applied sciences

MDPI

Article

Arabic Text Clustering Using Self-Organizing Maps and Grey Wolf Optimization

Souad Larabi-Marie-Sainte [1], Mashael Bin Alamir [2,]* and Abdulmajeed Alameer [3]

[1] Computer Science Department, College of Computer and Information Sciences, Prince Sultan University, Riyadh 11586, Saudi Arabia; slarabi@psu.edu.sa
[2] Graduate Unit, College of Computer and Information Sciences, Prince Sultan University, Riyadh 11586, Saudi Arabia
[3] Computer Science Department, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia; abalameer@ksu.edu.sa
* Correspondence: mshael.alameer@gmail.com

Abstract: Arabic text clustering is an essential topic in Arabic Natural Language Processing (ANLP). Its significance resides in various applications, such as document indexing, categorization, user review analysis, and others. After inspecting the current work on clustering Arabic text, it is observed that most researchers focus on applying K-Means clustering while hindering other clustering techniques. Our evaluation shows that K-Means has a weakness of inconsistent clustering results and weak clustering performance when the data dimensionality increases. Unlike K-Means clustering, Artificial Neural Networks (ANN) models such as Self-Organizing Maps (SOM) demonstrated higher accuracy and efficiency in clustering even with high dimensional datasets. In this paper, we introduce a new clustering model based on an optimization technique called Grey Wolf Optimization (GWO) used conjointly with SOM clustering to enhance its clustering performance and accuracy. The evaluation results of our proposed technique show an improvement in the effectiveness and efficiency in comparison with state-of-the-art approaches.

Keywords: natural language processing; text clustering; Arabic text; grey wolf optimization; document categorization; topic extraction

check for updates

Citation: Larabi-Marie-Sainte, S.; Bin Alamir, M.; Alameer, A. Arabic Text Clustering Using Self-Organizing Maps and Grey Wolf Optimization. *Appl. Sci.* **2023**, *13*, 10168. https://doi.org/10.3390/app131810168

Academic Editors: Yujin Lim and Hideyuki Takahashi

Received: 28 July 2023
Revised: 29 August 2023
Accepted: 2 September 2023
Published: 10 September 2023

1. Introduction

Clustering text documents is an important field in the area of Natural Language Processing (NLP) as it simplifies the tedious process of categorizing specific documents among millions of resources, especially when metadata such as key phrases, titles, and labels are not available. Text clustering is valuable for different applications, including topic extraction, spam filtering, automatic document categorization, user reviews analysis, and fast information retrieval.

The process of clustering text written in natural languages is complicated, especially for the Arabic language. One of the complications in Arabic is the language's morphological complexity. For instance, a word in Arabic can be written in several forms that might exceed ten forms [1]. Ambiguity is also another major complication in the Arabic language, which is caused by the richness and complexity of Arabic morphology [1,2]. There are various other factors in the Arabic language causing difficulty in text clustering. Among these factors are the different dialects for different regions. Texts from different regions may exhibit significant linguistic variations. Moreover, in the Arabic language, the ordering of words in a sentence provides quite different interpretations for that sentence [3,4].

Several Arabic text clustering techniques have been proposed by researchers to encounter these challenges. Among the various techniques, it has been concluded that the K-Means clustering algorithm is the most widely applied, and that is due to its simplicity and efficiency in comparison with other clustering algorithms [2,5–7]. However, the initiation process of K-Means weakens its accuracy results. The initiation starts with plotting the centers of the clusters randomly and then assigning documents to the nearest center. If the initiation process is inaccurate, then the clustering will be imprecise [8]. Researchers proposed the use of K-Means++, which is an improved algorithm for the initialization process of K-Means [9]. However, our experiments show that even with this smart initialization process, the accuracy of the clustering is low compared to other techniques. Researchers also proposed the use of other clustering techniques, such as Suffix Tree clustering [10] and SOM [11]. Suffix Tree clustering has a limitation of overlapping documents in different clusters [12], while SOM clustering techniques demonstrated high effectiveness in clustering text even with high-dimensional datasets [13–17].

In this paper, we introduce a new optimized SOM clustering approach that utilizes Grey Wolf Optimization (GWO) [18] to enhance the clustering performance and accuracy of the traditional SOM clustering. To the best of our knowledge, the integration of SOM and the GWO algorithm is the first of its kind. Hence, we also investigate its efficiency and effectiveness. We evaluate our proposed approach using different clustering metrics, such as the F1-score, precision, recall, and accuracy. More specifically, the contributions of this paper are as follows:

- A novel Arabic text clustering approach that is based on Self-Organizing Maps (SOM) and Grey Wolf Optimization (GWO).
- An extensive overview of the research that is related to our approach.
- An evaluation of our proposed approach that demonstrates its effectiveness and efficiency in comparison with other clustering techniques.
- A publicly available implementation of our proposed approach.

The remainder of this paper is organized as follows. In Section 2, we give an overview of the research related to our approach. Section 3 provides background information on the components used in our approach. In Section 4, we describe the details of our proposed Arabic text clustering approach. We provide a detailed experimental evaluation in Section 5, and we conclude in Section 6.

2. Related Work

A wide range of papers have been published aiming to enhance the clustering of Arabic text. In the next three subsections, we present an overview of the recent work related to our paper. In Section 2.1, we provide an overview of recent Arabic text clustering techniques. In Section 2.2, we discuss related work that applies SOM in clustering, and in Section 2.3, we provide an overview of the research related to GWO.

2.1. Arabic Text Clustering

Alharwat and Hegazi demonstrated the issue of data mining and data with high dimensions [19]. To overcome the addressed problem, the authors applied modeling techniques to the documents before clustering them. The authors used the Modern Standard Arabic (MSA) dataset [20], which has several versions with different preprocessed articles. The outcome of this study showed that normalized data provided better quality in clustering than unnormalized ones. With normalization, the purity of their clusters was 0.933, and the F1-score was 0.8732. Similar to Alharwat and Hegazi, Al-Azzawy et al. used K-Means to cluster an Arabic dataset corpus which contains 20 documents related to news and short anecdotes [21]. The highest clustering scores for the precision, recall, and F1-measure were 98%, 88%, and 93%, respectively. Mahmood and Al-Rufaye also addressed the problem of the high dimensionality of documents by minimizing the dimensionality of documents using the Term Frequency (TF), Inverse Document Frequency (IDF), and Term Frequency–Inverse Document Frequency (TF-IDF) feature selection approaches [22]. Following that, K-Means and K-Medoids were used for the clustering. The authors implemented their experiment on a 300-document corpus they built. The authors reported that K-Medoids provided more accurate results than K-Means; the first scored 60%, 78%, and 67% for the precision, recall, and F1-measure, respectively, while the second scored 80%, 83%, and 81%, respectively. Another group of researchers used K-Means clustering along with the TF-IDF and Binary Term Occurrence (BTO) feature selection approaches [23]. The authors used a dataset that contains 1121 Arabic tweets. The outcome of their work showed that the BTO feature selection approach outperformed the TF-IDF. The literature for clustering Arabic text using K-Means shows high variation in the performance scores for clustering Arabic text, which could be attributed to the instability and inconsistency of the K-Means clustering algorithm.

To overcome the limitations of the K-Means random initiation of cluster centroids, researchers used PSO-optimized K-Means to cluster Arabic text [24–26]. The use of Particle Swarm Optimization (PSO) contributes to selecting the initial seeds of K-Means. A group of researchers implemented their algorithm for the purpose of Quran verses theme clustering [24], whereas another group [25,26] used three different datasets, named BBC, CNN, and OSAC [27]. The outcome of these research papers demonstrated the effectiveness of applying optimization methods for enhancing the accuracy of the clustering models used.

