***Article Received:** 05 July 2023 **Revised:** 25 August 2023 **Accepted:** 15 September 2023*

Hybrid Spam Filtering using Monarch Butterfly Optimization Algorithm with Self-Adaptive Population

Deepika Mallampati [1, *], Dr. Nagaratna P. Hegde 2
[1]Research Scholar, Department of CSE,Osmania University
Assistant Professor, Department of CSE, Neil Gogte Institute of Technology
Hyderabad, Telangana, India.
e-mail: mokshhyd@gmail.com
[2]Professor, Department of CSE, Vasavi College of Engineering
Hyderabad, Telangana, India
e-mail: nagaratnaph@staff.vce.ac.in

Abstract: Spam causes bottlenecks and congestion, reducing the speed, processing power, available memory, and bandwidth. Existing spam email classification methods need to be more accurate because of the large dimensionality of hybrid spam datasets. This makes the need for a feature dimensionality reduction technique that uses only associated features of the problem instead of all features in the dataset. This paper presents a feature selection based on the monarch butterfly optimization (MBO) algorithm that emphasizes less complexity and few features. This method is efficient and produces a more accurate classification. To improve further standard MBO algorithm performance, we introduce the population size in both subpopulations 1 and 2 will experience dynamic variations as the algorithm proceeds along its linear way. As the idea of a self-adaptive and greedy strategy is modified, the self-adaptive population monarch butterfly optimization (SPMBO) method is introduced, and only newly generated SPMBO individuals are eligible for the next generations if they are better individuals earlier before. Later, this paper proposes an email classification system based on k-nearest neighbors (k-NN) based on two distance metrics, explicitly Euclidean, and Manhattan, that also uses the SPMBO technique. This method seeks to determine whether a hybrid email is a spam. The efficiency of the proposed SPMBO algorithm was compared with standard MBO based on three datasets Dredze, Image spam hunter, and Spambase. Thus, the use of SPMBO results has shown superior as related to other authors' works in relevant fields.

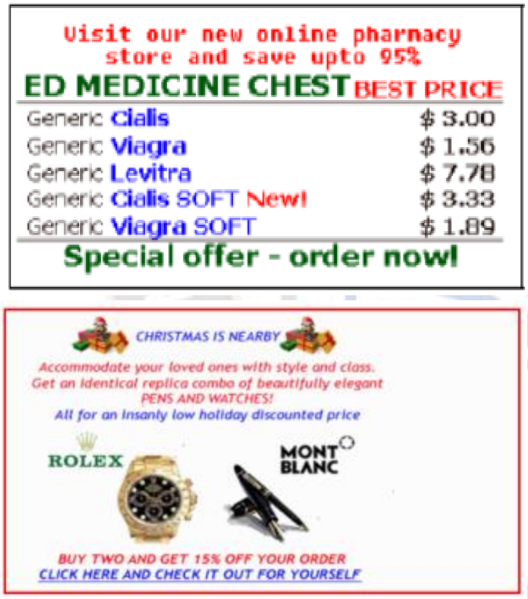
Keywords- E-mail, Monarch butterfly optimization, Migration operator, Spam Filtering, Classification.

I. INTRODUCTION

An e-mail has maintained its dominance in the global distribution of information. e-mail is popular because it is quick, cheap, and simple to use on personal computers, smartphones, and other modern electronic devices [1, 2]. Even though other forms of online communication, such as instant messaging and social networking, are becoming more popular, e-mails remain the most common way for business people to communicate. E-mails are still required when using other forms of online communication and conducting business. An e-mail has made it easier for groups to communicate with one another, as evidenced by the growth of companies worldwide [3]. Spammers' spam e-mails typically promote unwanted or irrelevant products that tend to hinder inboxes. This can make storing other essential and relevant e-mails difficult. E-mails about commercial offers, lotteries, bank offers, and other similar issues cheat trusting people before attempting to steal money from them. Because spam and non-spam e-mails can say different things to different people, spammers and scammers are doing terrifying things. Furthermore, these e-mails comprise harmful data that damages the systems permanently. Even though most people can recognize spam e-mails and understand how dangerous they can be, many people who take spam e-mails are ignorant of this and reply to them, allowing the sender to profit. In addition, learning how to detect spam e-mails has become increasingly important in a world where things change quickly, and people rely on technology. Some new methods [4–7] of detecting and blocking image spam have emerged in recent years. However, in recent years, spammers have discovered a new way to communicate their messages by creating multimedia spam. Text-based spam filters cannot detect this type because the text message is embedded in the image. Figure 1 depicts the various types of spam images.

Even though machine learning (ML) is used as the foundation for the vast majority of newly developed spam detection systems [8, 9], one of the most common issues is determining how to select the appropriate input feature subsets for the various classifiers. This is frequently accomplished through the use of FS processes, which are typically made more difficult by the problem of high data dimensionality inherent in FS processes. This issue hampers the performance of specific classifiers like

SVM, ANN, and NBC [10, 11]. If all of the potential subsets of the dataset are retrieved during the creation phase, the level of complexity and, as a result, the amount of time required for computation will be extremely high. As a result, researchers have attempted to develop strategies capable of resolving feature dimensionality constraints and delivering the best remedies to conventional methods. The use of metaheuristic algorithms [12–14] is an example of a procedure identified as one of the approaches. Metaheuristic algorithms are intelligent search algorithms that mimic natural processes [15]. Following the execution of the migration operator, the monarch butterfly will be recognized as a new that follows, regardless of the quality of the butterfly it produces. This will happen in the next generation, and there will be no change in the number or distribution of butterflies on either land one or land two throughout the optimization process. In the beginning, the value of the parameter p was used to determine this quantity so that it could be accurate. As possible solutions to the problems presented, self-adaptive and greedy upgrades to the fundamental process are proposed in this study.



