Sigma J Eng Nat Sci, Vol. 43, No. 3, pp. 1020–1037, June, 2025

Sigma Journal of Engineering and Natural Sciences
Web page info: https://sigma.yildiz.edu.tr
DOI: 10.14744/sigma.2025.00089

Research Article

Efficient hybrid approach for feature extraction using metaheuristic algorithm and majority voting in social media networks

Sonali LUNAWAT[1,*], Jyoti RAO[2], Pramod PATIL[2]

[1]Dr. D. Y. Patil Institute of Technology, Pimpri, Pune, 411018, India; PCET's Pimpri Chinchwad College of Engineering and Research, Ravet, Pune, 412101, India
[2]Dr. D. Y. Patil Institute of Technology, Pimpri, Pune, 411018, India

ARTICLE INFO

Article history
Received: 29 March 2024
Revised: 01 July 2024
Accepted: 06 November 2024

Keywords:
Data Mining; Gannet Optimization Algorithm; Machine Learning; Metaheuristic Algorithm; Majority Voting

ABSTRACT

Social media networking sites introduce specific difficulties to the researchers dealing with high-dimensional data. Particularly, this holds true when trying to find a typical users. Irrelevant or redundant features can significantly reduce classifier accuracy, increasing prediction time which in turn diminishes overall model effectiveness. In this respect, feature selection techniques usually are applied to mitigate such obstacles via removing irrelevant features, which in turn increases computational efficiency, improves accuracy, and applies simpler models. However, traditional methods of feature selection are computationally expensive and often come with reduced accuracy in classification since the redundant features may not have been removed-error-prone and generally filter and wrapper-based. While hybrid approaches are more efficient, they sometimes fail to consider interactions between features effectively, which can be complex. We identify the shortcomings of these representatives and propose an optimal hybrid approach that integrates GOA with Majority Voting. Then, the two-step process starts with a feature filter based on an information-theoretic measure that selects 24 features with existing approach out of 79 for the phishing dataset in order to capture the co-evolutionary behavior. GOA follows the second step by applying our hybrid approach selection top 10 most optimal features from GOA, keeping in consideration both maximum relevance and minimum redundancy.The strength of this hybrid approach lies in its versatility, which has been applied successfully across different datasets. We achieved an accuracy of 99.7% on the Phishing dataset, outperforming ten benchmark feature selection methods.This yielded accuracy, which outperformed some of the results of in-fashion classifiers for the KDD dataset, WSN-DS dataset, CICIDS2017 dataset. Obviously, these results assure that this optimized feature set enhances not only the accuracy but also reduces prediction time significantly, hence being very efficient for real-world applications in social media analytics and beyond. This not only advances the state-of-art in feature selection but also sets the ground for further research on how to optimize classifier performance in various domains. The advantages of our hybrid approach not only come from accuracy improvements but also from computational efficiency, making it a practically applicable powerful tool in high-dimensional data analysis.

Cite this article as: Lunawat S, Rao J, Patil P. Efficient hybrid approach for feature extraction using metaheuristic algorithm and majority voting in social media networks. Sigma J Eng Nat Sci 2025;43(3):1020–1037.

***Corresponding author.**
*E-mail address: sonali.lunawat@pccoer.in

This paper was recommended for publication in revised form by Editor-in-Chief Ahmet Selim Dalkilic

INTRODUCTION

Social networking sites are essentially online spaces where individuals may create groups and collaborate to create social networks involving individuals, organisations, and communities. Due to these websites, we are currently dealing with a number of issues like adolescent violence, cyberbullying, and cybercrime. rate, accuracy, and processing overhead, it is particularly challenging to identify anomaly patterns. Due to the numerous challenges researchers face, many developed models struggle to accurately detect anomalies. Something that is out of the ordinary or unexpected is called an anomaly. As more and more risks emerge n a daily basis, experts are forced to consider the safety of humans.

Feature selection (FS) is the process of identifying and selecting the most relevant features from a larger set, aiming to create a subset that best correlates with and influences the model's outcome [1].

Such benefits of feature selection are numerous, impactful, and very relevant to better quality data. The most relevant features will give way more accurate results that can be relied upon. Besides, at model prediction, the computational load reduces, thus enabling the process to be more efficient and faster. Feature selection also streamlines data preprocessing to turn it into a more effective and manageable task. Ultimately, this will end up building a more accurate model by turning attention to highly influential variables and reducing the complexity of the model, hence improving its interpretability and maintainability. Complex dimensional data lead to noisy, irrelevant, and redundant data, resulting in model performance degradation, which causes overfitting by increasing the error rate of the algorithm. To solve this problem, we coined the term "Dimensionality reduction," which is an important pre-processing stage. Dimensionality Reduction is categorized into Feature Selection (FS) and Feature Extraction (FE). FS is a cleanup approach for boosting model performance. The Types of Feature Selection techniques Classification of various dimensionality reductions is shown in Figure 1 above.

Feature Selection

Another important difference between the two techniques is how they work: while feature extraction creates a new modified feature space by transforming the original features, feature selection reduces the original number of features.

Feature Transformation

In which from original features selected informative and relevant features forms a new set of features,Feature extraction has a number of very useful benefits, such as in training a machine-learning model. It increases the accuracy of the model by better concentrating on only the most important aspects of the data, thereby avoiding issues with overfitting. This also brings about better efficiency in data visualization, making it easier to truly understand and interpret the data. With added speed in training, models can be built and deployed at much faster rates.

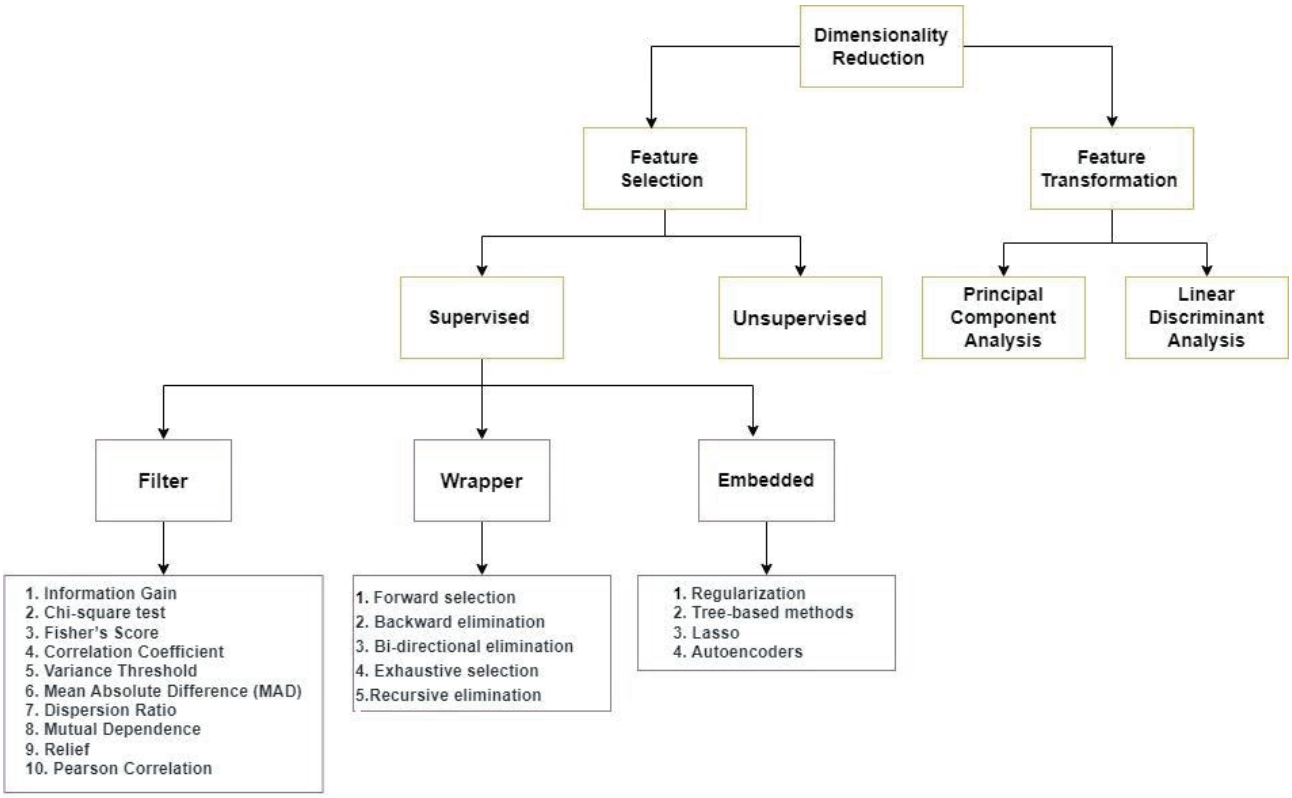

Figure 1. Classification of various feature selection techniques.

Supervised feature selection operates using the label information principle, selecting significant and pertinent elements from data by analysing it using information gain or the Gini index. The result is the best possible feature selection for a classifier's training [2].

Filter method

Works on the principle that information about the selected features based on their scores in various numerical tests and relevant features are considered by their correlation with a dependent variable [2].

Wrapper method

Works on the principle based on correlation of the features and the labelled class by considering the dependency with other features. algorithm can be optimized by considering a bias decision. The wrapper technique is associated with a high risk of overfitting. According to researchers' conclusions, it provides a more accurate result than the filter method [2].

