Journal of Sciences, Islamic Republic of Iran 31(2): 165 - 173 (2020) http://jsciences.ut.ac.ir
University of Tehran, ISSN 1016-1104

A Classification Method for E-mail Spam Using a Hybrid Approach for Feature Selection Optimization

Z. Hassani[1*], V. Hajihashemi[2], K. Borna[3], I. Sahraei Dehmajnoonie[4]

[1] *Department of Computer Science, Faculty of Sciences, Kosar University of Bojnord, Islamic Republic of Iran*
[2] *Faculty of Enginering, Kharazmi University, Tehran, Islamic Republic of Iran*
[3] *Faculty of Mathematics and Computer Science, Kharazmi University, Tehran, Islamic Republic of Iran*
[4] *Science and Research Branch, Islamic Azad University, Kerman, Islamic Republic of Iran*

Received: 13 September 2019 / Revised: 3 December 2019 / Accepted: 10 January 2020

Abstract

Spam is an unwanted email that is harmful to communications around the world. Spam leads to a growing problem in a personal email, so it would be essential to detect it. Machine learning is very useful to solve this problem as it shows good results in order to learn all the requisite patterns for classification due to its adaptive existence. Nonetheless, in spam detection, there are a large number of features to attend as they play an essential role in detection efficiency. In this article, we're working on a feature selection method to e-mail spam. This approach is considered a hybrid of optimization algorithms and classifiers in machine learning. Binary Whale Optimization (BWO) and Binary Grey Wolf Optimization (BGWO) algorithms are used for feature selection and K-Nearest Neighbor (KNN) and Fuzzy K-Nearest Neighbor (FKNN) algorithms are applied as the classifiers in this research. The proposed method is tested on the "SPAMBASE" datasets from UCI Machine learning Repesotries and the experimental results revealed the highest accuracy of 97.61% on this dataset. The obtained results indicateed that the proposed method is suitable and capable to provide excellent performance in comparison with other methods.

Keywords: Spam Mails; Whale Optimization Algorithm; Grey Wolf Optimization Algorithm; Fuzzy K-Nearest Neighbor algorithm (FKNN); Feature Selection.

Introduction

Electronic mail is one of the important means of communication. These dayes, Most of the peoples use E-mail for different purposes since it is the fast cheap and very easy manner to communicate. This is while invaders permanently attempt to attack this useful tool for different purposes. Spam emails are the unwanted e-mails that are daily sent to the inbox of many different users [1]. Spam emails may include numerous copies of similar messages, commercial advertisements, or any other unrelated posts of pornographic contents [2].

* Corresponding author: Tel: : +989181693056; Fax:+985832262863; Email: Hassani@kub.ac.ir

Spam emails of advertisements aim at advertising various products and services, including those of software, electronics, pharmaceuticals, stocks, loans, gambling, jewelry, pornography, or even for malware and phishing attempts. The disadvantages of spam mails include wasting mailbox space and network bandwidth besides consuming user's time as forcing him/her to identify and delete the unwanted messages. Hence, one of the big challenges for individuals and organizations is spam detection [3]. There are several techniques and methods to reduce the amount of spam. An anti-spam law has been applied by legislating penalty for distributing spam emails. Machine learning is another method for email spam detection which can be able to detect and classify email data into spam and ham email [4].

various machine learning algorithms have been applied for email spam detection including algorithms that are considered in a text classification [1], like Principle Component Analysis (PCA) [1], decision tree [3, 5], Naive Bayes [2, 6-10], Support vector machine (SVM) [5, 10, 11], k-Nearest-Neighbour (k-NN) [6,11], Random tree, Random Forest [8, 11], Artificial neural networks (ANN), Logistic Regression [11].

Moreover, feature selection plays an important role in classification like email spam detection so that is very effective in shortening the training time and improving the performance. In general, Methods of feature selection include Filter, Wrapper, and Hybrid approaches. In filter method, a subset of the features is selected without taking a specific approach of learning that depends on the general features of a dataset to evaluate and select a subset of the features. The wrapper method employs classification techniques and meta-heuristic algorithms to choose the optimal subset of features [5, 12, 13]. Various aprocroches of feauture selection are considered in email spam detection such as simulated annealing, ant colony optimization[5], particle swarm optimization [3,5,7], Improved Binary Particle Swarm Optimization (IBPSO) and Binary Flower Pollination Algorithm (BFPA) [6] .

