Proceedings of to the 9th Scientific Conference of University of Garmian: Pure science and Technology Applications (SCUG-PSTA-2022). Kalar (46021) / Sulaymaniyah (26-27 Oct.2022) (Special issue for Passer journal of basic and applied sciences) Paper No. 02, 17-39

Sentiment Analysis using SVM-based SSO Intelligence Algorithm

Omar Y. Abdulhammed[1*], Pshtiwan J. Karim[2]

[1*] Department of Computer Science, College of Science, University of Garmian, Kalar, Iraq.

[2] Department of Computer Science, College of Science, University of Garmian, Kalar, Iraq.

Omar.y@garmian.edu.krd; Pshtiwan.jabar@garmian.edu.krd

ABSTRACT

Facebook and Twitter, as two known social media become popular sources of big data that give the right to people to share and express their feedback about products, services, politicians, events, and every aspect of life in the form of short texts. The classification of sentiments could be automated through machine learning and enhanced using appropriate feature extraction methods. In this work, we collected the most recent tweets about (Biden, Benzema, Apple, and NASA) using Twitter-API and assigned sentiment scores using a rule-based lexicon approach; after pre-processing stage, each dataset is divided into 80% as a training set, and rest 20% as testing set. After that, the Distributed bag of words, Distributed memory mean, Distributed Memory Concatenation, and Term Frequency-Inverse Document Frequency models are used for feature extraction from pre-processed tweets. Depending on the Shark smell optimizer algorithm, the SVM technique was used to classify the extracted features. The SSO was used to tune and select the best value for SVM parameters to optimize the overall model performance. The results display that these optimizers have an essential impact on increasing the model accuracy. After optimization, the model accuracy reached 92.12%, while the highest accuracy without optimization was 88.69% for various feature extraction methods.

KEYWORDS: Sentiment Analysis, SVM, SSO, Twitter, Optimization, and classification.

1 INTRODUCTION

After the rise of the internet and the rapid progress of technologies, the origin of data such as web browsers, social media, blogs, smartphones, sensors, and more have increased, making them a rich source of data known as Big Data [1]. Extensive data analysis is an operation that inspects vast data collections

using various procedures and mechanisms to find meaningful trends, potential correlations, and concealed patterns for making evidence-based decisions to follow high-quality outcomes [2]. As a micro-blogging platform and social media, Twitter has become a rich significant data source. Twitter allows people to share and express their opinions about services, products, events, topics, brands, movies, markets, politics, etc. These opinions are so important for companies, politicians, governments, and other people to understand how people feel about them and their policies and products, and services through analyzing the data. This work can be carried out through a process called Sentiment Analysis (SA) or Opinion Mining (OM) [3,4].

SA or OM is a technique that is based on Natural Language Processing (NLP) that analyzes people's attitudes. The primary goal of SA is to determine the polarity of a writer's attitude from text toward various topics and group them into positive, negative, or neutral categories [4]. SA can be performed at three levels:

• Document-based level: this level summarizes that the whole document is assigned a positive or negative polarity.

• Sentence-based level: this level summarizes that the document is analyzed at the sentence level. Each sentence within the document is inspected independently and grouped as positive and negative.

• Aspect Level: also known as (Feature Level and Attribute Level). It concerns the more extensive investigation of text, identifying aspects in sentences and examining them, then classifying them as positive and negative [5]. Developing data analytics models cover some steps. One major step is feature extraction, a numerical description of tokens available in any document. Features contain noise because of the data gathering stage due to data-scraping techniques' incompleteness or redundant and unrelated information that already exists in the data for a particular issue. This will cause degradation of the entire learning process performance, and classification model accuracy raises the computational complexity of a model and causes over-fitting. So, the problem of high dimensionality should be managed by applying data mining and machine learning algorithms. To solve this case, the techniques of feature selection could be utilized for choosing the optimum features from the existing feature set for regression, classification, and clustering problems. Apart from the accuracy, that is a significant metric for evaluating the classification model. Feature selection is vital in enhancing model accuracy by recognizing and eliminating redundant and irrelevant features. Based on functionality, the feature selector methods were divided into three classes: filter, wrapper, and embedded or hybrid bases [6].

Machine learning techniques are broadly utilized to implement SA models. However, the work of these algorithms depends on some parameter(s) that have significantly impacted the overall model. In this work, Shark Smell Optimizer (SSO) is used to determine the ideal value for two parameters of the SVM algorithm, which is known as the regularization parameter /penalty denoted by (C) and gamma. Also, some feature extraction methods are used as DBoW, DMC, DMM, and TF-IDF. Then, the overall model was evaluated to explain the impact of SSO in improving the SVM accuracy.

2 RELATED WORKS

In recent years, SA has been active in the research field, and many works have been done in this field, and we focus on the most recent works that other researchers have fulfilled.

Shuai et al. (2018). [7], the authors carried out binary sentiment classification on hotels review in China; they collected 11600 hotel reviews from the website. According to their rating, the comments were split into two classes. To make a balanced data set, the data was half positive and half negative. Document to Vector (Doc2Vec), a Deep learning-based technique, was applied to extract features from the data. Some supervised (ML) algorithms, including Logistic Regression, SVM, and Naïve Bayes, are utilized to classify the comments. The performance of the models was assessed using the precision, recall, and f-score metrics. According to the results they awarded, SVM achieves the best average f-score of 81.16%, while the average f-score of Logistic Regression and Naïve Bayes classifiers are 79.6% and 73.6%, respectively.

Ahmad et al. (2018). [8] In this work, the authors performed sentiment classification on three datasets, two of the datasets had multi-class (3-class), and the other one had binary label class using Radial Bias Function (RBF) kernel-based SVM with 10-Fold cross-validation. They proposed a grid search method to select the ideal value for the RBF-SVM parameters. The performance of the presented method was evaluated for all three chosen datasets based on precision, recall, and f1-score metrics. The proposed model obtained 84.1% of the average f-score, while the best previous average f-score was 78.8% compared to their prior work.

Naz et al (2018) [9] The researchers performed a Twitter sentiment analysis on SemEval 2016 data set, after pre-processing, they used different weighting schemes such as TF, TF-IDF, and Binary Occurrence (BO) to the numerical representation of tweet tokens along with various n-gram ranges like Uni-gram, Bi-gram, Tri-gram, and union of them, after that, the extracted features were fed for SVM training, the best outcome was obtained was 79.6% for Uni-gram based TF-IDF. In addition, the sentiment Score Vector (SCV) package was used to compute the positivity and negativity of tweets to boost SVM's performance, with various weighting schemes and n-gram ranges, SCV with BO and Uni-gram range achieved an accuracy of 81% that is the highest compared to other methods.