Another work on clustering Arabic documents was based on the sentiment orientation and context of words in the data corpus [5]. The authors used the Brown clustering algorithm on user reviews of several topics, such as news, movies, and restaurants. The data in this research were collected from several sources [28–31]. The evaluation results of this approach showed that the subjectivity and polarity of the clustering documents provided rates of 96% and 85%, respectively. The evaluation results indicated that the number of clusters also affects the accuracy rates, showing that fewer clusters provide better results.

In another work [2], the authors used a combination of Markov Clustering, Fuzzy-C-Means, and Deep Belief Neural Networks (DBN) in an attempt to cluster Arabic documents. Two datasets were used in this study; the first was acquired from the Al-Jazeera news website with 10,000 documents and the second from a Saudi Press Agency [32] with 6000 documents. The clustering precision, recall, and F1-measure resulted in 91.2%, 90.9%, and 91.02%, respectively. The model that was used was highly impacted by the feature selection of the root words leading to imprecise clustering results.

Al-Anzi and Abuzeina [11] used Expectation-Maximization (EM), SOM, and K-Means algorithms to cluster Arabic documents. They built a corpus of 1000 documents extracted from a Kuwaiti newspaper website called Alanba [33]. The documents cover different topics, such as health, technology, sports, politics, and others. The authors then compared the evaluation of the three clustering algorithms. They reported that SOM obtained the highest accuracy between the three algorithms with a rate of 93.4%. From this study, it appears that the use of SOM in clustering Arabic text is promising.

The Bond Energy Algorithm (BEA) was also used by researchers to cluster Arabic text [34]. The results of this study showed that the BEA algorithm outperforms K-Means clustering in terms of precision, recall, and the F1-score.

In the broader field of text clustering, researchers also proposed the use of prototype-based models for text clustering [35]. The results of this work showed that it outperforms K-Means clustering.

To conclude, most of the current work on Arabic text clustering used K-Means clustering because it is a simple model and can be applied easily. However, the mechanism that K-Means follows has limitations. For instance, K-Means first initiates centers of clusters and then assigns documents to these clusters. If the initiation process of K-Means is not well formulated, then the risk of incorrect clustering arises. Moreover, techniques that integrate K-Means clustering with Particle Swarm Optimization [26] have promising results. This shows that optimization contributes positively to clustering models. In addition, previous work showed that the use of SOM provided better clustering results than K-Means for Arabic text [11]. We hypothesize that integrating SOM with an optimization method would result in better clustering as we are presenting in this paper. Table 1 presents a summary of the recent work regarding Arabic text clustering.

Table 1. Arabic text clustering related work comparison.

Ref.	Model	Dataset	Purity	F1-Score	Precision	Recall	Accuracy
[19]	K-Means	MSA	93.3%	87.32%	87.13%	87.52%	-
[21]	K-Means	Own corpus	-		93%	98%	88%
[26]	K-Means + (PSO)	BBC, CNN, OSAC	50%	47%	33%	-	-
[5]	Brown clustering algorithm	Own corpus	85%	-	-	-	-
[25]	K-Means	Arabic tweets	76.4%	-	-	-	-
[23]	TF-IDF + BTO	Arabic tweets	-	-	-	-	-
[22]	K-Medious	Own corpus	-	67%	60%	78%	-
[2]	Markov + Fuzzy-C-Means + DBN	Own corpus	-	91.02%	91.02%	90.9%	-
[10]	Suffix Tree	Own corpus	-	81.11%	80.3%	83.75%	-
[11]	SOM	Own corpus	-	-	-	-	93.4%

2.2. *Self-Organizing Maps*

Researchers have applied the Self-Organizing Maps (SOM) clustering algorithm in several domains, such as speech recognition [36], medical imaging and analysis [37], classification of satellite images [38], and others. The following presents some applications of the SOM algorithm.

He et al. attempted to resolve the issue of the sudden disabling of electronic car batteries [39]. The authors used SOM to provide the battery's performance. SOM were used to cluster the characteristics of the battery, including the battery's capacity, temperature, voltage, lifespan, internal resistance, and self-discharge rate. Afterward, the result of the clustering provides the driver with different information about the battery and its usage. Also, it alerts the driver when the battery should be replaced. The authors in this study did not mention the accuracy rate of their clustering. However, they compared the K-Means clustering algorithm with SOM and concluded that the latter outperformed K-Means.

Bara et al. applied the SOM algorithm to analyze students' E-Learning activities [40]. SOM are applied to provide clusters for the E-Learning activities to investigate the relation between the students' activities regarding the E-Learning portal and their academic performance. The authors obtained a dataset from the Universiti Teknologi Malaysia (UTM) Moodle LMS log records. Then, SOM were applied to cluster students according to their E-Learning activities. In their evaluation, the authors observed a correlation between the students' performance and their E-Learning activities, showing that the students' performance is affected by the activities positively.

SOM were also used by Simon and Elias to detect fake followers on Twitter [41]. The authors applied their model to a dataset of fake followers provided by the Institute of Informatics and Telematics of the Italian National Research Council. The dataset uses the accounts' related features to categorize the type of user, whether fake or real. The features used include the following count, followers count, favorites count, and others. The use of SOM in this study showed its effectiveness in detecting fake accounts.

Mei et al. used SOM to detect damaged lesions in the brain from the Magnetic Resonance Imaging (MRI) scan for Relapsing Remitting Multiple Sclerosis (RRMS) patients [42]. SOM are applied to cluster lesions based on the possessed damage. The authors used a dataset of 10 patients to perform their study. They concluded the effectiveness of analyzing the MRI scans through SOM automatically.

Similarly, in the research by Sarmiento et al. [43], the authors used SOM clustering as a disease early diagnosis tool. The authors used SOM to allocate significant gene paths in the human body. Gene paths can aid in the early diagnosis of different diseases, such as diabetes, heart diseases, and others. The authors used a dataset provided by the Kyoto Encyclopedia of Genes and Genomes (KEGG). They also applied the K-Means clustering algorithm but concluded that SOM performed better than K-Means.

To conclude, the discussed related work of SOM shows its applicability to various research areas. In addition, many researchers performed a comparison between SOM and

K-Means clustering algorithms [26,39,43]. These researchers mainly found that SOM are more effective than K-Means in clustering.

2.3. Grey Wolf Optimization Algorithm

The GWO algorithm has been used in several research areas, including engineering design, economy, astronomy, and others. Since its introduction [18], GWO has gained widespread attention from researchers due to its effectiveness in solving complex optimization problems. The following presents some of its applications.

Guo et al. used an optimized learning algorithm model to provide optimized solutions for mathematical nonlinear equation problems [44]. The authors compared the use of two optimization algorithms, GWO and PSO, and their results showed that GWO outperformed PSO.

In another work, the authors compared the use of several optimization algorithms including GWO, the Genetic Algorithm (GA), Ant-Lion Optimization (ALO), the Krill Herd Algorithm (KHA), and others for solving the problem of Combined Economic and Emission Dispatch (CEED) [45]. The result of this study showed that the use of GWO achieved higher performance and better solutions. Xiao et al. also compared the use of GWO with Particle Swarm Optimization (PSO) along with a machine learning classifier to detect far orbits through the extraction of image features [46]. The PSO-based classifier performance results were low due to the computational complexity, whereas the results of the GWO-based classifier outperformed the PSO-based classifier by 8%. The discussed research shows the effectiveness of using GWO for enhancing the performance of machine learning models.

Researchers have also employed GWO-based models to discover optimal solutions for various engineering problems. For instance, a group of researchers applied GWO to a civil engineering problem involving the design of water distribution networks. The objective was to minimize financial costs and reduce the number of network components, including pipe sizes, pump ratings, and other elements. This approach met the established expectations for both performance and cost perspectives [47]. On the other hand, Majeed and Rao [48] built a GWO-based model to automate the process of designing analog circuits. Through this application, they effectively showcased the utility of GWO by producing enhanced circuit designs in a minimal amount of time.