In 2013, Sharma et al. studied various machine learning technique such as Bays Net, Logic Boost, JRip, J48, Multilayer Perception, Kstar, Random tree, Random Forest, Random Committee. Their Approach is used for classifying the spam and they used spambase from UCI dataset. Algorithms Adaptions 94.28% Accuracy Achieved and best result was archived 94.28% accuracy of the Random committee [8]. BPSO with a mutation operator was proposed by Zhang for feature selection using a decision tree in an attempt to detect spam emails. In his proposed procedure, 6000 emails were prepared based on the same standard as the UCI machine learning repository except for changing the year 1999 to 2012 [3]. Feng et al. considered SVM, Naive Bayes and SVM based Naive Bayes Algorithm for filter spam emails and evaluated on DATAMALL dataset which The SVM-NB system achieved the best result [10].

A system of spam detection was proposed in a text classification mode by Esmaeili et al., who then attempted to filter out any written spam emails from the user's mailbox by using a Bayesian vs. Principle Component Analysis (PCA) method. Furthermore, forward and backward Feature Selection (FS) methods were introduced by them by finding the best tokens as the main features through a Genetic Algorithm (GA) [1]. In 2017, Naive Bayes algorithm was studied by Rusland et al. in an attempt to filter spam e-mails using the two Spam Data and SPAMBASE datasets and the accuracies of 91.13% and 82.54% were obtained for them, respectively [2]. In 2017, IBPSO and BFPA algorithms were investigated by Rajamohana et al. using Naive Bayes and k-Nearest-Neighbour (k-NN) classifiers. They also experimented the opinion spam dataset and dataset of hotel reviews, which is among the 20 most popular Chicago hotels [6].

In [5], detection of spam comments on the Facebook social network was studied using various optimization algorithms, such as simulated annealing, particle swarm optimization, ant colony optimization. Singh et al. proposed Correlation-based Feature Selection and Particle Swarm Optimization (CFS-PSO) and assessed them on WEBSPAM-UK2006 dataset [7]. Abdulhamid et al. studied different classification algorithms such as Bayesian Logistic Regression, Hidden Naïve Bayes, Radial Basis Function (RBF) Network, Multilayer Perceptron, Voted Perceptron, Lazy Bayesian Rule, Logit Boost, Rotation Forest, Logistic Model Tree, REP Tree, Random Tree, and J48. a performance analysis is done on spambase dataset which Rotation Forest obtained the best accuracy of 94.2% [9]. In 2018, Bassiouni et al. studied method of machine learning for the email spam detection that evaluated the spambase UCI dataset. They considered 10 different Classifiers(Random forest, Random Tree, Artificial neural networks(ANN), Logistic Regression, SVM, KNN, Decision Table, Bayes Net, Naïve bayas (NB), and Radial basis function (RBF)). The best performance is obtained 95.45% accuracy by using Random Forest technique [11].

In this paper, Binary Grey Wolf Optimizer (BGWO) was proposed together with KNN classifier so as to select the best feature of spam by assessing SPAMBASE database. This paper was organized in 3 main sections. In section II, the preliminaries of the

research were introduced. Also, Our proposed approach was presented in this section. Experimental results and discussions were given in section III. Finally, the conclusions were summed up.

Materials and Methods

Feature selection

Feature selection includes the process of selecting a subset of relative features to be used in model construction in the machine learning and the statistics. Three major approaches exist to select a subset of features that are recognized as Filter, Wrapper, and Hybrid approaches. The general properties of features are applied by the filter techniques for evaluating and selecting feature subsets of minimum members without having the output information.

Nonetheless, the wrapper techniques are more accurate than the filter methods since employing the output information. This method first apply an optimization algorithm for dividing features into various subsets and then utilizing a classification algorithm for evaluating the performance of those subsets in the output classification and finding the minimum members as the best classification result. Their only disadvantage is that they could be computationally expensive. Wrapper techniques have two different steps: application of an optimization algorithm that divides features into subsets and utilization of a classification algorithm. Taking these two steps could be the main reason for the slower speed of wrapper compared to filter techniques. Wrapper techniques are even involved in much slower speed when dealing with large and high-dimensional datasets.

Hybrid techniques attempt to take advantages of both the filter and wrapper techniques by simultaneously exploiting their strengths. These techniques usually apply the selected subsets by the filter techniques to be then processed by the wrapper techniques. Due to the time complexity for searching the selected subsets, wrapper techniques use an optimization method, especially meta-heuristics algorithm [12].