Naw (2018) [10] In this work, SVM and K-Nearest Neighbour (KNN) classifiers were used to cluster tweets about business, education, health, and crime into positive, negative, and neutral to analyze business bate, crime rate, educational rate and health rate that occurs in Singapore, Malaysia, Myanmar, and Vietnam. For these cases, SVM outperformed K-NN and obtained the highest accuracy in the education dataset which was 74.46% while the highest accuracy of KNN was 70.59% in the business dataset.

Rustam et al (2019) [11] in this work, the authors performed multi-class (3-class) sentiment classification on a dataset that contains 14640 user reviews on six United States airline companies. To carry out classification AdaBoost, Calibrated, Decision Tree, Extra Tree, Gaussian Naïve Bayes, Gradient

Boosting Machine, Logistic Regression, Random Forest, Stochastic Gradient Descent, SVM, and voting (combination of Logistic Regression + Stochastic Gradient Descent classifiers) classifiers are used. The tweets went through two pre-processing steps, they were named complete pre-processing and partial pre-processing. At the complete pre-processing method, they removed stop words and took the stem of words, in contrast, did not perform those two steps in partial pre-processing. The dataset was split into a training set of 75% and a testing set of 25%. After that, different techniques like TF-IDF, and (Word2Vec) were used for vector representation of pre-processed tweets with complete pre-processed tweets, and only (TF) and (TF-IDF) with partial pre-processed tweets. The voter classifier outperforms the other classifiers with an accuracy of 79.2% on complete pre-processing and 80.4% on partial pre-processing both with the TF-IDF feature extraction method, followed by the SVM classifier with an accuracy of 78.5% and 80.1% respectively.

In this work [12] the researchers designed a classifier system utilizing Machine Learning (ML) methods that can predict the polarity of a comment. The system has three operations: processing, data extraction, and 20odelling and uses the NLTK dataset with text mining methods to create and process the variables. The tweets are divided into positive and negative sentiments by a classifier using a supervised probabilistic machine-learning technique. The results proved the effectiveness of the suggested system

This paper [13] presents a novel efficient technique using the deep learning method of sentiment analysis by merging the "universal language model fine-tuning" (ULMFiT) with SVM to improve accuracy. The technique presents a modern deep learning-based strategy to determine people's opinions regarding specific products based on their moods expressed as "Comments" on Twitter. The outcomes prove that the proposed system has high performance and accuracy.

3 THE TECHNIQUES OF SENTIMENT ANALYSIS

SA is a process that uses computational techniques to classify the opinions presented as text sentences to know whether the opinions about a particular subject, product, etc., are negative, positive, or neutral. Researchers divide SA techniques into three strategies: machine learning, lexicon-based, and hybrid. ML strategy applies the popular algorithms of machine learning. The lexicon-based strategy depends on the sentiment lexicon, a set of general and pre-compiled sentiment terms. They were classified into dictionary-based and corpus-based strategies that apply statistical or semantic techniques to discover sentiment polarity. The hybrid strategy embeds both strategies, is more familiar with sentiment lexicons, and has a vital role in most strategies. [14]. Figure 1 illustrates different sentiment analysis techniques.

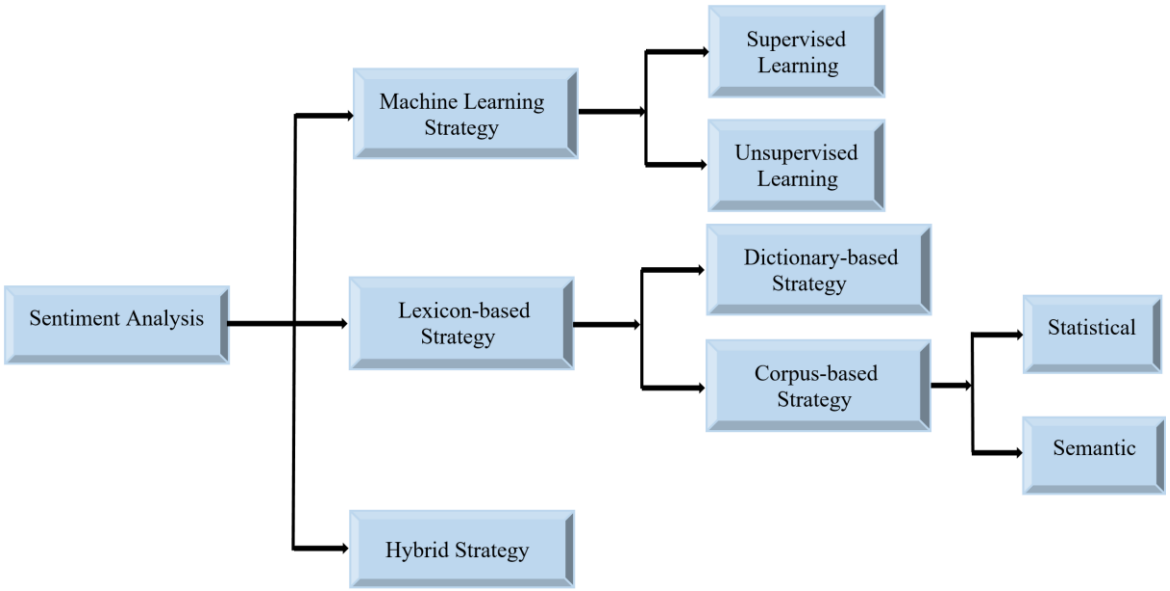

Figure 1: Sentiment Classification Techniques.

4 CONCEPT OF META-HEURISTIC TECHNIQUES

The word heuristic means to find or discover, it is described in the context of computing for problem-solving without the total application of a method. In other meaning, a method that (i) looks for an estimated solution, (ii) does not require discrete mathematical convergence proof, and (iii) doesn't need to investigate all available solutions in the search space before reaching the last solution. Consequently, it is efficient in terms of computationally [15].

5 WORD EMBEDDING AND DOC2VEC

Word embedding (Word2Vec) in short, is defined as a method that uniquely vector exhibits each token with the suitable meaning of the taken word. In contrast to a bag of words, which is a common way that transforms words to numbers based on a fixed-length feature vector, this way has some limitations such as not considering the word orders along with skipping the meaning of the words. For instance, "Strength," "Power," and "Chicago" are equally assessed and produce a high dimensional feature set, also requiring a large amount of memory space[16]. In the Word2Vec procedure, each word is converted to a vector in a limited vector space. Neural networks are used to learn these vectors. Word2Vec could be fulfilled with two forms: a continuous bag of words (CBoW) and skip-gram (SG). The first one predicts current words

based on the input of future words and remembered words. The second form maximizes the possibility of adjacent words given the current word being utilized in word embedding[16] [17].