In general, several researchers used optimization techniques to improve the results of different clustering approaches [49–52]. The results of these studies support our hypothesis that using optimization with clustering techniques can help in improving their effectiveness.

To conclude, the use of the GWO algorithm spans several research areas, including medical, engineering, astronomy, and others. The discussed studies show that the use of GWO enhances the performance and the accuracy of the results for the defined problems.

3. Background

To gain an understanding of the clustering model proposed in this paper, brief details about its components are discussed in this section as follows.

3.1. Self-Organizing Maps

SOM are an unsupervised type of Artificial Neural Network (ANN) [53]. It uses an unsupervised learning model to create a map of different groups [54]. As illustrated in Figure 1, SOM consist of two main layers: the input layer where inputs or neurons are inserted, and the output layer, also called a competitive layer, where groups of similar inputs are formed. The outputs of the SOM are generated by multiplying the inputs with SOM weights. These weights are randomly initiated when training a new network, and then at each iteration, the weights are updated in accordance to Equation (1). Correspondingly, Equation (2) is used to update the neighborhood function $\theta(t)$ in which the SOM network topology is defined.

$$w_{ij}(t+1) = w_{ij}(t) + \alpha(t) \times \theta(t) \cdot (x(t) - w_{ij}(t)) \tag{1}$$

- w_{ij}: the weights.
- t: the current iteration.
- $\alpha(t)$: the learning rate.
- $\theta(t)$: the neighborhood function.
- $x(t)$: the input

$$\theta(t) = exp\left(- \frac{distance\ from\ BMU^2}{2\alpha(t)^2} \right) \tag{2}$$

- BMU: The Best Matching Unit reflecting the closest weight for the input instance.

There are two crucial attributes behind the effectiveness of the SOM algorithm. The first attribute is its capability to diminish the input space by moving similar inputs close to eventually form clusters of outputs. The second attribute is forming topological ordering which is based on the location of the neurons in the SOM grid. This ordering is correlated to the input space features [38].

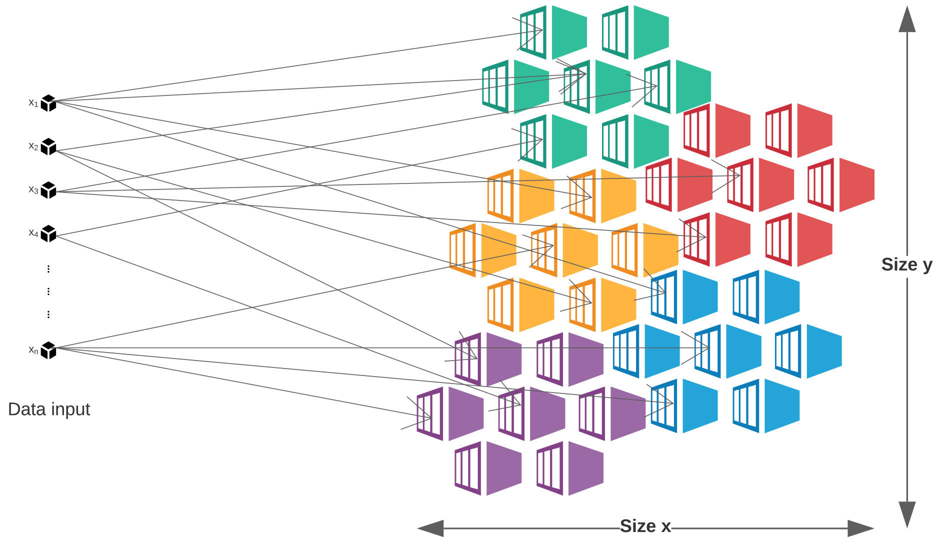


Figure 1. Self-Organizing Maps' Structure.

3.2. Grey Wolf Optimization Algorithm

Grey Wolf Optimization (GWO) is a nature-inspired meta-heuristic algorithm that reflects the social behavior of grey wolves when hunting to solve optimization problems [18]. Particularly, it reflects the hierarchy of leadership and hunting in grey wolves' packs. The hierarchy can be represented as a pyramid: The leader wolf, alpha, making all the decisions, is on top of the pyramid. The second level has the beta, assisting the alpha wolf in their decisions, and it can also substitute the place of the alpha when required. The level after that is called delta, which has the responsibility of protecting the tribe. The lowest level has several wolves, omega, that are dominated by all the other types in the pyramid. The Grey Wolf Optimization (GWO) algorithm consists of these steps:

- Initialization: Initialize a population of wolves representing potential solutions to the optimization problem. The number of wolves and their positions are generated randomly.
- Fitness evaluation: Evaluate the fitness of each wolf in the population by applying the fitness function of the optimization problem.
- Alpha, beta, delta, and omega determination: Identify the alpha, beta, and delta wolves based on the population fitness values.
- Update positions of the wolves: The beta and delta wolves adjust their positions based on their current position and the position of the alpha wolf, while the omega wolves explore new positions by more random movement in the solution space.

- Repeat the second step: Evaluate the fitness of the population again after the position updates and determine the new alpha, beta, delta, and omega wolves based on their new fitness values.
- Termination criteria: Check if the termination criteria are met. The termination criteria might include a maximum number of iterations or a fit-enough solution is obtained.

Once the termination criterion is met, the alpha wolf represents the best solution found by the GWO algorithm, which can be used as the optimal solution to the given optimization problem.

4. Proposed Approach

In this section, we present our proposed approach for clustering Arabic text, leveraging the GWO algorithm to enhance the clustering results for SOM. Instead of relying on the default random initiation process, our approach involves optimizing the clustering process by fine-tuning the SOM's initial weights. Our proposed approach has two main phases. First, we run GWO to find the optimized SOM initial weights. Then, we use the output of the previous phase to adjust the SOM's initial weights and run it. The following are the details of these two phases.

4.1. Phase1: Grey Wolf Optimization Algorithm

In order to run the GWO algorithm, we have to set the parameters that are required to execute it. GWO has six parameters as follows:

- Fitness function.
- Dimension.
- Lower bound.
- Upper bound.
- Dataset.
- Number of search agents.

Defining the fitness function for GWO is critical as it must be aligned with its purpose which, in this paper, is minimizing the clustering Quantification Error (QE) value. We compute the fitness function by running the SOM clustering with only ten epochs and computing the clustering QE. The dimension parameter represents the SOM's weights' shape. It is calculated by multiplying the number of features in the dataset with the SOM's dimension. Both values, the number of features and SOM dimension, are obtained from the dataset metadata [18,55]. The lower and upper bound values represent the range of possible SOM weights which is $[-1, 1]$. Furthermore, the dataset parameter represents the actual dataset that we are using for the clustering. The last parameter is the number of search agents which is chosen empirically and varies depending on the problem being solved (see Section 5 for details).

After setting the parameters, we run the GWO algorithm until the stopping criterion is met, which we define in our approach as having five consecutive iterations without improvement in the fitness value. By the end, the GWO algorithm returns the position of the alpha wolf which has the best fitness value. The position of the alpha wolf reflects the initial weight to be used for the SOM clustering algorithm.

4.2. Phase2: Self-Organizing Maps Optimization

Figure 2 illustrates the abstract idea of optimizing the SOM algorithm. In the second phase, we use the value of the best solution in the first phase as an initial weight for the SOM algorithm. Our hypothesis is that using the values that were obtained from the GWO algorithm will cause the SOM clustering algorithm to converge faster and provide better clustering results.