Visit our new online pharnacy
store and save upto 95%

ED MEDICINE CHEST BEST PRICE

Generic Cialis	$ 3.00
Generic Viagra	$ 1.56
Generic Levitra	$ 7.78
Generic Cialis SOFT New!	$ 3.33
Generic Viagra SOFT	$ 1.89

Special offer - order now!

CHRISTMAS IS NEARBY

Accommodate your loved ones with style and class.
Get an identical replica combo of beautifully elegant
PENS AND WATCHES!
All for an insanly low holiday discounted price

BUY TWO AND GET 15% OFF YOUR ORDER
CLICK HERE AND CHECK IT OUT FOR YOURSELF

Figure 1. Spam image examples. (a) an image containing text embedded within it; (b) an image containing text in addition to an image

During the optimization process, a self-adaptive mechanism modifies the number of butterflies in both land 1 and land 2. Only butterflies given a genetic boost by a migration operator have a chance of developing into new butterflies in the generation following them. This best strategy almost always increases the number of butterfly populations. This ensures that the newly generated population equals the one that came before it. The self-adaptive population MBO algorithm, also known as SPMBO, incorporates the previously mentioned improvements. Furthermore, SPMBO is assessed using different functions by widths ranging from 30 to 60. In the vast majority of cases, the SPMBO outdoes the MBO process. Furthermore, when applied to high-dimensional global optimization, the implementation of the self-adaptive strategy improves the performance of the fundamental MBO algorithm. This section investigates the k-NN classification technique and its combination with the proposed BIC classification technique. In addition, a thorough explanation of the various methods is provided.

II. RELATED WORK

In this section, we will look at several significant studies that used intelligent algorithms to detect how to find and classify e-mail spam. The enhanced Firefly Optimization Algorithm (EFOA) employs the fitness function to select acceptable features from an upper-dimensional space effectively. The authors presented their proposal for this technique in [16]. Once the EFOA has determined which feature space is the most successful, artificial neural networks are used to classify spam. Following the preprocessing of the e-mail spam dataset, the recovered textual features will be semantically reduced, and the feature weights will be adjusted using an optimized semantic WordNet. After applying EFOA to specific features, the results revealed that the ANN classifier could correctly classify e-mails as spam or not. According to the findings of this EFOA study, the proposed strategy significantly improved the method used for the SCA.

The authors of the paper [17] presented a particle swarm optimization (PSO) method that takes advantage of a logistically chaotic map to achieve better results. As a result, the dimensionality of the features is reduced, and the accuracy of spam e-mail classification is improved. A sigmoid function converts the features into the binary form so that each particle is assigned a feature. Then these features are fed to the SVM for classification. They used the spambase dataset for evaluation purposes using the Chaotic Binary PSO algorithm. Also, the classifier's efficiency and the feature vector's dimension that served as an input to the classifier are considered. This ensures that the assessment is as precise as possible. The trials' findings revealed that, despite having a limited set of characteristics to work with, the BPSO could produce good feature selection results.

The authors devised a spam-prevention strategy divided into two stages: selecting various features and classifying different types of e-mails [18] goes into greater detail about this approach. The first step of the process involves selecting wrapper features using Particle Swarm Optimization (PSO). This step reduces many measured characteristics by choosing the most accurate representative features. The features selected in the first step of the process are used in the second step to build a random forest spam filtering model. The experimental results showed the goodness of the model related to other works. Furthermore, the effectiveness of the spam filtering approach is evaluated using four distinct cost functions.

In light of the findings, the combination of PSO+RF is a valuable method for spam detection.

The author's proposed approach, presented in [19], used the Information Gain (IG) filter in conjunction with the Complement Naive Bayes (CNB) wrapper as a feature selector. This system included two distinct filtering models, dubbed "Filter" and "Wrapper," respectively. In this, they used four ML, classifier models. Also, they used the feature dimensionality technique for better performance. Following that, this layout is contrasted with other works in a similar vein using various parameters. The proposed method's accuracy was measured at 99%, which is the highest possible score and is considered optimal.

The authors of the paper [20] reduced the error rate of spam recognition by employing a technique known as the sine-cosine algorithm (SCA), which is a feature selection approach. According to the proposed plan, the SCA will be in charge of updating the feature vectors to select the features that will be most useful when ANN is trained. The results are shown on the spambase dataset, and Matlab is used for programming. Thus, the results obtained had a precision of 98.64%, an accuracy of 97.92%, and a sensitivity of 98.36%. In other words, when it comes to spam detection, the SCA is superior to MLP, DT, and RF models. Using the SCA during the testing process resulted in a 2.18 percent reduction in the amount of feature selection error produced by the MLP neural network.

Texture feature sets such as GLCM and RLM were used in [21] to present the author's proposed method for distinguishing between ham and spam in images. This method made use of numerous textural characteristics. Both SVM and KNN are used at various stages of the classification process. The Dredze dataset is used in the proposed approach's implementation. When the value of K is set to 20, the average accuracy is approximately 97.27%. The authors of the paper [22] devised a novel filtering strategy to remove spam photos from the content. This novel filtering method included ten elements similar to current results that distinguish spam images from normal images. These features are useful in classifying spam images. They used a strategy known as PCA for feature dimensionality reduction purposes. However, the SVM classifier's classification of spam-related communications. It was determined that the classification was correct 98% of the time.