Doc2Vec or paragraph vector (PV), is an extension of (Word2Vec), it transfers the whole paragraph to a unique vector presented by a column in matrix D and each word is vectorized uniquely in matrix W. The word and PVs are then linked together to anticipate the upcoming word. CBoW and SG procedures have been regulated to form Doc2Vec and transformed into a distributed bag of words of PVs (PV-DBOW) and distributed memory of PVs (PV-DM) versions [18]. The DBOW forces the model to anticipate the words randomly sampled from the paragraph in the output without consideration of the circumstance of the input words [16].

The DM model uses DM mean (DMM) and DM concatenation (DMC) approaches to anticipate the upcoming word in the context. The approaches work based on averaging (mean) and concatenating the paragraph, and word vectors respectively[16].

6 TERM FREQUENCY - INVERSE DOCUMENT FREQUENCY

Term Frequency – Inverse Document Frequency (TF-IDF); is a statistic-based numerical metric that could represent how much a word is 'significant' related to a document in a document set. It is an effective technique for automated text analysis that could be utilized in calculating scoring words for ML algorithms in Natural Language Processing (NLP). The technique is the combination of the multiplication of two other techniques listed below:

6.1 Term Frequency (TF)

Term Frequency calculates the occurrence of words in a document by showing that words often come in a document or "sentence" are likely more valuable than words that come hardly. Later, it goes through the normalization process by dividing it by the total words in the entire document. The purpose of this process is to inhibit a bias towards the longer documents. TF (t) = (times that term t appears in a document) / (Amount of available terms in the document).

6.2 Inverse Document Frequency (IDF)

IDF measure presents how a token is important by taking the total number of documents in the document collection and then dividing it by the number of documents that the token arises in it computed by: IDF (t) = log (total of documents in document set/amount of documents in document set contain the term t).

7 SUPPORT VECTOR MACHINE

Support Vector Machine (SVM) belongs to supervised ML algorithms that were introduced by V. Vapnik and C. Cortes in 1995. The SVM is mostly used in classification tasks. It can solve linear and non-linear problems and works effectually for several practical issues. It is effective when dealing with high-dimensional data such as images and text. The aim of SVM is clear, it tries to find a hyperplane that has the largest margin, i.e., the decision boundary that separates the support vectors to the farthest, it is responsible for finding the decision boundary to divide various classes and maximizing the margin [19].

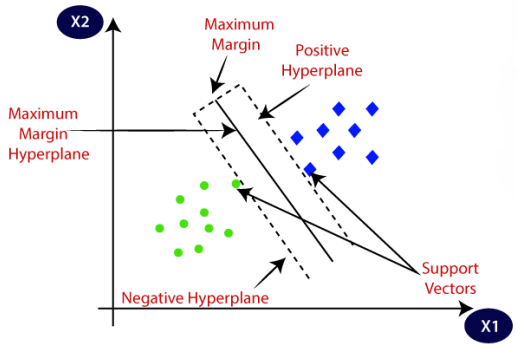

Figure 2: General structure of SVM

The above figure depicts the general form of SVM, the black line is the line that separates the data called the Decision Boundary (Hyper-plane) of SVM, and the other two lines (Hyperplanes) help us make the right decision boundary. The separating hyper-plane is defined as:

$$\omega.x + \beta = 0 \qquad (1)$$

Where the equation $\omega.x + \beta = 1$ for red circles, and $\omega.x + \beta = -1$ for blue circles. The ω is a weight vector and β is a bias (scalar). The maximal margin is mathematically expressed in equation 2.

$$M = \frac{2}{\|\omega\|} \qquad (2)$$

Where $\|\omega\|$ is the Equation norm of ω. one can notice that the data in figure 2 is linearly separable, this means that a straight line can separate? However, most data sets are linearly non-separable, for such cases, there is a kernel-based SVM algorithm that could transform features to much higher dimensional feature space by transformation φ. A kernel function represents a dot product of input data points. κ (Xi, Xj) $=< \varphi(Xi). \varphi(Xj) >$. There are four main types of kernels in SVM which are:

1. Linear kernel
2. Polynomial kernel
3. Sigmoid kernel
4. Radial Bias Function kernel

In this work, RBF-SVM is utilized as a classification algorithm. There are two parameters in RBF-SVM, known as hyper-plane soft margin parameters that are symbolized as $penalty(\mathrm{C})$ and complement parameter appears in the dual form of SVM:

$$f(x) = \omega^T x + b \quad (3)$$

$$\omega = \sum_i \alpha_i \, y_i x_i \quad (4)$$

By substituting 3 in 4.

$$f(x) = \sum_i \alpha_i \, y_i \left(x_i^T x\right) + b \quad (5)$$

$$\|\omega^2\| = \sum_i \sum_j \alpha_i \, \alpha_j \, y_i \, y_i \left(x_i^T x\right) + b \quad (6)$$

This leads to the dual form of the SVMs:

$$\underset{\alpha \geq 0}{\omega} = \sum_i \alpha_i \; - \frac{1}{2} \sum_i \sum_j \alpha_i \, \alpha_j \, y_i \, y_i \left(x_i^T x\right) \quad (7)$$

Where α_i is a Lagrangian multiplier and x_i are support vectors, such that:

$0 \leq \alpha_i \; \leq \mathrm{C}$ and $\omega = \sum_i \alpha_i \; y_i$

The classification with kernels is:

$$f(x) = \sum_i^N \alpha_i \, y_i \, k(x_i, x) + b \quad (8)$$

Where N is a training size, and the mathematical formula of RBF-SVM is:

$$k(x_i, x) = \exp\left(\frac{-\|(x_i - x)\|^2}{2\sigma^2}\right) \quad (9)$$

$(-\|(x_i - x)\|^2)$ Is Euclidean distance between support vectors and σ represents the kernel parameter (gamma).

The C parameter correctly controls the balance between decision boundary and categorizing training points, enabling the SVM how much to avoid mystifying each training instance. For higher values of it, SVM will select a narrower margin hyperplane. Conversely, a lower value of C will cause the SVM to look for a wider margin separating the hyperplane as shown in figures 3 and 4 [20].