Embedded method

Operates according to the idea of fusing the benefits of the filter and wrapper approaches . This approach takes into account a number of factors and is both quicker and more accurate than filter approaches. The chosen classification algorithm determined the best feature subset. When considering the complexity of the processing, it is more efficient [2].

Unsupervised feature selection [2] does not have label information regarding feature relevance but will consider descriptions to select related features.

Majority Voting

In particular, the majority voting principle is utilized to determine whether the feature is to be indexed in the list. In simple words, if the feature is chosen with maximum values, it is decided to be a highly valuable feature repaid with a high score [3-5].

Metaheuristic Algorithms

Optimization methods are those that guarantee the obtaining, at least, of good, if not the best, solutions to optimization problems. The approach of a Metaheuristic algorithm balances two conflicting objectives of exploring the search space and exploiting it for finding near-optimum solutions [6,7]. All FS problems in this study cannot be effectively addressed using a Metaheuristic-based approach alone. A new version of Metaheuristic algorithms that more properly balances the Exploration-Exploitation could be introduced as an optimization technique in order to solve the feature selection problem for enhancing efficiency. This motivation drives our work in developing a hybrid machine learning model that addresses specific challenges by minimizing the presence of weak or irrelevant features.

Feature Selection Approach Used in Optimization

In the context of feature selection within optimization, a dataset with n features will have 2^n possible feature subsets. When n is large the problem of feature selection gradually turns into an optimization problem [8]. The core question is how to select a subset of those feature combinations in order to improve the effectiveness of machine model training. This paper approaches the problem of feature selection as a process to select a subset of features from the complete set f of features, as illustrated in Figure 2. The objective function f(X) seeks to select the minimum number of features for which the classifier achieves best performance, providing optimal accuracy at the given point of f(X) is the best optimal optimal solution. Building on this provided intent and previous feature selection techniques, it becomes clear that the combination of Majority Voting with an Optimization algorithm may bring visible improvements in accuracy.

$$Max\ f(X)\ \ is\ achieved\ as\ X\ is\ \{x_1, x_2, , x_F\},\ x_i \in \{0,1\} \quad (1)$$

$$\text{Where} \quad \begin{array}{c} x_i\ \ is\ 0\ means\ feature\ not\ selected \\ and\ x_i\ \ is\ 1\ means\ feature\ is\ selected \\ final_{Subset} = 1,2, \ldots \ldots F \end{array} \quad (2)$$

$$where\ 1 \le \{final_{Subset}\} \le F$$

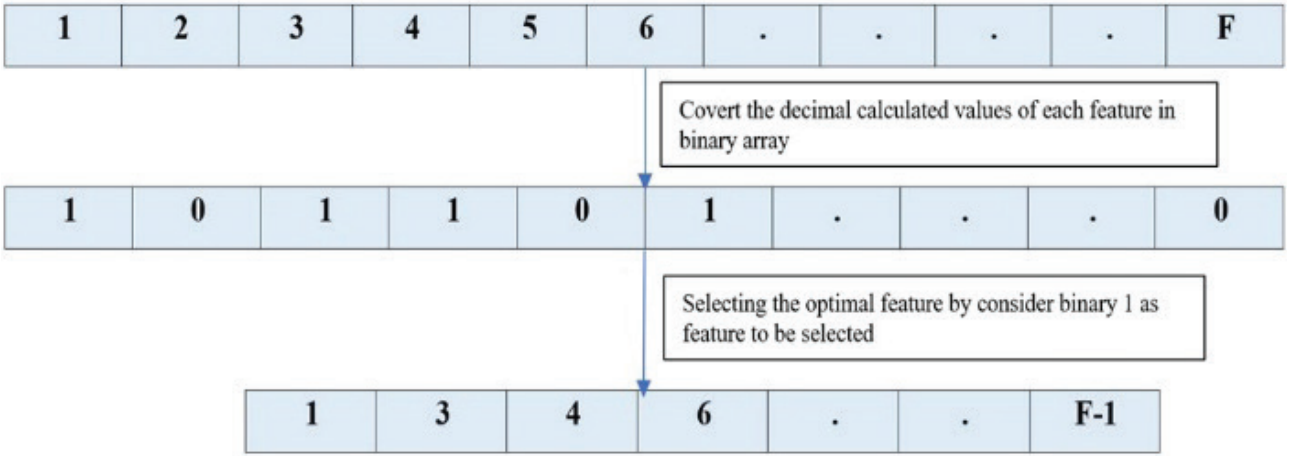

Figure 2. Process of feature Selection as Optimization Problem.

The proposed hybrid approach incorporates algorithms such as linear regression, support vector machine, k-nearest neighbour, Naïve Bayes, Gridboost, which proved to be much more accurate on both phishing and KDD datasets. In this paper, different feature selection methods over the two aforementioned datasets are compared, and it turns out that through the use of a proposed hybrid approach, an accuracy of up to 99.7 % could be obtained on the Phishing dataset and 97.34% on the KDD dataset. Why? Because the results are obtained both with the full feature set and with an optimized feature set. It demonstrates that, compared with other techniques of feature selection, the proposed method has high accuracy with a reduced prediction time.

Motivation

Most traditional techniques of feature selection are based on filter and wrapper methods. The main limitation of these methods lie in the problems associated with classification accuracy due to their high computational cost, which arises because of the presence of irrelevant and redundantly selected features. However, on the other hand, although hybrid feature selection methods are computationally very effective, they do not usually take into account interaction between features or influence on other features when one of the features is removed. In order to overcome these deficiencies, a hybrid approach combining majority voting with the two-step feature extraction technique of Gannet Optimization Algorithm has been developed. This approach tries to enhance feature selection by optimizing its accuracy and computational efficiency.

Contributions

Proposed a novel approach

The majority voting-based optimization method was developed by its integration with the Gannet Optimization Algorithm as a metaheuristic.Applied majority voting to information-theoretic measures in a way that considers the mutual behavior of features.

GOA application

GOA was applied to the selected features with the help of Majority Voting to select the best set of top 10 features.

Comparative analysis

Comparing the proposed feature selection method against 10 benchmark methods for impact on accuracy and validate the approach on the Phishing and KDD dataset.

Paper plan

The structure of the work is organized as follows: Section 2 gives an overview of the related works canvassed for this approach, Section 3 explains the proposed workflow along with the hybrid algorithms, Section 4 presents the results, and Section 5 concludes the study.

RELATED WORK

It surveys the current state of the art in feature selection methods, suggesting a number of approaches that are in use by researchers. Table 1 summarizes some of the important contributions from different researchers working in this particular field.

As evident from the summary of feature selection methods described above, there are only a limited number of studies conducted on incorporating Wrapper methods with majority voting and metaheuristic algorithms that have been proven to hold great promise in solving optimization problems in high-dimensional data. Presented below is a literature survey regarding how Metaheuristic algorithms work and their application in the field of feature selection.

According to Mallenahalli et al. [34], FS had the first step, which was required for the identification of only those statistically relevant features so that the prediction ability of classifiers could be improved. Abualigah et al. [35] proposed a feature selection approach for improving document grouping; their approach was based on PSO for enhancing the existing application of Bayesian calibration in building energy modelling. Chen et al. [36] proposed a water wave optimization-based text feature selection method called WWOTFS, which uses the water wave optimization-based feature selection approach. Chaokun et al. [37] used the taboo search and binary chemical reaction optimization hybridization process as one of the four basic reactions. Peng et al. [38] used the Ant Colony Optimization concept with feature selection

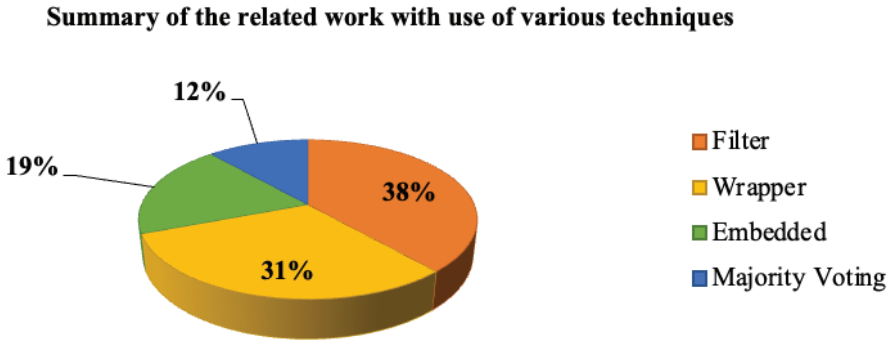

Figure 3. Summarized pie chart for various feature selection techniques.

