# Hybrid Phishing Detection Using Stochastic Gradient Descent and Naïve Bayes Optimized with the Mayfly Algorithm

Xiao Chen
Computer Engineering Technical College Guangdong Institute of Science and Technology, Zhuhai 519000, China
E-mail: 13277095763@163.com

**Keywords:** phishing, phishy attack, protecting sensitive data, machine learning, metaheuristic algorithms, hybrid models

**Received:** January 15, 2025

*Because hackers were able to access AOL user credentials in 1996, phishing, a malicious method of obtaining personal data, became a significant online threat. This fraudulent practice makes use of email and website spoofing techniques to trick victims into disclosing sensitive information. Advanced practices that make use of users' trust and web vulnerabilities, such as spear phishing and tab nabbing, may be hazardous to people's security. In the classification of phishing websites, this research used two prediction models: the Stochastic Gradient Descent (SGD) and the Naïve Bayesian Classification Algorithm (NBC). Hybrid models were developed by incorporating the Mayfly Optimization Algorithm (MOA), a sophisticated optimization method for improving predictive accuracy and overall performance. The dataset contained two stages with a total of 1,353 phishing, trustworthy, and dubious websites. Hyperparameters tuned using random search method for each hybrid model. The dataset contains nine input parameters and derived from previous studies.The results indicated that, with an accuracy of 0.921 during the testing phase, the hybrid model of SGD+MO fared best. On the other hand, the NBC model with Accuracy of 0.877 identified as the weakest model with 4.4% different compared to best model. Also, further improved performance was demonstrated by the numerical classification results for the various categories: it was observed that for phishing websites, the precision metric was 0.925; for suspicious websites, it was 0.933; while for legitimate websites, the precision was 0.911. These results point out the hybrid model's ability to enhance phishing detection systems by showing how well it classifies and detects different kinds of websites.*

*Povzetek: Prispevek predstavi hibridni model za odkrivanje ribarjenja, ki združuje SGD in Naïvni Bayes, optimiziran z algoritmom Mayfly, kar prinaša znatno izboljšano natančnost zaznave.*

## 1 Introduction

### 1.1 Study background

Phishing is a fraudulent tactic used to deceive individuals into disclosing sensitive data online [1,2]. Phishers are the attackers that organize phishing attacks. When hackers obtained AOL users' login credentials in 1996, phishing emerged as a significant online threat [3,4]. A successful phishing attack is typically carried out through website spoofing techniques [5,6] and email spoofing techniques [7]. Attackers who want to trick victims into believing they are communicating with legitimate organizations like banks, credit card companies, or government agencies [8,9] begin by sending spoof emails [10]. Since their source addresses are altered to resemble emails from reliable sources, the email addresses are spoofs. For instance, if a bank manager at bank "XYZ" has the email address bankmanager@xyz.co.in, the attacker will attempt to spoof that address to trick the user into thinking the email is legitimate and following the phisher's instructions [11]. Usually, the email requests that the recipient click on links to websites and reply to the message or the website with their banking information. Email spoofing uses open SMTP (Simple Mail Transfer Protocol) servers to send bogus emails to targets. Another deceptive technique is to create phishing websites that look and feel exactly like the targeted authentic websites since many users are reluctant to divulge personal information in response to an email [12–14].

Phishing is known as "Whaling" when it goes after well-known users [6]. Phishers employ diverse techniques to execute their deception successfully. These tips consist of the following: The first method involves manipulating links so that they appear to lead to a legitimate website, but in reality, they lead to a malicious or phished URL; the second method involves avoiding phishing detection filters [15]. (iv) utilizing pop-up windows to solicit user names and passwords; (iv) using Javascript to conceal the browser address bar and construct a custom address bar that presents a hard-coded legitimate URL to the user [12]. Use pictures rather than words, which might evade detection by several phishing filters [16].

Various researchers have investigated various phishing website identification methods based on differentiating characteristics in the recent past. Rami M et al. [17] performed a thorough investigation on automatic extraction and feature analysis that could differentiate between legitimate and phishing websites using

automated tools. Their analysis listed 17 important features, the most important being "Request URL," followed by "Age of Domain" and "HTTPS and SSL." For the better utilization of these features in phishing recognition systems, they created specific rules for each feature. They established in a series of experiments that the C4.5 algorithm performed more accurately than other rule-based classification algorithms such as RIPPER, PRISM, and CBA. They also improved the prediction accuracy by focusing on the nine most informative features; the lowest error rate was 4.75% with the CBA algorithm.

Abburous et al. [18] also worked on the classification techniques of data mining for phishing detection in e-banking. They indicated that factors such as "Page Style and content" and "Social Human Factor" had little influence on the final phishing recognition rate; however, they emphasized the critical roles that URL, Domain Identity, Security, and Encryption played. By highlighting the relationships between specific characteristics in their associative classification model, they were able to develop systems that could detect phishing. Tests indicated that Associative Classification methods outperformed more conventional algorithms, such as MCAR, which had an error rate of 12.622%. They suggested that future research apply various pruning techniques to raise the accuracy and efficiency of classifiers.

Ramesh et al. [19] finding the harmed domains is crucial in phishing detection; they presented an automated approach using a unique Target Validation algorithm to ensure the accuracy of finding target domains of phishing webpages. By analyzing a fake relationship, their results enhanced protection against online identity attacks and had more than 99% for detecting harmed domains.

The last new multi-label rule-based classification algorithm, EMCAC, was presented by Neda Abdelhamid [20]. Its purpose is to generate multi-class-labeled rules without needing recursive learning. Experimental results based on the phishing data demonstrated that the EMCAC outperformed algorithms such as CBA, MCAR, MMAC, PART, C4.5, and RIPPER. They discovered a more manageable, helpful set of features for website type detection based on Chi-square feature selection. Future research will apply EMCAC to unstructured data in text categorization.

These works collectively contribute to improving phishing detection methodologies in general by emphasizing effective features, improving classification algorithms, and proposing new methods to enhance the accuracy and efficiency of phishing detection systems.

Table 1 reports a summary of the existing articles in the study field.

Nomenclature

| | | | |
|---|---|---|---|
| SGD | Stochastic Gradient Descent | $F_i^{t+1}$ | Updated position of female iii at iteration t+1 |
| NBC | Naïve Bayesian Classification | $D_{mf}$ | Distance between male and female |
| MOA | Mayfly Optimization Algorithm | W | Random walk parameter (set to 1) |
| SVM | Support Vector Machines | r | Random number |
| DT | Decision Trees | $H(F_i)$ | Fitness value of female i |
| KNN | k-Nearest Neighbors | $H(M_i)$ | Fitness value of male i |
| PSO | Particle Swarm Optimization | $X_{i,j}^t$ | Position of individual iii in dimension j at iteration t |
| GA | Genetic Algorithms | $Y_{i,j}^t$ | Position of another reference individual in dimension j at iteration t |
| $M_i^t$ | Initial position of male i at iteration t | $\emptyset_{c_i}$ | Mean of feature $x_t$ for class $C_i$ |
| $M_i^{t+1}$ | Updated position of male iii at iteration t+1 | $\sigma_{c_i}$ | Standard deviation of feature $x_t$ for class $c_i$ |
| $V_i^t$ | Initial velocity of male iii at iteration t | A(Ci) | Prior probability of class $C_i$ |
| $V_i^{t+1}$ | Updated velocity of male i at iteration t+1 | A(X) | Normalization factor (marginal probability of X) |
| $V_{i,j}^t$ | Initial velocity component for male iii in dimension j at iteration t | TP | True Positives: Correctly classified positive samples. |
| $V_{i,j}^{t+1}$ | Updated velocity component for male iii in dimension j at iteration t+1 | TN | True Negatives: Correctly classified negative samples. |
| S1 | Personal learning parameter (set to 1) | FP | False Positives: Negative samples incorrectly classified as positive. |
| S2 | Social learning parameter (set to 1.5) | FN | False Negatives: Positive samples incorrectly classified as negative. |
| $\beta$ | Exponential decay parameter (set to 2) | $P_{best}(i,j)$ | Best personal position of male iii in dimension j |
| Dp | Cartesian distance between male position and personal best position | $Q_{best}(i,j)$ | Best global position in dimension j |
| Dg | Cartesian distance between male position and global best position | R | Random number in the range [-1,1] |
| $F_i^t$ | Initial position of female i at iteration t | d | Nuptial dance value (set to 5) |

Table 1: Summary of the previous studies.

| Authors | References | Techniques/Models Used | Dataset Used | Performance Metrics |
|---|---|---|---|---|
| Rami M et al. | [17] | C4.5, RIPPER, PRISM, CBA, Rule-based classification | Not specified | CBA: 4.75% error rate, C4.5 performed best |
| Abburous et al. | [18] | Associative Classification, MCAR, Data Mining techniques | E-banking data | MCAR: 12.622% error rate, Associative performed best |
| Ramesh et al. | [19] | Target Validation Algorithm, Automated phishing domain detection | Not specified | 99% accuracy in detecting harmed domains |
| Neda Abdelhamid | [20] | EMCAC (Multi-label classification), CBA, MCAR, MMAC, PART, C4.5, RIPPER | Phishing dataset | EMCAC outperformed others, Feature selection via Chi |

## 1.2 Objective of the study

Phishing attacks pose a significant and evolving cybersecurity threat, necessitating the development of accurate and robust detection systems. Traditional rule-based and blacklist-based approaches often fail to detect new and sophisticated phishing techniques due to their static nature. To address this challenge, this study proposes a hybrid machine learning-based approach to improve phishing website classification accuracy and adaptability.