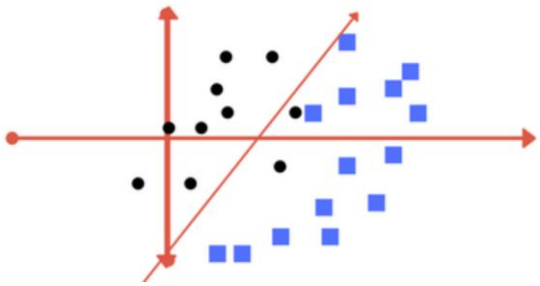

Figure 3: Low c value [21]

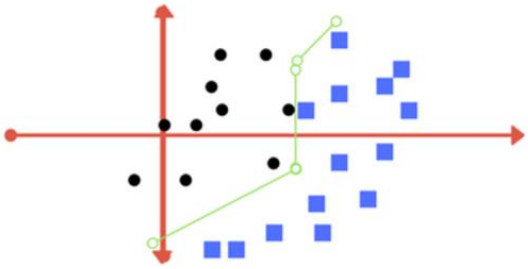

Figure 4: High C value [21]

The gamma parameter describes how far the effect of each training instance extends. A lower value indicates that every single point has a far reach. In contrast, a higher value depicts that every point has a close reach. But with high-rise value, the decision boundary depends on those points that are near to the line which efficiently outcomes in bypassing some of the furthest points from the decision boundary due to the nearer points getting more weight, resulting in a wavy curve as presented in the prior figure. In contrast, with quite low value, the further points get reasonable weight and a more linear curve is achieved. Figures 5 and 6 depict the gamma values.

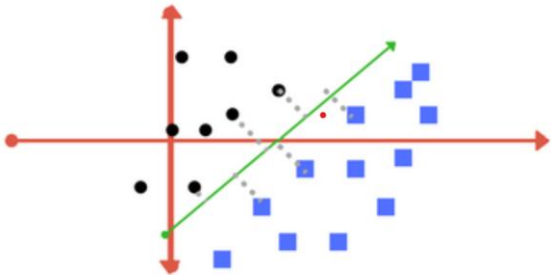

Figure 5: High gamma value [21]

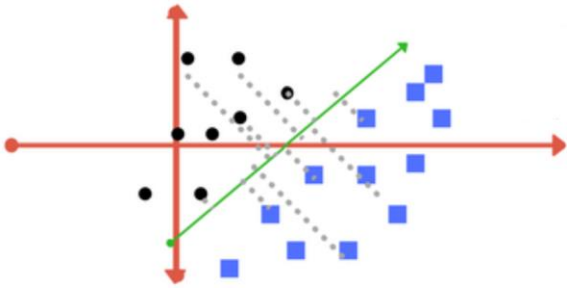

Figure 6: Low gamma value [21]

The performance of RBF-SVM roughly relies on the value of these parameters, and to obtain a preferable accuracy, one should accurately tune the value of these parameters [20].

8 SHARK SMELL OPTIMIZER

SSO is a member of SI where algorithms were developed and implemented by O. Abedinina, N. Amjady, and A. Ghasemi in 2014 based on the mimic of shark's hunting that has supremacy using a strong sense of smell in catching prey in a short time. The secret of this supremacy is the shark's ability to quickly discover the prey using its powerful smell sensing in a wide search space [22]. While the prey is injured and blood is flushing into the water, the shark moves in the direction of the prey by following the blood aroma. This movement relies on the intensity and flow of blood aroma in the water particles.[23].

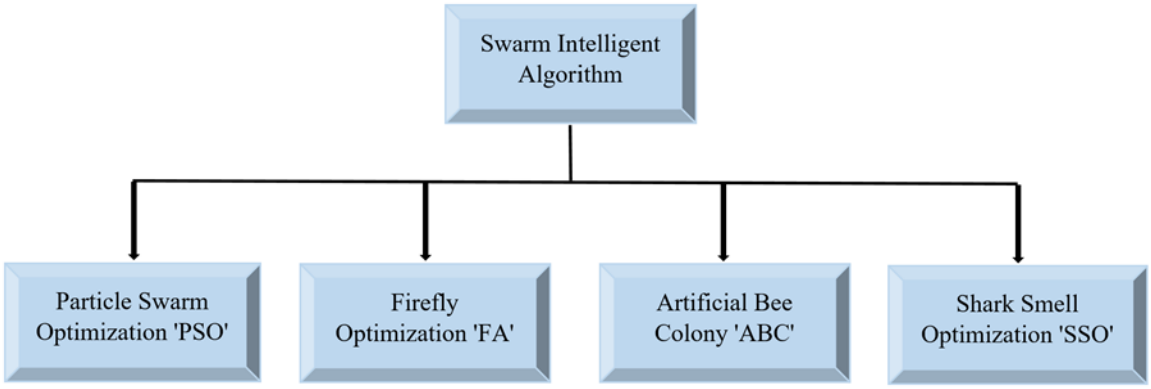

Figure 7: Some algorithms of swarm intelligence [24]

Intensity is considered the main point in directing the shark toward its prey. Additionally, the degree of intensity has a direct impact on the increase/decrease of sharks' movement [25][26]. The pseudo-code of the SSO algorithm has shown below.

Algorithm (1): SSO description [22]

Begin
Step 1. Initialization
Set parameters NP, k_{max}, n_k, α_k, and $(k = 1, 2, \dots, k_{max})$
Generate an initial population of all individuals
Generate each decision randomly within the allowable range
Initialize the stage counter k = 1
For $k = 1$: k_{max}
Step 2. Forward Movement
Calculate each component of the velocity vector, $V_{ij\ (i=1,\dots NP, j=1,\dots ND}$
Obtain the new position of shark based on forwarding movement, $Y_i^{k+1}(i = 1, \dots, NP)$
Step 3. Rotational movement
Obtain the position of the shark based on rotational movement, $Z_i^{k+1,m}(m = 1, \dots, M)$
Select the next position of the shark based on the two movements, $X_i^{k+1}(i = 1, \dots, NP)$
End for k
Set $k = k + 1$
Select the best position of the shark in the last stage which has the highest OF value
End

8.1 The working strategy of the SSO Algorithm

The worth king of the SSO algorithm comprises four major steps namely (population initialization, onward movement, rotational movement, and position update), each of them described in the below sections:

8.1.1 Population initialization

Like most SI algorithms, as the first step, the population of the initial solution should be produced randomly within the search space. Each member of these solutions denotes a particle of aroma that explains a possible position of the shark at the start of the exploring process. Time implementation is based on the mathematical shown formula shown in (10) and (11):

X^1= [X_1^2, X_2^1……, X_NP^1] (10)

Where X_i^1=ith is the initial position of the population vector and NP = population size. The associated optimization task can be presented as:

X_i^1=[X_(i,1)^2,X_(i,2)^1,X_(i,3)^1……,X_(i,ND)^1] (11)

Where X_(i,j)^2=jth represents the dimension of the shark's ith position, and ND = several decision variables [27].

