Journal of Education for Pure Science
Vol.15, No.2 (2025)
DOI: https://doi.org/10.32792/jeps.v15i2.526

Website: jceps.utq.edu.iq Email: jceps@eps.utq.edu.iq

MPA-SVM: An Effective Feature Selection Approach for High-Dimensional Datasets

Hadeel T. Ibrahim[†], Wamidh J. Mazher[‡]

[†]Information Technology Division, Col. of Education for Women, Shatrah University , Thi-Qar, Iraq, Hadeel.tariq@shu.edu.iq , https://orcid.org/0000-0001-9749-4024 .

[‡]Electrical Engineering Dept., Southern Technical University, Basra, Iraq, wamidh.mazher@stu.edu.iq , https://orcid.org/0000-0003-2092-3745

Received 13/8 /2025, Accepted 6 / 10 /2024, Published 1 / 6 /2025

Abstract

Finding the ideal number of superior features to optimize the learning algorithm's performance is the ultimate aim of feature selection. However, when a dataset's features count rises, this issue gets more difficult to solve. The Marine Predators Algorithm (MPA) is a novel metaheuristic that has proven effective in solving many optimization issues. In MPA, the fundamental exploratory and exploitative procedures are modified to choose the best and most significant features in order to get the most accurate categorization. The outcomes demonstrate the exceptional ability of the suggested MPA-SVM strategy to choose the most important and ideal attributes. Support vector machines (SVMs) are an essential method that are skillfully used to address classification problems. In this work the MPA is adjusted using the SVM as classifier. The present study proposes MPA- SVM as a solution to the issue of feature selection in high dimensional datasets. The suggested method's efficacy was confirmed using ten high-dimensional datasets acquired from Arizona State University (ASU) repository. The proposed method was compared with six state-of-the-art feature selection algorithms, including ASO and EO, demonstrating superior performance. Standard deviation is a measure of the robustness and stability of optimization algorithms, it is used in MPA-SVM and achieved lowest values. Across all datasets, MPA-SVM produces the lowest average error rates, minimum classification standard deviation (STD) values , FS rates and runtime.

Keywords

Marine Predators Algorithm, Feature selection , Support Vector Machine, Data mining , High Dimensional Datasets

1 Introduction

One important element that might impact how well a machine learning (ML) model performs in data mining activities is the dimensionality of the input. Upcoming sophisticated data collection tools have made massive amounts of data accessible for numerous uses in the near future[1] . However, working with these massive, high-dimensional datasets calls for computational resources. Furthermore, adding noisy data, such as unnecessary and redundant features, would significantly impair the ML model's performance. Since these noisy features have the potential to confuse the learning algorithms, they must be removed from the original dataset (Moorthy & Gandhi, 2021). To address the dimensionality problem, feature selection (FS) is suggested in this context.

In order to convey the goal concepts as effectively as possible, FS selects a collection of features from a dataset that are the most representative. It is advantageous for the readability and interpretability of the model in addition to removing redundant and unnecessary information [3]. In general, there are two primary types of FS: wrapper and filter. Statistical characteristics, mutual information, and distance measurement are used by filter algorithms to determine the ranking of the features. During the selecting process, they take no learning models into account [4]. Wrapper approaches, on the one hand, produce the results by taking into account every detail of a particular learning model.

The wrapper approaches are better at optimizing the learning algorithms' performance when compared to filter methods [5].

It is believed that the FS process is an NP-hard combinatorial optimization issue. To determine the optimal subset for a dataset containing N features, a total of 2^N feature subsets must be examined [6] . Therefore, for the high-dimensional dataset, it is not realistic to search through all of the potential subsets in pursuit of the ideal feature subset.

In recent times, a variety of engineering and optimization techniques have effectively utilized metaheuristics. Many research scientists use metaheuristic algorithms as selection mechanisms in the FS process because of their exceptional performance.

Differential evolution (DE) [7], Particle swarm optimization (PSO) [8], Genetic algorithms[9] , Ant colony optimization [10], Grasshopper Algorithm [11], Bat algorithms [12] and Salp algorithm [13] are a few instances of metaheuristic algorithms.

Marine Predators Algorithm (MPA) is primarily inspired by the widely used foraging strategy, specifically Lévy and Brownian movements, in ocean predators as well as the ideal encounter rate policy in the biological relationship between predator and prey [14]. MPA showed results that were extremely competitive. It demonstrates effectiveness in solving optimization issues. Numerous benefits come with MPA, including precise calculation, straightforward setting, easy implementation, and fewer parameters

[15]. Accordingly, MPA is used in solving the ORPD problem[16], identifying parameters of Triple-Diode[17], feature selection[15], forecasting COVID-19 cases[18], and wrapper-based feature selection[19].

The primary objective of this study is to increase the MPA algorithm's effectiveness when handling high dimensional FS situations. Our proposal is to introduce a modified Marine Predators Algorithm (MPA) with SVM as classifier to enhance MPA's search power in a higher dimension dataset. The underlying machine learning model (ML) used to evaluate the quality of the chosen features is SVM. Arizona State University's (ASU) 10 high-dimensional datasets were used to assess the suggested methods. Additionally, seven cutting-edge techniques were employed in this paper to confirm the efficacy of the suggested methodologies.

The following are some notable contributions made by this paper:

- Presenting a distinctive MPA-based algorithm for solving high-dimensional FS problem that adheres to the confronting rate strategy and ideal feeding strategy of marine environments between predators and prey.
- Contrasting the effectiveness of MPA with modified, well-known swarm intelligence algorithms as ASO [20], EO [21], EPO [22], MBO [23], SBO [24] and SCA [25] for FS applied on 10 high-dimensional datasets. Additionally, a fair comparison is made in terms of error rates, average FS and the standard deviation of features chosen.
- Support Vector Machines (SVM) [26] is used to realize the influence of the modified MPA-based classifier kind.

There are five sections in this article. The details of the suggested method MPA-SVM and how it is applied to FS in high dimensional datasets are provided in Section 2. The experimental findings are presented in Section 4. Section 5 concludes by summarizing and discussing the research findings.

2 Methodology

2.1 Marine Predator Algorithm (MPA)

MPA is proposed in 2020 by [14]. It is a novel algorithm that emulates the Lévy foraging strategy, Brownian movements in marine predators, and optimal encounters rate policy in predator-prey biological interactions. Furthermore, no algorithm exists that is able to learn the pattern of optimization outcomes by memory. The MPA method has an advantage compared to other algorithms in that it retains optimization outcomes. This is similar to how marine predators may remember important information, such as the site where they are effective foraging, thanks to their excellent memory.

The MPA method solves the feature selection problem in high dimensional datasets with fewer iterations and all of these advantages. In a marine ecological system, MPA stimulates the foraging behavior of ocean predators as well as the frequency of encounters between predators and prey. In MPA, prey and predator pursue one another while simultaneously searching for food. The creatures that are the prey and the predator are viewed as exhaustive review boards. They adhere to the survival of the fittest theory, which raises the likelihood that predators will find prey. For defining the MPA optimization process, the Lévy strategy and the Brownian process work best together.

For the best chance of survival in natural settings, predators must choose the best tactic to increase the frequency of encounters with prey [27]. Numerous animals in the wild employ the efficient random walk technique in their foraging routines. Scientists refer to this as a stochastic or random process. The animal's current position and the likelihood that it will move to the next place can be used to mathematically model where it will end up next [28]. Predators have selected these best practices because they have evolved naturally in the wild.

Lévy walks serve as the foundation for searching for patterns category of random walk techniques. The motions made by animals when foraging is referred to as the "Levy walk" [6]. A few benefits of the MPA algorithm include a low number of defined variables, straightforward process, minimal computational burden, significant convergence pace, near-global solution, freedom from the problem, and gradient-free style.

The MPA mathematics model's two primary random walks—(a) Lévy motion and (b) Brownian Lévy motion—are explained below.