This study investigates the effectiveness of integrating the Mayfly Optimization Algorithm (MOA) with two machine learning models: Naïve Bayes Classifier (NBC) and Stochastic Gradient Descent (SGD). The hypothesis tested is that the hybrid approach—leveraging the strengths of probabilistic and gradient-based learning models with a bio-inspired optimization algorithm—enhances predictive accuracy, convergence speed, and robustness against evolving phishing techniques compared to standalone models.

❖ **Justification for model selection:**

Stochastic Gradient Descent (SGD) was chosen due to its ability to efficiently handle large-scale datasets, making it well-suited for real-time phishing detection. Unlike traditional classifiers such as Support Vector Machines (SVM), which can be computationally expensive in high-dimensional spaces, SGD updates model parameters iteratively, ensuring fast adaptation to new patterns in phishing data. Compared to Decision Trees (DT) and k-Nearest Neighbors (KNN), which may struggle with scalability and feature complexity, SGD offers superior generalization, making it ideal for dynamic and evolving phishing attack scenarios. Similarly, the Naïve Bayes Classifier (NBC) was selected due to its probabilistic nature, which allows for interpretable decision-making and robust performance in high-dimensional spaces. Unlike deep learning models like Neural Networks, which require extensive training data and computational resources, NBC remains computationally efficient and effective even with limited training samples. Compared to Logistic Regression, which assumes a linear decision boundary, NBC handles non-linearity better due to its probabilistic assumptions, making it a strong candidate for phishing classification tasks.

❖ **Justification for optimizer selection:**

MOA was incorporated to enhance the predictive accuracy of SGD and NBC by optimizing hyperparameters and improving model convergence. Unlike traditional optimization techniques like Grid Search and Random Search, which are computationally expensive, MOA dynamically balances exploration and exploitation, enabling more efficient searching for optimal model parameters. Compared to Particle Swarm Optimization (PSO) and Genetic Algorithms (GA), MOA exhibits superior convergence speed and stability due to its biologically inspired swarming mechanism, making it well-suited for improving phishing detection models.

❖ **Novelty of the work:**

The novelty of this research lies in the hybridization of MOA with SGD and NBC to enhance phishing website classification. Unlike previous studies that rely on standalone machine learning models or traditional optimization techniques, this study introduces a bio-inspired optimization approach to fine-tune machine learning classifiers, improving their convergence and predictive performance. The proposed hybrid models demonstrate superior robustness against evolving phishing attacks, providing a scalable and interpretable solution for real-time cybersecurity applications.

By leveraging the efficiency of SGD, the interpretability of NBC, and the adaptive search capabilities of MOA, this study contributes a novel and effective approach to phishing detection. The results validate the hypothesis that the integration of optimization

algorithms with machine learning models enhances phishing detection accuracy, offering a more reliable and adaptable defense mechanism against cyber threats.

### 1.3 Paper organization

The structure of the article is: Section 2 describes the procedures of data collection and preprocessing in detail. Section 3 introduces the ML models for phishing detection, and Section 4 explains the metaheuristic algorithm for the optimization of these models. Section 5 presents the appraisal factors utilized to examine the execution of the model, including accuracy and F1-score. In Section 6, the results are presented both numerically and visually, and the findings are discussed. The study's summary of the main conclusions and recommendations for future research directions are given in Section 7. The references section concludes with a list of all cited works for additional reading.

## 2 Data collection

A reliable database derived from [21] for monitoring phishing websites, Phish Tank, provided 1353 inputs for the dataset used in this study. The features present in the dataset for identifying phishing websites include Prefix/Suffix, SFH, Request URL, Web Traffic, Anchor URL, URL Length, Domain Age, Sub Domain, and IP Address. These characteristics were chosen because of their potential to be useful during the classification process, where one tries to determine whether a website is phishing or authentic. To construct the models and improve them during the training phase, 947 inputs were used. Thus, the remaining 406 inputs were used during the testing phase to confirm the functionality of the models. This division makes sure the models are adequately trained and subjected to a tight evaluation process for determining generalizability and predictive accuracy.

Fig. 1 presents the impact of each input parameter on the website classification outcomes. Among the parameters, the prefix/suffix parameter has shown the most positive impact, with a significant value of 0.24. The domain age parameter then shows a positive influence with a value of 1.7. On the other hand, residual parameters have adverse effects on the classification results. While stressing that other features may reduce predictive performance, this analysis emphasizes the significance of particular features, such as the prefix/suffix and domain age, in improving the accuracy of phishing website detection.

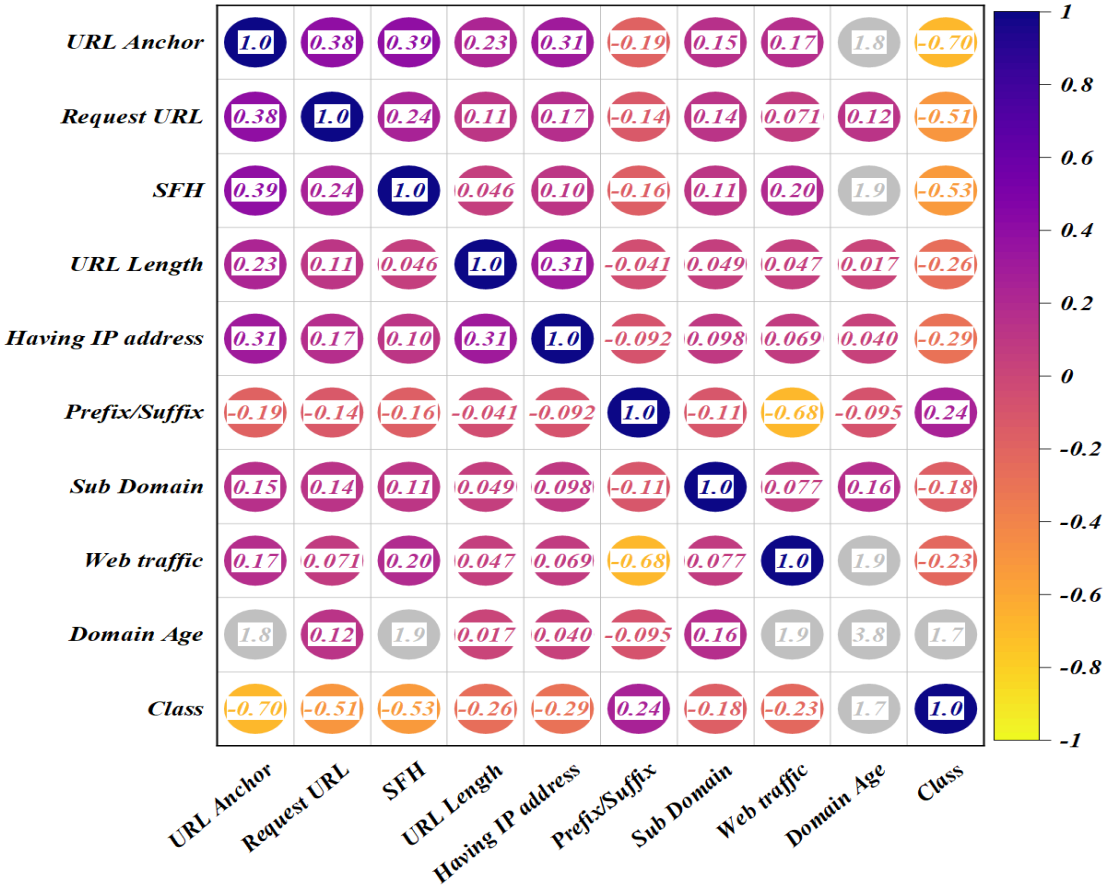



Figure 1: The plot illustrates the correlation between the input and output

# 3 Machine learning models

## 3.1 Stochastic gradient descent (SGD)

SGD is widely used and has demonstrated advanced execution on numerous ML tasks, as demonstrated in [22] and [23]. To specify the length of the subsequent stage to take when going in the gradient's direction, SGD incorporates a parameter known as the learning rate. A learning rate that is too small results in slow convergence, while a learning rate that is too large can impede convergence and cause the loss function to diverge or even oscillate around the minimum. For this work, a learning rate has been chosen that is modified by a predetermined schedule as the training progresses. However, n-SGD has a more significant step size than SGD because it is more difficult to take more significant steps in one point in SGD without running the risk of preventing convergence due to the noise on the gradient. The model in n-SGD is updated using minibatches, which are small collections of training samples. Here, the batch size is set to n, which is the total number of points to take into account when calculating the gradient.

## 3.2 Naïve Bayesian classification algorithm (NBC)

The NBC organization method functions on the assumption that every model's attributes are free within their respective groups. These are assumptions that most people find annoying, but NBC consistently performs well in verified scenarios. It can predict the possibility of being placed in a particular class based on tests from that class by computing contingent probabilities using Likelihood Bayes [24,25]. One advantage of this approach is that it expects property freedom, which means that order can be established with essentially a change in a variable at the class level instead of the entire covariance structure.

Determining the mean and standard deviation of the highlights in the preparation materials for each class is the most crucial step in this classification technique [26,27]. At that point, the processed mean and standard deviation are utilized to determine the probability $A(x_t|c_i)$.

$$A(x_t|c_i) = g\left(x_t, \varphi_{c_i}, \sigma_{c_i}\right) = \frac{1}{\sqrt{2\pi}\sigma_{c_i}} exp^{-\frac{(x_t-\varphi_{c_i})^2}{2\sigma_{c_i}^2}} \quad (1)$$

The Gaussian thickness capacity for the quality $x_t$ is addressed by the articulation $g\left(x_t, \varphi_{c_i}, \sigma_{c_i}\right)$. $\varphi_{c_i}$ and $\sigma_{c_i}$ represent the standard deviation of characteristic $x_t$ in the preparation information for the class.