8.1.2 Onward movement in SSO

When the victim's blood flushes into the water, the shark in a different location moves in the direction of particles that have a stronger aroma by a velocity "V," to be nearer the target (victim). Coincide with the position vector initialization; the velocity vector with its dimensional is a constituent element conveyed by (12) and (13).

V^1=[V_1^2,V_2^1,V_3^1……,V_NP^1] (12)

V_i^1=[V_(i,1)^2,V_(i,2)^1,V_(i,3)^1……,V_(i,ND)^1] (13)

Hence, the velocity dimensions are calculated based on formula (14):

V_(i,j)^k=nk.R1 (∂(OF))/∂xj|x_(i,j)^k (14)

Whereby k=1,2,….,k_max,(∂(OF))/∂xj|x_(i,j)^k is the objective function derivation (OFD) at position ¬¬x_(i,j)^k.

k_max= represents the maximum iteration for onward movement of the shark, k = number of iterations (stages), n= a value in the interval (0, 1), and R1 = a random number between (0, 1) [28].

The increase of the shark's velocity is specified by the rise in aroma severity. In each step of V_(i,j)^k, the controlling (limiting) of velocity utilized through adapting (14) and presented in (15):

V_(i,j)^k=nk.R1 (∂(OF))/∂xj|x_(i,j)^k+ α_k.R2.V_(i,j)^(k-1) (15)

Where nk is a velocity controller rate for stepk, α_k is the initial coefficient between (0, 1), and R2 also is a random number in the range of (0, 1).

Because of the shark's onward movement, the new position that is Y_(i,j)^k is calculated based on its older position(X_i^k) and velocity(V_i^k), formula (16) reveals its new position:

Y_i^(k+1)= X_i^k+ V_i^k.Δt_k (16)

Where Δt_k= represents is the time interval, for clarity, it supposed to be 1[29]. The onward movement of the shark in direction of prey showed in figure 8.

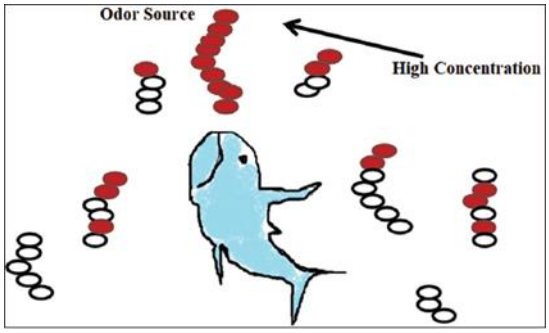

Figure 8: Shark's onward movement [29]

8.1.3 Rotational movement in SSO

Sharks have another movement technique known as "rotational movement", which uses in finding those particles that have a stronger aroma. This process is known as the local search of the SSO formed through (17).

$$Z_i^{(k+1,m)} = Y_i^k + \langle\!\langle R3\ Y \rangle\!\rangle_i^{(k+1)} \quad (17)$$

m = 1, 2, 3…, M, and R3 is a random number in the interval (−1, 1). To form the rotational movement, the number of points M in the local search is linked to making nearby contour lines [30].

8.1.4. Updating particle positions

The shark's search path will keep on with the rotational movement as it moves in direction of a point with a stronger aroma as shown in figure 9. This property could be formed mathematically as shown in 18.

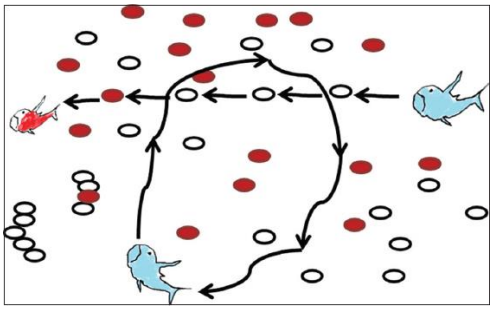

Figure 9: Shark's rotational movement [30]

$$X_i^{k+1} = \arg\max\{OF(Y_i^{k+1}), OF(Z_i^{k+1,i}), \dots, OF(Z_i^{k+1,M})\} \quad (18)$$

Where: X_i^{k+1} points to the shark's next location with the apex value of the objective function (OF). The process will keep on till k arrives at the maximum value (individual best) in the given population at a search space set for an optimization problem [31].

9 PROPOSED SYSTEM

In this stage, the implementation and analysis of the suggested method for classifying sentiment of Twitter data applying SSO and SVM are presented in detail. The proposed system for Twitter sentiment analysis consists of several steps, as presented in figure 10.

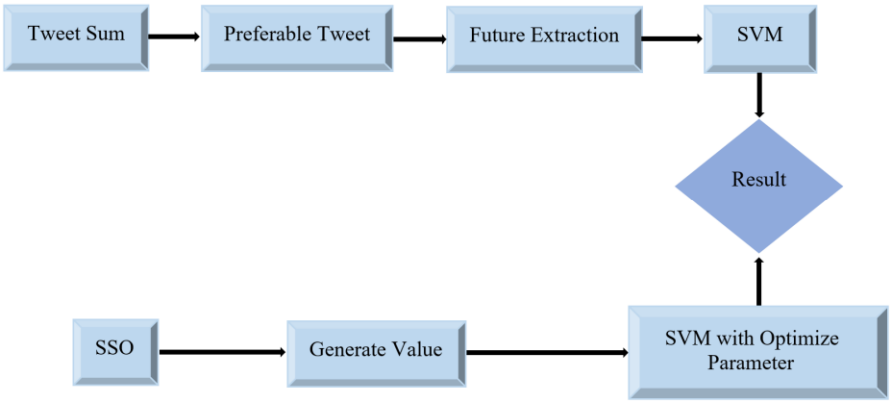

Figure 10: Block diagram of the proposed system

9.1 Data Collection

The prime step in every classification task is the existence of a dataset. To do this, Twitter Application Programming Interface (Twitter-API) named "Twitter-Sentiment-Analysis-20" is created. Twitter-API works as an interface between programmers and Twitter, which permits them to access Twitter via four secret keys and tokens. The API reads the most recent tweets (not re-tweet) for the proposed model's keywords: Biden, Benzema, Apple, and NASA. After collecting the tweets, they got labeled (positive) and (negative) based on their polarity using a lexicon-rule-based tool called "VADER." After that, protect the collected, labeled tweets in a comma-separated value (CSV) file for future steps.