III. MATERIALS AND METHODS

For text and image email spam classification, the proposed model employs the k-NN technique classifier and Monarch Butterfly Optimization, a bioinspired metaheuristic optimization method for feature selection purposes. Figure 2 depicts the text and image spam filtering model, which are available in the section below. Table 1 provides complete details for text and image.

TABLE I. Text and image dataset details

Authors	Dataset	Description
[23]	Dredze	Nonspam images=2550, spam images = 3239, spam archive images = 9503
[24]	Image Spam Hunter (ISH)	Nonspam images=810, spam images = 920
[25]	Spambase	Dataset size = 4601, spam=1813, ham= 2788

A. Feature Extraction

The spam email image was divided into two groups of features. These include traits derived from text sections as well as those derived from images. After extracting the image properties, the discovered characteristics must be quantified. This is commonly referred to as "data preparation." Feature vectors are a standard input format for machine learning algorithms, representing each feature numerically. As a result, image properties such as canny edges, histograms, and so on must be converted into numerical values. The number of edges in the canny edge image is extracted, as are histogram entropy, mean, variance, kurtosis, and other statistical techniques.

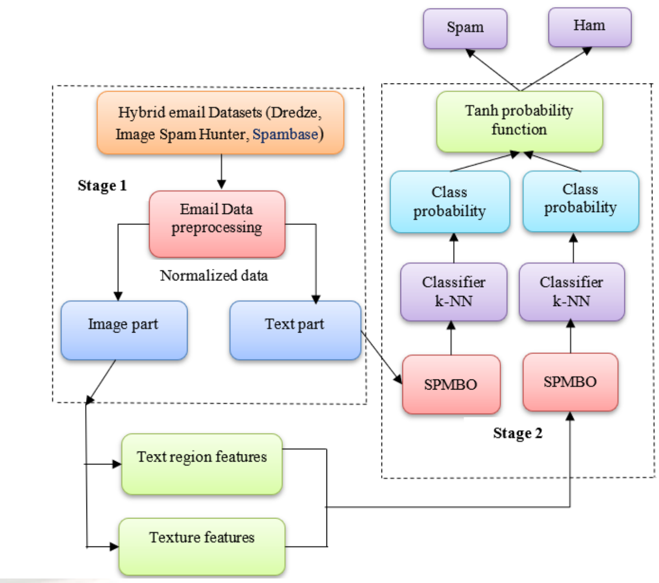


Figure 2: Proposed model for filtering hybrid email

Article Received: *05 July 2023* ***Revised:*** *25 August 2023* ***Accepted:*** *15 September 2023*

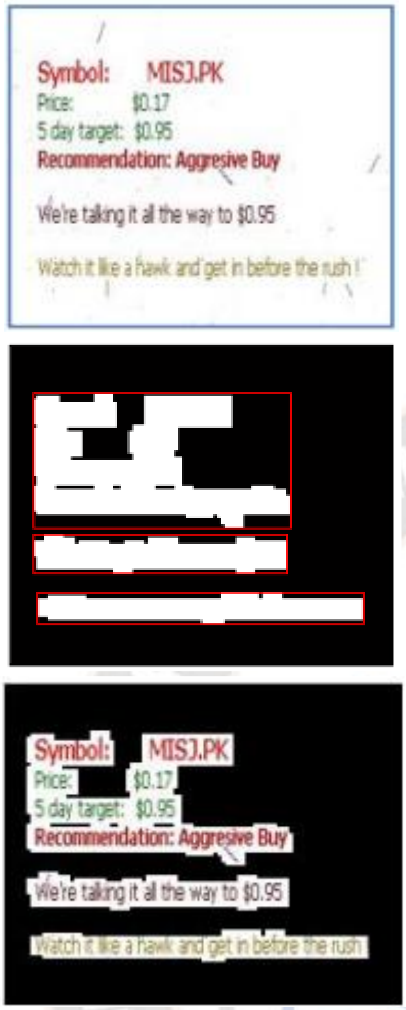

Symbol: MISJ.PK
Price: $0.17
5 day target: $0.95
Recommendation: Aggresive Buy
We're talking it all the way to $0.95
Watch it like a hawk and get in before the rush !

Figure 3. Detection of noisy text sections (a) spam example (b) labelled text regions (c) text region extraction

Following the creation of the feature vectors, a series of tests can be run to determine which traits should be incorporated into the final model to achieve higher accuracy. The initial spam image, shown in Figure 3a, contains noisy data with no text data. Figure 3c depicts the presence of noise pixels, perfect localization of text regions was achieved, whereas Figure 3b shows the correctly recognized text parts.

Image Feature Extraction Using the Local Binary Pattern

LBP is a technique for describing an image's texture based on variations between pixels close to and far from the image's center. This chunk of binary text represents a binary pattern. The value of the central pixel is set to threshold to generate a binary code, which is then applied to the value of the corresponding pixel in the image. As a result, if the value of the pixel next to it is larger than or equal to the threshold value, it will be set to 1; otherwise, it will be set to 0 (zero). A graph is used for binary value generation. On the other hand, (1) below provides the equation for LBP in its simplest form.

$$LBP(i_c, j_c) = \sum_{n=0}^{7} 2^n f\big(N_n - M(i_c, j_c)\big) \qquad (1)$$

$$f(i) = \begin{cases} 0 & i < 0 \\ 1 \; i & \geq \; 1 \end{cases} \qquad (2)$$

Where the values of LBP (ic, jc): LBP value at the centre pixel (ic, jc) $N(n)$: values of neighbor pixel values M (ic, jc): centre pixel n: index of neighboring pixels.