The likelihood $A(x_t|c_i)$ for each trademark in each class is replicated by the likelihood $A(X|C_i) = \pi_{(t=1)}^{n} A(x_t|C_i)$, which yields the probability $A(x_t|c_i)$ for each class. The back likelihood, or $A(x_t|c_i)$, can be found by multiplying this $A(x_t|c_i)$ by the earlier probability of each class.

To get the probability $A(x_t|c_i)$ for each class, the likelihood $A(x_t|c_i)$ for each trademark in each class is replicated by the likelihood $A(X|C_i) = \pi_{(t=1)}^{n} A(x_t|C_i)$. By multiplying this $A(x_t|c_i)$ by the earlier probability of each class, one may find the back likelihood or $A(C_i|X)$.

$$A(C_i|X) = \frac{(X|C_i)\, A(C_i)}{A(x)} \quad (2)$$

When $1 \leq j \leq m$ and $j$ isn't equal to $i$, the test data are categorized into a specific class according to whether or not they satisfy the measures $A(C_i|X) > A\left(C_j|X\right)$. In Fig. 2, the NBC flowchart is displayed.

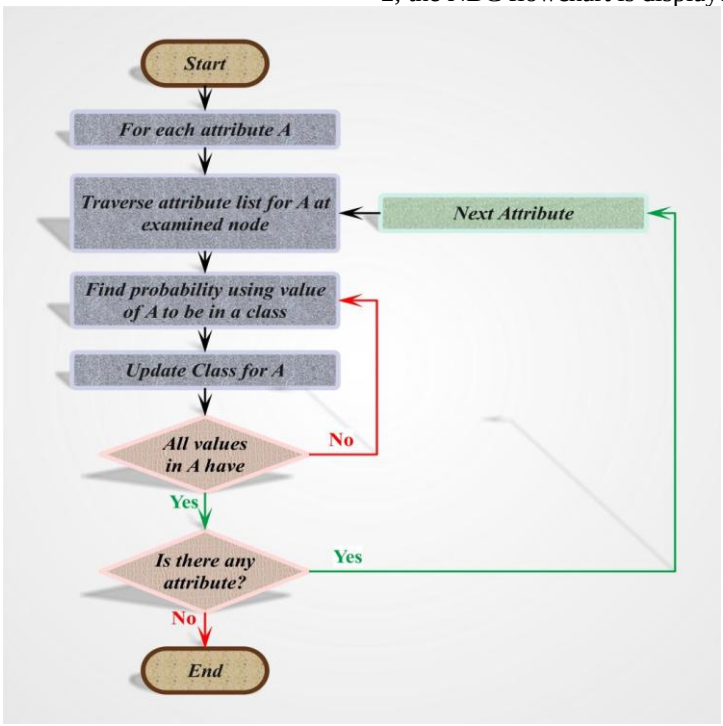



Figure 2: NBC Flowchart [28]

# 4 Metaheuristic algorithms

## 4.1 Algorithm for optimizing mayfly (MOA)

MOA is a population-based technique that was developed in 2020 [29,30]. The PSO is the source of the MOA and combines all of the main advantages of the PSO, FA, and GA, as stated in the authors' prior claim. The activities listed below comprise the MFO concept: i) a pair of equal numbers of male and female agents are first sent out; ii) the male mayfly chooses which $Q_{best}$ is appropriate for the current task; iii) the female mayfly searches and merges with the male mayfly at $Q_{best}$ ; iv) procreating, and v) the search is terminated and the results are displayed.

The total efficacy of the MOA is determined by the male's initial location and the distance at which it attracts the female. This approach allows an equal number of male and female agents to be freely initiated in the search space, allowing each Mayfly to converge toward the $Q_{best}$ with a rise in convergence. This procedure will terminate when every couple of agents produces the same number of children. Every generated offspring is assigned a zero velocity to end the process and prevent it from moving further. The remaining pertinent details are located in [29,30], and [31] provide access to the basic code.

In a d-dimensional search area, assume that there are an equal number of male (M) and female (F) Mayflies. The overall count of agents (Mayflys) is represented by the numbers i=1, 2..., N. Every agent is randomly initialized in the search locality during the optimization search, and as the iteration count rises, each agent is permitted to advance toward the best position ($Q_{best}$). By changing its position and speed, the male can get to the $Q_{best}$. The agent will travel in the direction of its target based on the Cartesian distance and the increasing iteration. Similar to this procedure is the FA discussed in [32]. The revised position and velocity are displayed in Eqs. (3) and (4);

$$M_i^{t+1} = M_i^t + V_i^{t+1} \tag{3}$$

$$V_{i.j}^{t+1} = V_{i,j}^t + S_1 \times e^{-\beta {D_p}^2}\left(P_{best_{i,j}} - M_{i,j}^t\right) + S_2 \times e^{-\beta {D_g}^2}\left(Q_{best_{i,j}} - M_{i,j}^t\right) \tag{4}$$

where $\beta = 2$, personal learning parameters $(S_1)$=1, $(S_2)$=1.5, $M_i^t$ and $M_i^{t+1}$ are the initial and modified positions, $V_i^{t+1}$ and $V_{i,j}^{t+1}$ are the initial and modified velocities, and $D_p$ and $D_g$ are the Cartesian distances. The FA and PSO values are combined to form the frame in (4). To seduce the female (F) with a distinctive nuptial dance (dancing up and down on a water surface), every man (M) will attain the $Q_{best}$ when the updation continues depending on the advancement in cycles. The definition of the velocity update during this procedure is as follows:

$$V_{i,j}^{t+1} = V_{i,j}^t + d \times R \tag{5}$$

where $R$ = random numeral [-1], [1], and nuptial dance value $(d) = 5$.

Each female (F) is then allowed to determine the guy who is at $G_{best}$ once the male has finished his optimal search. which a female (F) may use a random walk ($W = 1$) value to escape to a new location or move in the direction of the male ($D_{mf}$). The update of location and velocity for F may be expressed mathematically in Eq. (6) and Eq. (7):

$$F_i^{t+1} = F_i^t + V_i^{t+1} \tag{6}$$

$$V_{i,j}^{t+1} = \begin{cases} V_{i,j}^t + S_2 \times e^{-\beta {D_{mf}}^2}\left(X_{i,j}^t - Y_{i,j}^t\right) & if\ H(F_i) > H(M_i) \\ V_{i,j}^t + W \times r & if\ H(F_i) \le H(M_i) \end{cases} \tag{7}$$

$H$ stands for the maximized objective value.

As the number of iterations grows, each F will ultimately become the correct M, resulting in the birth generation. The product of M and F displays the overall count of offspring in the MOA. After mating, the progeny of M and F will thus always have an initial velocity value of zero. Only M and F's search performance is considered in the IMLT issue since the initial velocity operator of the offspring is neutralized. The best characteristics of the FA, PSO, and FA are merged to generate the recommended MOA, as shown by the previously described equations. Additional relevant information is available from [29,30]. Fig. 3 shows the flowchart of MOA.

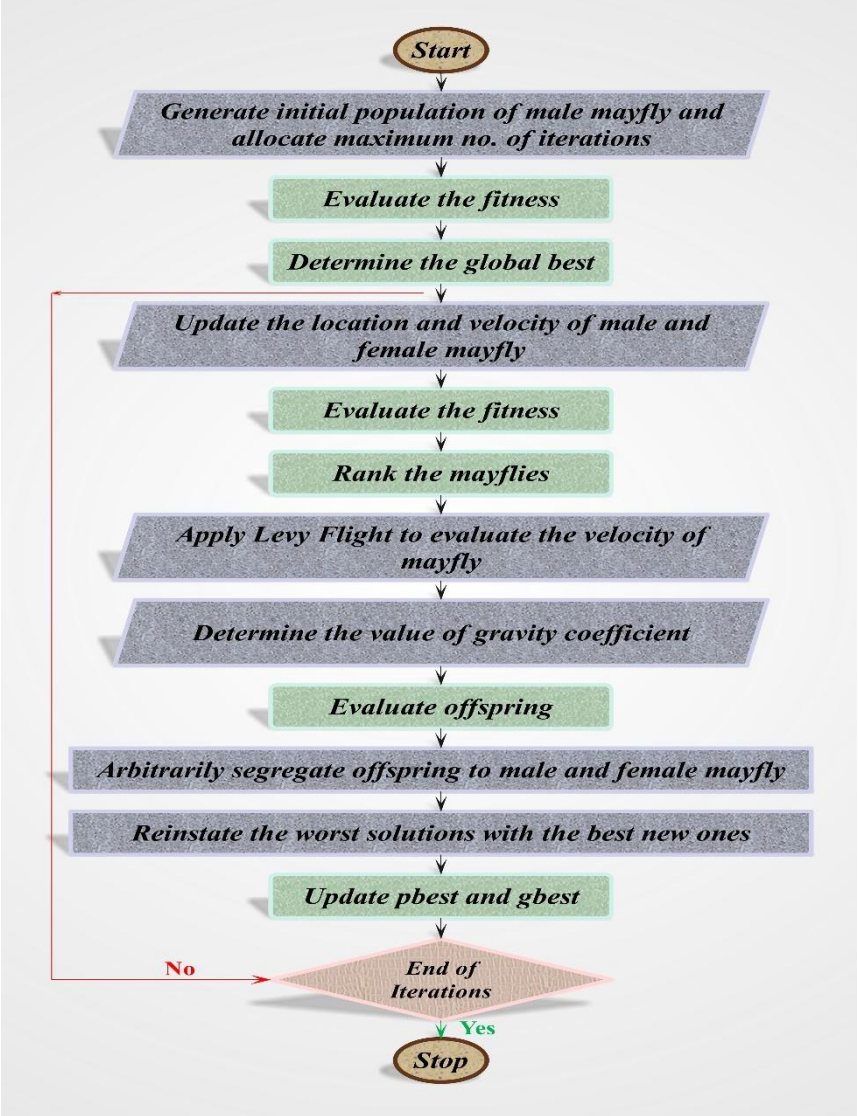



Figure 3: MOA flowchart

# 5 Performance evaluators

To assess the performance of the classification models, four key evaluation metrics are considered: Accuracy, Precision, Recall, and F1-Score. These metrics provide a quantitative measure of the model's effectiveness in classifying data and its resilience against misclassification.