Binary Whale Optimization Algorithm (BWOA)

Mirjalili et al. [14] proposed a new technique called grey wolf optimization, which mimics the behavior of a whale. The algorithm is induced by the bubble-net feeding behavior. Here, the first technique was applied to the swarming behavior through a numerical model as follows:

$$\vec{D} = |\vec{C}.\vec{X}^{*}(t) - \vec{X}(t)| \quad (1)$$

$$\vec{X}(t+1) = \vec{X}^{*}(t) - \vec{A}.\vec{D} \quad (2)$$

where t, X, and X^{*} denote the current iteration, position vector, and position vector of the best founded solution, respectively. A and C represent the coefficient vectors, which are calculated as follows:

$$\vec{A} = 2.\vec{a}.\vec{r} - \vec{a} \quad (3)$$

$$\vec{C} = 2.\vec{r} \quad (4)$$

where r is the random vector in [0, 1]. Here, a is linearly decreased from 2 to 0 over the iterations.

In the bubble-net attacking method, two separate techniques are incorporated to mimic the bubble-net behavior of the whale and obtain a mathematical equation as follows:

1. **Shrinking encircling mechanism**: This approach involves linear reduction of a vector value on the interval [0, 2], while a displays a random value between -1 and 1.

2. **Spiral updating position:** The distance between the whale and the prey (small fish) is computed via this method, which demonstrates a helix-shaped circulation as follows:

$$\vec{X}(t+1) = \vec{D}'.e^{bl}.\cos(2.\pi.l) + \vec{X^{*}}(t) \quad (5)$$

where $\vec{D}' = |\vec{X^{*}}(t) - \vec{X}(t)|$ indicating the distance of the i^{th} whale to the prey. l is a random number in the range of [-1, 1] and b stands for a constant defining the logarithmic spiral shape. In addition, the whale position is assumed to be calculated based on a 50% probability for choosing between the shrinking encircling mechanism or the spiral model. If $p > 0.5$, then it would select the shrinking encircling mechanism; otherwise, the spiral model would be chosen. Here, p demonstrates a uniformly distributed random number. Furthermore, the random searching of the whales for the prey would be significant. Besides randomly searching for the prey, they would change their positions proportionate to the positions of other searching agents. Moreover, random values within the range of $1 \prec A \prec -1$ are mathematically utilized so as to make the searching agent move away from Reference Whale A as formulated as follows:

$$\vec{D} = |\vec{C}.\vec{X}_{rand} - \vec{X}(t)| \quad (6)$$

$$\vec{X}(t+1) = \vec{X}_{rand} - \vec{A}.\vec{D} \quad (7)$$

where X_{rand} is a random position vector (a random whale) chosen from the current population [14].

WOA searches are performed in a continuous space, which needs the solutions to be limited to a binary value {0, 1} for the feature selection. In a word, to solve FS problems, it is necessary for the continuous space to be transformed into a binary space.

$$\text{sigmoid(a)} = \frac{1}{1 + e^{-10(x - 0.5)}} \quad (8)$$

Where x, a are position vector in a continuous space, and binary space, respectively. Therefore, the

transformation function indicates a sigmoid function. In [15], V-shaped functions have been found to have the best performance since avoiding any local minima and the convergence speed. A V-shaped function is displayed as follows:

$$y_k = |\tanh x_k| \quad (9)$$

$$X_i^d = \begin{cases} 0, & if\ rand\ < S(x_i^k(t+1)) \\ 1, & otherwise \end{cases} \quad (10)$$

where rand is a random value in the range of [0, 1] and Function S is the sigmoid function. The search agents are forced to move in a binary space [15, 16]. Algorithm 1 displays BWOA algorithm.

Binary Grey Wolf Optimization algorithm (BGWO)

In nature, grey wolves live in a group and are divided into the social hierarchy of α, β, δ, and ω high-to-low levels. α shows the best wolf (solution) in the GWO algorithmand β and δ represent the 2nd and 3rd best wolves, respectively. ω indicates the rest of the wolves [17]. The social hierarchy behavior of the grey wolf in the group of hunters is a very interesting social behavior. The algorithm based on it can be considered as a robust meta-heuristic one based on swarm optimization. The main steps of the grey wolf hunting are taken as follows:

- Track, chase, and approach the prey
- Pursue, encircle, and harass the prey until it finally stops moving
- Attack the prey

During the hunting time, the wolves of α, β, and δ guide those of ω to approach and surround, the prey and finally attack it. The mathematical model of the hunting wolves is expressed similar to the first equation and Eq. 1 of the WOA algorithm. The feature vector of a