Table 1. Analysis of various methods used for feature selection

Paper ID	Supervised feature selection type	Method Used	Accuracy	Time (in secs)	Findings/Limitations
[9]	Filter Method	Information Gain	88.50%		A thorough identification of the links between attributes and ranking is required.
[10]		Chi-squared feature selection	85.00%	2.25	Novel integrations of machine learnin algorithms with feature selection techniques, such as wrapper feature selection
[11]		Fisher's exact test, Chi-square, Information Gain, Correlation	98		Examining diverse kinds of datasets is required
[12]		Correlation Coefficient	Average: 90%		-
[13]		Correlation Coefficient	SVM: 98% KNN:99.05	-	The relationship between two variables hasn't entirely disappeared.
[14]		Relief	-		Innovations are essential for choosing the optimal application strategy.
[15]		Variance-Based Feature Selection	Fisher iris: 96%	3.21	improving the procedure and algorithms to use different kinds of data
[17]		Pearson Correlation-Based (Filter)	93.10%	-	Data augmentation can also be used to test and apply additional feature selection techniques.
[19]		Mutual Information	Precision :0.57		To use unsupervised feature selection
[18]	Wrapper Method	Forward-Backward Selection	-		Restricted to choosing the same quantity of variables
[20]		recursive feature elimination	Average :75%		Important considerations in feature selection include feature interaction and feature dependency.
[26]		Wrapper	-	-	It is noted that feature selection algorithms have a stability problem.
[6]		Metaheuristic Algorithm	Average 60 ot 70%		Everything you need to solve the feature selection problem with metaheuristic algorithms is right here.
[32]		Metaheuristic Algorithm	80.15		Since hybrid-based approaches are not thoroughly studied, they need to receive more attention.
[33]		Wrapper	99%		Employ a sophisticated classifier to achieve greater detection
[22]	Embedded Method	LASSO	98%	-	Adaptive approaches are helpful in achieving a better version of feature selection techniques.
[23]		LASSO	-		-
[24]		Autoencoders	91.76	-	to make the weights of the more salient and less salient traits diverge more in magnitude.
[25]		Autoencoders	-	-	-
[16]	Filter, Wrapper and Embedded Method	Hybrid	99.78%	0.78	Create the most appropriate and useful feature set possible from a complex and vast amount of data.
[21]		ExhauFS	65%	59	Our method's primary drawback is its high computational complexity.
[27]	Feature Extraction	PCA	-		In increase in the misclassification rate
[3]	Majority voting	Majority voting	99.5		Compared to other similar methods, it performs better in terms of attack detection accuracy.
[29]		Majority voting	72		Technique either improves the performance of defect detection or maintains.
[30]		Majority voting	-		Examine the potential for creating a unified framework by utilizing search and voting techniques.

in the FACO technique that seems to be an enhanced FS algorithm. The increase in feature sets and network data brings other security threats to networks, such as DDoS and APT. Tubishat et al. [39] improved the swarm method by solving the feature selection problem by the integration of a local search algorithm with a reverse science strategy. Hamidzadeh et al. [40] applied the cuckoo search algorithm in order to address the feature selection problem by including the opposite learning and destruction operators. Le Wang et al. [8] presented a new feature selection algorithm, CBCSEM—an improved Cuckoo Search Algorithm. Mohammad et al. [41] presented an improved edition of the Butterfly Optimization Algorithm for overcoming the challenge of feature selection.

Summary of related work

Popular classifiers now in use provide greater accuracy for smaller datasets, and there is still room to improve efficiency in classifier development. To enhance the accuracy of intrusion detection, it is recommended to create novel techniques to enhance model performance. Feature selection by considering mutual behaviour and extracting optimal features without compromising accuracy and reduction in time for model prediction remain challenging.

3. PROPOSED METHODOLOGY

As shown in figure 4 is the various steps carried to get the subset of best feature subset.

Step 1

Phishing Dataset have two types in this data set: malevolent (Phishing) and normal. The data set has 79 features. accessible on March 20, 2022, and contributed by Yazdi and his team [42]. Phishing Dataset contain 4,004 instances in total and 79 attributes in the phishing dataset. Each instance corresponds to a web resource that has been analyzed for its content and collection of phishing characteristic properties. The attributes are combined HTML and JavaScript characteristics, URL and domain characteristics, and content-based indicators in the identification of phishing attempts. It is labeled: each instance says whether it is phishing or not, an ideal situation for training and testing a machine learning model for phishing detection.

The dataset, KDD [48], provided as part of your file, is a very huge dataset that was designed to be used for intrusion detection systems assessment. It contains 494,020 entries and 42 features. Each entry of the dataset stands for a network connection described by a set of different attributes, denoting different properties of the connection's behavior. Among these attributes are the numerical features: 'duration', 'src_bytes', 'dst_bytes', etc.; 'count', 'serror_rate', 'rerror_rate'; and many others. Other examples of categorical features in the data set are protocol_type, service, and flag—features indicating the type of protocol, network services on the destination, and status flag respectively.

WSN-DS[51] dataset consist of 24 features and catogories of attacks are Blackhole, Grayhole, Flooding, and Scheduling among normal network traffic.

CICIDS2017[50] dataset that contains benign and seven attacks of network flows also provides details about port numbers, source and destination addresses, timestamps, and detected attack.

Step 2

Data cleaning is an important stage in the process of machine learning that involves the detection and elimination of duplicate, irrelevant, missing data, and so on. One of the major reasons for cleaning the data is making sure that a dataset is reliable, constant, and error-free, because if the data is at different levels, then its performance in a machine learning model might be reduced.

Step 3

Classifier anomaly detection is subsequently conducted using various classifier techniques. LR is a supervised machine learning technique used to find the linear relationship between dependent and one or more than one independent variables; the independent variables are considered as features [43]. K-nearest Neighbours (KNN), typically used for regression or classification, has an essential hyper-parameter, k, that defines the size of the prediction neighbourhood. One of the primary reasons for selecting KNN, however, is that it's a non-parametric algorithm; thus, it has no assumption regarding the distribution of data [43][49]. Support Vector Machine is another technique within a range of applications suitable for classification, regression, and outlier detection tasks. An SVM operates on the principle of finding a hyperplane that best separates data into binary groups. The class margin is set in this through distance calculations by the hyperplane from the nearest data points for each class [43]. Naive Bayes: Drawing its drive from Bayes' theorem for estimating the probability of evidence on the likelihood of a hypothesis, NB has been primarily used as a solution for classification problems. It assumes that the features are conditionally independent given any class label, and based on the training data, computes a probabilistic model of the likelihood of different features occurring for each class label. GridBoost is also applied as a more accurate classifier compared to the other techniques above. It integrates XGBoost with hyperparameter tuning by Grid search, improving overall classification performance [44][45][46].

Step 4: Feature Selection Techniques

1. Chi statistic (Chi) measures the goodness of fit for what is expected versus observed in categorical variables based upon a random sample; it is used in a statistical test called the chi-square test.

2. Pearson correlation coefficient (PCC) is a measure of the degree of linear correlation between two variables; its values range from -1 to 1. Here, -1 stands for a perfect negative linear correlation; 0 means no correlation, and +1 means perfect positive correlation.
3. Recursive feature elimination (RFE) elimination is a feature selection technique that considers the fitting of the model and recursively excludes those features that are the weakest until it reaches the specified number of features to be selected.
4. Forward sequential selection (FSS) represents 'forward sequential selection,' a form of forward selection where one variable is selected based on some criterion, and then at every subsequent step, the parameter that best satisfies the criterion is added.
5. Lasso stands for Least Absolute Shrinkage and Selection Operator. This is used in linear regression and other associated tasks of the feature selection model, fitting it by adding a penalty term to the cost function of the linear regression. It incites shrinking in coefficients of features not relevant enough; these hence turn into zero, resulting in them being removed from the model, thus leaving sparse solutions including only the most important features.
6. ANOVA-F belongs to the field of inferential statistics and is used to compare differences among several groups using their means; that is, it tries to find out if there are significant differences between group means.
7. Correlation-based feature selection (CBFS) is a filter method that is unaffected by the final classification model and evaluates feature subsets based on data intrinsic properties, specifically correlations. The goal is to identify a subset with low feature-feature correlation and high feature-class correlation, thereby retaining or enhancing predictive power.
8. Information gain uses a given value for a random variable in splitting a dataset and calculates reduction in the amount of entropy—in other words, the surprise—so that lower entropy in a group means higher information gain.
9. Principal component analysis is used as the primary tool while a large set of data is to be considered with many dimensions or features per observation. It enhances interpretability, gives maximum information, and makes multidimensional data visualization easier.

Step 5: Feature Selection Using Information-Theoretic Measures Fusion with Majority Voting

The fusion process of information-theoretic measures, Use of majority voting improves the feature selection using information-theoretic measures. That is, the results obtained from different information-theoretic measures, such as entropy, conditional entropy, relative entropy, relative conditional entropy, and information gain, will be fused in a process based on the majority voting principle. These information-theoretic measures are important in generating and identifying appropriate anomaly detection systems [31]. More specifically, among the identified measures within this process are entropy, conditional entropy, relative entropy, relative conditional entropy, and information gain, as illustrated in Figure 5. The input social network data A_n with dimension $i * j$ is taken as input. The used output is implied by T_n dimension $i * q$, such that,

$$\mathrm{T_n} \in \{\mathrm{T_1}, \mathrm{T_2}, \mathrm{T_3}, \mathrm{T_4}, \mathrm{T_5}\} \tag{3}$$

a) Entropy is referred to fundamental concept of an information theory that detects unexpected or anomalous of data item collections. For the dataset $\boldsymbol{A}$, wherein individual data item that belongs to class $\boldsymbol{z} \in \boldsymbol{L}_z$ entropy of $\boldsymbol{A}$ relation to this $|\boldsymbol{L}_z|$ wise classification can be defined by,

$$\mathrm{M(A)} = \sum_{\mathrm{z}\in \mathrm{L_z}} \mathrm{P(z)} \log \frac{1}{\mathrm{P(z)}} \tag{4}$$

Here, $P(z)$ represents probability of z in A and z indicates features. After computation of entropy, it chooses q features with low entropy value and thus, it is indicated as T_1 with dimension $i * q$.