- **Accuracy:** Accuracy is the most straightforward metric, representing the proportion of correctly classified instances in the dataset. It is defined as:

$$\text{Accuracy} = {(F^P + T^N)}/{(F^P + T^N + F^P + F^N)} \quad (8)$$

where **TP** (True Positives) and **TN** (True Negatives) denote correctly classified samples, while **FP** (False Positives) and **FN** (False Negatives) indicate misclassified samples. Although accuracy provides an overall performance measure, it may not be reliable for imbalanced datasets, where one class dominates.

- **Precision:** Precision evaluates how well the model identifies positive instances while avoiding false positives. It is given by:

$$\text{Precision} = {T^P}/{(T^P + F^P)} \quad (9)$$

A high precision score indicates that the model minimizes false positives, making it particularly valuable in applications where false alarms must be reduced.

- **Recall:** Recall (also known as Sensitivity) measures the model's ability to correctly identify all relevant instances. It is expressed as:

$$\text{Recall} = F^P R = {F^P}/{P} = {F^P}/{(F^P + F^N)} \quad (10)$$

A high recall means that the model effectively captures all true positive cases, reducing the likelihood of missing critical classifications. This metric is particularly important in scenarios where failing to detect positive instances has severe consequences, such as in medical diagnosis.

- **F1-Score**

The F1-Score provides a balanced measure of a model's performance by combining Precision and Recall into a single metric. It is calculated as the harmonic mean of the two:

$$F1-Score = (2 \times Recall \times Precision) / (Recall + Precisi \quad (11)$$

This metric is especially useful when dealing with imbalanced datasets, as it considers both false positives and false negatives in the evaluation.

By analyzing these evaluation metrics together, a comprehensive understanding of the classification model's performance is obtained, ensuring robust and reliable predictions. K-Fold Cross validation

K-fold cross-validation (KCV) is a commonly used technique for model selection and error estimation in classification tasks. It involves dividing the dataset into k subsets, where, in each iteration, some subsets are utilized for training while the remaining ones are used for testing the model's performance. In this study, a 5-fold cross-validation strategy (k=5) was implemented to improve the proposed algorithms by systematically varying the training and testing data. Fig. 4 illustrates the accuracy achieved across different folds. The results show that K5 recorded the highest accuracy (0.90909), followed by K2 (0.90613), while K4 had the lowest accuracy (0.8847). The variation observed among the folds demonstrates the effectiveness of cross-validation in evaluating model generalization.

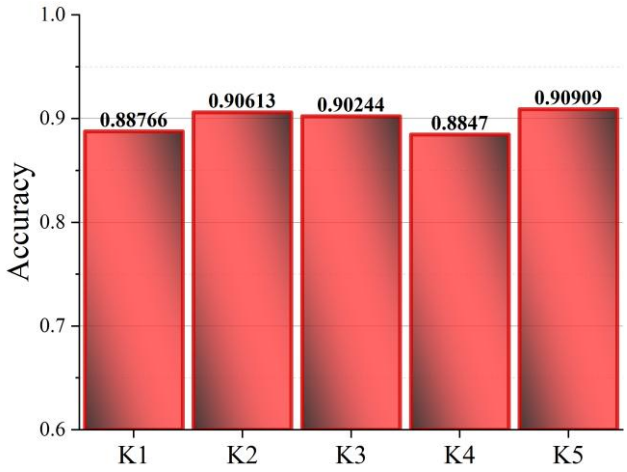



Figure 4: The results of 5-Fold cross validation

# 6 Results

## 6.1 Results of hyperparameters and convergence curves

Hyperparameters are predefined settings that control the learning process of a machine learning model. Unlike model parameters, which are learned from data, hyperparameters are set before training and have a significant impact on model performance. Various techniques exist for tuning hyperparameters to enhance model accuracy and efficiency. One widely used approach is random search, which was employed in this study to optimize the hyperparameters of the proposed hybrid models. Table 2 presents the optimized hyperparameter values for the SG + MO and NBC + MO models. For SG + MO, key hyperparameters include alpha (0.001), l1_ratio (0.014692), epsilon (0.022627), n_jobs (2), eta0 (0.999), and power_t (0.959754), which were fine-tuned to enhance the model's predictive performance. For NBC + MO, the most influential hyperparameters were alpha (62) and min_categories (4). These values were determined through random search, ensuring optimal model performance while balancing computational efficiency.

Table 2: Outcomes of hyperparameters for hybrid models.

| Hyperparameter | Models | |
|---|---|---|
| | SG + MO | NBC + MO |
| alpha | 0.001 | -- |
| l1_ratio | 0.014692 | -- |
| epsilon | 0.022627 | -- |
| n_jobs | 2 | -- |
| eta0 | 0.999 | -- |
| power_t | 0.959754 | -- |
| alpha | -- | 62 |
| min_categories | -- | 4 |

Two convergence graphs for hybrid models over 200 iterations are shown in Fig. 5. The y-axis shows convergence (accuracy), and the x-axis shows the count of iterations. The accuracy in the left graph, labeled SG+MO, starts at about 0.654 and rises stepwise until the 200th iteration, when it reaches an approximate accuracy of 0.92831. Comparably, the accuracy of the right graph, designated NBC+MO, begins at about 0.601 and increases stepwise until it reaches about 0.89579 by the 200th iteration. Both graphs demonstrate the convergence behavior of the hybrid models by showing a general trend of increasing accuracy with the number of iterations.

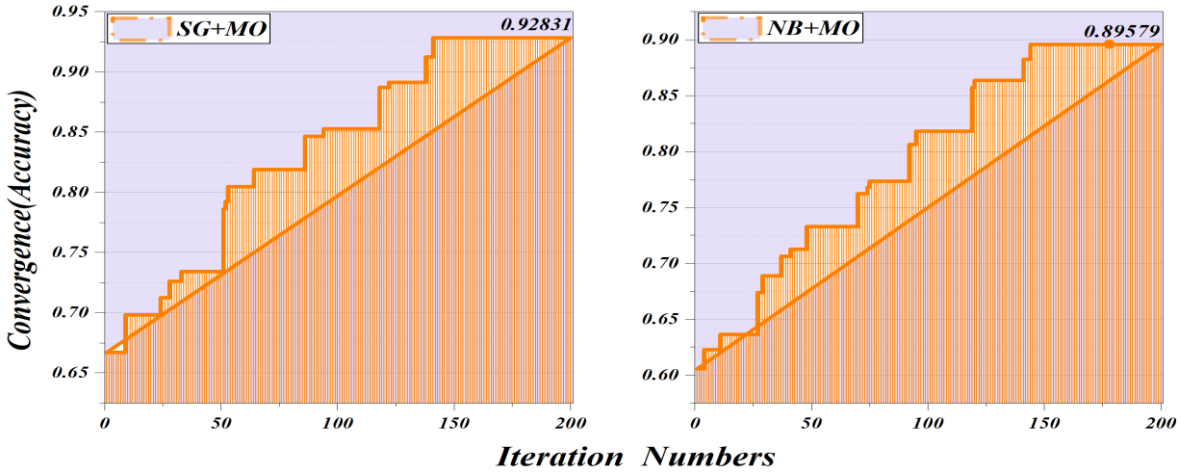



Figure 5: The convergence graphs of the hybrid models

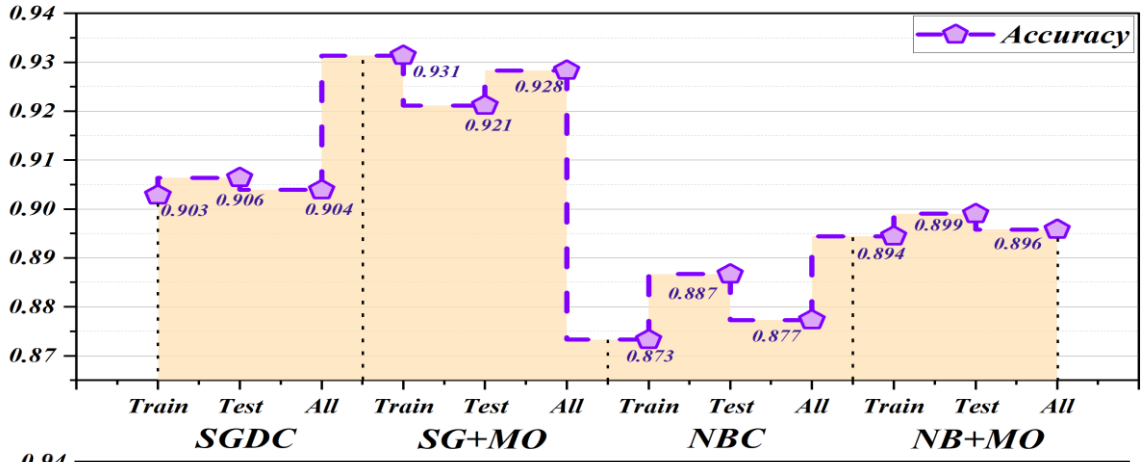
## 6.2 Results of metrics for predictive models