---

Algorithm 1: Algorithm BWOA [14]
Input: n number of whales and MaxIter number of iteration.
Output: Optimal whale position
(i) Initialize the whales population X_i (i = 1, 2,…, n) and number of iterations
(ii) Calculate the fitness of each search agent that is considered as the value of the estimator function
(iii) X^*=the best search agent
(iv) while (t<MaxIter)
for each search agent
Update a, A, C, l, and p
if1 (p < 0.5)
if2 (|A|< 1)
Update the position of the current search agent by equation (2)
else if2 (|A| >= 1)
Select a random search agent
Update the position of the current search agent by equation (7)
end if2
else if1 (p >= 0.5)
Update the position of the current search by equation (5)
end if1
end for
Check if any search agent goes beyond the search space and amend it, Calculate the fitness of each search agent
Update X^* if there is a better solution.
t=t+1
end while
(v) return X^*

Algorithm 2: Algorithm BGWOA [17]
Input: n Number of gray wolves in the pack and MaxIter number of iteration for optimization.
Output: x_α optimal gray wolf binary position, $f(x_\alpha)$ Best Optimal value.
(i) Initialize a population of n wolves positions at random ∈ [0, 1].
(ii) Find the α, β, δ solutions based on fitness.
(iii) While stopping criteria not met do
for each $Wolf_i$ ∈ pack do
Update $wolf_i$ position to a binary position by
Equation (10)
end
(1) Update a, A, C.
(2) Evaluate the positions of individual wolves.
(3) Update α, β, δ
end

candidate solution is represented by the position vector of the corresponding grey wolf. During the iteration course, the wolves of α, β, and δ represent the best and the 2nd and 3rd best candidate solutions, respectively, while those of ω are supposed to include the rest of the solutions [18]. The fundament of GWO has been explained via a flowchart well in Figure 1.

Similar to the algorithm displayed in the GWA algorithm, searches are performed in a continuous space, which needs the solutions to be limited to the binary value of {0, 1} for the feature selection. Algorithm 2 shows the main steps of the proposed BGWOA [19]. The value of stopping criteria is equal to the max iteration value.

K-Nearest Neighbour (KNN) algorithm

K-Nearest Neighbor (KNN) algorithm is a plain, slow-footed, and nonparametric classifier. It is preferred to any other classifiers since provided that all the features are continuous. Moreover, in pattern recognition, KNN algorithm is employed to provide the classification and regression processes. In both cases, the input is made of k training samples in the feature space. Applications of KNN in the classification and regression processes would determine the output type. Classification is made possible by identifying the nearest neighbor and thus distinguishing the class of an unknown sample. It is elected from among other algorithms due to its high convergence speed and simplicity. KNN classification consists of two steps:

- Detect the k number of the example in the dataset that is nearby, for instance, S
- Take this sample k number to vote and specify the class, for instance, S

The accuracy of KNN depends on the distance metric and K value. A characteristic k-NN algorithm is susceptible to the local data structure. To appraise the new unknown sample, KNN computes its K nearest neighbors and locates a class by voting of the majority [20].

Fuzzy K-Nearest Neighbour (FKNN) algorithm

Fuzzy K-Nearest Neighbour (FKNN) classifier presented by Keller et al. [21] that FKNN is an improved KNN classifier so increases performance classifier. In FKNN, the training set is evaluated by class memberships then every sample of the test set is calculated. Class membership of training instance x is precalculated from class j as follows:

$$\mu_j(x) = \begin{cases} 0.51 + \left({}^{n_j}/_{k}\right) * 0.49 & if\ j = i \\ \left({}^{n_j}/_{k}\right) * 0.49 & otherwise \end{cases} \quad (11)$$

where k is k nearest neighbor of training set and n_j is Nearest neighbors of x from class j. Then class membership of instance testing x is evaluated as follows:

$$\mu_j(x) = \frac{\sum_{i=1}^{k} \mu_j(X_i)\left({}^{1}/_{\|x-X_i\|^{2}/_{b-1}}\right)}{\sum_{i=1}^{k}\left({}^{1}/_{\|x-X_i\|^{2}/_{b-1}}\right)} \quad (12)$$

where value of b is an integer and b>1. X_i is nearst neighbor of x. If instance x belongs to class j then the value is 1, otherwise 0 [21, 22, 23].