(a) Lévy motion

A stochastic process with independent, stationary increments is referred to as a Lévy process, for the mathematician Paul Lévy. The Lévy flight is a sort of random walk where the step sizes are represented by a probability function that is specified by the power of law tailed of the Lévy distribution, as stated in Eq. (2.1):

$$L(a_j) \approx \left|a_j\right|^{1-c} \qquad (2.1)$$

where $1 < c \leq 2$ is the power-law exponent and $a_j$is the flight length.

whereby d chooses the scale unit and c displays the distribution index and manages the scale parameters of the process. Equation (2.2) illustrates how to compute the integral form of the probability of density for the Levy stable process:

$$f_L(a;c,d) = \frac{1}{\pi}\int_0^\infty \exp(-dp^c)\cos(pa)\,dp \qquad (2.2)$$

whereby d chooses the scale unit and c displays the distribution index and manages the scale parameters of the process. A Gaussian distribution is displayed when c=2, while a Cauchy distribution is represented when c=1 [6]. Eq. (2.2) gives an analytical solution in a small number of instances.

In Eq. (2.3), the series expansion method is typically used to solve the problem only in cases where a has a large value, as stated below:

$$f_L(a;c,d) \approx \frac{d\Gamma(1+c)\sin\left(\frac{\pi c}{2}\right)}{\pi a^{(1+c)}}, a \to \infty \qquad (2.3)$$

Gamma function Γ, where Γ (1+c) = c! for integer c numbers.

(b) BROWNIAN MOTION

The process known as Brownian motion is one wherein the variance (z2=1) and step length (b=1) are determined by the probability function of a Gaussian distribution when the mean is equal to zero. The

Probably Density Function (PDF) is defined in point an of this motion as follows in Eq. (2.4) [29]: (a; c, d).

$$f_B(a;b,z) = \frac{1}{\sqrt{2\pi z^2}} \exp - \left(\frac{(a-b)^2}{2z^2}\right)$$
$$= \frac{1}{\sqrt{2\pi}} exp\left(-\frac{a^2}{2}\right) \qquad (2.4)$$

A novel and useful methodology has been proposed in a paper [30], which employs the Magneta method to obtain random numbers from the Lévy distribution. This is demonstrated in equation (2.5), where the random values of the index distribution (c) fall between 0.3 and 1.99 [31].

$$Levy(c) = 0.05 \times \frac{a}{|a\prime|^{1/c}} \qquad (2.5)$$

where the standard deviations of za are defined as follows in Eqs. (2.6), (2.7) and (2.8), and where the normally distributed variables are a and a' :

$$a = Normal(0, z_a^2) \qquad (2.6)$$

$$a' = Normal\left(0, z_{a'}^2\right) \qquad (2.7)$$

$$z_a = \left[\frac{\Gamma(1+c)\, sin\left(\frac{\pi c}{2}\right)}{\Gamma\left(\frac{(1+c)}{2}\right) c2^{\frac{(c-1)}{2}}}\right]^{1/c} \quad and\ z_{a'} = 1\ and\ c = 1.5 \qquad (2.8)$$

2.2 The MPA Optimization Mechanism

This mechanism is broken down into three primary optimization stages that are based on different velocity ratios and concurrently mimic the behavior of predators and prey in the wild. The following is a definition of these stages:

- **The first stage**: occurs when the prey outpaces the predator in speed and there is a high-speed ratio. Moving is not the ideal approach for Predator when there is a high-speed ratio (s ≥ 10), according to the extracted standards. Exploration is crucial during this first optimization stage. This rule's mathematical model is used as shown in eq(2.9):

$$\text{While } Itr < \frac{1}{3}\max\_Itr$$
$$pacesize_x = \overrightarrow{A_B} \otimes \left(Elite_x - \overrightarrow{A_B}\,\overline{\otimes\, Prey_x}\right) \quad x = 1, \dots n$$
$$\overrightarrow{Prey_x} = \overrightarrow{Prey_x} + B.\vec{A} \otimes pacesize_x \qquad (2.9)$$

Arbitrary numbers make up the vector of $\overrightarrow{A_B}$in the Brownian motion normal distribution. The gait of the prey is mimicked by multiplying $\overrightarrow{A_B}$by Prey. A is a uniform randomness vector in [0,1], and B is a constant number equal to 0.5. This stage occurs every 1/3 of iterations, during which high rate of movement permits high levels of exploration (Itr being the current iteration, Max_Itr being the maximal one).

- **The second stage** : occurs when the prey and predator are traveling at the same time. When exploration tries to be quickly transformed into exploitation, this is what happens. This stage includes exploration and exploitation issues; predators are in charge of exploration, and prey is in charge of exploitation.

Predator's optimal approach is Brownian if the unit velocity ratio ($s \approx 1$) is met and moving in Lévy is the best course of action for Prey, according to the rule that was established. Equation (2.10) illustrates how this rule's mathematical model is used.

$$\text{While } 1/3 \max\_Itr \prec Itr \prec 2/3 \max\_Itr$$

$$\overrightarrow{pacesize_x} = A_L \otimes \left(\overrightarrow{Elite_y} - A_L \otimes \overrightarrow{prey_y}\right) \quad x = 1, \dots {}^{n}/_{2}$$

$$\overrightarrow{prey_x} = \overrightarrow{prey_x} + B.\vec{A} \otimes pacesize_x \qquad (2.10)$$

Based on the Lévy distribution, A_y represents a vector of random numbers for the first part of the population. The exponential growth of A_y mimics the Lévy-style motion of the prey. Pace size can be added to the prey place to better replicate the gait of the prey.

In the Lévy distribution, the majority of pace sizes are tiny. According to this study, the remaining half of the population is represented by:

$$pacesize_x = \overrightarrow{A_G} \otimes \left(A_G \otimes \overrightarrow{Elite_x} - \overrightarrow{prey_x}\right) \quad x = {}^{n}/_{2}, \dots n$$

$$\overrightarrow{prey_x} = \overrightarrow{Elite_x} + B.CD \otimes \overrightarrow{pacesize_x} \qquad (2.11)$$

Where $CD = \left(1 - \frac{Itr}{Max\_\{Itr\}}\right)^{2\frac{Itr}{Max\_\{Itr\}}}$

Whereas CD is an adapted parameter that regulates the size of the step taken by predators. The multiplication of $A_G$simulates the Brownian movement of the predator. Additionally, Elite replicates the mobility of Prey by adjusting its location in response to predators' Brownian motion.

- **The third stage:** in which the predator has a low-speed ratio and is outpacing the prey. This stage is linked to strong exploitation ability and aids in it.

When the speed-ratio is modest (s= 0.1), Lévy is the most effective Predator approach. In this stage, the multiplication of A_L simulates the movement of the Predator in the Lévy strategy. Additionally, Elite is emulated by adding the pace size to the Elite location, whereas the Predator's mobility assists Prey in updating its current location. This is shown as:

Where $Itr \prec {}^{2}/_{3}$ max _Itr

$$pacesize_x = \overrightarrow{A_L} \otimes \left(\overrightarrow{Elite_x} - \overrightarrow{Prey_x}\right) \quad x = 1, \dots. n$$

$$\overrightarrow{Prey_x} = \overrightarrow{Elite_x} + B.CD \otimes \overrightarrow{pacesize_x} \qquad (2.12)$$

There is a set amount of iteration time designated for each stage. These stages are distinguished by the laws that regulate the movements of the predator and prey, which are mimicked in the actions of the predator and prey. These three stages mimic the size of a predator's step as it approaches its target. During a predator's lifelong, the guidelines presume that the ratio of Lévy and Brownian motion will not change. In the first stage, the Predator is motionless, and in the second, it moves in a Brownian manner. It employs the Levy approach in the third stage. This situation also affects prey considering prey additionally has the capacity to be a predator. In particular, during the first stage Prey moves in a Brownian manner, while during the second stage it exhibits Lévy behavior [15].