b) Conditional entropy for anomalous detection, conditional entropy can be utilized as the measure of sequential dependency reliability. The conditional entropy of A given K as entropy of probability distribution $P(z|k)$ can be given as follows,

$$M(A|K) = \sum_{z,k\in L_z,L_k} P(z,k)\, log\, \frac{1}{P(z|k)} \tag{5}$$

Here, $P(z,k)$ denotes joint probability of z and k whereas $P(z|k)$ refers to conditional probability of z given k. z Implies candidate feature and k indicates target. Afterwards computation of conditional entropy, it selects q features with smaller values for each feature and can be represented by T_2 with dimension $i \times q$, where $j > q$.

c) The relative entropy calibrates a distance of regularities amongst two databases. The relative entropy among two probability distributions $p(z)$ and $t(z)$can be modelled over a same $z \in L_z$is,

$$T_3(p|t) = \sum_{z\in L_z} p(z)\, log\, \frac{p(z)}{t(z)} \tag{6}$$

Thereafter computing relative entropy for individual feature, it selects q feature with small values. The relative entropy is denoted as T_3 with dimension $i \times q$, where $j > q$.

d) The relative conditional entropy is defined as entropy among two probability distributions. The relative conditional entropy among two probability distributions $p(z|k)$ and $t(z|k)$ which are determined over same $z \in L_z$and $k \in L_k$ can be illustrated by,

$$T_4(p|t) = \sum_{z,k\in L_z,L_k} p(z,k)\, log\, \frac{p(z|k)}{t(z|k)} \tag{7}$$

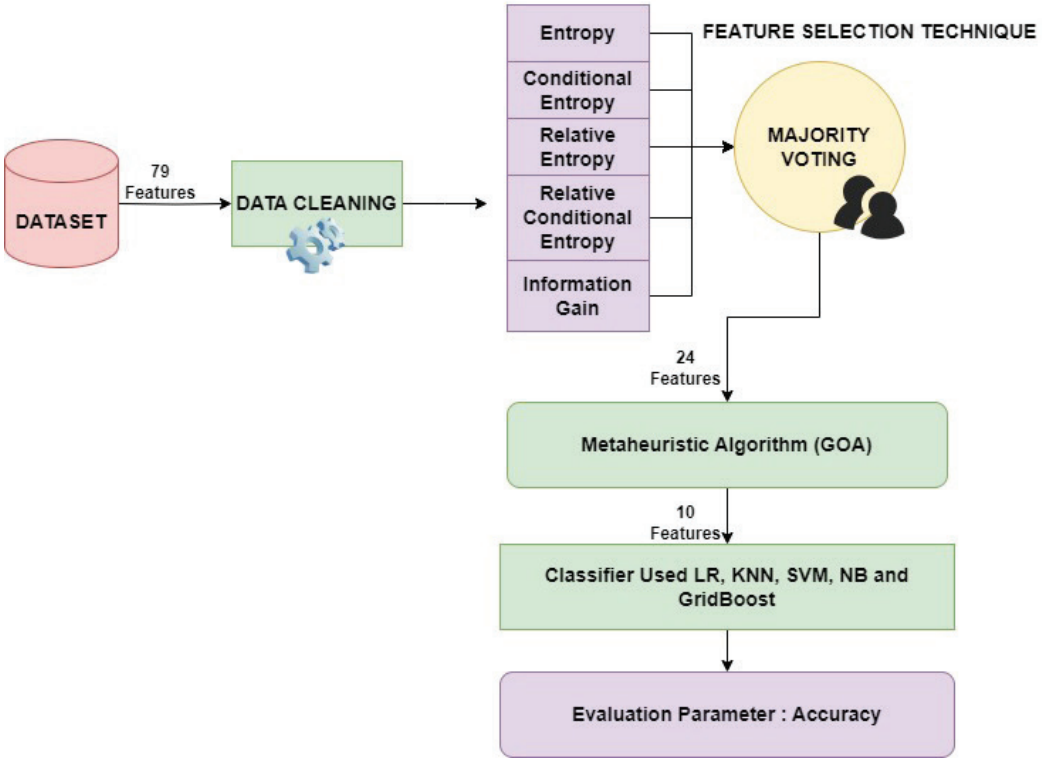

Figure 4. Proposed workflow.

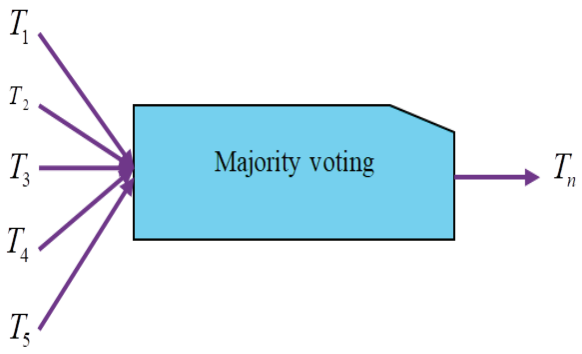

Figure 5. Fusion utilizing majority voting.

After the computation of relative conditional entropy, it chooses q features with low relative conditional entropy value. The relative conditional entropy value is specified as T_4 with dimension $i \times q$.

e) An information gain is stated as reduction of the entropy while a database is partitioned in accordance to feature values. It can be expressed as follows,

$$T_5(A, G) = M(A) - M(A/G) \quad (8)$$

Here, $M(A)$ symbolizes entropy of class label whereas $M(A/G)$ indicates probability of class A for a given feature G. A denotes class label and G indicates feature. Thereafter, computation of information gain, it selects q feature with small values for each feature. The information gain can be signified as T_5 with dimension $i \times q$.

Step 6: Gannet Optimization Algorithm

The Gannet Optimization Algorithm (GOA) [28] is a newly invented Metaheuristic algorithm, taking inspiration from how Gannets naturally forage for food. Mathematically, the GOA models major distinctive actions of any gannet to explore and exploit any search space. GOA applies U-shaped and V-shaped diving patterns in the exploration of optimum regions while sudden shifts and random walks help in finding better solutions. The algorithm has two phases: an exploration phase, where the algorithm broadly explores the space with dive patterns, and an exploitation phase, where solutions are refined through sudden rotations and random walks. Then, select position update formulae based on catching ability. This process starts by randomly generating a set of initial solutions, iteratively updated using one of the four position update algorithms. It alternates between the exploration and exploitation procedures with equal probability in refining solutions until an optimal or near-optimal result is reached, as shown in Figure 6. Through its balanced approach, GOA will effectively navigate across a search space to find high-quality solutions.

In the described algorithm, different phase allows the algorithm to explore different positions of individuals-potential solutions-in the search space.

Random number generation

In each iteration, a random number as *rand is selected* between 0 and 1 to update the position of the individual.

Position update based on random number

If *rand >= 0.5: where a1* is randomly set within the range [*-x, x*]and *a2* depends on two parameters, which are the difference between the position of the current individual M_{cur} and a randomly selected population individual $M_{run}(pos)$ update the current position of the individual accordingly.

else *rand < 0.5*: Then, *b1* and *b2* will be used to update the position by the current individual. *b1* is any random value in the range of [*-y, y*] and *b2* depends on the difference of the current individual's position M_{cur} concerning the average position of the population $M_{avg\text{-}pos}$.

Mathemathical explanation

Randomness in position updating for flexibility, providing an opportunity to an individual to explore different parts of the search space, where *X* and *Y* are random values calculated as *(2 * rand_1 - 1) * x* and *(2 * rand_1 - 1)*