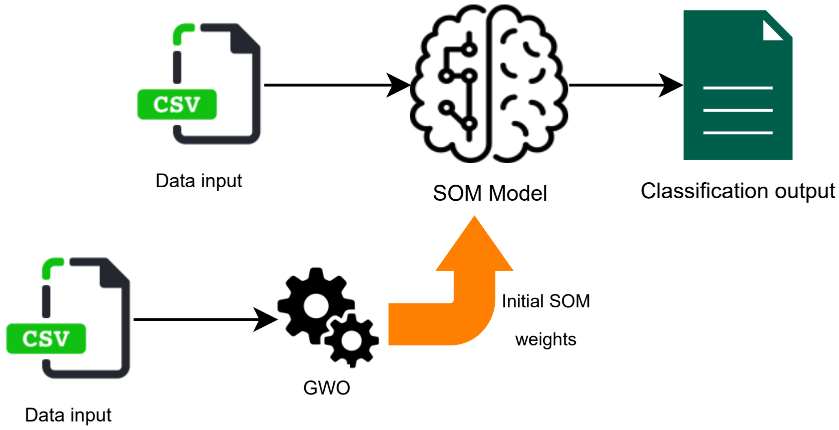


Figure 2. Using Grey Wolf Optimization to Optimize Self-Organizing Maps.

In order to implement the optimized version of SOM, we used a Python library called MiniSom [55]. The process of clustering starts with setting the initial weights of the SOM to the weights computed in the first phase. Then, we run the SOM clustering using different parameters' values, which we empirically evaluate in Section 5.

5. Experimental Evaluation

To assess the effectiveness of our proposed model for clustering Arabic text, we implemented it to evaluate its accuracy and efficiency. The full source code of our implementation is publicly available to ensure the reproducibility of our results [56]. In this section, we address the following research questions:

- RQ1: What is the effectiveness of GWO-optimized SOM in clustering Arabic text compared to other models?
- RQ2: How efficient are GWO-optimized SOM in clustering Arabic text compared to other models?
- RQ3: What is the impact of data representation techniques on the effectiveness and efficiency of clustering Arabic text?

To answer these research questions, we carried out an empirical evaluation on two Arabic datasets, the MSA dataset [25] and NADA dataset [57], and compared our results with two other models for clustering Arabic text, K-Means and standard non-optimized SOM. Figure 3 shows an illustration of the experiment process we used in this paper. The experiment starts with collecting data sources for Arabic text. Then, each of these sources is preprocessed and represented using two different representation techniques. After that, the outcome of the representation is trained using three clustering techniques, K-Means, SOM, and GWO-optimized SOM (our model). The final step is to evaluate the clustering in terms of the accuracy, F1-score, precision, recall, and training time for each experiment. All the experiments are executed using a MacBook Pro with a 2.3 GHz Intel Core i5 processor and 8 GB 2133 MHz LPDDR3 RAM. The following subsections show the details of each of these steps.

5.1. Data Collection

For the purpose of evaluating our proposed model, we carried out our experiments in two datasets, the MSA corpus [20] and NADA corpus [57]. Note that both datasets are processed in the same way.

The MSA corpus includes nine subjects related to technology, sports, religion, politics, literature, law, health, economy, and art. There are five versions of each category: The first has the original dataset, and the second contains the data after removing stop words and punctuation. The third and fourth versions contain the data after applying a stemmer. The last version contains the data after extracting the roots. Each category contains 300 documents which results in 2700 documents for each version. In our experiments, we used the first version and applied our own preprocessing procedure (described in Section 5.2). Table 2 presents the subject distribution for the MSA corpus.

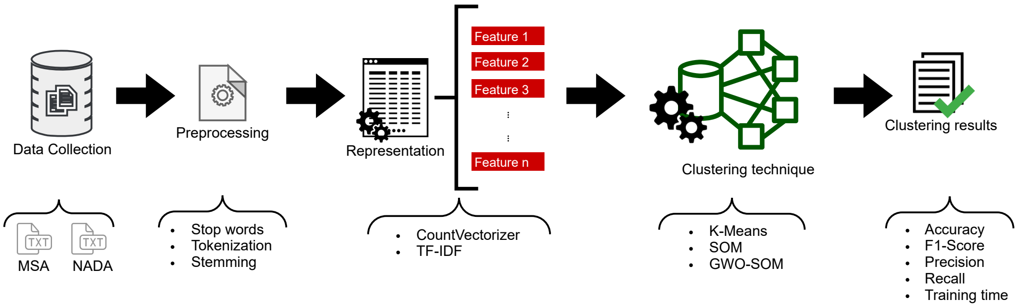


Figure 3. Experiment process.

Table 2. MSA corpus subject distribution.

Category	Number of Text Files
Technology	300
Religion	300
Politics	300
Law	300
Sports	300
Art	300
Literature	300
Economy	300
Health	300
Total	2700

The second corpus is NADA. It is a new corpus built by integrating topics from other Arabic corpus sources, such as OSAC [27] and the Diab Dataset [58]. NADA includes ten categories of text files which are about Arabic literature, economical social sciences, political social sciences, law, sports, art, Islamic religion, computer science, health, and astronomy, with a total number of 7310 text files. The distribution of the text files is illustrated in Table 3.

Table 3. NADA corpus subject distribution.

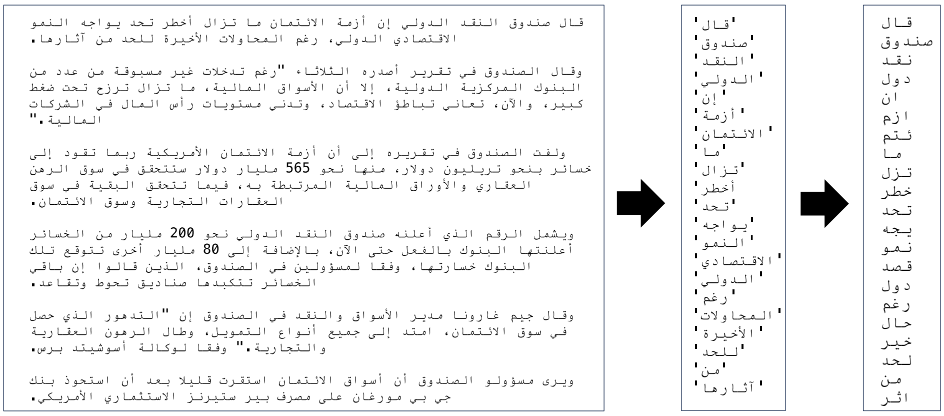
Category	Number of Text Files
Arabic Literature	400
Economical Social Sciences	1307
Political Social Sciences	400
Law	1644
Sports	1416
Art	400
Islamic Religion	515
Computer Science	400
Health	428
Astronomy	400
Total	7310

5.2. Data Preprocessing

After collecting the data, the second step is preprocessing the datasets. The preprocessing of Arabic text poses significant challenges due to the language's complex morphological structure and syntactical rules. In Arabic, a single word can comprise multiple independent tokens, and morphological knowledge of the language needs to be incorporated into the tokenizer [59]. Stemming is also challenging given the diverse range of morphological configurations and diacritics in the language. To address these challenges, our preprocessing has two stages:

- Text tokenization.
- Text stemming and morphological analysis.

In our experiments, cleaning and preprocessing the datasets is executed using PyArabic [60] and ISRIStemmer [61]. Figure 4 presents a sample text file before and after the preprocessing phase.



قال صندوق النقد الدولي إن أزمة الائتمان ما تزال أخطر تحد يواجه النمو
الاقتصادي الدولي، رغم المحاولات الأخيرة للحد من آثارها.

وقال الصندوق في تقرير أصدره الثلاثاء "رغم تدخلات غير مسبوقة من عدد من
البنوك المركزية الدولية، إلا أن الأسواق المالية، ما تزال ترزح تحت ضغط
كبير، والآن، تعاني تباطؤ الاقتصاد، وتدني مستويات رأس المال في الشركات
المالية."

ولفت الصندوق في تقريره إلى أن أزمة الائتمان الأمريكية ربما تقود إلى
خسائر بنحو تريليون دولار، منها نحو 565 مليار دولار ستتحقق في سوق الرهن
العقاري والأوراق المالية المرتبطة به، فيما تتحقق البقية في سوق
العقارات التجارية وسوق الائتمان.