Feature Selection

Because we have so many features, irrelevant features have to reduce from the total number of features. Also, for image processing purposes, features have to be extracted. Thus, feature extraction has become a task that takes a lot of processing power.

Monarch butterfly optimization algorithm (MBOA):

A population-based algorithm-based algorithm is one of several types. These algorithms indicate behaviour species that collect in large groups, such as bees, butterflies, and other similar organisms. The MBO algorithm falls into the swarm class.

Migration Operator

To know more about migration operators' total population of butterflies at each location is calculated using the formulas on lands 1 and 2, respectively. We will use the abbreviations SP1 and SP2 to refer to subpopulations 1 and 2, respectively. In this case, the ceil(x) function rounds x down to the nearest integer greater than x. As a result, when r is less than p, then obtained and can be written in mathematical form as [26]:

$$x_{i,k}^{t+1} = x_{r1,k}^{t} \qquad (3)$$

In equation (3) $x_{i,k}^{t+1}$ and $x_{r1,k}^{t}$ denotes kth element of x_i, x_{r1} respectively. Randomly selected Butterfly $r1$ from SP1 can be written as:

$$r = rand \ * peri \qquad (4)$$

where peri is the migration period. In comparison, when $r > p$, then $x_{r1,k}^{t}$ is given by

$$x_{i,k}^{t+1} = x_{r2,k}^{t} \qquad (5)$$

where $x_{r2,k}^{t}$ denotes kth element of x_{r2}, and randomly selected Butterfly $r2$ from SP2.

As rand < p, the kth element for butterfly j can be written as

$$x_{j,k}^{t+1} = x_{best,k}^{t} \qquad (6)$$

where $x_{j,k}^{t+1}$ signifies kth element of x_j. Also, $x_{best,k}^{t}$ signifies *k*th element of the best individual x_{best}. Subsequently, when rand > *p*, which can put in mathematically form as:

$$x_{j,k}^{t+1} = x_{r3,k}^{t} \qquad (7)$$

where $x_{r3,k}^{t}$ signifies *k*th element of x_{r3}. However, *r*3 ϵ {1, 2, . . . , *NP*₂*}*.

In this case, when *rand* is bigger than *BAR*, it can be calculated in another form [88]:

$$x_{j,k}^{t+1} = x_{j,k}^{t} + \alpha \times (dx_k - 0.5) \qquad (8)$$

SPMBO Algorithm

However, as we have previously stated, the MBO employs a fixed number of butterflies on Land 1 and Land 2, and the migration operator accepts each new butterfly individual that it creates. In this article, we will propose a new MBO algorithm that incorporates both self-adaptive and greedy techniques.

Self-Adaptive Strategy

In MBOA, the butterfly's population in both lands 1 and 2 depends on ceil function of their subpopulations (i.e., ceil (*p**NP*) (*NP*1, subpopulation 1) makes butterflies remain static throughout the optimization method, but there is dynamic migration of butterflies as per the value of factor *p* which receives its updates as follows:

$$p = a + bt \qquad (9)$$

where *t* is current generation, and *a* and *b* are constants

$$a = \frac{p_{min} t_m - p_{max}}{t_m - 1}, \qquad (10)$$

$$b = \frac{p_{max} - p_{min}}{t_m - 1}, \qquad (11)$$

where t_m denotes maximum generation, p_{min} denotes lower bound, and p_{max} denotes the upper bound of factor *p*. Notably, p_{min} and p_{max} belong to the [0, 1] class. The butterfly adjustment operator updates all butterflies in the standard MBO algorithm when *p*=0, whereas the migration operator updates all butterflies when *p*=1. Despite these two exceptions, to broaden the range of the factor *p*, we assign pmin and pmax to 0.1 and 0.9 in the subsequent trials. We can see from Equation (9) that the factor *p* changes linearly from the lower limit p_{min} to the higher bound p_{max}.

Greedy Strategy

In this process, all newly formed butterfly individuals to the pool of butterfly individuals are to be utilized in the next generation. This pool of butterfly individuals is used to create the next generations of butterflies. Suppose the newly-generated individual of the butterfly is of poorer quality than the one that came before it. In that case, the population will suffer due to this update, and the convergence rate will be slowed down. In general, if something occurs in the later stages of the search, it will cause a shift in the population. In the context of this work, the fundamental MBO algorithm was enhanced by adding a greedy strategy. Only newly-created butterflies that have significantly increased their overall degree of fitness are allowed for the subsequent generation. This selection method promises that the newly generated population will not be inferior to the one that came before it and that the algorithm will develop in the desired manner. After the performance of the greedy technique, the new butterfly is fragmented down into its parts as follows to reduce the risk of any unexpected problems:

The effectiveness of the SPMBO algorithm is tested by applying the proposed approach to discover solutions to three benchmark datasets. This is done to evaluate the performance of the proposed feature selection method (i.e., SPMBO). Because each of the three benchmark datasets only has a few attainable functions. This is because each problem only has a few functions.

$$x_{i.new}^{t+1} = \begin{cases} x_i^{t+1}, f(x_i^{t+1}) < f(x_i^t) \\ x_i^t, \quad else \end{cases} \qquad (12)$$

where $x_{i.new}^{t+1}$ denotes newly-generated butterfly, thus moves to the next generation, $f(x_i^{t+1})$ and $f(x_i^t)$ denotes fitness of butterfly x_i^{t+1} and x_i^t, respectively. Subsequently applying greedy strategy to the migration operator, the updated process is discussed in Algorithm 1.