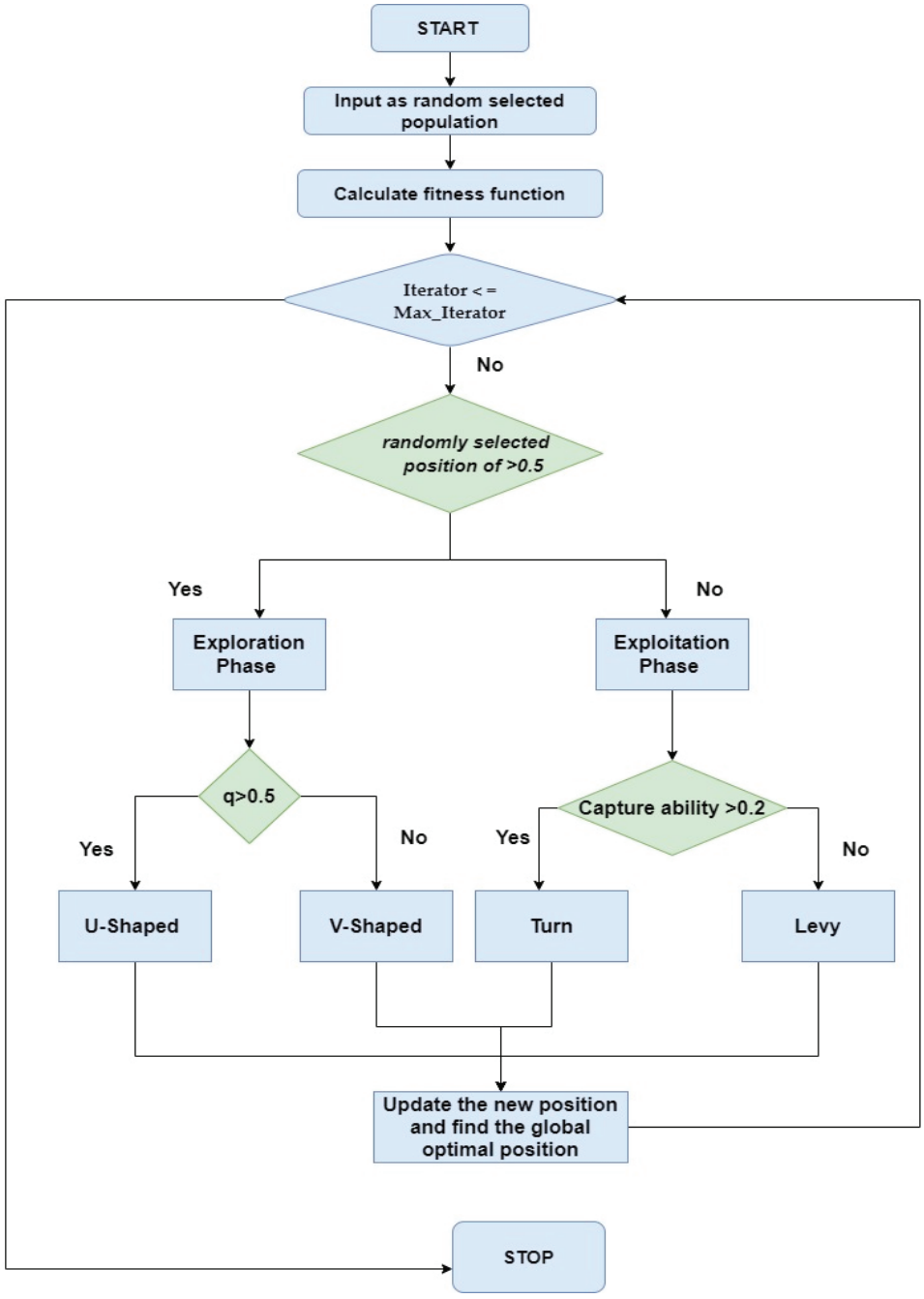

Figure 6. Flowchart of the proposed hybrid feature selection method.

Algorithm of proposed hybrid approach

Input: *Read Phishing Dataset*

Output: *Optimal Feature Set*

1. *Calculate values entropy, conditional entropy, relative entropy, relative conditional entropy, and information gain and store in* ***separate and sorted array as per eq. 4 to 8.***
2. *Using eq. 3 select the highest voting features from step 1.*
3. *Generate random population subsets of size 10 each.*
4. *Initialize random solutions as Matrix M and create a copy of same in TM which is Temporary Matrix*
5. *Calculate fitness using* $sum \mathrel{+}= solution[i][j]$ *where* i *and* j *are iterator till* N
6. $pos = 1 - Iterator/Max\_Iterator$
7. *while (Iterator < = Max_Iterator):*
8. *if rand > 0.5 where rand is a random number between 0 and 1 at each change of position of the gannet individuals will be stored in the temporary matrix TM. After evaluation by the fitness function as using step 5, if the temporary matrix* TM_{cur} *individual performs > current solution* M_{cur} *individual, then* TM_{cur} *is used instead of* M_{cur} *;else solution in the M matrix is used.*
9. *Assuming x, y = 5, 10*
10. *a1 = any value from -x, x*
11. *b1 = any value from -y, y*
12. *X = (2 * rand_1 - 1) * x*
13. *Y = (2 * rand_1 - 1) * y, where rand_1 is any random value between 0 & 1*

Exploration phase:

14. *if rand_con >= 0.5:*
    $TM_{cur}(pos+1) = M_{cur}(pos) + a1 + a2$
15. $a2 = X * \left(M_{cur}(pos) - M_{ran}(pos)\right)$, *Where* $M_{cur}(pos)$ *is the ith individual in the current population,* $M_{ran}(pos)$ *is a randomly selected individual in the current population*
16. *else:*
    $TM_{cur}(pos+1) = M_{cur}(pos) + b1 + b2$
17. $b2 = Y * \left(M_{cur}(pos) - M_{avg\_pos}(pos)\right)$, *Where* $M_{cur}(pos)$ *is the ith individual in the current population,* $M_{avg\_pos}(pos)$ *average position of individuals in the current population*
18. *end if*

Exploitation phase:

19. *else*
20. $Capture_{ability} = \frac{1}{[2.5\ kg * 1.5\frac{m^2}{s}/(0.2+(2-0.2)*r]*pos}$, *where 2.5 kg is weight of gannet and 1.5 m/s is speed of the gannet and r is any random value between o and 1 and pos is as calculated in step 6*
21. $\Delta = Capture_{ability} * |M_{cur}(pos) - M_{best}(pos)|$
    if $Capture_{ability}$ *>= 0.2:*
22. $TM_{cur}(pos+1) = pos * \Delta * \left(M_{cur}(pos) - M_{best}(pos)\right) + M_{cur}(pos)$
23. *else*
    $TM_{cur}(pos+1) = M_{cur}(pos) - M_{best}(pos)) * Levy(\dim) * pos$, *where Levy(dim) is searching function for get next target and* X_{best} *is best performing individual from the population*
24. *end if*
25. *end if*
26. *end While*

* *y* for randomness. To find best solutions by adjustment of positions is done with regards to random individuals or by the average position of the population that allows the algorithm to explore new regions for an improved solution space which prevents it from converging too early to suboptimal solutions, thereby making it more likely that the algorithms find a global optimum. In this proposed method random population is selected of 10 and which undergoes different iteration and the best optimal feature set is selected.

EXPERIMENTAL RESULTS

The computing platform used in the experiment was an Intel Core i5 with 16 GM RAM and a processing speed of 2.40 GHz. The programming environment was Python, running on Jupyter Notebook within the Windows 10 operating system. Table 2 (a) & (b) shows the results obtained from experimentation of 10 different Feature Selection techniques across various classifiers. In other words, accuracy-wise, the proposed hybrid method really outperformed other state-of-the-art feature selection techniques and classifiers. We compared these techniques in

Total Feature Set Opted after Stage 1: (Highlighted are the selected features)

0 id	39 f38_Normalized_Frequent_Link
1 url string	40 f39_contentSSL1
2 f1_NumberOf_iframe	41 f40_contentSSL2
3 f2_NumberOf_frame	42 f41_contentSSL3
4 f3_NumberOf_form	43 f42_contentSSL4
5 f4_NumberOf_input	44 f43_Number_TFIDF_Word_Occures_in_Main_SLD
6 f5_HavingObject	45 f44_havePersen
7 f6_CodebseAttrInObject	46 f45_suspeciouse1
8 f7_HavingApplet	47 f46_suspeciouse2
9 f8_CodebseAttrInApplet	48 f47_suspeciouse3
10 f9_HavingLink	49 f48_suspeciouse4
11 f10_HrefAttrInLink	50 f49_suspeciouse5
12 f11_ActionAttrInform	51 f50_suspeciouse6
13 f12_HavingScript	52 f51_having_input_pass
14 f13_haveAtSign	53 f52_Identity1
15 f14_haveUnderLine	54 f53_Identity2
16 f15_havedash	55 f54_Sensetive_Words
17 f16_haveQuestionsign	56 f55_bad_action_field
18 f17_haveEqualsign	57 f56_Nonmatching_URLs
19 f18_haveSpecialSymbol	58 f57_Out_of_position_brand_name
20 f19_haveCodedURL	59 f58_Identity3
21 f20_IP_address	60 f59_Identity4
22 f21_MLDLength	61 f60_Login_Form_identity
23 f22_countOccurrencesDot	62 f61_Domain_name_identity
24 f23_PathLength	63 f62_Domain_name_in_the_path_of_the_URL
25 f24_domain	64 f63_Nil_anchors
26 f25_Host_length	65 f64_ID_foreign_anchors
27 f26_fileLength	66 f65_foreign_anchors
28 f27_Out_of_position_top_level_domain	67 f66_ID_foreign_request
29 f28_Embedded_domain	68 f67_foreign_request
30 f29_SSL_certificate	69 f68_Using_forms_with_Submit_button
31 f30_Levenshtein_distance_Normalize1	70 f69_Sub_domain_in_URL
32 f31_Levenshtein_distance_Normalize2	71 f70_Segments_of_URLs
33 f32_Levenshtein_distance_Normalize3	72 f71_Numerical_Primary_Domain
34 f33_Levenshtein_distance_Normalize4	73 f72_tidf_keywords_contain_in_path_portion_of_URL
35 f34_Number_of_AllLinks	74 f73_number_of_terms_in_the_host_name_of_the_URL
36 f35_Number_of_SLD	75 f74_NumberOf_redirect
37 f36_Number_of_AllLinks	76 f75_OnMouseOver
38 f37_Number_of_uniq_Links	77 label
	78 class

Total Feature Set Opted after Stage 2: (Highlighted are the selected features)