Variations in surroundings can also influence how marine predators behave. The impacts of Fish Aggregating Devices (FADs), sometimes referred to as eddy generation, are one illustration of this. Whales spend over 80% of their daily lives in the vicinity of FADs; the remaining 20% of their day is likely spent making lengthy plunges in several dimensions in search for various prey habitats [32]. As regional optima, overall FADs have the effect of keeping these sites in the search space. These lengthier hops prevent holding up in local optima through the experiment. The following defines the mathematical model of the effect of FAD:

$$\overline{prey_y} = \begin{cases} \overline{prey_x} + CD\left[E_{min} + \vec{A} \otimes (\bar{E}_{max} - \bar{E}_{min})\right] \otimes \vec{V} & if\ h \leq FADs \\ \overrightarrow{\overline{prey_y}} + [FADs(1-h) + h](\overline{prey_{r1}} - \overline{prey_{r2}}) & if\ h > FADs \end{cases} \quad (2.13)$$

2.3 MPA exploitation and exploration phases

In the first phase, the Prey moves in a Brownian motion, according to the optimization phases that were previously discussed. At the beginning of the search, the prey is dispersed uniformly across the search field. Due of the comparatively vast distance between the predator and the prey, Brownian motion is used to attain meaningful range exploration. Prey is able to independently explore the surroundings as a result. Then, if the current position is more advantageous than the one it substituted, every Prey is assessed for fitness. The prey's locations can be regarded as important food regions, and the prey's capacity to recall important areas of food is comparable to the saving process. When the Prey becomes more adept at finding foods on its own, it can be classified as a Predator.
This implies that the Prey replaces the leading Predator if its fitness value is greater than that of the latter. The second phase of the optimization begins when the Predator goes foraging. The optimization moves from exploration to exploitation within this phase. In this phase, the two species of Predator and the Prey need to look for foods in order to succeed in exploitation as well as exploration. In this period, fifty percent of the population is responsible for exploration and the remaining half for exploitation. The Brownian method is one that predators employ to find their prey. Simultaneously, Prey begins to explore its immediate vicinity. If it is unable to locate any food in the vicinity, it employs the Levy technique, which involves a lengthy leap. During the long steps in the Lévy strategy and the increasing proximity of the predator and prey sites compared to the preceding phase, the impact of FADs aids in improving MPA efficiency and preventing stagnation of local optima.

In the last phase of the optimization process, the MPA algorithm demands a high level of exploitation capability.

During this phase, the Predator begins to employ the Lévy approach instead of the Brownian technique in order to seek certain areas more effectively. In a given neighborhood, predators assist in limiting the search areas for exploitation by utilizing the adaptively set convergence factor (CF). By utilizing the extended pace sizes of the Lévy approach for the domain's non-promising places CF additionally avoids wasteful searching effort. The MPA pseudo-code is displayed in Algorithm 1.

Algorithm 1: The MPA pseudo-code(Faramarzi, Heidarinejad, Mirjalili, et al., 2020)

Establish the initial population of agents of searching (Prey) with x = 1, 2,...,n.
while the halt condition is not satisfied **do**
Determine the starting fitness function and create the Elite matrix.
If Itr < Max_Itr/3 **then**
Refresh the prey using Equation (3.9)
If Itr < Max_Itr/3 < Itr < 2*Max_Itr/3 **then**
Regarding the initial portion of the population (x=1,...,n/2)
Refresh the prey using Equation (3.10)
Regarding the second portion of the population (x=1,...,n/2)
Refresh the prey using Equation (3.11)
If Itr > 2 * Max_Itr/3 **then**
Refresh the prey using Equation (3.12)
End if
End if
End if
Finish the Elite upgrade and memory saver
Applying the FADs effect and updating with Equation (3.13)
End while

2.4 SVM outline

SVM was invented by Vapnik [33], who included into account density estimating, regression, and classification in its whole. Let + and - be two different categories of things that are present in two different dimensional planes. Our goal is to create a barrier between these two regions. It needs to be able to shift in both dimension and orientation. Furthermore, we often identify the widest border that maintains object separation. The hyperplane was shown as follows:

$$z^T a + c = 0$$

$z^T a$ found the plane direction in which c regulates the origin's migration. Two hyperplanes can be used for describing the margin. By adjusting the default vector's angle, we can rotate the edge. Margin modified by adjusting the c values. . The margin size is equivalent to $\frac{2}{\|z\|}$ these incomes that the width is inverse relative to the distance of regular vector. Now for the points a_m, a_n with b_m, by_n labels, where: $a_m = [-1,1]^T$, $b_m = +1$, $a_n = [1,-1]^T$, $b_n = -1$

The goal of this optimization task is to increase the margin's size:

$$max \frac{2}{\|z\|} \rightarrow \min \frac{\|z\|}{2} \rightarrow min \frac{1}{2}\|z\|^2$$

The limitations are:

$$z^T a_m + c \geq 1 \qquad since + \ class\ is\ over\ the\ boundary$$

$$z^T a_n + c \leq -1 \qquad since - \ class\ is\ under\ the\ boundary$$

$b_m(z^T a_m + c) \geq 1 \quad and \quad b_n(z^T a_n + c) \geq 1$ were the result of multiplying each limitation by the labels.

Ultimately, we must optimize: $min \frac{1}{2}\|z\|^2 \quad subject\ to: \quad b_i(z^T a_m + c) \geq 0$

We used Lagrange multipliers to multiply the inequalities in order to solve it. Lagrange multipliers are inserted with $\alpha_{\,i}$:.

$$G_{ef} = \frac{1}{2}\|z\|^2 - \sum_{i=1}^{N} \alpha_i [b_i(z^T a_i + c) - 1] \qquad (2.14)$$

Several partial derivations were discovered by using the Karush-Kuhn-Tucker criteria[34], as follows: $\frac{\mu G_{ef}}{\mu z} = 0$, $\frac{dG_{ef}}{dc} = 0$, $\alpha_i \geq 0$

$$\alpha_i [b_i(z^T a_i + c) - 1] = 0 \qquad (2.15)$$

Where $\frac{\mu G_{ef}}{\mu z} = 0 \longrightarrow z = \sum_{i=1}^{N} \alpha_i\, b_i a_i$, $\frac{dG_{ef}}{dc} = 0 \longrightarrow \sum_{i=1}^{N} \alpha_i\, b_i = 0$

Employing Karush-Kuhn-Tucker on the primal-dual, we obtained:

$$G_d = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i \alpha_j\, b_i b_j\, {a_i}^T a_j + \sum_{i=1}^{N} \alpha_i \qquad (2.16)$$

By using knowns in place of: $b_m = 1,\ b_n = -1, a_m = [-1, 1]^T,\ a_n = [1, -1]^T$

$$G_d = -\alpha_m^2 - \alpha_n^2 - 2\alpha_m \alpha_n + \alpha_m + \alpha_n \qquad (2.17)$$

$\frac{\mu G_d}{\mu \alpha} = 0 \longrightarrow \alpha_m + \alpha_n = {}^1/_2$

$\sum_{i=1}^{N} \alpha_i b_i = 0 \longrightarrow \alpha_m = \alpha_n$, $where\ \alpha_m + \alpha_n = {}^1/_2 \longrightarrow \alpha_m = \alpha_n = {}^1/_4$

The normal vector z can be found by:

$$z = \sum_{i=1}^{N} \alpha_i b_i a_i = [{}^{-1}/_2, {}^1/_2]^T \qquad (2.18)$$

To determine bias, bs:

$\alpha_i [b_i(z^T a_i + bs) - 1] = 0$, $bs = {}^1/_{b_i} - z^T a_i \quad \forall i \quad S.T. \quad \alpha \neq 0$, $bs = 0\ \forall i$

3 The proposed MPA-SVM system

The ability of the MPA technique to recall patterns of optimization findings using the sites for effective foraging by marine predators and their partners is one of its numerous advantages. Furthermore, MPAs need a lot less iterations. Its many benefits include being gradient-free, having a straightforward process, requiring less computing power, accelerating convergence, and being insulated from the problem. These essential benefits are all very helpful in resolving high dimensionality issues. The steps to build MPA-SVM system are listed below:

3.1 Features normalization

First of all , we need to normalize the inputted features using a vector of real values. Based on Min-Max normalization [35]features are randomly mapped onto the interval [0,1] using eq. (3.1). The variable is scaled to a percentage of the whole range of the original dataset by this division. The adjusted value thus lies between 0 and 1. As a result, if the component value is higher or equal to 0.5, it will be replaced by 1 and the feature is selected; if not, the value is calculated to be 0 and the feature is not selected.

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (3.1)$$

3.2 Fitness function

The feature subset with the lowest classification error rate and the fewest features chosen is the optimal one. The next equation illustrates how the fitness function is utilized in MPA-SVM to assess individual searching agents:

$$Fit_{min} = ac_L(DS) + b\frac{|L|}{|O|} \quad (3.2)$$

Where $ac_L(DS)$ is the error rate which related to the decision on selection DS. The chosen features subset's length is denoted by L, where O is the overall count of datasets' features. The variables a and b are equivalent to the classification quality significance and chosen subset's feature length regarding that $a\epsilon\ [0,1]\ and\ b = 1 - a$ as approved in [36].