The proposed Approach

Our proposed method aimed to enhance the classifier performance for selecting the best feature of a spam email. BWOA and BGWOA algorithms were used in

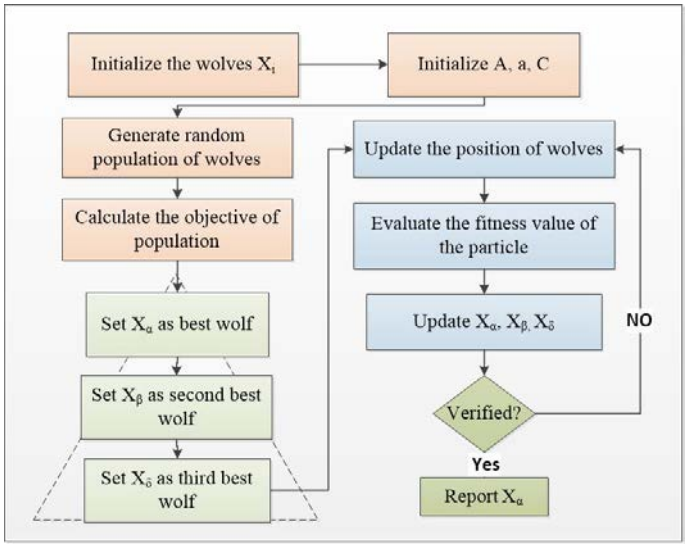

Figure 1. Diagram of algorithm GWO

this approach. The processes of the proposed method can be expounded in the five steps:

First step: The dataset was collected from the UCI Machine Learning Repositories and then the data preprocessing was started for data normalization data and creation of the best performance by the classifier. The feature values had to be within the interval of [0, 1]. The normalization formula is as follows:

$$X = (x - x_{MIN})/(x_{MAX} - x_{MIN}) \quad (13)$$

where x_{MIN} and x_{MAX} are the minimum and maximum values of each feature normalization, respectively, thus improving the results.

Second step: Parameters of algorithm are Initialized which max iterations and population sizes are set. Also, initialize a population of n agents randomly that are whales of BWOA and wolves of BGWOA. Then fitness of each initial agent is calculated based on classifiers and the best agent is found. Every population is a solution for the problem.

Third step: FS was regarded as a problem of binary optimization and the solutions were limited to the binary values of {0, 1}. The vectors of a representing 0 and 1 indicated the solutions of the problem in the forms of unselected and selected features, respectively. The length of the solution vector was shown by the number of features in the original dataset. The next preprocessing included selection of the optimal feature through the BWOA and BGWOA algorithms. Every agent of optimization algorithms is d dimension and an equal number of features datasets that each feature subset can be seen as a position of an agent

Fourth step: In this step, the fitness of each particle is calculated which is processed based on classifiers. The accuracy result of the classification was used to improve the feature selection in the wrapper-based methods which was sent to the optimization algorithm. The classifications are evaluated by the KNN and FKNN algorithms based on the accuracy criteria that was performed via 10-fold cross-validation runs of KNN and FKNN algorithms. Also, value k is considered with 3 for them.

Fifth step: The processes 3 and 4 were repeated until the stopping condition was met and the best result was obtained as the final result. HBGWOA-FKNN/KNN and HBWOA-FKNN/KNN algorithms are shown in Figures 2 and 3.

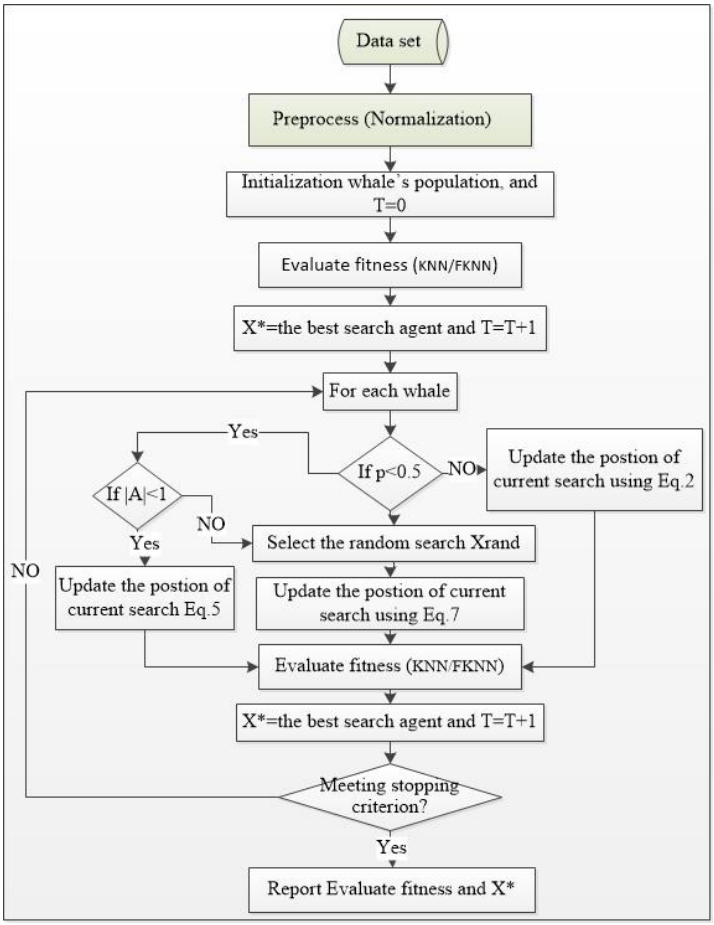

Figure 2. An overview of proposed method based on hybrid BWOA and KNN/FKNN (HBWOA -FKN/KNN)