0 id	39 f38_Normalized_Frequent_Link
1 url string	40 f39_contentSSL1
2 f1_NumberOf_iframe	41 f40_contentSSL2
3 f2_NumberOf_frame	42 f41_contentSSL3
4 f3_NumberOf_form	43 f42_contentSSL4
5 f4_NumberOf_input	44 f43_Number_TFIDF_Word_Occures_in_Main_SLD
6 f5_HavingObject	45 f44_havePersen
7 f6_CodebseAttrInObject	46 f45_suspeciouse1
8 f7_HavingApplet	47 f46_suspeciouse2
9 f8_CodebseAttrInApplet	48 f47_suspeciouse3
10 f9_HavingLink	49 f48_suspeciouse4
11 f10_HrefAttrInLink	50 f49_suspeciouse5
12 f11_ActionAttrInform	51 f50_suspeciouse6
13 f12_HavingScript	52 f51_having_input_pass
14 f13_haveAtSign	53 f52_Identity1
15 f14_haveUnderLine	54 f53_Identity2
16 f15_havedash	55 f54_Sensetive_Words
17 f16_haveQuestionsign	56 f55_bad_action_field
18 f17_haveEqualsign	57 f56_Nonmatching_URLs
19 f18_haveSpecialSymbol	58 f57_Out_of_position_brand_name
20 f19_haveCodedURL	59 f58_Identity3
21 f20_IP_address	60 f59_Identity4
22 f21_MLDLength	61 f60_Login_Form_identity
23 f22_countOccurrencesDot	62 f61_Domain_name_identity
24 f23_PathLength	63 f62_Domain_name_in_the_path_of_the_URL
25 f24_domain	64 f63_Nil_anchors
26 f25_Host_length	65 f64_ID_foreign_anchors
27 f26_fileLength	66 f65_foreign_anchors
28 f27_Out_of_position_top_level_domain	67 f66_ID_foreign_request
29 f28_Embedded_domain	68 f67_foreign_request
30 f29_SSL_certificate	69 f68_Using_forms_with_Submit_button
31 f30_Levenshtein_distance_Normalize1	70 f69_Sub_domain_in_URL
32 f31_Levenshtein_distance_Normalize2	71 f70_Segments_of_URLs
33 f32_Levenshtein_distance_Normalize3	72 f71_Numerical_Primary_Domain
34 f33_Levenshtein_distance_Normalize4	73 f72_tidf_keywords_contain_in_path_portion_of_URL
35 f34_Number_of_AllLinks	74 f73_number_of_terms_in_the_host_name_of_the_URL
36 f35_Number_of_SLD	75 f74_NumberOf_redirect
37 f36_Number_of_AllLinks	76 f75_OnMouseOver
38 f37_Number_of_uniq_Links	77 label
	78 class

terms of accuracy when using all features versus using optimal features extracted (top 10) and the time taken by the classifier to predict anomalies using the trained model. It was found that the accuracy with these extracted features is a little better than having all the features, while the time of prediction reduced. Clearly, it can be concluded from the graph that the more the number of features gets reduced, the better or good both in processing time and accuracy.

This hybrid approach, as noted in the chart, shows there to be a visible improvement of prediction time upon the addition of coevolutionary behaviour. Without coevolutionary behaviour, the time of prediction varies with spikes

occurring at times. If one puts in place the coevolutionary behaviour, it drops the time for prediction through the board. This proves the robustness of the proposed hybrid approach by considerably taking advantage of techniques in coevolutionary methods to make it more efficient and stable **in terms of computational time, hence improving overall performance.**

The Figure 7 graph shows a comparison of prediction times with and without co-evolutionary behavior among different classifiers and feature selection techniques. The analyzed classifiers include Logistic Regression, Naïve Bayes, K-Nearest Neighbor, Support Vector Machine, and GridBoost, paired with various feature selection methods such as the Chi-squared Test, Pearson Correlation Coefficient, Recursive Feature Elimination, Lasso, Forward

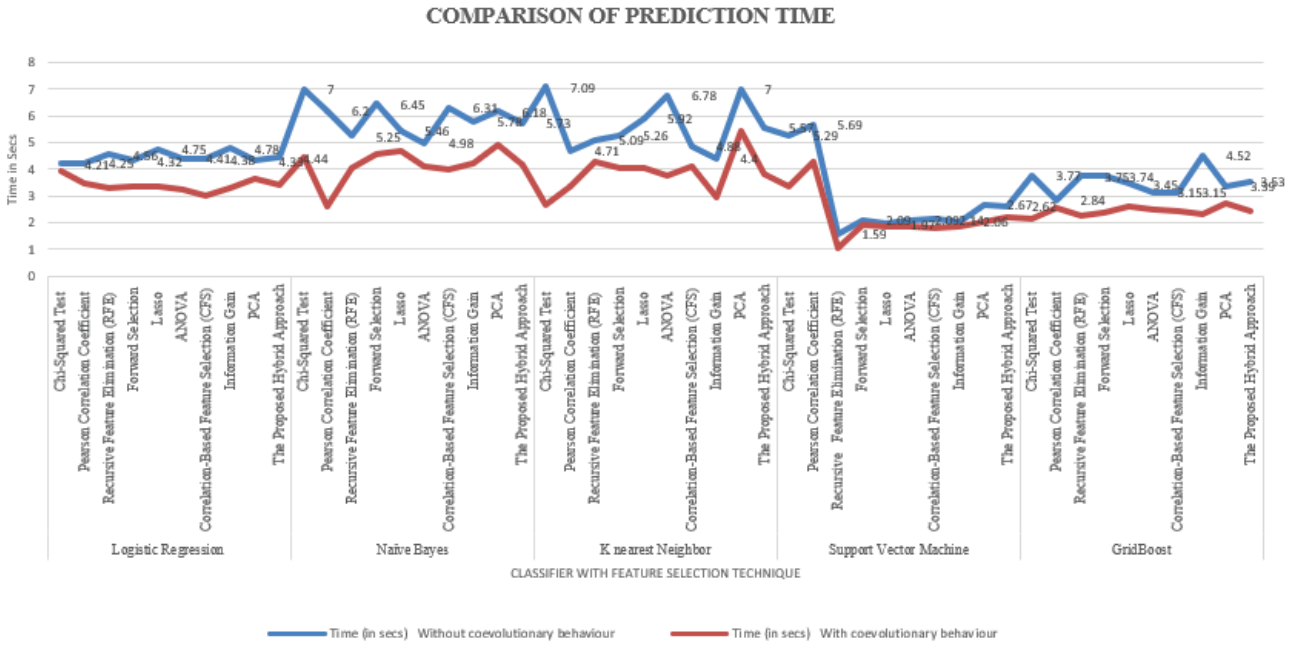

Figure 7. Comparison of predication time with feature selection.

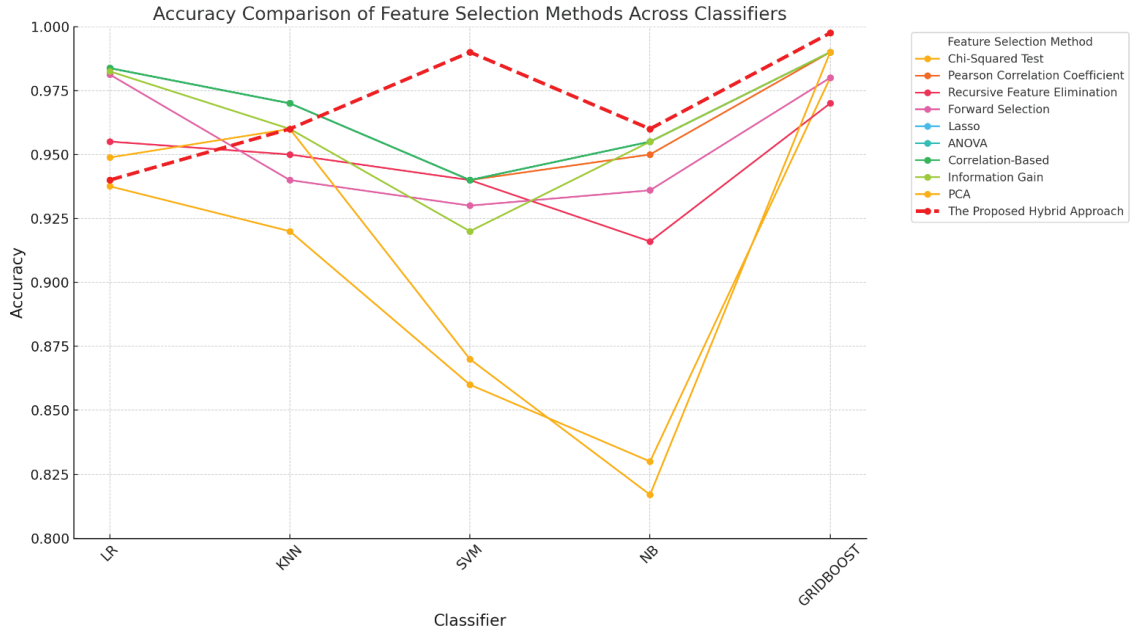

Figure 8. Comparison of Accuracy considering with and without coevolutionary behaviour.

Selection, ANOVA, Correlation-based Feature Selection, Information Gain, PCA, and a Hybrid approach. The Y-axis represents prediction time in seconds, while the X-axis shows the different combinations of classifiers and feature selection techniques.

The graph presents two lines: one in blue, representing prediction times without coevolutionary behavior, and another in red, representing times with coevolutionary behavior. Generally, the red line is below the blue line in most combinations, indicating that prediction time is generally shorter with the inclusion of coevolutionary behavior. The disparities in prediction times between both conditions are particularly noticeable for certain classifiers like K-Nearest Neighbor and Support Vector Machine. In summary, the graph indicates that co-evolutionary behavior enhances prediction efficiency across all tested classifiers and feature selection techniques.