9.2 Pre-Processing

The prepared sentiment analysis is achieved in the second step, which is a major and significant stage in the natural language processing algorithm (NLP). It transforms text into some preferable form so that machine learning methods can perform better by reducing the text's noise. This process speeds up the classification and improves the performance of the classifier. This stage consists of some procedures which are:

1. Remove duplicate tweets
2. Lowercase all texts
3. Replace emojis with their meaning
4. Remove URLs
5. Remove the Hashtag sign
6. Remove email address
7. Remove punctuation marks such as", ",?., etc.
8. Remove numbers
9. Remove extra white spaces
10. Tokenization
11. Remove Stop Words

9.3 Future Extraction

One of the major steps of text classification is feature extraction. Feature extraction of a text is a numerical representation of individual tokens and vectorising them, used to train and test the classifier algorithm. The input feature has a major influence on the performance of machine learning algorithms. Therefore, an important aspect of building a model is to choose an appropriate feature extraction technique.

In this research, after pre-processing steps, the datasets are separated into the training set and testing sets by the rates of 80% and 20% for each, using the hold-out technique, which gave out the best outcome among all other methods tested. Then, Doc2Vec and Term Frequency-Inverse Document frequency (TF-IDF) techniques have been utilized to convert every tweet into numerical form.

9.4 Sentiment Classification Algorithm

Sentiment analysis is the function of determining the polarity of the author's attitudes that present ted in short text format, and classifying them into positive, and negative. Here, the proposed model performs binary document-level sentiment classification. The SVM-based Function kernel classifier chose as a

classification algorithm. SVM belongs to supervised machine learning algorithms that are widely for classification and regression; it has a large amount of data.

In this experiment, the SVM classifier was implemented in two stages. In the first stage, the classifier with default parameter values has been implemented. The classifier has two hyperplane parameters which are known as Soft Margin parameters and denoted as C and gamma. The value of each parameter has a major influence on the accuracy of SVM. However, identifying the ideal value of parameters is a challenging function and impractical to be performed manually. Instead, in this work, the presented technique uses the SSO algorithm to aid in selecting the best parameter values.

9.5 SVM with SSO Optimizer

The performance of RBF-kernel-based SVM mainly relays on the value of C and gamma regularization parameters. Selecting those two parameter values is so crucial to have a major impact on enhancing the performance of overall the system. SSO, which is a population-based meta-heuristic optimizer, is used to solve an optimization problem that has been implemented to determine the best value for the two above-mentioned parameters. SSO-SVM has been implemented as follows:

1. Reading pre-processed tweets of each dataset from a CSV file.
2. Splitting the dataset using the hold-out technique into 80% as training-set and 20% "as test-set.
3. Using Doc2Vec architectures (DBoW, DMC, and DMM) to extract features from the training set and test set.
4. Extracting features from training-set and test-set based on Term Frequency-Inverse Document frequency (TF-IDF).
5. Initializing swarm size with 15 sharks along with their velocities. Then randomly distributing them in search space, initialize accelerator parameters (C1, C2, W), and maximum iteration number with 10 iterations.
6. At each iteration, the fitness value of each shark will be calculated inside a function. In this function, C and gamma take the value of each particle’s position, and the best accuracy of SVM for each particle position in the current iteration will be saved in an array.
7. At each iteration, the fitness value of each shark will be evaluated.
8. After reaching the maximum iteration, the best accuracy was achy the optimized RBF-SVM will be printed with the respect to best C and gamma values. Figure 11 shows how to optimize RBF-SVM by using SSO.

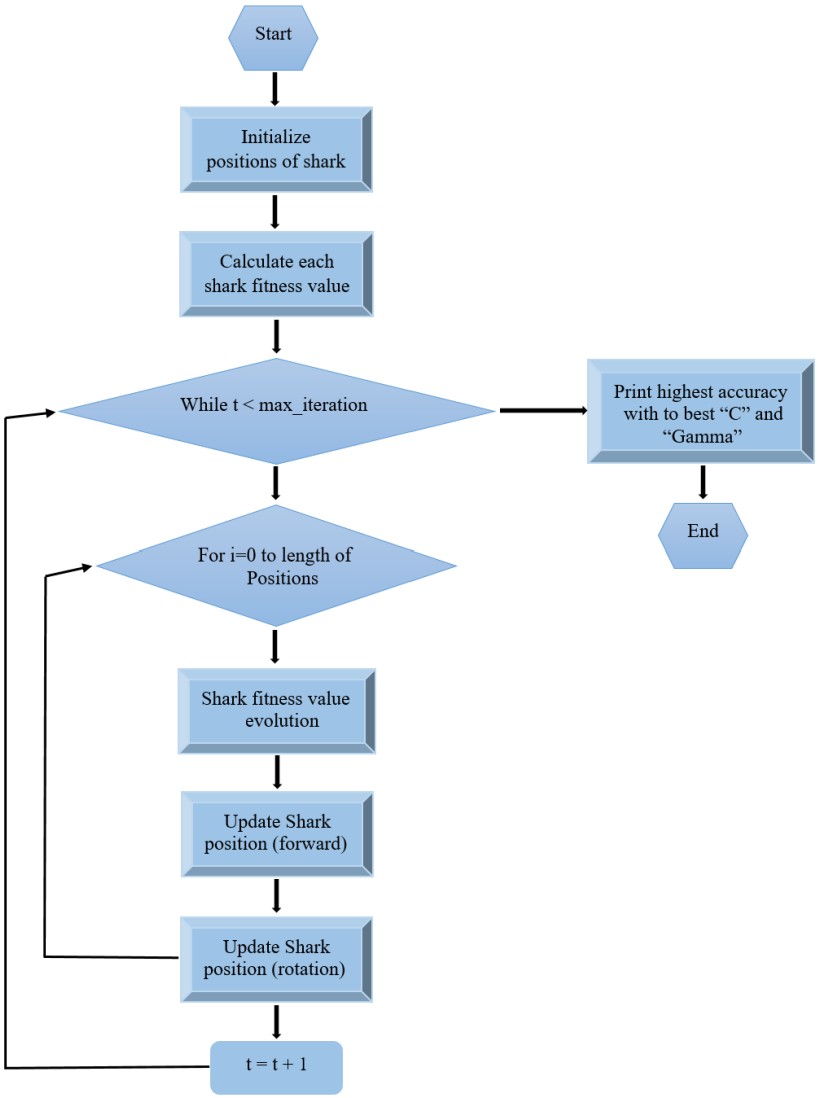

Figure 11: RBF-SVM with SSO

10 RESULTS AND DISCUSSIONS

In this stage, the accuracy of classification with traditional SVM compared to optimized SVM classifier will be visualized and discussed in detail based on the achieved results.

At first, the traditional classifier is trained with train-set data and evaluated with test-set data using its default parameter values. In the second phase, hybrid SVM-SSO respectively is applied to determine the best value for each parameter. The proposed model was evaluated based on the Accuracy metric [32]:

$$\text{Accuracy} = TP + TN / (TP + TN + FP + FN) * 100 \quad (19)$$

Where TP, TN, FP, and FN refer to the True Positive, True Negative rate, False Positive, and False Negative rates respectively. The details of each keyword or data have has been showing in table 1.