Algorithm 1: Migration operator updating.

```
for $i$ = 1 to NP1
  for $k$ = 1 to D do
    Generation of rand.
    $r$ = rand $*$ peri.
    if $r \leqslant p$ then
      Choosing $r_1$ randomly from SP1;
      Produce $x_{i.k}^{j+1}$ using eq. (3);
    else
      Choosing $r_2$ randomly from SP2;
      Produce $x_{i.k}^{t+1}$ using eq. (5);
    end if
  end for
  Produce $x_{i,new}^{t+1}$, using eq. (12).
end for
```

In the similar manner SPMBO process is discussed in Algorithm 2 along with its complete pseudo code.

Algorithm 2: SPMBO**.**

```
Initialization. Fix generation $t$ = 1, and $t_m$, {NP$_1$, NP$_2$, BAR, $peri$, $p_{min}$, $p_{max}$}.
Compute required objective fitness function.
    while $t < t_m$ do
    Categorize all the individuals of butterfly population.
    Using eq. (9) calculate factor
    Fix NP$_1$ and NP$_2$.
    Separate SP1 and SP2.
        for $i$ = 1 to NP$_1$ do
            From Algorithm 1 execute updated migration operator.
        end for
        for $j$ = 1 to NP$_2$
            Execution of standard MBO process.
        end for
    Obtain the newly-generated butterfly fitness function.
    $t = t + 1$.
    end while
Store the updated result.
```

Classification:

***k*-nearest neighbor (k-NN)**

The *k*-nearest neighbor algorithm was used in this study to classify data into image-based spam e-mails. The data classification was accomplished using a feature set and trained data. This data has m rows and n columns that have been divided into two subsets: D_{train} (*p* rows and *n* columns) and X_{est} (m *n) (q*n). The data for the supervised training is initially divided into two classes: spam and valid *e*-mails. The classification values saved in the *Y* train's class column are used to determine which classes these items belong to. The flow process and optimization algorithm (OA) of *k*-Nearest Neighbors is discussed in Algorithm 3.

Algorithm 3: *k*-Nearest Neighbors'

```
Input: $X_{train}$, $X_{test}$, $Y_{train}$ and $Y_{test}$
Output: $\zeta$
$r \leftarrow \{a_1, a_2, \ldots, a_p\} \in D_{train}$
$s \leftarrow \{b_1, b_2, \ldots, b_q\} \in D_{test}$
$v \leftarrow \{c_1, c_2, \ldots, c_p\} \in B_{train}$
$w \leftarrow \{d_1, d_2, \ldots, d_q\} \in B_{test}$
$Initialize : \phi_i \leftarrow X^{\leftrightarrow}rand \mid i \in \{1, 2, 3, \ldots, 12\}$
$\zeta i \leftarrow 0 \mid i \in \{1, 2, \ldots, q\}$
for $i$ = 1 to $p$ do
    if $v_i = 1$ then
        $D_{spami} \leftarrow r_i$
    else
        $D_{legiti} \leftarrow r_i$
    end if
end for
$x_{spam} \leftarrow mean(D_{spam})$
$x_{legit} \leftarrow mean(D_{legit})$
$\{\phi_1, \phi_2, \phi_3\} \leftarrow \mathrm{OA}(D_{spam}, x_{spam}, E(a, b))$
$\{\phi_4, \phi_5, \phi_6\} \leftarrow \mathrm{OA}(D_{spam}, x_{spam}, M(a, b))$
$\{\phi_7, \phi_8, \phi_9\} \leftarrow \mathrm{OA}(D_{legit}, x_{legit}, E(a, b))$
$\{\phi_{10}, \phi_{11}, \phi_{12}\} \leftarrow \mathrm{OA}(D_{legit}, x_{legit}, M(a, b))$
$\zeta \leftarrow kNN(\phi, s, k)$
```

This approach uses two different distance metrics, which will later be substituted in the function Dist, to compute many class representatives for each of the two classes (a, b). During this research, the Euclidean distance was initially regarded as the best option for use as a distance metric. According to equation (13), the Euclidean distance is calculated among *a* and *b* of two points that can be put in mathematical form.

$$E(a,b) = \sqrt{\sum_{i=1}^{n}(a_i - b_i)^2} \qquad (13)$$

The Manhattan distance is the second distance metric used, based on the idea that the distance between *a* and *b* two points is the absolute difference between the coordinates. This metric, used to calculate distances, replaces the traditional metric used in Euclidean geometry with a new metric.

$$M(a,b) = \sum_{i=1}^{n}|a_i - b_i| \qquad (14)$$

The dataset used to determine class representatives is considered new training data and calculates how far a testing data point is from the class representatives. The data set used to determine the class representatives is regarded as new training data to calculate how far a testing data point is from the class representatives. This approach can be used to compute the distance between two points which is discussed in Algorithm 4. To do so, calculate the distance between each point in the training set and each point in the testing set.

Algorithm 4: k-NN

```
Input: $\phi$, $s$ and $k$
Output: $\zeta$
for $i$ = 1 to $q$ do
    for $j$ = 1 to 18 do
        $\psi_{neighbors} \leftarrow$ Compute $Dist(\phi j, si)$
    end for
    $X \leftarrow sort\ ascending(\psi_{neighbors})$
    if majority class among first $k$ distances in $X$ then
        $\zeta i \leftarrow 1$
    else
        $\zeta i \leftarrow 0$
```

```
end if
end for
```

A given data point is considered to belong to a class that is more prevalent among its neighbors if it is close to its neighbors as "*k*." This *k* value is used as input for k-NN. Then the obtained class is saved in a variable I for each instance of *the Dtest*. When combined with k-NN, these three approaches produce positive results due to the ease with which they can be implemented and the plethora of essential qualities of k-NN. *K-Nearest* Neighbors is effective, requires little to no training time, and does not require any prior knowledge of the data set. This is because no previous knowledge of the data set is needed. Furthermore, the procedure is straightforward to follow.