The chart clearly shows that the proposed hybrid approach significantly improves the accuracy of a Logistic Regression classifier in case of co-evolutionary behavior application - from 94.28% up to 98.37%. As a result, the method was effective at both increasing accuracy and computational efficiency.

Figure 8: gives the line plot of the comparative performance of various feature selection methods against different classifiers. The proposed Hybrid Approach was plotted in red and showed to have the highest accuracy against all classifiers, hence proving that the choice made is robust and

Table 2. Comparison of Accuracy using different datasets

	Accuracy without Feature Selection using Gridboost Classifier	No of Features	Accuracy with existing Feature Selection using Gridboost Classifier	No of Features using existing Feature selection	Accuracy with proposed hybrid approach using Gridboost Classifier	Optimal features using the Proposed Hybrid Approach
Phishing Dataset	98% [79]	79	99.62%	24	99.7%	10
KDD	96% [43]	43	96.9%	17	97.34%	10
CICIDS2017	85%[79]	79	86.2%	24	98.99%	10
WSN-DS	86%[19]	19	87.5%	17	99.30%	8

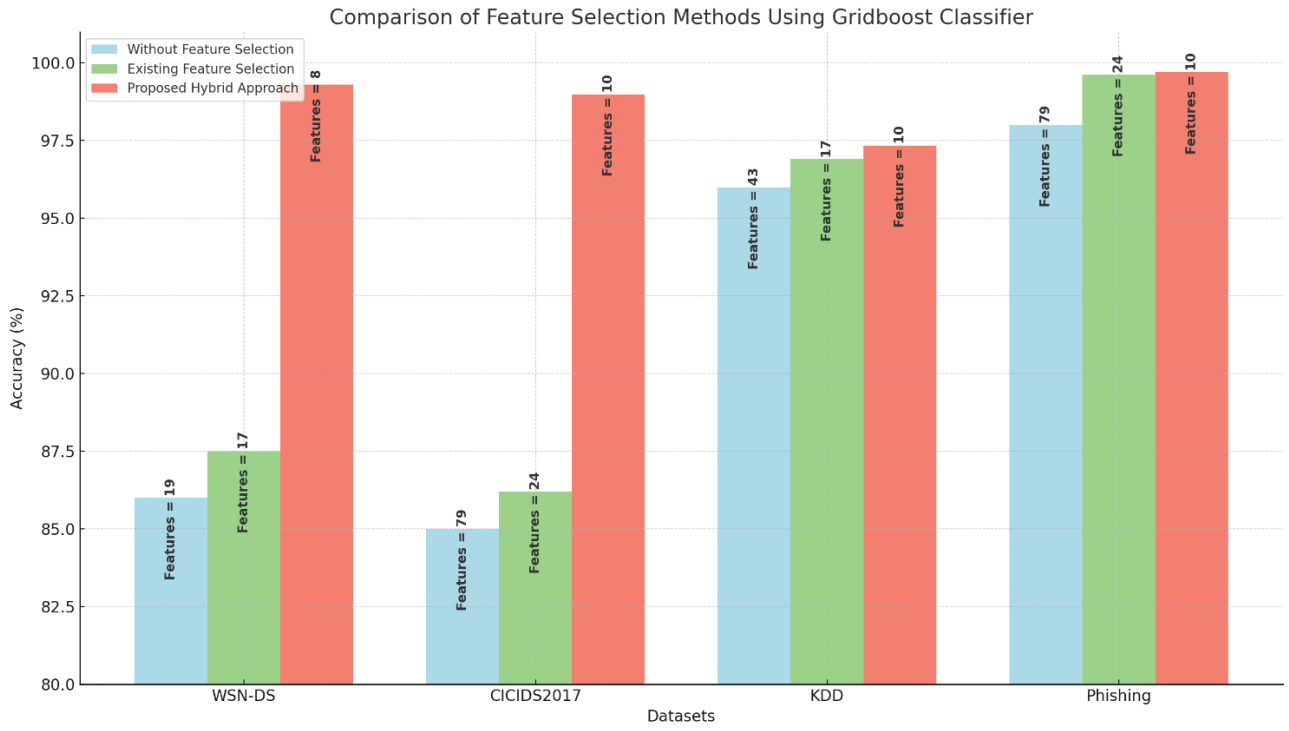

Figure 9. Comparison of accuracy using different datasets.

very effective in choosing optimal features. Methods such as Pearson Correlation Coefficient and Recursive Feature Elimination performed well but did not return the consistency that the Proposed Hybrid Approach projected. This analysis indicates that a sophisticated feature selection method, like the Proposed Hybrid Approach, is one of the necessary steps toward achieving better model performance for a range of classifiers.

The bar graph plots the performance improvement of various feature selection methods using the Gridboost Classifier across the four datasets: WSN-DS, CICIDS2017, KDD, and Phishing. It compares feature selection without optimization in light blue, existing feature selection techniques in light green, and the Proposed Hybrid Approach in salmon. One main observation made from this chart is that, in the entire datasets, the best accuracy achieved has been from the Proposed Hybrid Approach, hence outperforming the standard and existing feature selection methods. Next is that this also uses the least number of features, indicating the high point wherein it improves not only the accuracy but with leaner feature sets. For instance, in the case of the WSN-DS dataset, the accuracy with the Proposed Hybrid Approach is nearly 100% using just 8 features, while without feature selection, the accuracy was 86% with 19 features and 87.5% with 17 features using traditional feature selection methods. The CICIDS2017 dataset also presents a similar situation, whereby at 10 features, the hybrid approach attains an accuracy of 98.99%, with other methods using more features below this. This is the trend until in KDD and Phishing, Proposed Hybrid Approach attains the highest accuracy with a minimum number of features. That is, the proposed hybrid approach improves model performance in an efficient fashion by selecting the most relevant features, hence by keeping computational costs low at an optimum rate. This consistent superior performance underlines the capability to perform dynamic optimization of feature selection by the proposed hybrid approach, reached through a highly accurate and efficient model; hence, this is the best choice compared with referred methods.

T-test Result

T-statistic: The greater the magnitude of T, the larger the difference between the means of the two groups.P-value: If the p-value is less than a set value ; we reject the null hypothesis and conclude there is a significant difference between the two sets with and without co-evolutionary behavior.This t-test now will show whether the improvement in accuracy is statistically significant.The p-value turns out to be low, then we can say with statistical significance that co-evolutionary behaviour yields significantly better results.Once the T-test for all feature selection methods is computed, you can plot these differences as bar plots or even a heatmap showing how each method is influenced by the coevolutionary approach.

The Figure 10 shows graph to depicted by the bar chart that three feature selection methods are used in this case: Without Feature Selection, Existing Feature Selection, and the Proposed Hybrid Approach. The Proposed Hybrid Approach yielded a mean accuracy of 98.33% with the smallest variability, hence proving effective and yielding consistent results from the most relevant features. In contrast, Existing Feature Selection methods raise accuracy to 92.55% but have moderate variability, showing less reliable performance. Then again, without Feature Selection, the accuracy drops to a lowly 91.25% with the maximum variability. This shows that feature selection has a number of irrelevant and/or redundant features. Summing up,

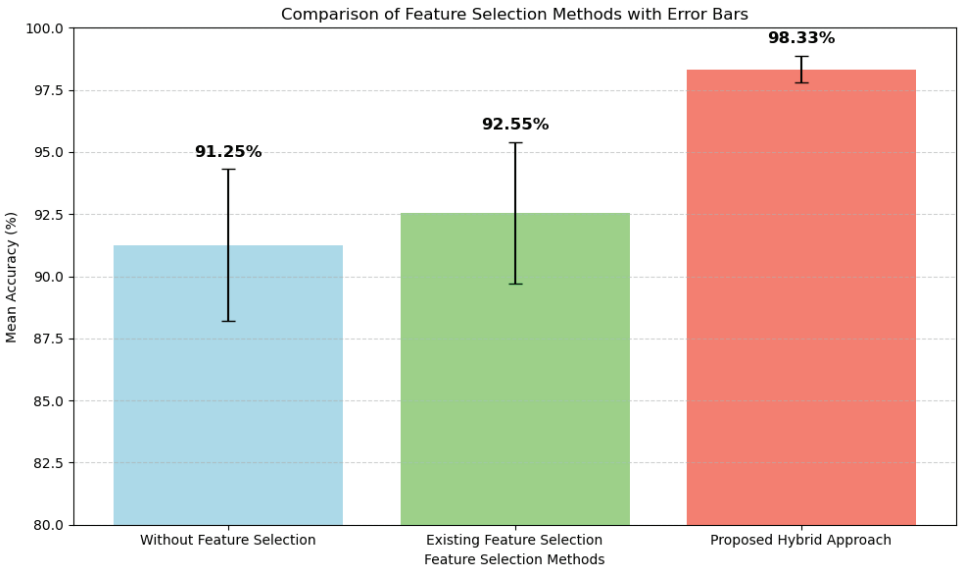

Figure 10. T-test accuracy comparison and differences between hybrid and existing methods.

performances of the Proposed Hybrid Approach turned out to be significantly better as compared to others and hence can be termed robust and efficient.