3.3 System architecture

The MPA-SVM architecture is shortened in **Fig. 1**, which reveals the relatives amongst the key portions of the system. As seen, MPA-SVM begins by collecting datasets and then, as previously mentioned, applies normalization to such datasets. Subsequently, the optimized features undergo model validation (cross validation) through the application of training and testing procedures. In order to generate the specified subset feature, the features in this case need to be marked with 1s and deployed to the entire dataset (training and testing sets). The MAP pseducode is shown before in subsection 2.3. The fitness evaluation is then used following the computation of the error rate. Until the maximal iteration is reached and the termination condition is met, the process is carried out repeatedly.

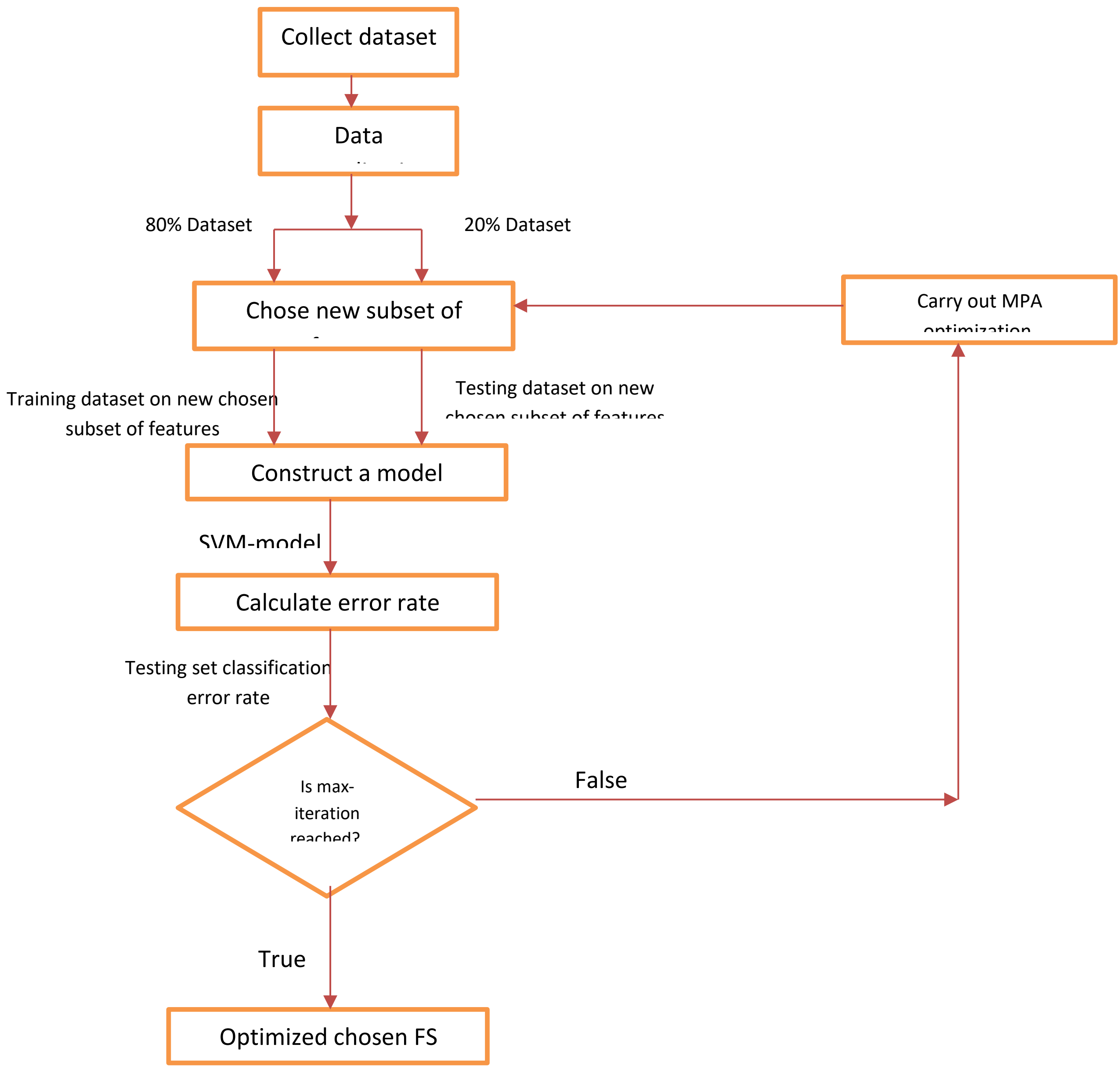

Fig .1 Proposed MPA-SVM architecture

4 Experiments results & discussions

In order to verify and assess the efficacy of the suggested MPA-SVM algorithm, MPA-SVM was juxtaposed with six well-known and contemporary optimization algorithms, such as ASO [20], EO [21], EPO [22], MBO [23], SBO [24] and SCA [25]. Every experiment was run using 10 high dimensional benchmark datasets that were taken from Arizona State University (ASU) repository [37].The following stages present the used datasets and complete experiment details:

4.1 Datasets specifications

Ten high dimensional benchmark datasets from the ASU repository were employed in this study. These datasets' specifics are shown in TABLE 1. . Every test was done with the setting listed in TABLE 2. .

TABLE 1. Employed ASU datasets.(Datasets | Feature Selection @ ASU, n.d.)

No.	Dataset	No. of features (attributes)	No. of instances
1	CLL-SUB-111	111	11340
2	20newsgroups	171	5748
3	GLA-BRA-180	180	49151
4	GLI-85	85	22283
5	orlraws10P	100	10304
6	pixraw10P	100	10000
7	SMK-CAN-187	187	19993
8	TOX-171	171	5748
9	AR10P	130	2400
10	PIE10P	210	2420

4.2 The parameter setting for algorithms and experiments

The suggested algorithm's fit performance was verified in each experiment using an SVM classifier based on the wrapper technique. TABLE 2. and TABLE 3. also display the used PC descriptions and the parameter settings for the additional baseline optimization methods, which are ASO, EO, EPO, MBO, MPA, SBO, and SCA, respectively. Furthermore, the population size for each method was set to 10, and the highest number of iterations permitted was 100.

TABLE 2. PC descriptions

Name	Descriptions
CPU	Intel(R) Core(TM) i7-5500U
RAM	2.40 GHz, 8 GB RAM
OS	Windows 10
APPLICATION	MATLAB R2015a

TABLE 3. Optimization algorithms' parameter set in use.

Algorithm name	Parameters setting
ASO[20]	α= 50 β=0.4
EO[21]	a1=3 (constant 1) a2=2 (constant 2) GP=0.7 (Generation probability)
EPO[22], [23]	M = 4 (movement parameter)
MBO[23]	Per = 1.2 (migration period) p = 5/12 (ratio) Smax= 1 (maximum step) BAR= 5/12 (butterfly adjusting rate)

MPA[14]	N1= 4 (number of butterflies in land1) P= 0.6 (constant) FADs= 0.4 (fish aggregating devices effect)
SBO[24]	α= 0.95 (constant) z= 0.01 (constant) MR = 0.04 (mutation rate)
SCA[25]	α= 2 (constant)

4.3 Evaluation metrics

All algorithm's ultimate accuracy in classification is assessed using the well-known Support Vector Machine (SVM) classifier[38], together with wrapper-based feature selection. A four of evaluation metrics is employed here to evaluate various facets of performance:

a. **Classification error rate:** to calculate a classifier's error rate using test data is to divide the number of erroneously categorized objects the by total number of items [39].

b. **Standard Deviation (STD):** the variable "std" represents the variance of the best results that were achieved after a random optimizer was executed for a number of runs. Std is used to measure the resilience and stability of optimization; lesser values of Std indicate that the algorithm always ends to the same solution, whereas bigger values of Std indicate significantly more irregular performance [40].

c. **Average selected features percentage:** the secondary goal of the fitness function that is being employed is average selected features percentage, which is the average ratio of the features that have been chosen to the entire number of features multiplied by 100.

d. **Algorithms runtimes:** runtime allows us to measure how long an algorithm takes to execute in relation to the size of the input. It gives us a methodical and unambiguous approach to evaluating and contrasting algorithmic efficiency.