Evaluation Metrics

The proposed method performance was determined using several evaluation metrics, including accuracy, recall, precision, and the F1-score. The evaluation will be based on the following indicators:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \qquad (15)$$

$$Recall = \frac{TP}{TP + FN} \qquad (16)$$

$$Precision = \frac{TP}{TP + FP} \qquad (17)$$

$$Precision = \frac{2 \times (Precision + Recall)}{Precision + Recall} \qquad (18)$$

IV. RESULTS AND DISCUSSION

This paper's evaluation was performed on a personal computer with the following settings: The memory available is 8GB, and the Intel core i7 processor runs at 2.6GHz. The technique was implemented in the MATLAB programming language, also used to write it. Matlab R2020a was used to help with the feature selection process. The program was run with the proper settings after installing the upgraded version of Matlab that included the MBO toolbox. The WEKA was employed during the classification phase because it was relatively simple compared to other machine learning software. Using 10-fold cross-validation, the k-nearest neighbor classifier was used on the entire datasets before and after using feature selection in the analysis. Following that, the effectiveness of the suggested strategies was assessed using the newly generated confusion matrix.

Table 2 and Figure 4 show that SPMBO is demonstrably superior in terms of the number of features it can remove compared to traditional works. It outperforms the MBO on every single dataset. In the experiments conducted on the two datasets, it is important to note SPMBO feature selection is superior to the baseline MBO approach in all three datasets. The selection size rate for the features of the Dredze dataset is 7.32, the Image Spam Hunter dataset is 8.34, and Spambase was 6.51. Given these results and the accuracy results previously reported, it is possible to conclude that the SPMBO was effective for feature selection of accurately describing the greatest extent possible.

TABLE 2. Comparison of number of features selected SPMBO and MBO and methods

Sno	Dataset	MBO	SPMBO
1	Dredze	11.64	7.32
2	Image Spam Hunter	12.87	8.34
3	Spambase	9.21	6.51

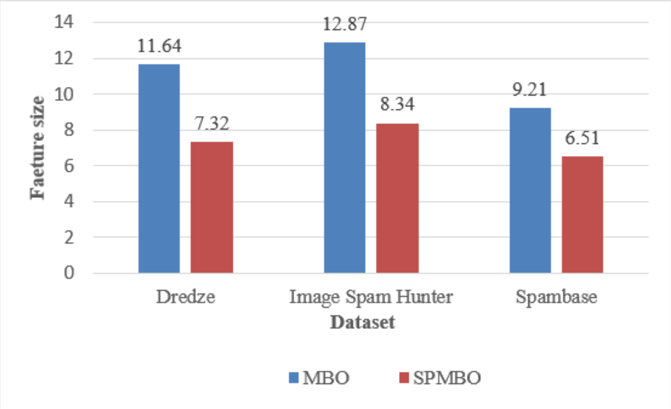


Figure 4. Features selected for SPMBO and MBO

Tables 2 and 3 illustrate the resultant metric values of different evaluation measures for the distance metrics Euclidean and Manhattan respectively. These values are computed for all discussed algorithms using 10-fold cross validation.

Table 3: Performance evaluation measures values obtained by MBO and SPMBO algorithms for Dredze dataset with Euclidean and Manhattan.

Algorithm	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-measure (%)**
MBO-kNN	97.8	98.23	97.11	96.37
SPMBO-kNN (Our work)	98.8	99.24	98.87	98.45

Figure 5 shows that the proposed SPMBO-kNN attained the highest accuracy of 98.8%, a precision of 99.24%, a recall of 98.87%, and an F1-measure of 98.45% as related to the classic

MBO-kNN algorithm on Dredze dataset

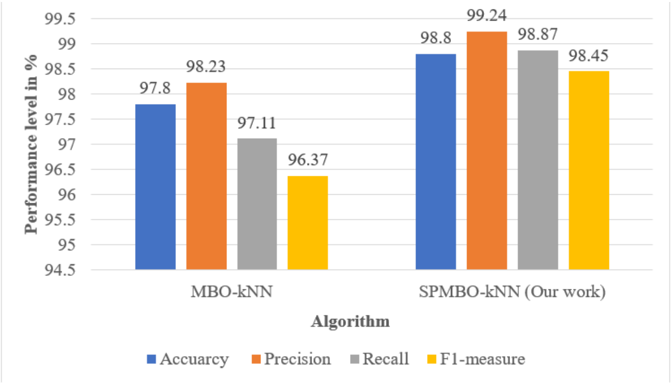


Figure 5. Results of the proposed SPMBO and MBO algorithms

Table 4: Performance evaluation measures values obtained by MBO and SPMBO algorithms for Image Spam Hunter dataset with Euclidean and Manhattan

Algorithm	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-measure (%)**
MBO-kNN	98.20	98.84	97.14	96.51
SPMBO-kNN	98.5	99.34	98.98	98.65

Figure 6 shows that the proposed SPMBO-kNN attained the highest accuracy of 98.5%, the precision of 99.34%, recall of 98.98%, and the F1-measure of 98.65% as related to the classic MBO-kNN algorithm on the Image Spam Hunter dataset benchmark.