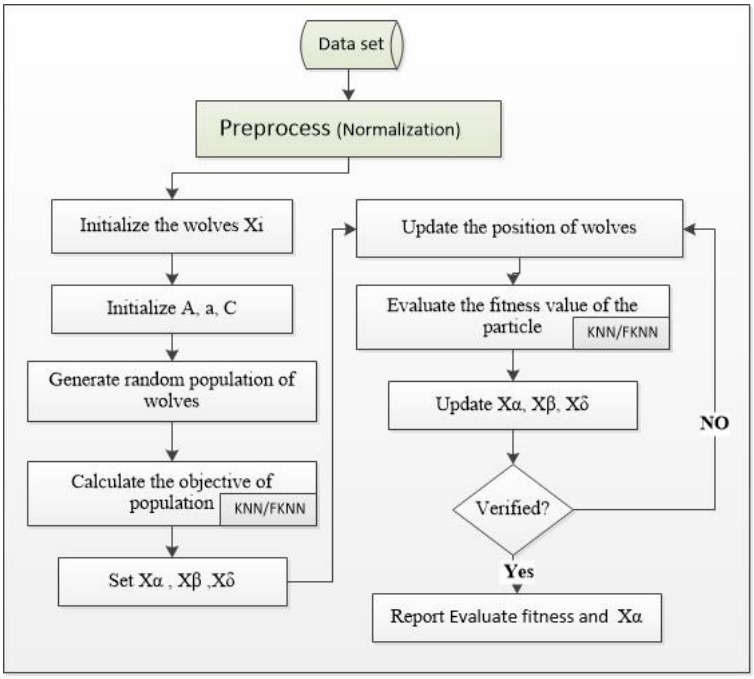

Figure 3. An overview of proposed method based on hybrid BGWOA and KNN/FKNN (HBGWOA -FKN/KNN)

Results and Discussion

This section introduces spambase database which is used for evaluating the performance of the proposed method. Afther that, the results obtained are discussed by executing proposed method. In addition to accuracy, three other criteria, i.e., precision, sensitivity, and specificity, were calculated to further evaluate the proposed hybrid method.

Dataset description

The spambase database taken from UCI machine learning repository and compiled by George Forman, Erik Reeber, Mark Hopkins, and JaapSuermondt was selected for the study. The dataset contained 4601 email messages with 195 samples, each of which consisted of 57 attributes with a special attribute forming a class. The attribute of spambease are given in Table 1.

The dataset includes the spam and non-spam emails that spam e-mails came from their postmaster and individuals who had filed spam. also, non-spam emails was collected from single e-mail accounts, personal e-mails, and filled work. This dataset included selected mail messages, which were suitable for testing spam filtering systems. The frequencies of corresponding characters and words the instance in the emails were represented by most attributes [24].

The evaluation metric

In this paper, we used evaluation metrics for investigating the efficiency of the proposed model. Accuracy, precision, sensitivity, and specificity are important evaluation metrics as shown in Eqs. 14-17:

$$\text{ACCURACY} = (\text{TP} + \text{TN})/(\text{TP} + \text{TN} + \text{FP} + \text{FN}) \quad (14)$$

$$\text{SENSITIVITY} = \text{TP}/(\text{TP} + \text{FN}) \quad (15)$$

$$\text{SPECIFICITY} = \text{TN}/(\text{TN} + \text{FP}) \quad (16)$$

$$\text{PRECISION} = \text{TP}/(\text{TP} + \text{FP}) \quad (17)$$

True Positives (TPs) and False Positives (FPs) respectively representing the numbers of correctly and

Table 1. Description of spambase dataset

Attribute Number	Attribute Type	Description Attribute
1 - 48	word_freq_WORD	percentage of words in the e-mail that match WORD.
49 - 54	char_freq_CHAR	percentage of characters in the e-mail that match CHAR
55	capital_run_length_average	average length of uninterrupted sequences of capital letters
56	capital_run_length_longest	length of longest uninterrupted sequence of capital letters.
57	capital_run_length_total	sum of length of uninterrupted sequences of capital letters = total number of capital letters in the e-mail
58	Class attribute	enotes whether the e-mail was considered spam or not

incorrectly classified records were put in the positive class. Again, True Negatives (TNs) and False Negatives (FNs) respectively demonstrating the numbers of correctly and incorrectly classified records were considered as the negative class [2, 4].