4.4 Experiments discussion

Here, we provide an overview and the findings from each experiment. The MPA-SVM technique was customized to handle the feature selection problem in high dimensional datasets through three key experiments, as mentioned in last subsection. The suggested adaptive algorithm, or MPA-SVM, was put into practice on a PC, and its specifications are given in TABLE 2. The algorithms tested on ten publicly available high dimensional datasets included Atom Search Optimization (ASO), Equilibrium Optimizer (EO), Emperor Penguin Optimizer (EPO), Monarch Butterfly Optimization (MBO), Satin Bowerbird Optimizer (SBO), and Sine Cosine Algorithm (SCA).

TABLE 3 lists the parameters that are set in each algorithm. The experiments repeated for 100-iteration and ten search agents in all employed algorithms. Based on the rate of classification error, Table 4. compares the performance of all methodologies. The results in Table 4. show that the proposed MPA-SVM has the best (less) classification error rate in 80% of all datasets and lowest average error rate (4.521). MPA showed results that were extremely competitive. It demonstrates effectiveness in optimization issues. Numerous benefits come with MPA, including precise calculation, straightforward setting, easy implementation, and fewer parameters. In the last stage of the optimization process, the MPA algorithm demands a high level of exploitation ability. During this phase, the Predator begins to employ the Lévy approach instead of the Brownian technique in order to seek a specific neighborhood more effectively. Accordingly, MPA couldn't conduct the lowest classification error rates in 20% of datasets.

Table 4. is visualized in **Fig. 2** .

TABLE 4. Comparing the suggested methods depending on the rate of classification error

Datasets	ASO	EO	EPO	MBO	MPA	SBO	SCA
CLL-SUB-111 (111*11340)	22.728	18.182	13.637	22.728	13.637	31.819	**9.091**
20newsgroups (171*5748)	8.824	8.824	14.706	11.765	**2.942**	17.648	8.824
GLA-BRA-180 (180*49151)	13.889	16.667	13.889	30.556	**5.556**	19.445	25.000
GLI-85 (85*22283)	**0.000**	**0.000**	**0.000**	**0.000**	**0.000**	17.648	5.883
orlraws10P (100*10304)	10.000	**0.000**	**0.000**	5.000	**0.000**	10.000	**0.000**
pixraw10P (100*10000)	5.000	**0.000**	**0.000**	**0.000**	**0.000**	5.000	**0.000**
SMK-CAN-187(187*19993)	10.811	18.919	13.514	21.622	**2.703**	16.217	21.622
TOX-171 (171*5748)	11.765	**2.942**	14.706	11.765	11.765	17.648	5.883
AR10P (130*2400)	42.308	30.770	19.231	42.308	**3.847**	30.770	11.539
PIE10P (210*2420)	4.762	7.143	2.381	2.381	4.762	9.524	**0.000**
Average error rate	13.009	10.345	9.206	14.813	**4.521**	17.572	8.784

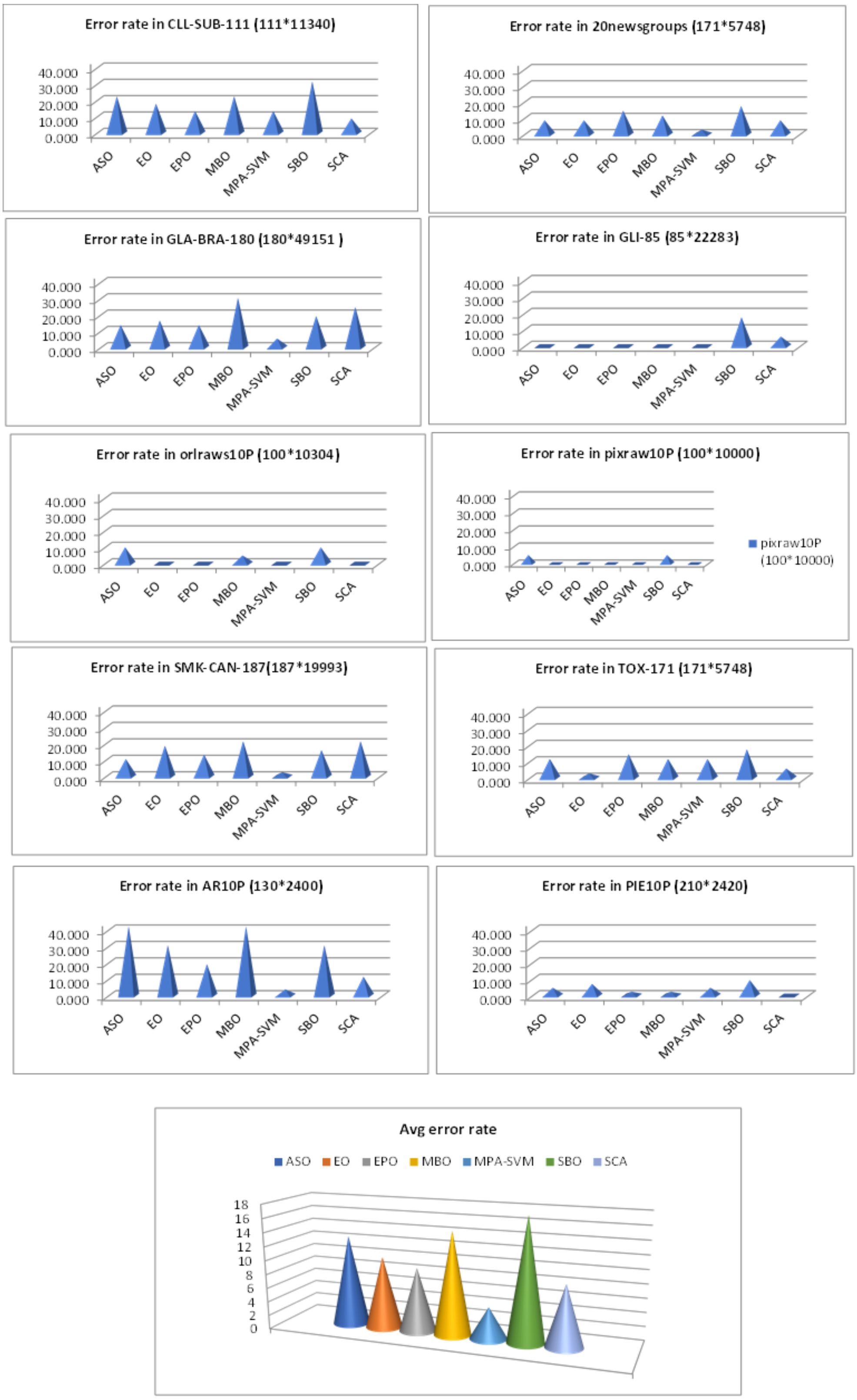

Fig .2 Comparing the suggested methods depending on the rate of classification error

TABLE 5. Comparing the suggested methods depending on the STD

Datasets	ASO	EO	EPO	MBO	MPA-SVM	SBO	SCA
CLL-SUB-111 (111*11340)	0.028	0.037	0.044	0.013	**0.003**	0.011	0.073
20newsgroups (171*5748)	0.019	0.061	0.009	0.028	**0.005**	0.010	0.037
GLA-BRA-180 (180*49151)	**0.006**	0.025	0.080	0.008	0.033	0.002	0.038
GLI-85 (85*22283)	0.033	0.033	0.016	0.011	**0.001**	0.730	0.013
orlraws10P (100*10304)	0.000	0.001	0.007	0.080	**0.000**	0.047	0.036
pixraw10P (100*10000)	0.450	0.015	0.006	0.770	**0.000**	0.077	0.023
SMK-CAN-187(187*19993)	0.020	0.042	0.034	**0.003**	0.045	0.009	0.015
TOX-171 (171*5748)	0.026	0.030	0.018	0.021	**0.002**	0.024	0.028
AR10P (130*2400)	0.044	0.024	0.061	0.011	**0.000**	0.011	0.047
PIE10P (210*2420)	0.000	0.015	0.060	0.002	0.003	**0.000**	0.014
Average STD rate	0.063	0.028	0.034	0.095	**0.009**	0.092	0.032

According to the second metric, the standard deviation (STD), MPA-SVM achieved the lowest STD in 70% of all datasets and less STD rate, as listed in TABLE 5. and visualized in **Fig. 3** . These outcomes show that the suggested method can manage high-dimensional data collections. Furthermore, for the majority of the data sets, MPA-SVM displays decreased Std values, confirming the algorithm's robustness.