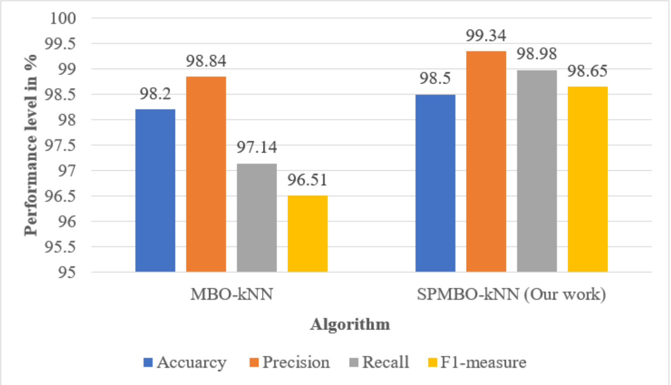


Figure 6. Results of the proposed SPMBO and MBO algorithms on Image Spam Hunter

Table 5: Performance evaluation measures values obtained by MBO and SPMBO algorithms for Spambase dataset with Euclidean and Manhattan

Algorithm	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1-measure (%)**
MBO-kNN	96.20	95.23	96.24	95.57
SPMBO-kNN	97.21	96.45	97.34	96.77

Figure 7 shows that the proposed SPMBO-kNN attained the highest accuracy of 97.21%, the precision of 96.45%, recall of 97.34%, and F1-measure of 96.77% as related to the classic MBO-kNN algorithm on the Spambase dataset benchmark

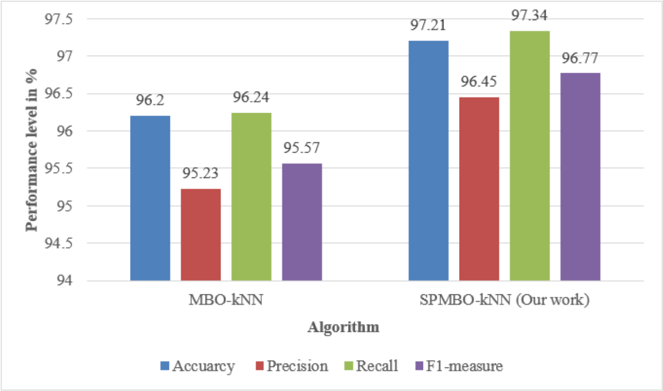


Figure 7: Results of the proposed SPMBO and MBO algorithms on Spambase

TABLE 6: Accuracy comparison on Dredze, ISH, and Spambase datasets

Author and method	**Dataset 1**	**Dataset 2**	**Dataset 3**
Author in [26]	98	97	96
Author in [27]	97.9	98.3	97.5
Our proposed model	98.8	98.5	98.8

Our paper and tests also concentrated on additional spam image datasets that are readily available to the general public, such as the well-known Dredze and Image Spam Hunter datasets. After that, our findings are contrasted with those acquired by various methodologies, each using a different collection of machine learning characteristics and procedures [28]. The use of metadata and OCR are two examples of these approaches and features, which range from fundamental to sophisticated. As seen in Table 6, the model we presented performed admirably in this domain as well. By utilizing various machine learning techniques, it not only achieved a high level of accuracy but also managed to outperform the nearly flawless results that earlier authors had reported.

V. CONCLUSION

Currently, hybrid e-mail spam filtering is a tedious job due to more features in the datasets. In the standard MBO algorithm, we presented two strategies in this paper that can be used to avoid self-adaptive and greedy modes of operation. The factor p-value is fine-tuned linearly. As a result, the starting value of the p will be used in the computation of the total number of butterflies found on both lands, 1 and 2. Furthermore, only individuals from the next generation of butterflies who have improved their fitness function are passed to the next generation. Thus proposed SPMBO results of the experiments show that search capability outperforms the standard MBO algorithm on several majorities of evaluation metrics. The results showed the highest accuracy of SPMBO is 98.8% for Dredze, 98.5% for ISH, and 97.21% for Spambase datasets, respectively. Also, the selection size rate for the features of the Dredze dataset is 7.32, the Image Spam Hunter dataset is 8.34, and Spambase was 6.51, which can be treated to better result in this research area. In future work, we plan to implement new optimizing algorithms to further reduce the features in the dataset for better e-mail spam filtering.

References

[1] Carmona-cejudo JM, Castillo G, Baena-garcía M, Morales-bueno R (2013) A comparative study on feature selection and adaptive strategies for email foldering using the ABC-DynF framework. Knowl-Based Syst 46:81–94. https://doi.org/10.1016/j.knosys.2013.03.006

[2] Kumar RK, Poonkuzhali G, Sudhakar P (2012) Comparative study on email spam classifier using data mining techniques. In: Proceedings of the international multiConference of engineers and computer scientists, vol 1, pp 14–16.

[3] Idris I, Selamat A (2014) Improved email spam detection model with negative selection algorithm and particle swarm optimization. Appl Soft Comput J 22:11–27. https://doi.org/10.1016/j.asoc.2014.05.002.

[4] Liu, Q., Qin, Z., Cheng, H., Wan, M.: Efficient modeling of spam images. In: 2010 Third International Symposium on Intelligent Information Technology and Security Informatics (IITSI), pp. 663–666 (2010)

[5] Li, P., Yan, H., Cui, G., Du, Y.: Integration of Local and Global Features for Image Spam Filtering. Journal of Computational Information Systems **8**(2), 779–789 (2012)

[6] Krasser, S., Tang, Y., Gould, J., Alperovitch, D., Judge, P.: Identifying image spam based on header and file properties using C4.5 decision trees and support vector machine. In: IEEE Workshop on Information Assurance (2007)

[7] Liu, T., Tsao, W., Lee, C.: A high performance image-spam filtering system. In: Ninth International Symposium on Distributed Computing and Applications to Business, Engineering and Science 2010, pp. 445–449 (2010)

[8] Chang, Matthew & Poon, Chung. (2009). Using phrases as features in email classification. Journal of Systems and Software. 82. 1036-1045. https://doi.org/10.1016/j.jss.2009.01.013.