ويشمل الرقم الذي أعلنه صندوق النقد الدولي نحو 200 مليار من الخسائر
أعلنتها البنوك بالفعل حتى الآن، بالإضافة إلى 80 مليار أخرى تتوقع تلك
البنوك خسارتها، وفقا لمسؤولين في الصندوق، الذين قالوا إن باقي
الخسائر تتكبدها صناديق تحوط وتقاعد.

وقال جيم غارونا مدير الأسواق والنقد في الصندوق إن "التدهور الذي حصل
في سوق الائتمان، امتد إلى جميع أنواع التمويل، وطال الرهون العقارية
والتجارية." وفقا لوكالة أسوشيتد برس.

ويرى مسؤولو الصندوق أن أسواق الائتمان استقرت قليلا بعد أن استحوذ بنك
جي بي مورغان على مصرف بير ستيرنز الاستثماري الأمريكي.

'قال'
'صندوق'
'النقد'
'الدولي'
'إن'
'أزمة'
'الائتمان'
'ما'
'تزال'
'أخطر'
'تحد'
'يواجه'
'النمو'
'الاقتصادي'
'الدولي'
'رغم'
'المحاولات'
'الأخيرة'
'للحد'
'من'
'آثارها'

قال
صندوق
نقد
دول
ان
ازم
ئتم
ما
تزل
خطر
تحد
يجه
نمو
قصد
دول
رغم
حال
خير
لحد
من
اثر

Figure 4. A sample Arabic text preprocessed using the tokenization and stemming steps.

5.2.1. Text Tokenization

This step is used to separate and identify words in the text by breaking down the words that comprise multiple tokens, eliminating white spaces, punctuation marks, and mark-ups. Text tokenization defines the word boundaries as described in [59].

5.2.2. Stemming and Morphological Analysis

Stemming is used to eliminate the suffix and prefix from the words in the text files. The steps to perform the stemming are as follows:

1. Remove the diacritics from the Arabic word.
2. Remove the prefixes from the Arabic word.
3. Remove the suffixes from the Arabic word.
4. Remove the connective letters from the Arabic word.
5. Normalize the initial Hamza to bare Alif.

Following the stemming is the morphological analysis which returns the root of the words. The Stem function from the ISRIStemmer library [61] combines the stemming and the morphological analysis by directly returning the root of the words.

5.3. Data Representation

The outcome of the preprocessing phase is used for representation. Data representation is an essential phase because datasets cannot be executed directly by the clustering algorithm. In this phase, characters are mapped into predefined vectors of real numbers, meaning that the words in the text files are mapped into vectors using a Term Frequency model. Two representation methods are used, the CountVectorizer (CV) and Term Frequency–Inverse Document Frequency (TF-IDF) Vectorizer.

5.3.1. CV

The representation CountVectorizer (CV) counts the words in a document. The CV is a simple technique that provides satisfactory results. This representation method counts the number of times a word appears in a document and uses this count as a weight for the word. The final output of this representation method is a matrix of the words in a document along with the number of occurrences [62].

5.3.2. TF-IDF Vectorizer

Term Frequency–Inverse Document Frequency (TF-IDF) [63] is a representation that measures how relevant a word is to a document in a collection of documents. The term has two parts which are multiplied with each other. The first is the Term Frequency (TF), also called Term Score, which considers all the files in the dataset as a bag of words. The TF is calculated by dividing the number of word occurrences in a document by the total number of words in the same document. The second part is the Inverse Document Frequency (IDF) which is calculated by obtaining the exponential logarithm of the total number of documents in the dataset divided by the number of documents having a specified word w.

5.4. Clustering Techniques

The output of the data representation phase is the input for the next phase which is the clustering. Clustering is performed to identify the different clusters each document belongs to. Because there are two representation techniques, CV and TF-IDF, as discussed in Section 5.3, we apply the clustering techniques to all representations for the purpose of our evaluation. The clustering techniques we used in our evaluation are as follows:

1. The K-Means clustering technique.
2. The Self-Organizing Maps clustering technique.
3. The Grey Wolf-Optimized Self-Organizing Maps clustering technique (our proposed model).

In the following, we show the details of the implementation for each clustering technique.

5.4.1. K-Means Clustering Technique

K-Means is a clustering algorithm that provides k numbers of clusters. Its mechanism is based on placing center points and each center represents a cluster. Data are passed to the K-Means algorithm and then the distance between the centers and each sample of data is calculated. The sample is then assigned to the center with the shortest distance. Applying K-Means for the two datasets is straightforward because the number of clusters for each dataset is known. For the purpose of our experiment, we use the implementation of K-Means that is available in the scikit-learn Python library [64]. This implementation of K-Means uses K-Means++ initialization, which is an enhanced initialization process for K-Means. Experiments have shown that K-Means++ initialization has improved performance compared to the standard K-Means [9].

5.4.2. Self-Organizing Maps Clustering Technique

The mechanism in which the Self-Organizing Maps (SOM) work is discussed in Section 3.1. For the purpose of our experiment, we use the implementation of SOM that is available in the MiniSom Python library [55]. Clustering using SOM requires setting six parameters:

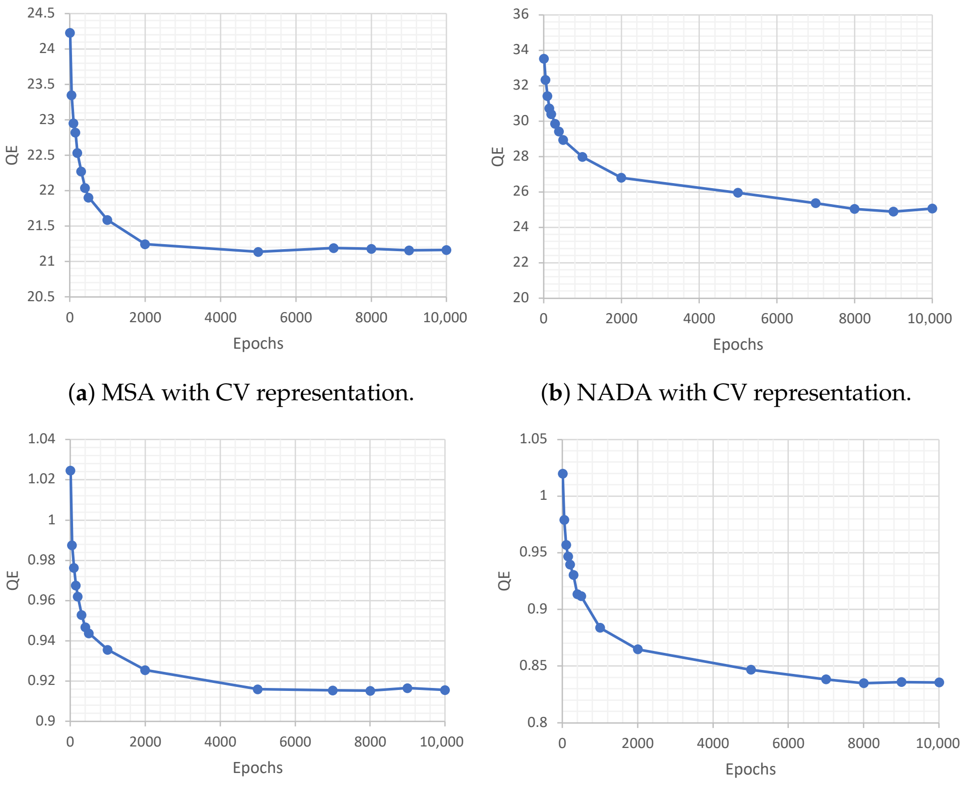
- dim_x: the x dimension of the map.
- dim_y: the y dimension of the map.
- D: the number of features in the dataset.
- sig: the sigma value representing the radius of the different neighbors in the SOM representing the different categories.
- $n(t)$: the learning rate value.
- t: the number of iterations (epochs).