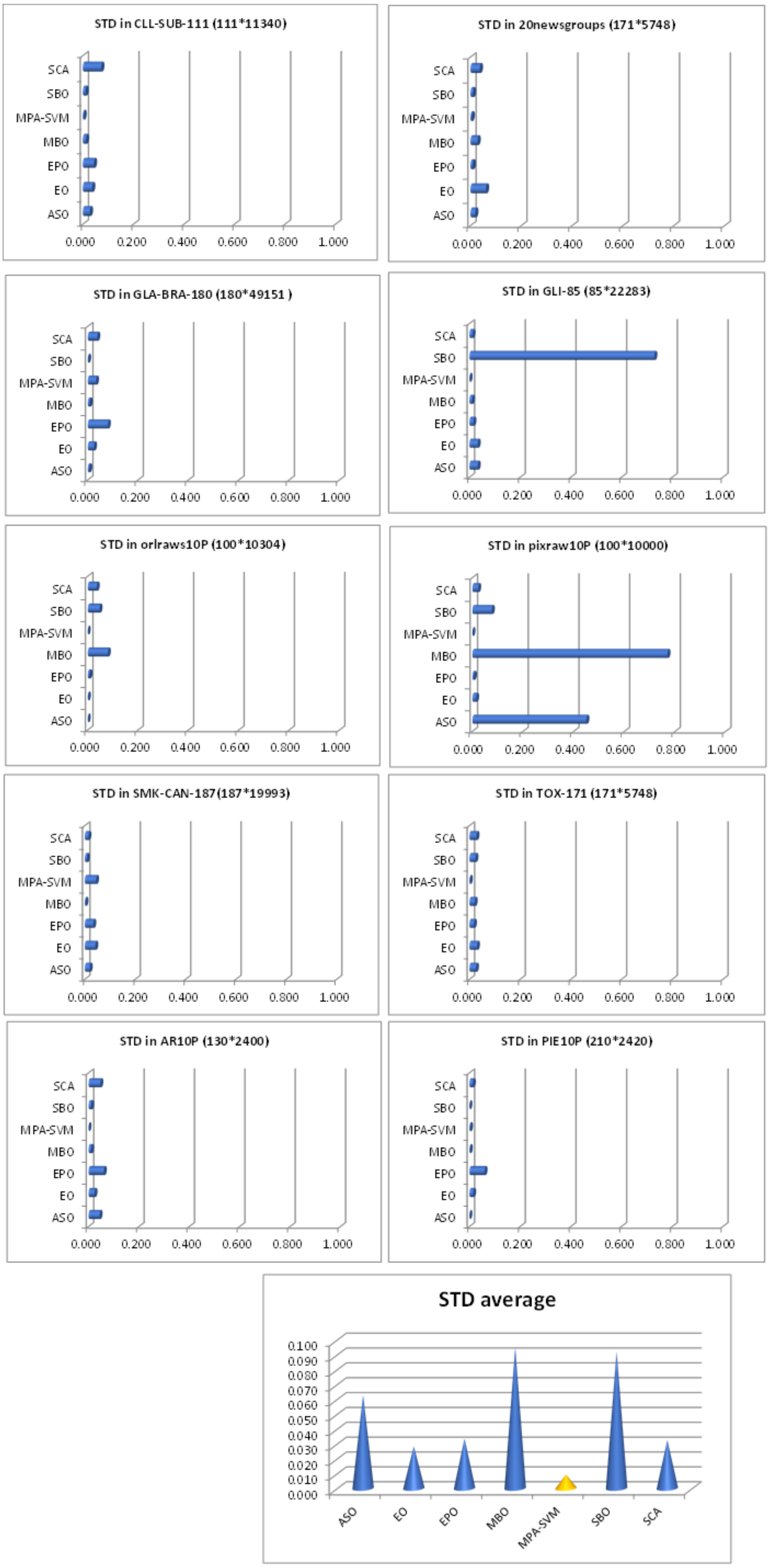

Fig. 3 Comparing the suggested methods depending on the STD

Finally, by employing the third metric, average selected features percentage, we found that MPA-SVM achieved the lowest FS percentage over 80% all 10 datasets in comparison with the other six state of art algorithms in addition to achieving 0.97% as average FS. The results of current metric are listed in TABLE 6. and visualized in **Fig. 4** . Table 6 displays the number of selected features that each approach obtained through evaluation. The MPA-SVM approach yields a minimum number of meaningful selected features for all datasets, indicating its high efficiency and suitability for the FS process especially for high dimensional datasets.

TABLE 6. Comparing the suggested methods depending on the average selected features percentage

Datasets	**ASO**	**EO**	**EPO**	**MBO**	**MPA-SVM**	**SBO**	**SCA**
CLL-SUB-111 (111*11340)	48.40%	11.00%	0.50%	43.30%	**0.29%**	48.30%	2.00%
20newsgroups (171*5748)	48.00%	14.30%	1.30%	44.20%	**0.90%**	47.90%	6.00%
GLA-BRA-180 (180*49151)	48.70%	2.70%	1.80%	43.30%	**1.90%**	48.70%	2.10%
GLI-85 (85*22283)	49.70%	5.10%	2.00%	43.40%	**1.50%**	48.30%	1.70%
orlraws10P (100*10304)	46.80%	0.20%	5.00%	39.30%	**0.10%**	46.70%	0.50%
pixraw10P (100*10000)	46.00%	**0.08%**	3.00%	39.10%	0.30%	47.20%	0.20%
SMK-CAN-187(187*19993)	49.90%	5.60%	**0.20%**	44.00%	0.80%	44.00%	2.70%
TOX-171 (171*5748)	48.20%	17.30%	1.30%	43.90%	**1.00%**	48.20%	4.40%
AR10P (130*2400)	45.70%	6.70%	2.20%	37.70%	**1.70%**	46.90%	4.80%
PIE10P (210*2420)	44.10%	3.10%	6.00%	37.20%	**1.20%**	44.70%	1.70%
Average FS percentage	47.55%	6.61%	2.33%	41.54%	**0.97%**	47.53%	2.61%