[9] Ayodele, Taiwo & Zhou, Shikun & Khusainov, Rinat. (2010). Email Classification Using Back Propagation Technique. International Journal of Intelligent Computing Research. 1. https://doi.org/10.20533/ijicr.2042.4655.2010.0001.

[10] Liu, Hui & Shi, Xiaomiao & Guo, Dongmei & Zhao, Zuowei & Min, Yi. (2015). Feature Selection Combined with Neural Network Structure Optimization for HIV-1 Protease Cleavage Site Prediction. BioMed research international. 2015. 263586. https://doi.org/10.1155/2015/263586.

[11] Zhao, Mingyuan & Fu, Chong & Ji, Luping & Tang, Ke & Zhou, Mingtian. (2011). Feature selection and parameter optimization for support vector machines: A new approach based on genetic algorithm with feature chromosomes. Expert Syst. Appl.. 38. 5197-5204. https://doi.org/10.1016/j.eswa.2010.10.041.

[12]. Yusta SC (2009) Different metaheuristic strategies to solve the feature selection problem. Pattern Recogn Lett 30:525–534

[13] Tahir MA, Smith J (2010) Creating diverse nearest-neighbour ensembles using simultaneous metaheuristic feature selection. Pattern Recogn Lett 31:1470–1480

[14] Kumar L, Bharti KK (2019) An improved BPSO algorithm for feature selection. In: Khare A, Tiwary US, Sethi IK, Singh N (eds) Recent trends in communication, computing, and electronics, ed: Springer, pp 505–513

[15] Yang XS (2010) Nature-inspired metaheuristic algorithms: Luniver press

[16] Poonkodi, Et. (2021). E-Mail Spam Filtering Through Feature Selection Using Enriched Firefly Optimization Algorithm. Turkish Journal of Computer and Mathematics Education (TURCOMAT). 12. 1248-1255. https://doi.org/10.17762/turcomat.v12i5.1791.

[17] Saleh, Hadeel & Ali Alheeti, Khattab M. & Saad, Saif & Assaf, Omer & Jassam, Noor. (2019). An Enhanced Particle Swarm Optimization algorithm for E-mail Spam Filtering. AUS. 26. 245-251. https://doi.org/10.4206/aus.2019.n26.2.31.

[18] Faris, Hossam & Aljarah, Ibrahim & Al-Shboul, Bashar. (2016). A Hybrid Approach Based on Particle Swarm Optimization and Random Forests for E-Mail Spam Filtering. 9875. https://doi.org/10.1007/978-3-319-45243-2_46.

[19] Pourhashemi, Seyed. (2013). E-mail spam filtering by a new hybrid feature selection method using IG and CNB wrapper. Computer Engineering and Applications Journal. 2. https://doi.org/10.18495/comengapp.v2i3.29.

[20] Pashiri, Rozita & Rostami, Yaser & Mahrami, Mohsen. (2020). Spam detection through feature selection using artificial neural network and sine–cosine algorithm. Mathematical Sciences. 14. https://doi.org/10.1007/s40096-020-00327-8.

[21] Khalid, Yasmine & Ali, Suhad & Naser, Mohammed. (2019). Spam image email filtering using K-NN and SVM. International Journal of Electrical and Computer Engineering (IJECE). 9. 245. 10.11591/ijece.v9i1.pp245-254.

[22] Zuo, Haiqiang & Hu, Weiming & Wu, Ou & Chen, Yunfei & Luo, Guan. (2009). Detecting image spam using local invariant features and pyramid match kernel. 1187-1188. 10.1145/1526709.1526921.

[23] M. Dredze, R. Gevaryahu, and A. Elias-Bachrach, “Learning fast classifiers for image spam,” in Proceedings of the CEAS 2007 <e Fourth Conference on Email and Anti-Spam,pp. 2007–2487, Mountain View, CA, USA, August 2007.

[24] Y. Gao, M. Yang, X. Zhao et al., “Image spam hunter,” in Proceedings of the 2008 IEEE international conference on

acoustics, speech and signal processing, pp. 1765–1768, IEEE, Las Vegas, NV, USA, April 2008.

[25] M. Bassiouni, M. Ali & E. A. El-Dahshan (2018) Ham and Spam E-Mails Classification Using Machine Learning Techniques, Journal of Applied Security Research, 13:3, 315-331, 10.1080/19361610.2018.1463136

[26] A. Chavda, K. Potika, F. D. Troia, and M. Stamp, “Support vector machines for image spam analysis,” in Proceedings of the 15th International Joint Conference on e-Business and Telecommunications - Volume 1: BASS, pp. 431–441, Porto, Portugal, July 2018

[27] T. Kumaresan, S. Sanjushree, K. Suhasini, and C. Palanisamy, “Image spam filtering using support vector machine and particle swarm optimization,” International Journal of Computer Application, vol. 1, pp. 17–21, 2015.

[28] Mallampati, D., Hegde, N.P. (2022). Feature extraction and classification of email spam detection using IMTF-IDF+Skip-thought vectors. Ingénierie des Systèmes d’Information, Vol. 27, No. 6, pp. 941-948. https://doi.org/10.18280/isi.270610