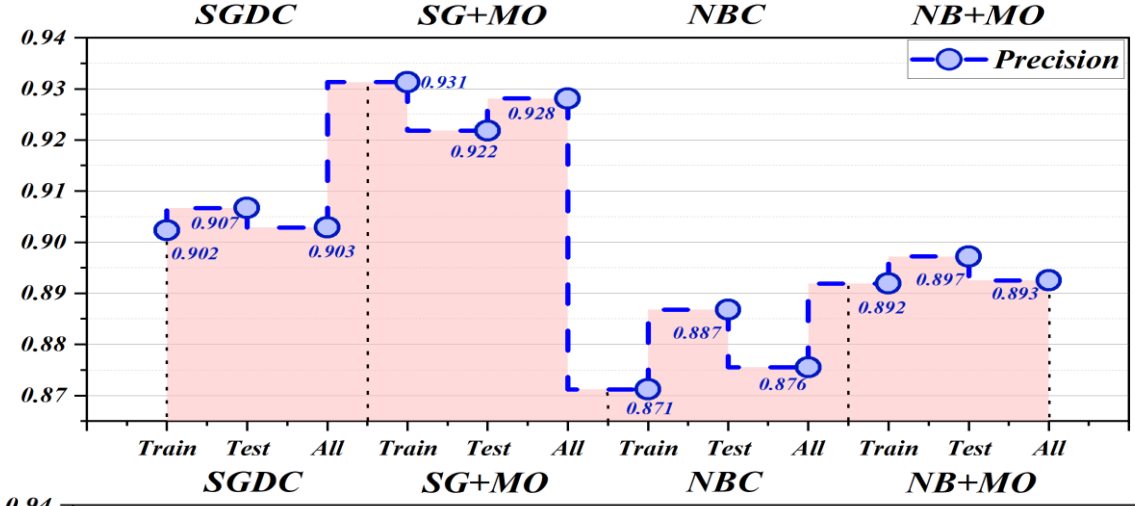
In the overall accuracy evaluation over all datasets, as shown in Table 3, the single SGDC model performed admirably, with an accuracy of 0.904, which was higher than the NBC model's 0.877 accuracy. This pattern continued with the hybrid models, where the SG+MO configuration outperformed the NBC+MO model with an accuracy of 0.928, surpassing its performance of 0.896. The SG+MO model demonstrated a significant performance difference, surpassing the NBC+MO model by 3.57%. Beyond accuracy, the SG+MO model also performed better on other critical metrics, such as precision, recall, and F1-score, all of which registered at an excellent 0.928. These findings, which are presented in detail in Table 3 and visually represented in Fig. 6, confirm the effectiveness of the SG+MO model when compared to other configurations and demonstrate its strong performance across a range of evaluation parameters.

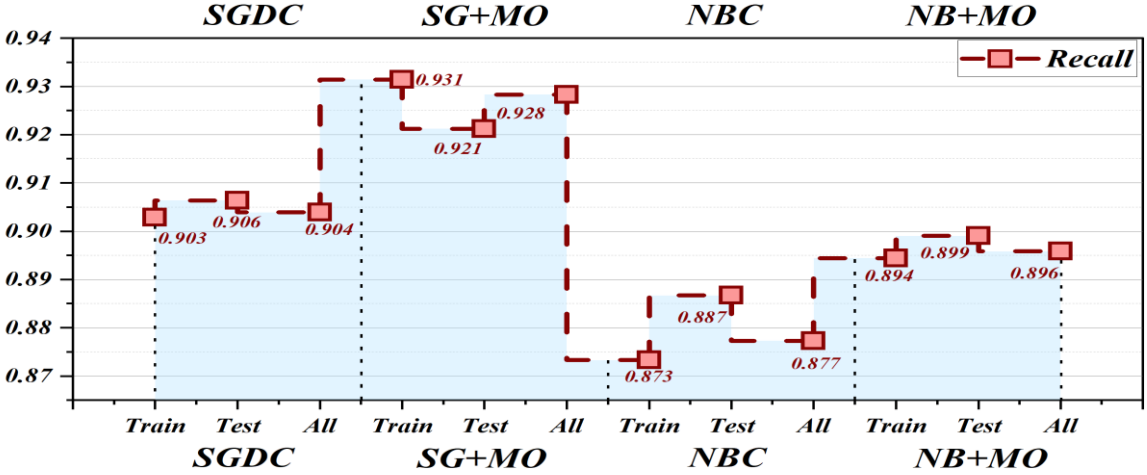
Table 3: Outcomes of the presented developed models.

| Section | Model | Metric values | | | | | |
|---|---|---|---|---|---|---|---|
| | | Accuracy | Precision | Recall | F1-score | AUC | MCC |
| Train | SGDC | 0.903 | 0.902 | 0.903 | 0.902 | 0.881 | 0.8252 |
| | SG+MO | 0.931 | 0.931 | 0.931 | 0.931 | 0.915 | 0.8760 |
| | NBC | 0.873 | 0.871 | 0.873 | 0.866 | 0.799 | 0.7681 |
| | NBC+MO | 0.894 | 0.892 | 0.894 | 0.890 | 0.835 | 0.8085 |
| Test | SGDC | 0.906 | 0.907 | 0.906 | 0.906 | 0.881 | 0.8340 |
| | SG+MO | 0.921 | 0.922 | 0.921 | 0.921 | 0.915 | 0.8607 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | NBC | 0.887 | 0.887 | 0.887 | 0.881 | 0.799 | 0.7965 |
| | NBC+MO | 0.899 | 0.897 | 0.899 | 0.897 | 0.835 | 0.8200 |
| All | SGDC | 0.904 | 0.903 | 0.904 | 0.903 | 0.881 | 0.8277 |
| | SG+MO | 0.928 | 0.928 | 0.928 | 0.928 | 0.915 | 0.8714 |
| | NBC | 0.877 | 0.876 | 0.877 | 0.870 | 0.799 | 0.7769 |
| | NBC+MO | 0.896 | 0.893 | 0.896 | 0.892 | 0.835 | 0.8118 |



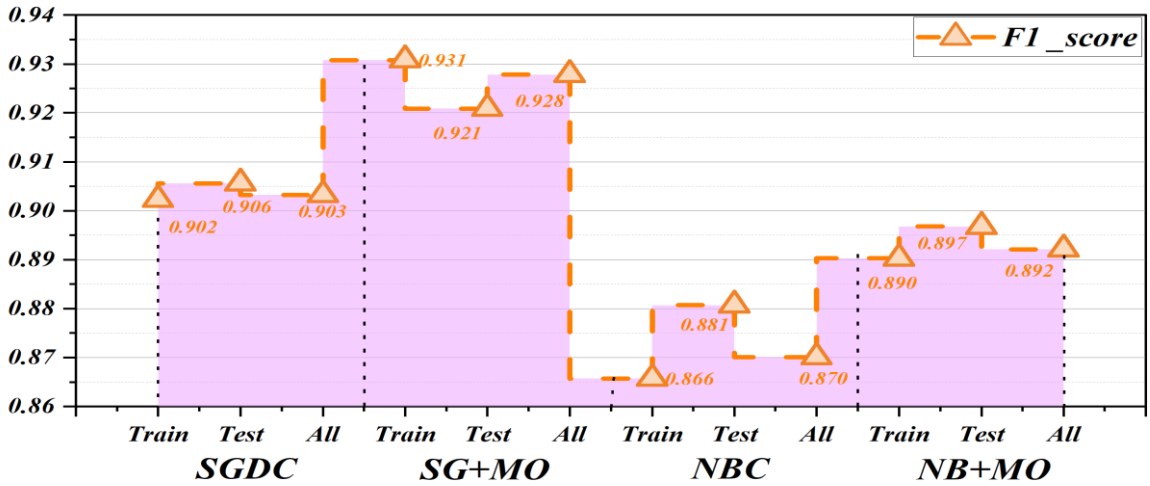
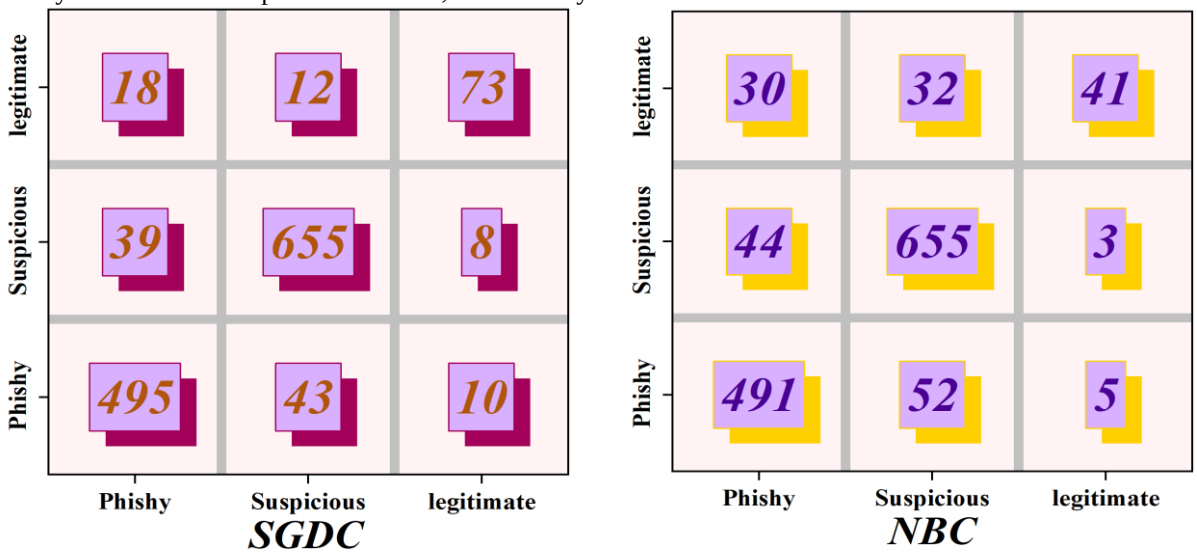



Figure 6: Plot showing the performance of the models across different phases

The confusion matrix, shown in Fig. 7, compares the actual and predicted classifications for three categories: Phishy, Suspicious, and Legitimate. It is a tool used to assess the effectiveness of classification models. The following is revealed by a thorough analysis of the misclassifications by rows for the top-performing SG+MO model: While correctly identifying 82 legitimate instances, the model misclassified 12 as phishy and nine as suspicious for legitimate instances. Although the model correctly identified 669 suspicious instances, it incorrectly classified 29 as phishy and four as legitimate. The model correctly identified 505 phishy instances but incorrectly classified 39 as suspicious and four as legitimate. The most notable misclassification in the matrix is that of phishy instances being classified as suspicious (39), while the model has the highest number of correct classifications for suspicious instances (669). Furthermore, when comparing phishy and suspicious instances to legitimate ones, the model shows comparatively fewer misclassifications.