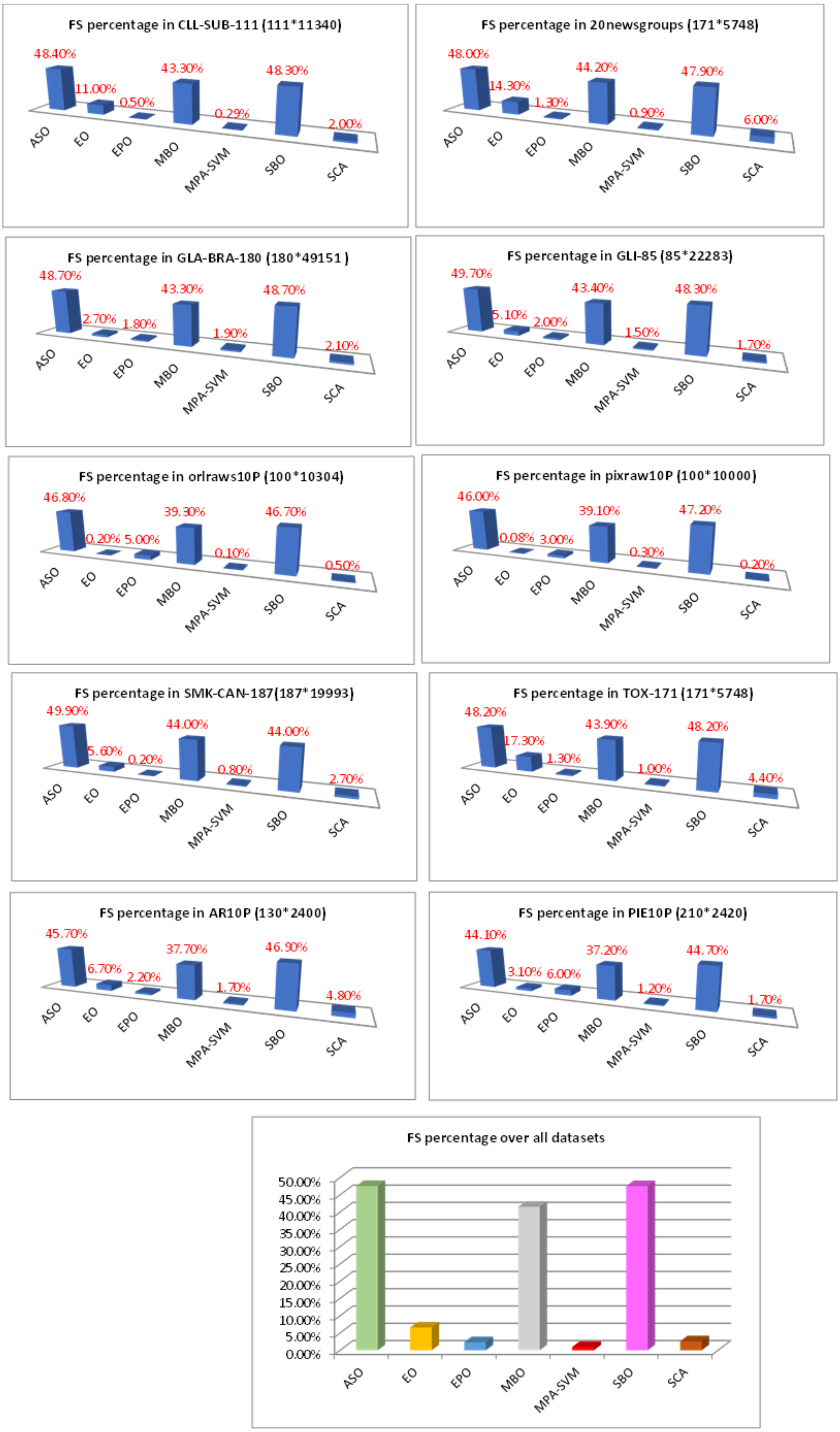

Fig. 4 Comparing the suggested methods depending on the average selected features percentage

According to the fourth criteria, the runtime for each algorithm, proposed MPA-SVM achieved the lowest runtime over the ten datasets. The MPA algorithm is superior to other algorithms. because it retains optimization results in memory, referencing that marine predator has the benefit of having a strong memory in keeping their colleagues and the location of prosperous gathering [15]. Consequently, MPA algorithm needs less iterations which is the main cause of its high speed. Table 7 lists the runtimes of MPA-SVM with other six algorithms. Obviously, MPA-SVM approved its supremacy by conducting the lowest runtimes over all datasets. Such achievement is visualized in Fig. 5.

TABLE 7. Comparing the proposed MPA-SVM based on algorithms runtimes

Datasets	ASO	EO	EPO	MBO	MPA-SVM	SBO	SCA
CLL-SUB-111 (111*11340)	29.100	24.711	13.427	89.453	6.624	91.720	91.720
20newsgroups (171*5748)	26.184	34.040	13.678	59.655	10.322	58.781	58.781
GLA-BRA-180 (180*49151)	200.833	91.683	29.388	505.659	12.607	506.153	506.153
GLI-85 (85*22283)	44.538	28.294	17.968	162.993	12.199	165.975	165.975
orlraws10P (100*10304)	24.615	18.261	14.112	80.503	4.155	80.098	80.098
pixraw10P (100*10000)	25.523	18.134	13.655	77.521	6.442	74.041	74.041
SMK-CAN-187(187*19993)	76.004	52.889	18.427	204.081	7.883	191.024	191.024
TOX-171 (171*5748)	22.982	24.351	13.631	62.247	11.373	58.511	58.511
warpAR10P (130*2400)	14.507	15.921	12.406	30.704	5.706	26.950	26.950
warpPIE10P (210*2420)	12.761	17.955	12.679	32.525	10.614	34.295	34.295
Average	47.705	32.624	15.937	130.534	**8.793**	128.755	128.755

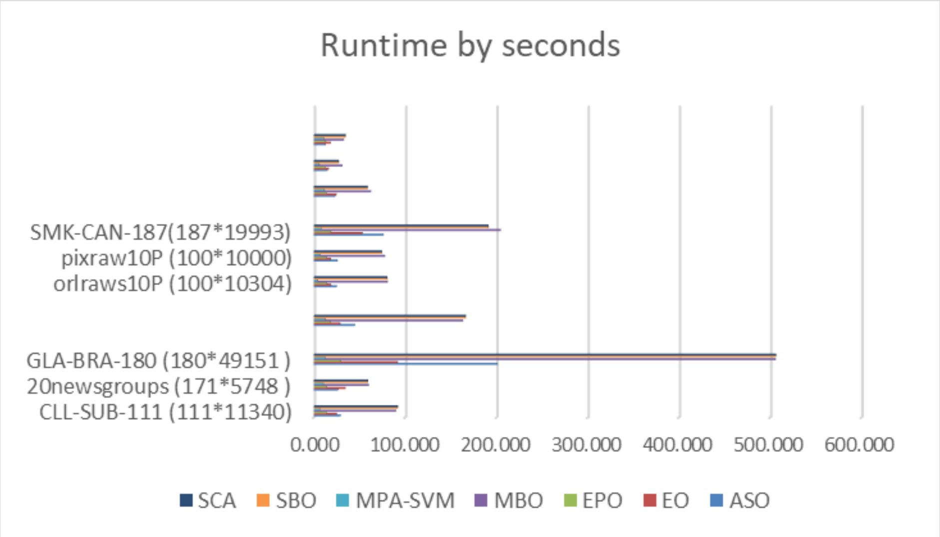

Fig. 5 Comparing the proposed MPA-SVM based on algorithms runtimes

5 Conclusions

The purpose of this work was to suggest a modified method of feature selection in high dimensional datasets depending on the Marine Predators Algorithm (MPA). Four evaluation criteria are evaluated to examine various aspects of the performance of comparison algorithms, and the experiments are done on ten high dimensional benchmark datasets from ASU datasets to investigate the performance of the suggested MPA-SVM technique. The experimental findings demonstrated that the suggested MPA-SVM technique outperformed the six well-known meta-heuristic algorithms ASO, EO, EPO, MBO, MPA, SBO from current literature in terms of results. The findings demonstrated that the MPA produced the lowest error rate with the less classification STD and minimum number of important features chosen for the majority of datasets when used with SVM as the classifiers. The MPA-SVM proved to be much more advantageous for comparatively large datasets, such achievement comes from that MPA needs less iterations. Compared to other algorithms, MPA has the advantage of memorizing optimization outcomes, which is related to the fact that marine predators have an excellent memory for remembering where successful foraging is. We get to the conclusion that the suggested MPA-SVM technique reduced the number of important features chosen while achieving excellent performance in comparison to the other tested methods. Solving other optimization problems in different disciplines is recommended for more evaluation of MPA. Since MPA is a velocity-based algorithm developing a binary and multi-objective version of MPA would be a valuable contribution.

Author's Contributions: The study's inception and design were aided by the contributions of all authors. Hadeel was primarily responsible for data collecting and material preparation. All of the authors conducted the findings analysis. Each contributor made an equal contribution to the manuscript's authoring. The final manuscript was read and approved by all authors.

Fundings: The authors did not receive support from any organization for the submitted work.

References

[1] N. Dupin and E. G. Talbi, “Machine learning-guided dual heuristics and new lower bounds for the refueling and maintenance planning problem of nuclear power plants,” *Algorithms*, vol. 13, no. 8, Aug. 2020, doi: 10.3390/A13080185.

[2] U. Moorthy and U. D. Gandhi, “A novel optimal feature selection technique for medical data classification using ANOVA based whale optimization,” *J Ambient Intell Humaniz Comput*, vol. 12, no. 3, pp. 3527–3538, Mar. 2021, doi: 10.1007/s12652-020-02592-w.

[3] M. Agyeman, A. Guerrero, Q. V.-I. Access, and undefined 2022, “A review of classification techniques for arrhythmia patterns using convolutional neural networks and Internet of Things (IoT) devices,” *ieeexplore.ieee.org*, Accessed: Oct. 13, 2022. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9832886/

[4] K. Jha and S. Saha, “Incorporation of multimodal multiobjective optimization in designing a filter based feature selection technique,” *Appl Soft Comput*, vol. 98, Jan. 2021, doi: 10.1016/j.asoc.2020.106823.

[5] R. Kohavi and G. H. John, “Wrappers for feature subset selection,” *Artif Intell*, vol. 97, no. 1–2, pp. 273–324, 1997, doi: 10.1016/S0004-3702(97)00043-X.