For the purpose of tuning the parameters of the SOM, the dataset was divided into three parts, training, validation, and testing, with 60%, 20%, and 20%, respectively. The first two parts were used to tune the parameters while the last part was used to generate the final clustering results.

The first two parameters, dim_x and dim_y, define the dimension of the map. There are two methods to set the SOM's dimension. The first method is considered to be a rule of thumb where the dimension is obtained from the number of features in the dataset using Equation (3).

$$\sqrt{5 \times \sqrt{D}} \tag{3}$$

On the other hand, the second method requires tuning the SOM to obtain the least consistent QE value. We used both methods and concluded that the first method where the dimension is obtained from Equation (3) provided better clustering results. The third parameter is the number of features in the dataset which is equivalent to the number of columns in the dataset's representation version. The sigma and learning rate have default values of 1 and 0.5 consequently. The final parameter represents the number of epochs for training the clustering model. For each dataset, the number of training epochs was tuned while fixing the other five parameters. The numbers of tested epochs were 10, 50, 100, 150, 200, 300, 400, 500, 1000, 2000, 5000, 7000, 8000, 9000, and 10,000. For each epoch, we observed the relation between the SOM's dimension and the QE. Eventually, the number of epochs is selected when the QE reaches the lowest consistent value. Figure 5 shows the results for fine-tuning the number of epochs parameter.



(**a**) MSA with CV representation.

(**b**) NADA with CV representation.

(**c**) MSA with TF-IDF representation.

(**d**) NADA with TF-IDF representation.

Figure 5. Results of fine-tuning the number of epochs parameter for Self-Organizing Maps.

5.4.3. Grey Wolf-Optimized Self-Organizing Maps Clustering Technique

The mechanism we used to optimize the SOM is described in Section 4. The following are the steps we used for evaluating the optimized SOM. Note that these steps are executed on all datasets (MSA and NADA) with both representation methods (CV and TF-IDF), and 10-fold cross-validation is performed to validate the obtained results:

1. Divide the dataset into three portions: 60% for training, 20% for validation, and 20% for testing.
2. Set the parameters of the GWO as follows:

- Fitness function: We used the QE function provided by the MiniSom python library as a fitness function for the Grey Wolf Optimization algorithm.
- Dimension: We set the dimension to be the product of the total number of corpus features by the selected SOM dimensions, dim_x and dim_y.
- The number of search agents: For the purpose of this paper, we chose the number of search agents to be five. Note that the initial experiments with different numbers of search agents were performed and the difference between their results was negligible.

3. Initiate the positions of the most powerful wolves, alpha, beta, and delta. The initial positions are set as a matrix of 0 s.
4. Initiate random values for the search agents with the same GWO dimension.
5. Calculate the fitness value using the QE, on the validation data portion, provided by the MiniSom Python library.
6. With every fitness calculation, the result is compared against the alpha, beta, and delta wolves. Accordingly, if the new fitness is less than the values of these wolves, update the wolves' values with the new fitness.
7. When a stopping criterion is met, whether completing the maximum number of iterations or having no improvement in fitness for five consecutive iterations, the GWO function terminates by passing the value of the fitness into the MiniSom to train the model.
8. Test the model using the testing data portion and report the results as shown in Table 4.

Table 4. Final clustering results.

Dataset	Representation	Clustering Technique	F1-Score	Precision	Recall	Accuracy	Time in Minutes
MSA	CV	K-Means	73.00%	75.00%	89.60%	73.00%	-
	TF-IDF	K-Means	90.80%	89.80%	94.20%	93.80%	-
	K-Means in [19]		87.32%	87.13%	87.52%	-	-
	CV	SOM	86.59%	87.06%	86.66%	86.70%	07:18
	TF-IDF	SOM	93.37%	93.54%	93.33%	93.30%	06:27
	CV	SOM + GWO	98.14%	98.21%	98.14%	98.00%	07:00
	TF-IDF	SOM + GWO	98.33%	98.36%	98.33%	98.30%	04:48
NADA	CV	K-Means	44.30%	34.30%	31.90%	44.30%	-
	TF-IDF	K-Means	57.70%	60.20%	71.60%	57.70%	-
	CV	SOM	91.84%	92.01%	91.79%	91.80%	54:27
	TF-IDF	SOM	93.33%	93.47%	93.29%	93.30%	18:22
	CV	SOM + GWO	98.70%	98.77%	98.70%	98.70%	18:10
	TF-IDF	SOM + GWO	98.44%	98.53%	98.42%	98.40%	16:00

5.5. Results and Discussion

In this section, we present the results of our experiments and discuss our research questions. Table 4 shows the overall results of our experiments that we executed as we described in the previous subsections, noting that we performed 10-fold cross-validation on the training and validation data in our experiments. The following is a discussion of each research question with more details.

To answer RQ1, we used four different metrics to evaluate the effectiveness of GWO-optimized SOM in comparison with K-Means clustering and traditional SOM. These metrics are the F1-score, precision, recall, and accuracy. As can be seen in Table 4, each of the clustering techniques is evaluated for both datasets, MSA and NADA, and also with both representation methods. We also show our results in comparison with the K-Means clustering implementation in [19]. As highlighted in the table, optimizing the SOM using the GWO algorithm resulted in improving the clustering for both datasets with both representation methods. The results show a consistent improvement in all the effectiveness metrics we used in our evaluation, indicating the effectiveness of the introduced model.

The use of GWO for initializing the weights for SOM allows it to explore a larger space of possible initializations, increasing the chance of finding a more suitable starting point that leads to better final clustering results.

To answer RQ2, we used the training time as a metric to evaluate the efficiency of GWO-optimized SOM in comparison with K-Means clustering and traditional SOM. As can be seen in Table 4, GWO-optimized SOM are faster than traditional SOM in training time for both the MSA and NADA datasets and in both the CV and TF-IDF data representations. We also noticed that the training time for all the clustering approaches is significantly higher in the NADA corpus when compared against the MSA corpus. This is mainly due to the larger size of the NADA corpus. After further investigation of the training time of the SOM and optimized SOM techniques, we found that the overhead of finding the best weights in the optimized SOM is high, but the SOM training time after initializing the weights is significantly reduced, resulting in an overall shorter training time compared to traditional SOM. As can be seen in Table 5, initializing the weights consists of about 30% of the overall running time for the GWO-optimized SOM.

Table 5. Breakdown of the GWO-optimized SOM running time (in minutes).

Dataset	Representation	Weights Initialization (GWO Optimization)	Training	Total Time
MSA	CV	02:23	04:37	07:00
	TF-IDF	01:38	03:10	04:48
NADA	CV	05:00	13:00	18:00
	TF-IDF	04:53	12:08	17:01

To answer RQ3, we compared the two representations, TF-IDF and CV, using the same metrics as in RQ1 and RQ2. As can be seen in Table 4, we observe that the TF-IDF representation technique reported better effectiveness for both datasets in all the evaluated metrics. However, in the GWO-optimized SOM clustering technique, the difference in the effectiveness results for the two representations was negligible. When evaluating the efficiency of the two representation techniques, we found that TF-IDF representation is overall more efficient than CV representation.

6. Conclusions

In this paper, we introduced a new model for clustering Arabic text. The model is consolidated by SOM and the GWO algorithm. The purpose of applying GWO is feeding SOM with new weights during the initiation of the network. In our experiment, the model was tested on two datasets, MSA and NADA. To obtain an equitable clustering evaluation, several experiments were executed on the same datasets. These included using SOM independently, with optimization, and also using different clustering techniques (i.e., K-Means). After evaluating all the results, we concluded that our introduced model resulted in a significant improvement in clustering accuracy and training time. We hope these results advance the broader Arabic Natural Language Processing (ANLP) field by giving researchers insights on how to improve the results of the clustering they might apply in their research. In the future, we plan to evaluate optimization techniques other than GWO and measure their impact for clustering Arabic texts. We also plan to gather new Arabic datasets encompassing a wider range of topics and subtopics. These additions will allow us to conduct more comprehensive evaluations, ultimately enriching the field of Arabic Natural Language Processing (ANLP).