Table 1: Dataset Description

Data set	Positive	Negative	Total
Biden	981	1176	2157
Benzema	2362	648	3010
Apple	1818	1876	3694
NASA	3425	563	3988

Table 2 presents the accuracy of the traditional SVM classifier with each keyword dataset using Doc2Vec and (TF-IDF) feature extraction methods.

Table 2: Traditional Classification

Keywords	DBoW	DMC	DMM	TF-IDF
Biden	74.86	73.92	74.97	75.21
Benzema	78.23	77.79	78.67	79.38
Apple	87.76	87.03	88.20	88.69
NASA	85.54	84.93	86.91	86.55

The results show that traditional SVM for the Biden keyword obtained the highest accuracy in the TF-IDF feature extraction method, followed by DMM and DBoW in the second and third levels. For the Benzema keyword, SVM provides better accuracy with TF- IDF methods, followed by DMM, DBoW, and DMC methods. For the Apple keyword, SVM, with TF-IDF and DMM achieves better accuracy compared to DMC and DBoW methods For the NASA keyword, the difference from Biden, Benzema, and apple keywords, the features of DMM methods are better than TF-IDF, DMC, and DBoW methods. Eventually, traditional SVM provides better results with TF-IDF methods in all datasets except in the NASA dataset as shown in figure 12.

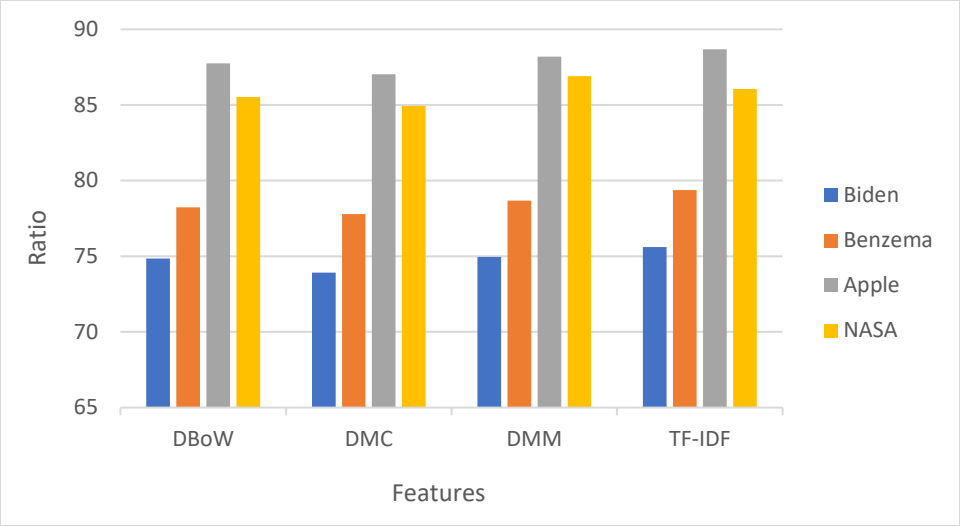

Figure 12: Result of SVM on datasets without optimization

Table 3 shows SVM classification accuracy for each keyword, while optimized by using SSO with DBoW, DMC, DMM, and TDF-IF feature extraction methods.

Table 3: Optimize Classification

Keywords	**DBoW**	**DMC**	**DMM**	**TF-IDF**
Biden	76.77	75.81	77.02	77.21
Benzema	80.11	79.43	80.90	81.59
Apple	90.95	90.15	91.32	92.12
NASA	88.96	88.34	90.01	89.84

For Biden, Benzema, and Apple keywords, optimized SVM provides better accuracy for more than 2% with TF- IDF methods when compared with traditional SVM, followed by DMM, DBoW, and DMC methods.

For NASA keywords, the difference between Biden, Benzema, and apple keywords, the features of DMM methods are better than TF-IDF, DMC, and

DBoW methods also the optimized SVM increased the accuracy by more than 2% compared with traditional SVM. Eventually, optimized SVM provides better results with TF-IDF methods in all datasets except in the NASA dataset also raising the accuracy by more than 2% when compared with traditional SVM as shown in figure 13.

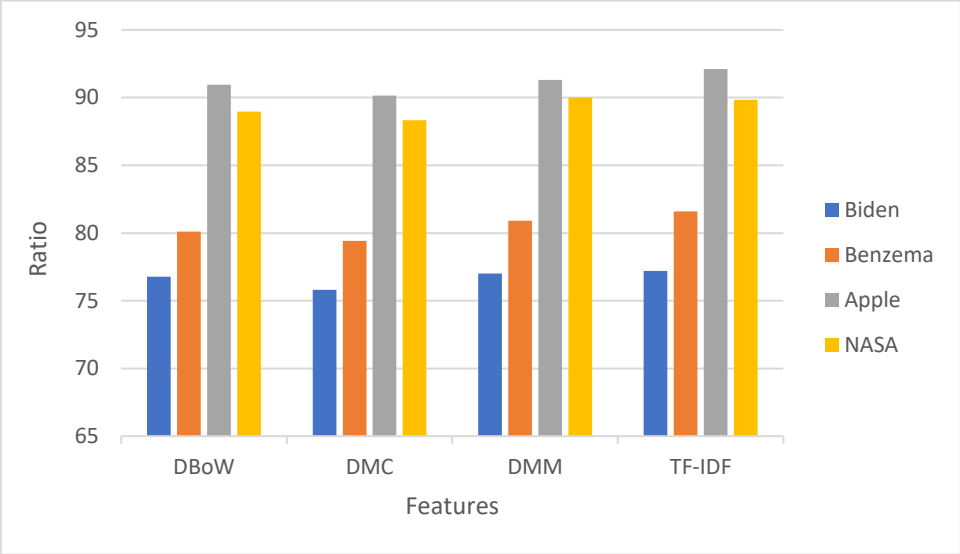

Figure 13: Result of SVM on datasets optimization

11 CONCLUSION

This paper aims to build a big data sentimental analysis model using the most current feature extraction method, Doc2Vec, and TF-IDF, along with an optimized machine algorithm through SSO. The results that the classification algorithm has achieved before and after optimization are compared to each other to show the importance of the optimization algorithms in enhancing the overall model accuracy. The results proved that the TF-IDF features extraction method is more effective than other methods. The optimizer increased the SVM accuracy with all Doc2Vec and TF-IDF feature extraction methods, increasing accuracy by 2% compared to traditional SVM.