[6] E. Emary and H. M. Zawbaa, “Feature selection via Lèvy Antlion optimization,” *Pattern Analysis and Applications*, vol. 22, no. 3, pp. 857–876, Aug. 2019, doi: 10.1007/s10044-018-0695-2.

[7] R. Storn and K. Price, “Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces,” *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997, doi: 10.1023/A:1008202821328.

[8] J. Kennedy and R. Eberhart, “Particle swarm optimization,” *Neural Networks, 1995. Proceedings., IEEE International Conference on*, vol. 4, pp. 1942–1948 vol.4, 1995, doi: 10.1109/ICNN.1995.488968.

[9] G. Winter, J. Periaux, M. Galan, and P. Cuesta, “Genetic Algorithms in Engineering and Computer Science,” 1996, [Online]. Available: http://dl.acm.org/citation.cfm?id=547504

[10] M. Dorigo and T. Stützle, “Ant colony optimization: overview and recent advances,” *Techreport, IRIDIA, Universite Libre de Bruxelles*, no. May, 2009.

[11] H. T. Ibrahim, W. J. Mazher, O. N. Ucan, and O. Bayat, “A grasshopper optimizer approach for feature selection and optimizing SVM parameters utilizing real biomedical data sets,” *Neural Comput Appl*, 2018, doi: 10.1007/s00521-018-3414-4.

[12] M. A. Tawhid and K. B. Dsouza, “Hybrid binary bat enhanced particle swarm optimization algorithm for solving feature selection problems,” *Applied Computing and Informatics*, vol. 16, no. 1–2, pp. 117–136, Apr. 2018, doi: 10.1016/j.aci.2018.04.001.

[13] “Feature Selection using Salp Swarm Algorithm for Real Biomedical Datasets,” *IJCSNS International Journal of Computer Science and Network Security*, Dec. 2017.

[14] A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi, “Marine Predators Algorithm: A nature-inspired metaheuristic,” *Expert Syst Appl*, vol. 152, Aug. 2020, doi: 10.1016/j.eswa.2020.113377.

[15] D. S. A. Elminaam, A. Nabil, S. A. Ibraheem, and E. H. Houssein, “An Efficient Marine Predators Algorithm for Feature Selection,” *IEEE Access*, vol. 9, pp. 60136–60153, 2021, doi: 10.1109/ACCESS.2021.3073261.

[16] M. A. M. Shaheen, D. Yousri, A. Fathy, H. M. Hasanien, A. Alkuhayli, and S. M. Muyeen, "A Novel Application of Improved Marine Predators Algorithm and Particle Swarm Optimization for Solving the ORPD Problem," *Energies (Basel)*, vol. 13, no. 21, Oct. 2020, doi: 10.3390/en13215679.

[17] M. A. Soliman, H. M. Hasanien, and A. Alkuhayli, "Marine Predators Algorithm for Parameters Identification of Triple-Diode Photovoltaic Models," *IEEE Access*, vol. 8, pp. 155832–155842, 2020, doi: 10.1109/ACCESS.2020.3019244.

[18] M. A. A. Al-Qaness, A. A. Ewees, H. Fan, L. Abualigah, and M. A. Elaziz, "Marine predators algorithm for forecasting confirmed cases of COVID-19 in Italy, USA, Iran and Korea," *Int J Environ Res Public Health*, vol. 17, no. 10, May 2020, doi: 10.3390/ijerph17103520.

[19] Z. Beheshti, "BMPA-TVSinV: A Binary Marine Predators Algorithm using time-varying sinus and V-shaped transfer functions for wrapper-based feature selection," *Knowl Based Syst*, vol. 252, Sep. 2022, doi: 10.1016/j.knosys.2022.109446.

[20] W. Zhao, L. Wang, and Z. Zhang, "Atom search optimization and its application to solve a hydrogeologic parameter estimation problem," *Knowl Based Syst*, vol. 163, pp. 283–304, Jan. 2019, doi: 10.1016/j.knosys.2018.08.030.

[21] A. Faramarzi, M. Heidarinejad, B. Stephens, and S. Mirjalili, "Equilibrium optimizer: A novel optimization algorithm ☆," vol. 191, p. 105190, 2020, doi: 10.1016/j.knosys.

[22] G. Dhiman and V. Kumar, "Emperor penguin optimizer: A bio-inspired algorithm for engineering problems," *Knowl Based Syst*, vol. 159, pp. 20–50, Nov. 2018, doi: 10.1016/j.knosys.2018.06.001.

[23] G. G. Wang, S. Deb, and Z. Cui, "Monarch butterfly optimization," *Neural Comput Appl*, vol. 31, no. 7, pp. 1995–2014, Jul. 2019, doi: 10.1007/s00521-015-1923-y.

[24] S. H. Samareh Moosavi and V. Khatibi Bardsiri, "Satin bowerbird optimizer: A new optimization algorithm to optimize ANFIS for software development effort estimation," *Eng Appl Artif Intell*, vol. 60, pp. 1–15, Apr. 2017, doi: 10.1016/j.engappai.2017.01.006.

[25] S. Mirjalili, "SCA: A Sine Cosine Algorithm for solving optimization problems," *Knowl Based Syst*, vol. 96, pp. 120–133, Mar. 2016, doi: 10.1016/j.knosys.2015.12.022.

[26] A. Mammone, M. Turchi, and N. Cristianini, "Support vector machines," *Engineering*, no. December, pp. 1–39, 2009, doi: 10.1002/wics.049.

[27] M. A. Al-Betar, M. A. Awadallah, A. A. Heidari, H. Chen, H. Al-khraisat, and C. Li, "Survival exploration strategies for Harris Hawks Optimizer," *Expert Syst Appl*, vol. 168, no. December 2019, p. 114243, 2021, doi: 10.1016/j.eswa.2020.114243.

[28] M. A. Tawhid and K. B. Dsouza, "Hybrid binary bat enhanced particle swarm optimization algorithm for solving feature selection problems," *Applied Computing and Informatics*, vol. 16, no. 1–2, pp. 117–136, Apr. 2018, doi: 10.1016/j.aci.2018.04.001.

[29] A. Einstein and A. D. Cowper, "INVESTIGATIONS O N THE THEORY .OF ,THE BROWNIAN MOVEMENT R. F ü R T H TRANSLATED BY."

[30] R. N. Mantegna, "Fast, accurate algorithm for numerical simulation of Levy stable stochastic processes," 1994.

[31] R. N. Mantegna, “Fast, accurate algorithm for numerical simulation of Lévy stable stochastic processes,” *Phys Rev E*, vol. 49, no. 5, pp. 4677–4683, May 1994, doi: 10.1103/PhysRevE.49.4677.

[32] R. P. Parouha and K. N. Das, “A memory based differential evolution algorithm for unconstrained optimization,” *Appl Soft Comput*, vol. 38, pp. 501–517, Jan. 2016, doi: 10.1016/J.ASOC.2015.10.022.

[33] V. V, *The nature of statistical learning theory*. Springer, New York, 1995.

[34] A. F. Izmailov, “Solution sensitivity for Karush-Kuhn-Tucker systems with non-unique Lagrange multipliers,” *Optimization*, vol. 59, no. 5, pp. 747–775, 2010, doi: 10.1080/02331930802434922.

[35] S. Gopal, K. Patro, and K. Kumar Sahu, “Normalization: A Preprocessing Stage.” [Online]. Available: www.kiplinger.com,

[36] M. M. Mafarja and S. Mirjalili, “Hybrid Whale Optimization Algorithm with simulated annealing for feature selection,” *Neurocomputing*, vol. 260, pp. 302–312, Oct. 2017, doi: 10.1016/j.neucom.2017.04.053.

[37] “Datasets | Feature Selection @ ASU.” Accessed: Aug. 01, 2024. [Online]. Available: https://jundongl.github.io/scikit-feature/OLD/datasets_old.html

[38] S. Mukherjee *et al.*, “Feature selection for SVMs,” 2000. [Online]. Available: https://www.researchgate.net/publication/221619995

[39] Z. Xu, G. Huang, K. Q. Weinberger, and A. X. Zheng, “Gradient boosted feature selection,” in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Association for Computing Machinery, 2014, pp. 522–531. doi: 10.1145/2623330.2623635.

[40] L. Jia, W. Gong, and H. Wu, “An Improved Self-adaptive Control Parameter of Differential Evolution for Global Optimization.”