Author Contributions: Conceptualization, M.B.A. and S.L.-M.-S.; methodology, M.B.A.; software, M.B.A.; validation, M.B.A., S.L.-M.-S. and A.A.; formal analysis, M.B.A.; investigation, M.B.A.; resources, M.B.A.; data curation, M.B.A.; writing—original draft preparation, M.B.A.; writing—review and editing, A.A.; visualization, M.B.A.; supervision, S.L.-M.-S.; project administration, S.L.-M.-S.; funding acquisition, A.A. All authors have read and agreed to the published version of the manuscript.

Funding: The authors would like to acknowledge the support of Prince Sultan University for paying the Article Processing Charges (APC) of this publication.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Farghaly, A.; Shaalan, K. Arabic natural language processing: Challenges and solutions. *ACM Trans. Asian Lang. Inf. Process. (TALIP)* **2009**, *8*, 14. [CrossRef]
2. Jindal, V. A Personalized Markov Clustering and Deep Learning Approach for Arabic Text Categorization. In Proceedings of the ACL 2016 Student Research Workshop, Berlin, Germany, 7–12 August 2016; pp. 145–151.
3. Habash, N.Y. Introduction to Arabic natural language processing. *Synth. Lect. Hum. Lang. Technol.* **2010**, *3*, 1–187.
4. Wenchao, L.; Yong, Z.; Shixiong, X. A novel clustering algorithm based on hierarchical and K-means clustering. In Proceedings of the Control Conference, Zhangjiajie, China, 26–31 July 2007; pp. 605–609.
5. Alotaibi, S.; Anderson, C. Word Clustering as a Feature for Arabic Sentiment Classification. *IJ Educ. Manag. Eng.* **2017**, *1*, 1–13. [CrossRef]
6. Goodfellow, I.; Bengio, Y.; Courville, A.; Bengio, Y. *Deep Learning*; MIT Press: Cambridge, UK, 2016; Volume 1.
7. Kriesel, D. A Brief Introduction on Neural Networks. 2007. Available online: https://www.dkriesel.com/en/science/neural_networks (accessed on 26 July 2023).
8. Mahdavi, M.; Abolhassani, H. Harmony K-means algorithm for document clustering. *Data Min. Knowl. Discov.* **2009**, *18*, 370–391. [CrossRef]
9. Arthur, D.; Vassilvitskii, S. K-Means++: The Advantages of Careful Seeding. In Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA'07, New Orleans, LA, USA, 7–9 January 2007; pp. 1027–1035.
10. Sahmoudi, I.; Lachkar, A. Towards a linguistic patterns for arabic keyphrases extraction. In Proceedings of the Information Technology for Organizations Development (IT4OD), Fez, Morocco, 30 March–1 April 2016; pp. 1–6.
11. Al-Anzi, F.S.; AbuZeina, D. Big data categorization for arabic text using latent semantic indexing and clustering. In Proceedings of the International Conference on Engineering Technologies and Big Data Analytics (ETBDA 2016), Bangkok, Thailand, 21–22 January 2016; pp. 1–4.
12. Pujari, P.S.; Waghmare, A. A Review of Merging based on Suffix Tree Clustering. In Proceedings of the National Conference on Advances in Computing, Roorkee, India, 13–15 February 2020.
13. Cottrell, M.; Olteanu, M.; Rossi, F.; Villa-Vialaneix, N. Self-organizing maps, theory and applications. *Rev. Investig. Oper.* **2018**, *39*, 1–22.
14. Yoshioka, K.; Dozono, H. The classification of the documents based on Word2Vec and 2-layer self organizing maps. *Int. J. Mach. Learn. Comput.* **2018**, *8*, 252–255. [CrossRef]
15. Yang, H.C.; Lee, C.H.; Wu, C.Y. Sentiment discovery of social messages using self-organizing maps. *Cogn. Comput.* **2018**, *10*, 1152–1166. [CrossRef]
16. Gunawan, D.; Amalia, A.; Charisma, I. Clustering articles in bahasa indonesia using self-organizing map. In Proceedings of the 2017 International Conference on Electrical Engineering and Informatics (ICELTICs), Banda Aceh, Indonesia, 18–20 October 2017; pp. 239–244.
17. Liu, Y.C.; Liu, M.; Wang, X.L. *Application of Self-Organizing Maps in Text Clustering: A Review*; IntechOpen: London, UK, 2012; Volume 10.
18. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey wolf optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [CrossRef]
19. Alhawarat, M.; Hegazi, M. Revisiting K-Means and Topic Modeling, a Comparison Study to Cluster Arabic Documents. *IEEE Access* **2018**, *6*, 42740–42749. [CrossRef]
20. Abuaiadh, D. Dataset for Arabic Document Classification. 2014. Available online: http://diab.edublogs.org/dataset-for-arabic-document-classification (accessed on 26 July 2023).
21. Al-Azzawy, D.S.; Al-Rufaye, F.M.L. Arabic words clustering by using K-means algorithm. In Proceedings of the New Trends in Information & Communications Technology Applications (NTICT), Baghdad, Iraq, 7–9 March 2017; pp. 263–267.

