International Conference on Advances in Electrical Engineering and Computer Applications (AEECA), Dalian, China, pp. 1030-1033. https://doi.org/10.1109/AEECA52519.2021.9574324
[19] Salloum, S.A., AlHamad, A.Q., Al-Emran, M., Shaalan, K. (2018). A survey of Arabic text mining. In Intelligent Natural Language Processing: Trends and Applications, pp. 417-431. https://doi.org/10.1007/978-3-319-67056-0_20
[20] Ahmed, A.A., Hasan, M.K., Jaber, M.M., Al-Ghuribi, S.M., Abd, D.H., Khan, W., Sadiq, A.T., Hussain, A. (2023). Arabic text detection using rough set theory: Designing a novel approach. IEEE Access, 11: 68428-68438. https://doi.org/10.1109/ACCESS.2023.3278272
[21] Shaalan, K., Siddiqui, S., Alkhatib, M., Abdel Monem, A. (2019). Challenges in Arabic natural language processing. In Computational Linguistics, Speech and Image Processing for Arabic Language, pp. 59-83. https://doi.org/10.1142/9789813229396_0003
[22] Salman, K.A., Khafaji, H.K. (2022). Arabic document clustering: A survey. In 2022 4th International Conference on Current Research in Engineering and Science Applications (ICCRESA), Baghdad, Iraq, pp. 59-64. https://doi.org/10.1109/ICCRESA57091.2022.10352511
[23] Daoud, A.S., Sallam, A., Wheed, M.E. (2017). Improving Arabic document clustering using K-means algorithm and Particle Swarm Optimization. In 2017 Intelligent Systems Conference (IntelliSys), London, UK, pp. 879-885. https://doi.org/10.1109/IntelliSys.2017.8324233
[24] Sangaiah, A.K., Fakhry, A.E., Abdel-Basset, M., El-henawy, I. (2019). Arabic text clustering using improved clustering algorithms with dimensionality reduction. Cluster Computing, 22(Suppl 2): 4535-4549. https://doi.org/10.1007/s10586-018-2084-4
[25] Alhawarat, M., Hegazi, M. (2018). Revisiting k-means and topic modeling, a comparison study to cluster Arabic documents. IEEE Access, 6: 42740-42749. https://doi.org/10.1109/ACCESS.2018.2852648
[26] Al-Sarrayrih, H.S., Al-Shalabi, R. (2009). Clustering Arabic documents using frequent itemset-based hierarchical clustering with an N-grams. In the 4th International Conference on Information Technology, Amman, Jordan.
[27] Mohamed, A.A. (2020). An effective dimension reduction algorithm for clustering Arabic text. Egyptian Informatics Journal, 21(1): 1-5. https://doi.org/10.1016/j.eij.2019.05.002
[28] Salman, K.A., Khafaji, H.K. (2024). A new algorithm for Arabic document clustering utilizing maximal wordsets. Revue d'Intelligence Artificielle, 38(3): 805-813. https://doi.org/10.18280/ria.380307

[29] Saad, M.K., Ashour, W. (2010). OSAC: Open source Arabic corpora. In 6th International Conference on Electrical and Computer Systems (EECS'10), Lefke, North Cyprus, pp. 55. https://doi.org/10.13140/2.1.4664.9288