In this study, novel hybrid method is presented that combination of optimization algorithms and classifiers to select optimal features. BWOA and BGWOA algorithms are employed to select subset of features, then subset of features are evaluted by classiifiers of KNN and FKNN. The proposed model was implemented by Matlab software, version R2014a, on a computer specified with Intel core i7. The mentioned hybrid model was implemented on a spambase dataset. We performed the experiment by using different max iterations and population sizes which iterations set 50 and 20 and population sizes set 10 and 20.

Tables 2 and 3 show the results obtained of hybrid BWOA and BGWOA with KNN classification algorithm, respectively. Table 4 and 5 display the results obtained of hybrid BWOA and BGWOA with FKNN classification algorithm. The best values of the evaluation indicators for every feature selection algorithm are displayed in bold letters. According to the comparison results in Table 2 and 4, the accuracy 97.28% and the precision of 95.94% with 25 selected features have been obtained for the BWOA algorithm, respectively. Similarly, BGWO algorithm shows the accuracy and precision of 97.61% and 96.27% with 19 selected features, respectively. determination of thet least optimal feature is another benefit of the proposed model. The results illustrate that performance of hybrid optimization algorithms and FKNN classifiers are favorable results which helps in research to obtain a improve performance while the feature selection methods are utilized.

Sharma et al. considered various machine learning technique with spambase dataset. They resulted 94.28% Accuracy from Random committee [8]. Spam email filtering on SPAMBASE datasets through Naive Bayes algorithm was studied by Rusland et al., who obtained 82.54% accuracy for spambase [2]. Abdulhamid et al. investigated different classification algorithms on spambase dataset. They achieved the best performance in Rotation Forest with 94.2%. accuracy [9]. In 2018, [11] investigated email spam detection with different classifier algorithms that Random Forest was obtained best performance 95.45% accuracy. The comparison results of BWOA and BGWOA algorithms are shown in Table 6. As shown in this table, both HBWOA_FKNN and HBGWOA_FKNN algorithms have better results obtained from other studied [2, 8, 11]. The performance of the proposed model was compared to other researches in this area. One of the positive points of the

Table 2. Experimental results of HBWOA -KNN

Iterations	Agents	Accuracy	Precision	Sensitivity	Specificity
50	20	91.74	95.32	92.01	92.44
	10	92.05	91.4	94.45	87.43
20	20	92.07	91.76	93.77	87.7
	10	92.09	95.34	93.66	91.61

Table 3. Experimental results of HBGWOA - KNN

Iterations	Agents	Accuracy	Precision	Sensitivity	Specificity
50	20	92.02	92.83	93.84	89.13
	10	92.15	91.37	94.42	87.43
20	20	92.32	93.91	92.58	90.4
	10	92.57	90.32	94.38	86.01

Table 4. Experimental results of HBWOA -FKNN

Iterations	Agents	Accuracy	Precision	Sensitivity	Specificity
50	20	97.17	100	95.60	100
	10	97.28	100	95.94	100
20	20	97.07	100	95.56	100
	10	97.28	100	95.78	100

Table 5. Experimental results of HBGWOA - FKNN

Iterations	Agents	Accuracy	Precision	Sensitivity	Specificity
50	20	97.17	100	95.71	100
	10	96.96	100	95.40	100
20	20	97.61	100	96.27	100
	10	96.63	100	94.75	100

Table 6. Comparison proposed method with other work

Refrence	Database	Method	Accuracy
[8] 2013	SPAMBASE	Random committee	94.28
[2] 2017		Naive Bayes	82.54
[9] 2018		Rotation Forest	94.2
[11] 2018		Random Forest	95.45
Proposed models		BWOA-FKNN	97.28
		BGWOA-FKNN	97.61

proposed method was its higher accuracy.

Conclusions

A hybrid model has been proposed in this paper to achieve an effective and efficient detection of spam email by selecting the optimal features. our proposed method combining BWOA and BGWOA optimization algorithms and KNN and FKNN's classifiers to select the optimal feature. It is tested on UCI Machine Learning Repositories "SPAMBASE" dataset. In compliance with the other method, the results of our method show better performance on this dataset. The accuracies of the proposed model were obtained to be 97.28% and 97.61% based on the BWOA and BGWO algorithms with FKNN classifier, respectively.