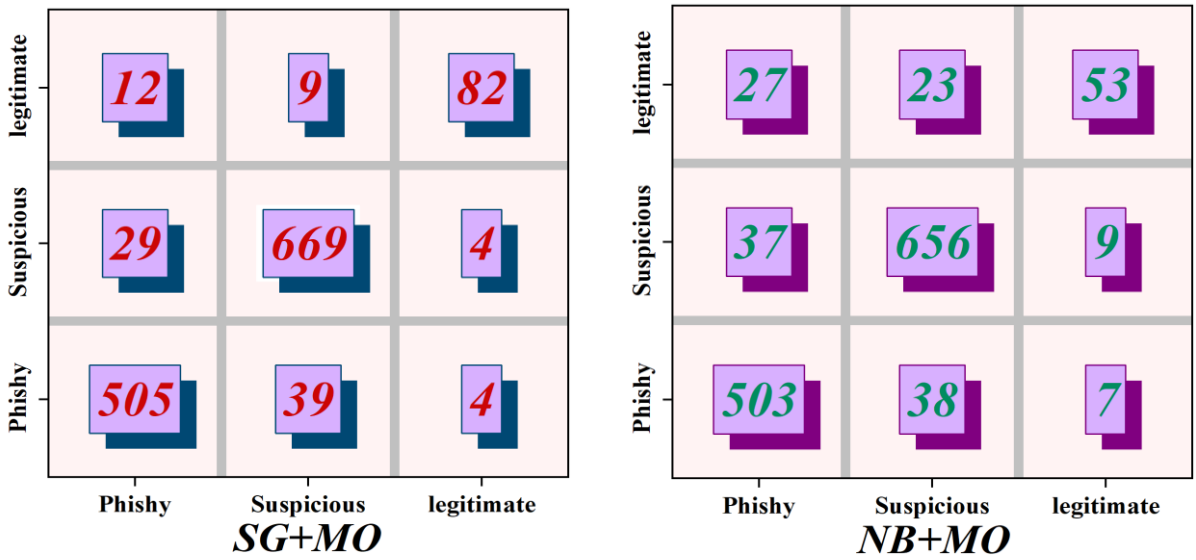



Figure 7: Confusion matrix showing the accuracy of the models under four specified conditions

The numerical classification results for each model are displayed in Table 4, where the SG+MO model performs better on many metrics. In particular, the SG+MO model shows the highest accurate results (0.923 for phishy, 0.943 for suspicious, and 0.850 for legitimate instances) when it comes to the F1-score. Furthermore, the model achieves 0.922 for phishy, 0.953 for suspicious, and 0.796 for legitimate classifications, according to a comparison of the Recall results. The SG+MO model continues to perform well when analyzing the Precision results, scoring 0.925 for phishy, 0.933 for suspicious, and 0.911 for legitimate instances. Together, these metrics show that the SG+MO model achieves high F1 scores by maintaining a balanced trade-off between precision and recall, in addition to being highly effective at correctly identifying positive cases in all three categories.

Table 4: Grading-based classification of the performance of the developed model

| Metric values | Grade | Model | | | |
|---|---|---|---|---|---|
| | | SGDC | SG+MO | NBC | NBC+MO |
| Precision | Phishy | 0.897 | 0.925 | 0.869 | 0.887 |
| | Suspicious | 0.923 | 0.933 | 0.886 | 0.915 |
| | legitimate | 0.802 | 0.911 | 0.837 | 0.768 |
| Recall | Phishy | 0.903 | 0.922 | 0.896 | 0.918 |
| | Suspicious | 0.933 | 0.953 | 0.933 | 0.935 |
| | legitimate | 0.709 | 0.796 | 0.398 | 0.515 |
| F1-score | Phishy | 0.900 | 0.923 | 0.882 | 0.902 |
| | Suspicious | 0.928 | 0.943 | 0.909 | 0.925 |
| | legitimate | 0.753 | 0.850 | 0.540 | 0.616 |
| AUC | Phishy | 0.847 | 0.895 | 0.696 | 0.751 |
| | Suspicious | 0.916 | 0.935 | 0.902 | 0.919 |
| | legitimate | nan | nan | nan | nan |
| MCC | Phishy | 0.735 | 0.840 | 0.556 | 0.605 |
| | Suspicious | 0.832 | 0.871 | 0.800 | 0.834 |
| | legitimate | 0.849 | 0.880 | 0.807 | 0.842 |

A comparison of the measured and predicted values for the three categories of suspicious, phishing, and legitimate is shown in Fig. 8. The measured value for the Suspicious category is 702. The SG+MO model outperforms the SGDC and NBC models, which both come in at 655, and the NBC+MO model, which comes in at 656, in making the closest prediction at 669. The measured value in the Phishy category is 548. The closest prediction, once again from the SG+MO model, is 505, followed by NBC+MO at 503, SGDC at 495, and NBC at 491. With a measured value of 103, the Legitimate category displays a more notable discrepancy; in this case, the SG+MO model predicts 82, which is greater than SGDC at 73, NBC+MO at 53, and NBC at 41. This graph highlights the SG+MO model's superior performance over the other models by showing that it consistently produces predictions that are closer to the actual measured values across all categories.

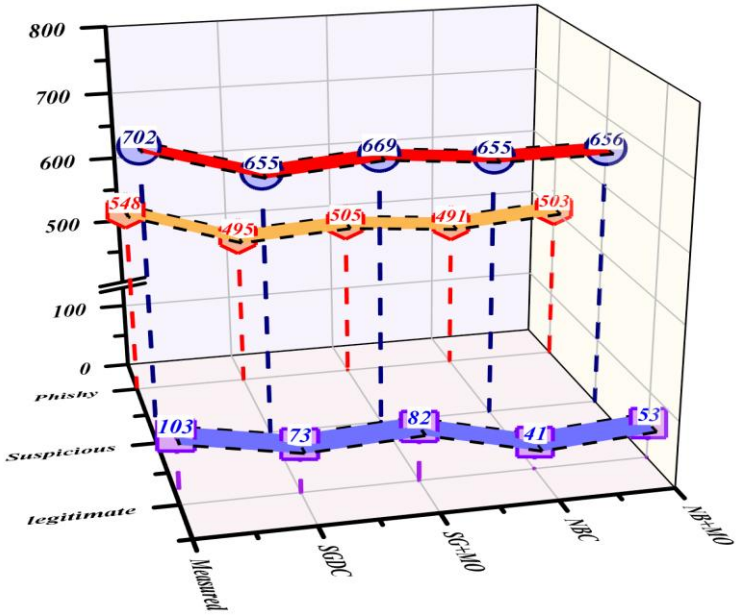



Figure 8: Visualization depicting the performance evaluation of the developed models

The ROC curve results are shown in Fig. 9, which is essential when assessing how well models perform in terms of achieving an actual positive rate of 1. The model that performs better is the one that reaches this rate the quickest. We can see that on the ROC curve plot for the category Phishy, the green line first reaches an actual positive rate of 1, demonstrating its high efficacy in this category for the model. Following that, the ROC curve for the suspicious category and the micro-average ROC curve reaches this rate, which goes on to say that the performance generally is excellent across a variety of classes and particularly good in detecting suspicious instances. The macro-average ROC curve represents the third ROC curve that reaches an actual positive rate of one. It provides an objective evaluation of the effectiveness of the model across all categories. Finally, the ROC curve for the legitimate category approaches this rate, showing its ability to classify the legitimate ones correctly. These ROC curves collectively present the relative efficaciousness of the models and their advantages within a broad spectrum of classification tasks.