REFERENCES

[1] P. Gupta, A. Sharma, and R. Jindal, “Scalable machine-learning algorithms for big data analytics: a comprehensive review,” Wiley Interdiscip. Rev. Data Min. Knowl. Discov., vol. 6, no. 6, pp. 194–214, 2016, Doi: 10.1002/widm.1194.

[2] “What is Big Data Analytics?” https://intellipaat.com/blog/big-data-analytics/ (accessed Aug. 10, 2022).

[3] "Twitter Statistics, Twitter Usage Statistics." https://www.internetlivestats.com/twitter-statistics/ (accessed Aug. 11, 2022).

[4] D. G. Aggarwal, "Sentiment Analysis An insight into Techniques, Application, and Challenges," Int. J. Comput. Sci. Eng., vol. 6, no. 5, pp. 697–703, 2018.

[5] H. Kaur, V. Mangat, and others, "A survey of sentiment analysis techniques," in 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2017, pp. 921–925.

[6] N. Krishnaveni and V. Radha, "Feature selection algorithms for data mining classification: a survey," Indian J Sci Technol, vol. 12, no. 6, 2019.

[7] Q. Shuai, Y. Huang, L. Jin, and L. Pang, "Sentiment Analysis on Chinese Hotel Reviews with Doc2Vec and Classifiers," Proc. 2018 IEEE 3rd Adv. Inf. Technol. Electron. Autom. Control Conf. IAEAC 2018, no. Iaeac, pp. 1171–1174, 2018, Doi: 10.1109/IAEAC.2018.8577581.

[8] M. Ahmad, S. Aftab, M. S. Bashir, N. Hameed, I. Ali, and Z. Nawaz, "SVM optimization for sentiment analysis," Int. J. Adv. Comput. Sci. Appl., vol. 9, no. 4, 2018.

[9] S. Naz, A. Sharan, and N. Malik, "Sentiment classification on Twitter data using support vector machine," in 2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI), 2018, pp. 676–679.

[10] N. Naw, "Twitter sentiment analysis using support vector machine and K-NN classifiers," IJSRP, vol. 8, pp. 407–411, 2018.

[11] F. Rustam, I. Ashraf, A. Mehmood, S. Ullah, and G. S. Choi, "Tweets classification on the base of sentiments for US airline companies," Entropy, vol. 21, no. 11, p. 1078, 2019.

[12] M. Hayouni and S. Baccar, "Sentiment Analysis Using Machine Learning Algorithms," 2021. DOI: 10.1109/IWCMC51323.2021.9498965.

[13] B. AlBadani, R. Shi, and J. Dong, "A Novel Machine Learning Approach for Sentiment Analysis on Twitter Incorporating the Universal Language Model Fine-Tuning and SVM," Appl. Syst. Innov., vol. 5, no. 1, 2022, Doi: 10.3390/asi5010013.

[14] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," Ain Shams Eng. J., vol. 5, no. 4, pp. 1093–1113, 2014.

[15] S. Bandaru and K. Deb, "Metaheuristic techniques," in Decision sciences, CRC Press, 2016, pp. 709–766.

[16] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in international conference on machine learning, 2014, pp. 1188–1196.

[17] M. Bilgin and \. Izzet Fatih \cSentürk, “Sentiment analysis on Twitter data with semi-supervised Doc2Vec,” in 2017 international conference on computer science and engineering (UBMK), 2017, pp. 661–666.

[18] A. Rane and A. Kumar, “Sentiment classification system of Twitter data for US airline service analysis,” in 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC), 2018, vol. 1, pp. 769–773.

[19] C. Cortes and V. Vapnik, “Support-vector networks,” Mach. Learn., vol. 20, no. 3, pp. 273–297, 1995.

[20] R. Pupale, “Support Vector Machines (SVM) — An Overview.” https://towardsdatascience.com/https-medium-com-pupalerushikesh-svm-f4b42800e989 (accessed Aug. 11, 2022).

[21] J. Daniel, “The Support Team — SVM.” https://towardsdatascience.com/the-support-team-svm-555d2c30b1b3 (accessed Aug. 12, 2022).

[22] S. Mohammad-Azari, O. Bozorg-Haddad, and X. Chu, “Shark smell optimization (SSO) algorithm,” in Advanced optimization by nature-inspired algorithms, Springer, 2018, pp. 93–103.

[23] B. T. Ahmed and O. Y. Abdulhameed, “Fingerprint authentication using shark smell optimization algorithm,” UHD J. Sci. Technol., vol. 4, no. 2, pp. 28–39, 2020.

[24] M. Mavrovouniotis, C. Li, and S. Yang, “A survey of swarm intelligence for dynamic optimization: Algorithms and applications,” Swarm Evol. Comput., vol. 33, pp. 1–17, 2017.

[25] M. Ehteram, H. Karami, S.-F. Mousavi, A. El-Shafie, and Z. Amini, “Optimizing dam and reservoirs operation-based model utilizing shark algorithm approach,” Knowledge-Based Syst., vol. 122, pp. 26–38, 2017.

[26] O. W. Salami, I. J. Umoh, E. A. Adedokun, M. B. Mu’azu, and L. A. Ajao, “Efficient method for discriminating flash event from DoS attack during internet protocol traceback using shark smell optimization algorithm,” in 2019 2nd International Conference of the IEEE Nigeria Computer Chapter (NigeriaComputConf), 2019, pp. 1–10.

[27] H. Hosseinzadeh and M. Sedaghat, “Brain image clustering by wavelet energy and CBSSO optimization algorithm,” J. Mind Med. Sci., vol. 6, no. 1, pp. 110–120, 2019.

[28] N. Gnanasekaran, S. Chandramohan, P. S. Kumar, and A. M. Imran, “Optimal placement of capacitors in radial distribution system using shark smell optimization algorithm,” Ain Shams Eng. J., vol. 7, no. 2, pp. 907–916, 2016.

[29] O. Abedinia, N. Amjady, and A. Ghasemi, “A new metaheuristic algorithm based on shark smell optimization,” Complexity, vol. 21, no. 5, pp. 97–116, 2016.

[30] H. Hosseinzadeh, “Automated skin lesion division utilizing Gabor filters based on shark smell optimizing method,” Evol. Syst., vol. 11, no. 4, pp. 589–598, 2020.

[31] S. A. L. I. JUMA, “Optimal radial distribution network reconfiguration using modified shark smell optimization,” JKUAT-PAUSTI, 2018.

[32] M. K. Das, B. Padhy, and B. K. Mishra, “Opinion mining and sentiment classification: A review,” in 2017 International Conference on Inventive Systems and Control (ICISC), 2017, pp. 1–3.