22. Mahmood, S.; Al-Rufaye, F.M.L. Arabic text mining based on clustering and coreference resolution. In Proceedings of the Current Research in Computer Science and Information Technology (ICCIT), Sulaymaniyah, Iraq, 26–27 April 2017; pp. 140–144.
23. Al-Rubaiee, H.; Alomar, K. Clustering Students' Arabic Tweets using Different Schemes. *Int. J. Adv. Comput. Sci. Appl.* **2017**, *8*, 276–280. [CrossRef]
24. Bsoul, Q.; Atwan, J.; Salam, R.A.; Jawarneh, M. Arabic Text Clustering Methods and Suggested Solutions for Theme-Based Quran Clustering: Analysis of Literature. *J. Inf. Sci. Theory Pract. (JISTaP)* **2021**, *9*, 15–34.
25. Abuaiadah, D.; Rajendran, D.; Jarrar, M. Clustering Arabic tweets for sentiment analysis. In Proceedings of the Computer Systems and Applications (AICCSA), Hammamet, Tunisia, 30 October–3 November 2017; pp. 449–456.
26. Daoud, A.S.; Sallam, A.; Wheed, M.E. Improving Arabic document clustering using K-means algorithm and Particle Swarm Optimization. In Proceedings of the Intelligent Systems Conference (IntelliSys), London, UK, 7–8 September 2017; pp. 879–885.
27. Saad, M.K.; Ashour, W.M. OSAC: Open source arabic corpora. In Proceedings of the 6th International Conference on Electrical and Computer Systems (EECS'10) , Lefke, North Cyprus, 25–26 November 2010; Volume 10.
28. Newspaper, E.S. Electronic Sabq Newspaper. 2020. Available online: https://sabq.org/ (accessed on 26 July 2023).
29. Souq Online Shopping. 2022. Available online: https://saudi.souq.com/sa-en/ (accessed on 26 July 2023).
30. Al-Subaihin, A.A.; Al-Khalifa, H.S.; Al-Salman, A.S. A proposed sentiment analysis tool for modern arabic using human-based computing. In Proceedings of the 13th International Conference on Information Integration and Web-based Applications and Services, Ho Chi Minh City, Vietnam, 5–7 December 2011; pp. 543–546.
31. Farra, N.; Challita, E.; Assi, R.A.; Hajj, H. Sentence-level and document-level sentiment mining for arabic texts. In Proceedings of the Data Mining Workshops (ICDMW), Ho Chi Minh City, Vietnam, 5–7 December 2010; pp. 1114–1119.
32. Al-Harbi, S.; Almuhareb, A.; Al-Thubaity, A.; Khorsheed, M.; Al-Rajeh, A. Automatic Arabic text classification. In Proceedings of the 9th International Conference on the Statistical Analysis of Textual Data, Lyon, France, 12–14 March 2008.
33. Alanba News. 2022. Available online: https://www.alanba.com.kw (accessed on 26 July 2023).
34. Alazzam, H.; AbuAlghanam, O.; Alsmady, A.; Alhenawi, E. Arabic Documents Clustering using Bond Energy Algorithm and Genetic Algorithm. In Proceedings of the 2022 13th International Conference on Information and Communication Systems (ICICS), Irbid, Jordan, 21–23 June 2022; pp. 4–8. [CrossRef]
35. Zeng, J.; Yin, Y.; Jiang, Y.; Wu, S.; Cao, Y. Contrastive Learning with Prompt-derived Virtual Semantic Prototypes for Unsupervised Sentence Embedding. In Proceedings of the Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, 7–11 December 2022; pp. 7042–7053. [CrossRef]
36. Kohonen, T. The'neural'phonetic typewriter. *Computer* **1988**, *21*, 11–22. [CrossRef]
37. Grajciarova, L.; Mares, J.; Dvorak, P.; Prochazka, A. Biomedical image analysis using self-organizing maps. In Proceedings of the Annual Conference of Technical Computing: Prague, Czech Republic, 1 January 2012.
38. Miljković, D. Brief review of self-organizing maps. In Proceedings of the MIPRO 2017, Opatija, Croatia, 22–26 May 2017.
39. He, Z.; Chen, J.; Gao, M. Feature time series clustering for lithium battery based on SOM neural network. In Proceedings of the 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA), Wuhan, China, 31 May–2 June 2018; pp. 358–363.
40. Bara, M.W.; Ahmad, N.B.; Modu, M.M.; Ali, H.A. Self-organizing map clustering method for the analysis of e-learning activities. In Proceedings of the Majan International Conference (MIC), Muscat, Oman, 19–20 March 2018; pp. 1–5.
41. Simon, N.T.; Elias, S. Detection of fake followers using feature ratio in self-organizing maps. In Proceedings of the 2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), San Francisco, CA, USA, 4–8 August 2017; pp. 1–5.
42. Mei, P.A.; de Carvalho Carneiro, C.; Kuroda, M.C.; Fraser, S.J.; Min, L.L.; Reis, F. Self-organizing maps as a tool for segmentation of Magnetic Resonance Imaging (MRI) of relapsing-remitting multiple sclerosis. In Proceedings of the Self-Organizing Maps and Learning Vector Quantization, Clustering and Data Visualization (WSOM), Nancy, France, 28–30 June 2017; pp. 1–7.
43. Sarmiento, J.A.R.; Lao, A.; Solano, G.A. Pathway-based human disease clustering tool using self-organizing maps. In Proceedings of the Information, Intelligence, Systems & Applications (IISA), Larnaca, Cyprus, 27–30 August 2017; pp. 1–6.
44. Guo, L.; Xie, Q.; Huang, X.; Chen, T. Time difference of arrival passive location algorithm based on grey wolf optimization. In Proceedings of the Computer and Communications (ICCC), Chengdu, China, 13–16 December 2017; pp. 877–881.
45. Mostafa, E.; Abdel-Nasser, M.; Mahmoud, K. Application of mutation operators to grey wolf optimizer for solving emission-economic dispatch problem. In Proceedings of the Innovative Trends in Computer Engineering (ITCE), Aswan, Egypt, 19–21 February 2018; pp. 278–282.
46. Xiao, J.; Zou, G.; Xie, J.; Qiao, L.; Huang, B. Identification of Shaft Orbit Based on the Grey Wolf Optimizer and Extreme Learning Machine. In Proceedings of the 2018 2nd IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), Xi'an, China, 25–27 May 2018; pp. 1147–1150.
47. Sankaranarayanan, S.; Swaminathan, G.; Sivakumaran, N.; Radhakrishnan, T. A novel hybridized grey wolf optimzation for a cost optimal design of water distribution network. In Proceedings of the Computing Conference, London, UK, 18–20 July 2017; pp. 961–970.
48. Majeed, M.A.M.; Rao, P.S. Optimization of CMOS Analog Circuits Using Grey Wolf Optimization Algorithm. In Proceedings of the 2017 14th IEEE India Council International Conference (INDICON), Roorkee, India, 15–17 December 2017. [CrossRef]

49. Mjahed, S.; Bouzaachane, K.; Taher Azar, A.; El Hadaj, S.; Raghay, S. Hybridization of fuzzy and hard semi-supervised clustering algorithms tuned with ant lion optimizer applied to Higgs boson search. *Comput. Model. Eng. Sci.* **2020**, *125*, 459–494.
50. Khan, A.R.; Khan, S.; Harouni, M.; Abbasi, R.; Iqbal, S.; Mehmood, Z. Brain tumor segmentation using K-means clustering and deep learning with synthetic data augmentation for classification. *Microsc. Res. Tech.* **2021**, *84*, 1389–1399. [CrossRef] [PubMed]
51. Khan, Z.; Koubaa, A.; Fang, S.; Lee, M.Y.; Muhammad, K. A Connectivity-Based Clustering Scheme for Intelligent Vehicles. *Appl. Sci.* **2021**, *11*, 2413. [CrossRef]
52. Shah, Y.A.; Aadil, F.; Khalil, A.; Assam, M.; Abunadi, I.; Alluhaidan, A.S.; Al-Wesabi, F.N. An Evolutionary Algorithm-Based Vehicular Clustering Technique for VANETs. *IEEE Access* **2022**, *10*, 14368–14385. [CrossRef]
53. Hassoun, M.H. *Fundamentals of Artificial Neural Networks*; MIT Press: Cambridge, UK, 1995.
54. Kohonen, T. The self-organizing map. *Proc. IEEE* **1990**, *78*, 1464–1480. [CrossRef]
55. Vettigli, G. MiniSom: Minimalistic and numpy based implementation of the self organizing maps. 2018. Available online: https://github.com/JustGlowing/minisom/ (accessed on 26 July 2023).
56. GitHub—An Implementation of an Optimized Arabic Text Clustering Technique. Available online: https://github.com/majeedalameer/ArabicTextClustering (accessed on 26 July 2023).
57. Alalyani, N.; Marie-Sainte, S.L. NADA: New Arabic Dataset for Text Classification. *Int. J. Adv. Comput. Sci. Appl.* **2018**, *9*, 206–212. [CrossRef]
58. Abuaiadah, D.; El Sana, J.; Abusalah, W. On the impact of dataset characteristics on arabic document classification. *Int. J. Comput. Appl.* **2014**, *101*, 31–38. [CrossRef]
59. Attia, M.A. Arabic tokenization system. In Proceedings of the 2007 Workshop on Computational Approaches to Semitic Languages: Common Issues and Resources, Prague, Czech Republic, 28 June 2007; Association for Computational Linguistics: Stroudsburg, PA, USA, 2007; pp. 65–72.
60. Zerrouki, T. PyArabic, An Arabic language library for Python. 2010. Available online: https://pypi.python.org/pypi/pyarabic (accessed on 26 July 2023).
61. Syarief, M.G.; Kurahman, O.T.; Huda, A.F.; Darmalaksana, W. Improving Arabic Stemmer: ISRI Stemmer. In Proceedings of the 2019 IEEE 5th International Conference on Wireless and Telematics (ICWT), Yogyakarta, Indonesia, 25–26 July 2019; pp. 1–4.
62. Manning, C.; Schutze, H. *Foundations of Statistical Natural Language Processing*; MIT Press: Cambridge, UK, 1999.
63. Jones, K.S. A statistical interpretation of term specificity and its application in retrieval. *J. Doc.* **1972**, *28*, 11–21. [CrossRef]
64. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.