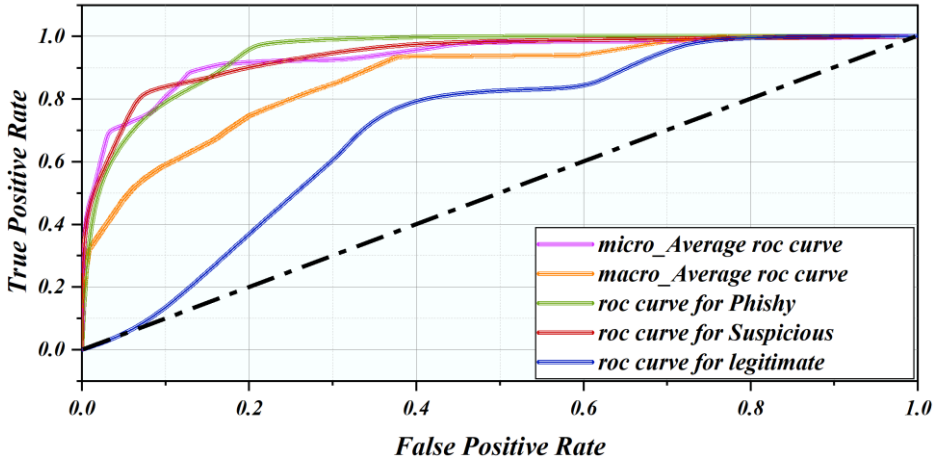



Figure 9: ROC curves illustrating the performance of the most effective hybrid models

**Run time comparison**

The computational efficiency of the models was also evaluated by recording their execution times. The runtime results indicate that the standalone Naïve Bayes Classifier (NBC) and Stochastic Gradient Descent (SGD) models exhibited significantly lower execution times, with 0.57 seconds and 0.87 seconds, respectively. However, when integrated with the MOA framework, the computational cost increased considerably, with NBC_MOA requiring 598.5 seconds and SGD_MOA taking 913.5 seconds. These findings highlight the trade-off between optimization and computational efficiency, suggesting that while the hybrid models may improve prediction accuracy, their feasibility for real-time applications depends on the available computational resources.

## 6.3 Sensitivity analysis based on FAST and SHAP

As its name suggests, fast sensitivity analysis provides a quick way of determining how changes in the input parameters will affect a model's output. Highlighting such important variables is crucial to feature selection, model performance enhancement, and, importantly, model complexity reduction. Sensitivity analysis enhances interpretability by reducing the model to just the most important features while informing data collection activities. Moreover, it offers a straightforward approach toward an iterative model enhancement process that ensures good adaptation and optimization without high computational loads.

The pie chart segments in Fig. 10 give a comprehensive look into the different features and each of their contributions to the model in identifying phishing attempts. The highest ranking, with 35.8% and in teal, is the URL Anchor segment, which is highly sensitive and influential. This feature is critical because it helps identify if the anchor text matches the actual URL, which is a general strategy that phishers use. The second most salient feature is the requested URL, in light teal color, making up 18.7%. This shows how important it is to examine the requested URL for deviations from expected patterns, as these can be a strong indicator of phishing attempts. The light brown (at 18%) SFH (Server Form Handler) feature is also essential because it deals with form submissions on a server, which is a crucial component in identifying phishing sites. Given that phishing sites frequently use longer, more complicated URLs to mask their true nature, URL length (dark teal, 8.3%) is deemed to be moderately significant. Another moderate factor is having an IP address (light brown, 7.4%), as legitimate websites hardly ever use raw IP addresses in URLs. Prefix/Suffix (dark brown, 4.8%) and subdomain (light orange, 3.4%) offer extra indications, but their significance is less than that of the leading features. Domain Age (light teal, 0.8%) has the least bearing because while newer domains can be more suspicious, it's not a very strong indicator. Web Traffic (dark orange, 3.3%) indicates that lower-traffic sites are more suspicious.

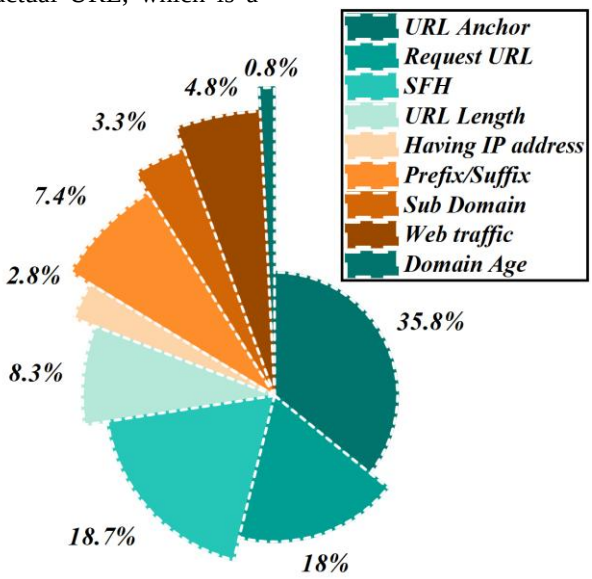



Figure 10: Sensitivity analysis conducted using FAST for the best model

The results of the SHAP sensitivity analysis summarized in Table 5. The SHAP sensitivity analysis ranks input features based on their average influence across three phishing categories: *Phishy*, *Suspicious*, and *Legitimate*. The URL Anchor feature has the highest overall impact (AVG = 1.1625), indicating its strong influence on classification decisions, particularly in distinguishing *Phishy* and *Legitimate* websites. SFH (Server Form Handler) follows as the second most significant feature (AVG = 0.8308), suggesting its relevance in identifying phishing threats. Request URL ranks third (AVG = 0.6089), highlighting its role in determining website legitimacy.

URL Length shows moderate importance (AVG = 0.5065), being more impactful in *Phishy* and *Suspicious* classifications than *Legitimate*. Having an IP Address (AVG = 0.3826) and Sub Domain (AVG = 0.3733) contribute similarly to predictions, but their influence is lower. The least impactful feature is Prefix/Suffix (AVG = 0.2635), indicating minimal significance in phishing detection.

Overall, the analysis confirms that *URL-based attributes*, particularly *URL Anchor*, *SFH*, and *Request URL*, play a crucial role in phishing classification, whereas domain-related attributes like *Prefix/Suffix* contribute less.

Table 5: Result of SHAP sensitivity analysis.

| Rank | Inputs | Phishy | Suspicious | legitimate | AVG |
|---|---|---|---|---|---|
| 1 | URL Anchor | 1.718901889 | 0.361819914 | 1.406823515 | 1.162515 |
| 2 | SFH | 1.091688147 | 0.292571162 | 1.108129482 | 0.830796 |
| 3 | Request URL | 0.850812044 | 0.156678455 | 0.819171597 | 0.608887 |
| 4 | URL Length | 0.643498015 | 0.613921318 | 0.262064841 | 0.506495 |
| 5 | Having IP address | 0.375322899 | 0.424208786 | 0.348136872 | 0.382556 |
| 6 | Sub Domain | 0.309529929 | 0.46483848 | 0.345516127 | 0.373295 |
| 7 | Prefix/Suffix | 0.17633227 | 0.266295892 | 0.347764885 | 0.263464 |
| 8 | Web traffic | 0.167606507 | 0.065772981 | 0.18652572 | 0.139968 |
| 9 | Domain Age | 0.098777995 | 0.058722238 | 0.051814727 | 0.069772 |

### 6.4 Wilcoxon test

The Wilcoxon test is a non-parametric statistical test used to compare paired or independent samples to determine whether there is a significant difference between them. Unlike parametric tests, it does not assume a normal distribution of data, making it particularly useful for small sample sizes or non-normally distributed datasets. The Wilcoxon signed-rank test is used for paired samples, whereas the Wilcoxon rank-sum test (also known as the Mann-Whitney U test) is applied to independent samples.

In this study, the Wilcoxon test was conducted to compare the performance differences between models, specifically examining the statistical significance of their differences through p-values and test statistics.

Table 6 presents the p-values and test statistics for different models under comparison. The p-value indicates whether there is a statistically significant difference between models, with a typical significance threshold set at 0.05 (i.e., if $p < 0.05$, the difference is considered significant).

- SGD_MOA ($p = 0.265464$, statistic = 2081.5): Since the p-value is greater than 0.05, there is no significant difference in performance when using the SGD_MOA model.
- SGD ($p = 0.753526$, statistic = 4127.5): The high p-value suggests no statistically significant difference, indicating that the SGD model's performance is not substantially different from the others.
- NBC_MOA ($p = 0.917888$, statistic = 4957): This is the highest p-value in the table, confirming that NBC_MOA does not show a significant difference compared to other models.
- NBC ($p = 0.36308$, statistic = 6385.5): Again, the p-value is well above 0.05, indicating no statistically significant performance difference.

Table 6: Result of Wilcoxon test.

| Difference of models | Parameter | |
|---|---|---|
| | p_value | statistic |
| SGD_MOA | 0.265464 | 2081.5 |
| SGD | 0.753526 | 4127.5 |
| NBC_MOA | 0.917888 | 4957 |
| NBC | 0.36308 | 6385.5 |

## 7 Discussion

### 7.1 Limitations of the study

Despite the promising results of the hybrid phishing detection models, this study has certain limitations. First, the dataset used, while diverse, may not fully capture the constantly evolving nature of phishing websites. Cybercriminals frequently adapt their techniques, introducing new evasion strategies that may reduce model effectiveness over time. Additionally, the study relied on a dataset with a limited number of features (nine input parameters), which may not fully encompass all relevant attributes influencing phishing website classification. Expanding the feature set could enhance model robustness and generalizability.

Another limitation lies in the reliance on MOA for hyperparameter optimization. While MOA demonstrated superior performance compared to traditional techniques, it may not always guarantee the absolute best hyperparameters, as optimization outcomes depend on initial conditions and algorithm-specific parameters. Further research could explore hybrid or ensemble optimization strategies to enhance performance.

Furthermore, real-time deployment of the models was not tested in a live cybersecurity environment. The study primarily focused on offline classification accuracy, leaving questions about computational efficiency and adaptability in real-world phishing detection scenarios. Future studies should evaluate the models' performance in real-time detection systems to assess latency, adaptability, and scalability.

### 7.2 Benefits of each hybrid model

Each hybrid model presented distinct advantages that contribute to phishing detection performance. The SGD+MOA model emerged as the most effective, achieving the highest accuracy of 92.1%. This model benefits from the ability of SGD to process large-scale datasets efficiently while adapting quickly to new phishing patterns. The integration of MOA further improved convergence and predictive capability by fine-tuning hyperparameters, making it a strong choice for real-time phishing detection applications.