References

1. Esmaeili M., Arjomandzadeh A. and Shams R., An Anti-Spam System using Naive Bayes Method and Feature Selection Methods. Int. J. Comput. Appl. **165**(4):1-5 (2017).
2. Rusland NF., Wahid N., Kasim S. and Hafit H., Analysis of Naive Bayes Algorithm for Email Spam Filtering across Multiple Dataset. Iop Conf. Ser. Mater. Sci. Eng. **226**: 1-9 (2017).
3. Zhang Y., Wang S., Phillips P. and Genlin J., Binary PSO with mutation operator for feature selection using decision tree applied to spam detection. Knowl. Based. Syst. **64**: 22-31 (2014).
4. Idris I., Selamat A. and Omat S., Hybrid email spam detection model with negative selection algorithm and differential evolution. Eng. Appl. Artif. Intel. **28:** 97–110 (2014).
5. Sohrabi MK. and Karim F., A Feature Selection Approach to Detect Spam in the Facebook Social Network. Arab J. Sci. Eng. **43**: 949–958 (2018).
6. Rajamohana SP. and Umamaheswari K., A Hybrid Approach to Optimize Feature Selection Process Using iBPSO- BFPA for Review Spam Detection. Appl. Math. Inform. Sci. **11**(5): 1443-1449 (2017).
7. singh S. and singh Ak., web-spam features selection using CFS-PSO. Procedia. Comput. Sci. **125**: 568–575 (2018).
8. Sharma S. and Arora A., Adaptive approach for spam detection. Int. J. Comput. Sci. Netw. **10**(4): 23-26 (2013).
9. Abdulhamid SM., Shuaib M., Osho O., Ismaila I. and Alhassan JK., Comparative Analysis of Classification Algorithms for Email Spam Detection. I. J. Computer Network and Information Security 60-67 (2018).
10. Feng W., Sun J., Zhang L., Cao C. and Yang Q., A support vector machine based naive Bayes algorithm for spam filtering. 2016 IEEE 35th International Performance Computing and Communications Conference (IPCCC) 1-8 (2016).
11. Bassiouni M., Ali M., and El-Dahshan EA., Ham and Spam E-Mails Classification Using Machine Learning Techniques. J. Appl. Secur. Res. **13**(3): 315-331 (2018).
12. Oreski S. and Oreski G., Genetic algorithm-based heuristic for feature selection in credit risk assessment. Expert. Syst. Appl. **41**: 2052–2064 (2014).
13. Roberto HW., George DC. and Renato FC., A global-ranking local feature selection method for text categorization. Expert. Syst. Appl. **39**(17): 12851–12857 (2012).
14. Mirjalili S. and Lewis A., the whale optimization algorithm. Adv. Eng. Softw. **95**: 51-67 (2016).
15. Hussien AG., Houssein EH., Hassanien AE., A binary whale optimization algorithm with hyperbolic tangent fitness function for feature selection. 2017 Eighth International Conference on Intelligent Computing and Information Systems (ICICIS) 166-172 (2017).
16. Amine LM, and Nadjet K., A Multiobjective Binary Bat Algorithm. Proceedings of the International Conference on Intelligent Information Processing, Security and Advanced Communication(IPAC '15) **75:** 1-5 (2015).
17. Lua C., Gaob L. and Yic J., Grey Wolf Optimizer with Cellular Topological Structure. Expert. Syst. Appl. **107**: 89-114 (2018).
18. Dipayan G., Kumar RP. and Subrata B., Load frequency control of large scale power system using quasi-oppositional grey wolf optimization algorithm. Eng. Sci. Technol. Int. J. **19**: 1693–1713 (2016).
19. Emary E., Zawbaa HM. and Hassanien AE., Binary grey wolf optimization approaches for feature selection. Neurocomputing **172**: 371-381 (2016).
20. Jabbar M., Prediction of heart disease using k-nearest neighbor and particle swarm optimization. Biomed. Res. **28**(9):1-10 (2017).
21. Keller JM., Gray MR. and Givens JA., A fuzzy k-nearest neighbor algorithm. IEEE T. Syst. Man Cyb. **15**(4): 580–5 (1985).
22. Patel H. and Thakur GS., An Improved Fuzzy K-Nearest Neighbor Algorithm for Imbalanced Data using Adaptive Approach. Iete. J. Res. **65**(6): 1-10 (2018).
23. Shang W., Huang H., Zhu h., Lin Y., Qu Y. and Wang Z., A novel feature selection algorithm for text categorization. Expert. Syst. Appl. **33:** 1–5 (2007).
24. UCI Machine Learning Repository Spambase Dataset. http://archive.ics.uci.edu/ml/datasets/Spambase.