On the other hand, the NBC+MOA model, while exhibiting slightly lower accuracy (87.7%), offered advantages in terms of interpretability and computational efficiency. The probabilistic nature of NBC allows for clearer decision-making processes, which can be useful in cybersecurity applications where transparency is essential. Additionally, NBC requires fewer computational resources compared to complex deep learning models, making it a viable option for systems with hardware constraints.

By combining machine learning models with bio-inspired optimization, both hybrid models demonstrated significant improvements over their standalone counterparts. The results indicate that optimization algorithms such as MOA can substantially enhance model accuracy, reduce misclassification rates, and provide more reliable phishing detection systems.

### 7.3 Practical implications of the study

The findings of this study have several practical implications for cybersecurity, particularly in enhancing phishing detection mechanisms. The demonstrated effectiveness of hybrid models suggests that organizations can integrate such approaches into their cybersecurity frameworks to improve email filtering systems, web security tools, and fraud detection software. By leveraging SGD+MOA, companies can deploy real-time phishing detection models that efficiently adapt to evolving cyber threats, reducing the risk of data breaches and identity theft.

For organizations with resource constraints, the NBC+MOA model offers a practical alternative, ensuring phishing detection with lower computational overhead while maintaining strong classification performance. This is particularly beneficial for small businesses, financial institutions, and government agencies that require cost-effective cybersecurity solutions.

Moreover, the study highlights the importance of bio-inspired optimization in machine learning applications, reinforcing the idea that intelligent optimization techniques can enhance traditional classification models. As phishing tactics become more sophisticated, continuously improving detection models through advanced optimization strategies can provide a proactive defense against cyber threats.

Future implementations could focus on integrating these models with browser extensions, email security tools, and AI-driven security systems to enhance real-time phishing detection. Additionally, regulatory bodies and cybersecurity professionals can utilize the findings to establish more effective security policies, training programs, and automated detection mechanisms to combat phishing attacks more effectively.

## 8 Conclusion

Phishing is a severe risk to both individuals and organizations because it uses deceptive tactics to trick victims into disclosing private information like passwords, bank account information, or personal information. Cybercriminals frequently use phishing tactics to trick unsuspecting victims into unintentionally disclosing their personal information using phishy emails, websites, or messages. Identity theft, money loss, and security compromise are possible outcomes of this.

ML is essential for securing personal information from phishing attacks. Using advanced algorithms and predictive models, ML may analyze data patterns to identify or flag potential phishing websites or suspicious activities. These models learn by identifying subtle cues or anomalies from enormous volumes of data indicative of a phishing attempt and provide proactive defense against cyber threats.

It was derived from the study that the MOA, combined with ML models, such as NBC and SGD, were able to classify phishing websites more accurately. The hybrid models derived showed a marked increase in the differentiation of authentic fraudulent websites by optimizing their predictive capability and reducing error rates.

The results showed that MO increased the accuracies of the SGD and NBC models by 2.65% and 2.17%, respectively. With the lowest error rates, the resultant hybrid model, namely SG+MO, turned out to be the most effective. In particular, out of 548 instances, it correctly predicted 92.15% of phishy websites while labeling 39 as suspicious and four as legitimate. The model misclassified 29 as phishy and four as legitimate out of 702 instances, hence giving an actual prediction rate of 95.3% for suspicious websites. The model achieved an actual prediction rate of 79.61% when it came to legitimate websites, misclassifying 12 as phishy and nine as suspicious out of 103 instances.

Future research could examine bigger and more varied datasets and more optimization strategies to improve model performance and further improve cybersecurity measures. Enhancing the precision and effectiveness of phishing detection systems can also involve addressing feature engineering and selection techniques.

## Competing interests

The scholars claim no competing interests.

## Authorship contribution statement

Xiao CHEN: Writing-Original draft preparation Conceptualization, Supervision, Project administration.

## Data availability

The scholars will make the raw data supporting this article's conclusions available without undue reservation.

## Declarations

Not applicable.

## Author statement

The manuscript has been read and approved by all the authors, the requirements for authorship, as stated earlier in this document, have been met, and each author believes that the manuscript displays honest work.

## Ethical approval

All scholars have been personally and actively involved in substantial work leading to the paper and will take public responsibility for its content.

# References


[1] Lastdrager EEH. Achieving a consensual definition of phishing based on a systematic review of the literature. Crime Sci 2014;3:1–10.

[2] Mohammad RM, Thabtah F, McCluskey L. Tutorial and critical analysis of phishing websites methods. Comput Sci Rev 2015;17:1–24.

[3] Garera S, Provos N, Chew M, Rubin AD. A framework for detection and measurement of phishing attacks. Proceedings of the 2007 ACM workshop on Recurring malcode, 2007, p. 1–8.

[4] Khonji M, Iraqi Y, Jones A. Phishing detection: a literature survey. IEEE Communications Surveys & Tutorials 2013;15:2091–121.

[5] Varshney G, Sardana A, Joshi RC. Secret information display based authentication technique towards preventing phishing attacks. Proceedings of the International Conference on Advances in Computing, Communications and Informatics, 2012, p. 602–8.

[6] Hong J. The state of phishing attacks. Commun ACM 2012;55:74–81.

[7] Pandove K, Jindal A, Kumar R. Email spoofing. Int J Comput Appl 2010;5:27–30.

[8] Jagatic TN, Johnson NA, Jakobsson M, Menczer F. Social phishing. Commun ACM 2007;50:94–100.

[9] Almomani A, Gupta BB, Atawneh S, Meulenberg A, Almomani E. A survey of phishing email filtering techniques. IEEE Communications Surveys & Tutorials 2013;15:2070–90.

[10] Drake CE, Oliver JJ, Koontz EJ. Anatomy of a Phishing Email. CEAS, 2004.

[11] Varshney G, Misra M, Atrey PK. A survey and classification of web phishing detection schemes. Security and Communication Networks 2016;9:6266–84.

[12] Varshney G, Joshi RC, Sardana A. Personal secret information based authentication towards preventing phishing attacks. Advances in Computing and Information Technology: Proceedings of the Second International Conference on Advances in Computing and Information Technology (ACITY) July 13-15, 2012, Chennai, India-Volume 1, Springer; 2012, p. 31–42.

[13] Gupta S, Kumar P. A desktop notification based scheme for preventing online frauds attempts to cloud users S| pp| S. 2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI), IEEE; 2013, p. 255–60.

[14] Workman M. Wisecrackers: A theory-grounded investigation of phishing and pretext social engineering threats to information security. Journal of the American Society for Information Science and Technology 2008;59:662–74.

[15] Bergholz A, De Beer J, Glahn S, Moens M-F, Paaß G, Strobel S. New filtering approaches for phishing email. J Comput Secur 2010;18:7–35.

[16] Islam R, Abawajy J. A multi-tier phishing detection and filtering approach. Journal of Network and Computer Applications 2013;36:324–35.

[17] Das A, Baki S, El Aassal A, Verma R, Dunbar A. SoK: a comprehensive reexamination of phishing research from the security perspective. IEEE Communications Surveys & Tutorials 2019;22:671–708.

[18] Aburrous M, Hossain MA, Dahal K, Thabtah F. Predicting phishing websites using classification mining techniques with experimental case studies. 2010 seventh international conference on information technology: New generations, IEEE; 2010, p. 176–81.

[19] Ramesh G, Krishnamurthi I, Kumar KSS. An efficacious method for detecting phishing webpages through target domain identification. Decis Support Syst 2014;61:12–22.

[20] Abdelhamid N. Multi-label rules for phishing classification. Applied Computing and Informatics 2015;11:29–46.

[21] Phishing Data Phishtank n.d. https://www.kaggle.com/datasets/latheeshmangeri/phishing-data-phishtank.

[22] Bottou L. Large-scale machine learning with stochastic gradient descent. Proceedings of COMPSTAT'2010: 19th International Conference on Computational StatisticsParis France, August 22-27, 2010 Keynote, Invited and Contributed Papers, Springer; 2010, p. 177–86.

[23] Ahn S, Korattikara A, Liu N, Rajan S, Welling M. Large-scale distributed Bayesian matrix factorization using stochastic gradient MCMC. Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining, 2015, p. 9–18.

[24] Manning CD. An introduction to information retrieval. Cambridge university press; 2009.
[25] Rosen G, Garbarine E, Caseiro D, Polikar R, Sokhansanj B. Metagenome Fragment Classification Using ?-Mer Frequency Profiles. Adv Bioinformatics 2008;2008.
[26] Han J, Kamber M. Data mining concepts and techniques San Francisco Moraga Kaufman 2001.
[27] Pekuwali AA, Kusuma WA, Buono A. Optimization of Spaced K-mer Frequency Feature Extraction using Genetic Algorithms for Metagenome Fragment Classification. Journal of ICT Research & Applications 2018;12.
[28] Kotsiantis S, Patriarcheas K, Xenos M. A combinational incremental ensemble of classifiers as a technique for predicting students' performance in distance education. Knowl Based Syst 2010;23:529–35.
[29] Zervoudakis K, Tsafarakis S. A mayfly optimization algorithm. Comput Ind Eng 2020;145:106559.
[30] Bhattacharyya T, Chatterjee B, Singh PK, Yoon JH, Geem ZW, Sarkar R. Mayfly in harmony: A new hybrid meta-heuristic feature selection algorithm. IEEE Access 2020;8:195929–45.
[31] Kadry S, Rajinikanth V, Koo J, Kang B-G. Image multi-level-thresholding with Mayfly optimization. International Journal of Electrical & Computer Engineering (2088-8708) 2021;11.
[32] Raja NSM, Rajinikanth V, Latha K. Otsu based optimal multilevel image thresholding using firefly algorithm. Modelling and Simulation in Engineering 2014;2014:37.