CONCLUSION

In this work, the Phishing Dataset, KDD dataset, WSN-DS dataset and CICIDS2017, comparison of achieve accuracy by Phishing Dataset 99.30%, KDD: 97.34%, CICIDS2017: 98.99%, WSN-DS: 98.99% was used for evaluation with and without feature selection. It is clear that the results obtained from the study showed our approach improved accuracy significantly up to 99.7% for Phishing Dataset, KDD: 97.34%, CICIDS2017: 98.99%, WSN-DS: 98.99%. This is a rather high improvement compared to results obtained using other very popular classifiers like KNN, LR, SVM, NB, and GridBoost. Literature review identified some of the critical challenges in feature selection, such as avoiding redundancy and irrelevant features, finding the best local solution, and seeking a better balance between exploration and exploitation. Guided by these challenges, a hybrid feature selection technique has been developed based on information-theoretic measures, majority voting, and the Gannet optimization algorithm. This is a truly innovative approach that aims to identify and let in only the optimal set of features by considering the impact of one feature with another which is co-evolutionary behaviour that would accomplish dual goals of simplification of processing time and increase in accuracy. In this work, we test the efficiency of our proposed method rigorously against ten benchmark feature selection methodologies on the all four standard datasets. Clearly emerging was that our hybrid approach outperformed traditional techniques in accuracy, although this improvement came only when the optimal combination of features selected was applied. This underlines further how critical a step selecting the right features actually is in elevating model performance. This reduction in the number of features can clearly be seen in the following graph above, which directly means improved processing time and accuracy. In this way, our approach will ensure that the model can work with maximum efficiency and effectiveness based on optimal features.

In the future, work on data augmentation techniques will be incorporated to improve the model's accuracy. These results will contrast independent results from many other optimization procedures, key in on key parameters that return the best results. This clearly denotes that commitment, ongoing to feature selection optimization, comes as a necessity in the accomplishments of superior accuracy in phishing detection models.

AUTHORSHIP CONTRIBUTIONS

Authors equally contributed to this work.

DATA AVAILABILITY STATEMENT

The authors confirm that the data that supports the findings of this study are available within the article. Raw data that support the finding of this study are available from the corresponding author, upon reasonable request.

CONFLICT OF INTEREST

The author declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

ETHICS

There are no ethical issues with the publication of this manuscript.

REFERENCES

[1] Venkatesh B, Anuradha J. A review of feature selection and its methods. Cybern Inf Technol 2019;19:3-26. [CrossRef]

[2] Sharma A, Dey S. A comparative study of selection and machine learning techniques for sentiment analysis. Proc 2012 ACM Res Appl Comput Symp RACS. 2012:1–7. [CrossRef]

[3] Patil D, Pattewar T. Majority voting and feature selection-based network intrusion detection system. ICST Trans Scalable Inf Syst. 2022;2022:173780. [CrossRef]

[4] Borandag E, Ozcift A, Kilinc D, Yucalar F. Majority vote feature selection algorithm in software fault prediction. Comput Sci Inf Syst 2018;16:515–539. [CrossRef]

[5] Chamakura L, Saha G. An instance voting approach to feature selection. Inf Sci 2019;504:449-469. [CrossRef]

[6] Agrawal P, Abutarboush HF, Ganesh T, Mohamed AW. Metaheuristic algorithms on feature selection: A survey of one decade of research (2009–2019). IEEE Access 2021;9:26766–26791. [CrossRef]

[7] Almasoudy FH, Al-Yaseen WL, Idrees AK. Differential evolution wrapper feature selection for intrusion detection system. Procedia Comput Sci 2020;167:1230–1239. [CrossRef]

[8] Wang L, Gao Y, Li J, Wang X. A feature selection method by using chaotic cuckoo search optimization algorithm with elitist preservation and uniform mutation for data classification. Discrete Dyn Nat Soc 2021. [CrossRef]

[9] Alhaj TA, Siraj MM, Zainal A, Elshoush HT, Elhaj F. Feature selection using information gain for improved structural-based alert correlation. PLoS One 2016;11:e0166017. [CrossRef]

[10] Spencer R, Thabtah F, Abdelhamid N, Thompson M. Exploring feature selection and classification methods for predicting heart disease. Digit Health 2020;6:2055207620914777. [CrossRef]

[11] Rajeswari S, Suthendran K. Feature selection method based on Fisher's exact test for agricultural data. Int J Recent Technol Eng 2019;8:558–566. [CrossRef]

[12] Potharaju SP, Marriboyina S. Correlation coefficient-based feature selection framework using graph construction. Gazi Univ J Sci 2018;31:775–787.
[13] Zhou H, Wang X, Zhu R. Feature selection based on mutual information with correlation coefficient. Appl Intell 2022;52: 5457–5474. [CrossRef]
[14] Urbanowicz RJ, Meeker M, Cava WL, Olson RS, Moore JH. Relief-based feature selection: Introduction and review. J Biomed Inform 2018;85:189–203. [CrossRef]
[15] Lakshmi PD, Vishnuvardhan B. Variance-based feature selection for enhanced classification performance. Inf Syst Des Intell Appl 2018:543–550. [CrossRef]
[16] Malhotra H, Sharma P. Intrusion detection using machine learning and feature selection. Int J Comput Netw Inf Secur 2019;4:43–52. [CrossRef]
[17] Nasir IM, Khan MA, Yasmin M, Shah JH, Gabryel M, Scherer R, et al. Pearson Correlation-Based Feature Selection for Document Classification Using Balanced Training. Sensors (Basel) 2020;20:6793. [CrossRef]
[18] Borboudakis G, Tsamardinos I. Forward-backward selection with early dropping. J Mach Learn Res 2019;20:1–39.
[19] Vandana CP, Chikkamannur AA. Feature selection: An empirical study. Int J Eng Trends Technol 2021;69: 165–170. [CrossRef]
[20] Jeon H, Oh S. Hybrid-recursive feature elimination for efficient feature selection. Appl Sci 2020;10:3211. [CrossRef]
[21] Nersisyan S, Novosad V, Galatenko A, Sokolov A, Bokov G, Konovalov A, et al. ExhauFS: Exhaustive search-based feature selection for classification and survival regression. PeerJ 2022;10:e13200. [CrossRef]
[22] Muthukrishnan R, James C. Feature selection through robust LASSO procedures in predictive modelling. Adv Appl Math Sci 2022;21:6103–6115.
[23] Muthukrishnan R, Rohini R. LASSO: A feature selection technique in predictive modeling for machine learning. IEEE Int Conf Adv Comput Appl ICACA. 2016. [CrossRef]
[24] Ramjee S, El Gamal A. Efficient wrapper feature selection using autoencoder and model-based elimination. IEEE Comput Soc LOCS. 2020.
[25] Sharifipour S, Fayyazi H, Sabokrou M, Adeli E. Unsupervised feature ranking and selection based on autoencoders. ICASSP 2019 - 2019 IEEE Int Conf Acoust Speech Signal Process ICASSP. 2019:Brighton UK. [CrossRef]
[26] El Aboudi N, Benhlima L. Review on wrapper feature selection approaches. Int Conf Eng MIS ICEMIS. 2016. [CrossRef]
[27] Malhi A, Gao RX. PCA-based feature selection scheme for machine defect classification. IEEE Trans Instrum Meas 2004;53:1517–1525. [CrossRef]
[28] Pan JS, Zhang LG, Wang RB, Snášel V, Chu SC. Gannet optimization algorithm: A new metaheuristic algorithm for solving engineering optimization problems. Math Comput Simul 2022;202:343–373. [CrossRef]
[29] Borandağ E, Ozcift A, Kilinç D, Yucalar F. Majority vote feature selection algorithm in software fault prediction. Comput Sci Inf Syst 2018;16:515-539. [CrossRef]
[30] Chamakura L, Saha G. An instance voting approach to feature selection. Inf Sci 2019;504:449-469. [CrossRef]
[31] Lee W, Xiang D. Information-theoretic measures for anomaly detection. Proc IEEE Symp Secur Priv 2000:130–143.
[32] Almasoudy FH, Al-Yaseen WL, Idrees AK. Differential evolution wrapper feature selection for intrusion detection system. Procedia Comput Sci 2020;167:1230–1239. [CrossRef]
[33] Kshirsagar D, Kumar S. Towards an intrusion detection system for detecting web attacks based on an ensemble of filter feature selection techniques. Cyber Phys Syst 2021;9:1-16. [CrossRef]
[34] Mallenahalli N, Sarma TH. A tunable particle swarm size optimization algorithm for feature selection. IEEE Congr Evol Comput CEC 2018:1–7. [CrossRef]
[35] Abualigah LM, Khader AT, Hanandeh ES. A new feature selection method to improve the document clustering using particle swarm optimization algorithm. J Comput Sci 2018;25:456–466. [CrossRef]
[36] Chen H, Hou Y, Luo Q, Hu Z, Yan L. Text feature selection based on water wave optimization algorithm. Int Conf Adv Comput Intell 2018:546–551. [CrossRef]
[37] Yan C, Ma J, Luo H, Wang J. A hybrid algorithm based on binary chemical reaction optimization and tabu search for feature selection of high-dimensional biomedical data. Tsinghua Sci Technol 2018;23:733–743. [CrossRef]
[38] Peng H, Ying C, Tan S, Hu B, Sun Z. An improved feature selection algorithm based on ant colony optimization. IEEE Access 2018;6:69203–69209. [CrossRef]
[39] Tubishat M, Idris N, Shuib L, Abushariah MAM, Mirjalili S. Improved salp swarm algorithm based on opposition based learning and novel local search algorithm for feature selection. Expert Syst Appl 2020;145:113122. [CrossRef]
[40] Kelidari M, Hamidzadeh J. Feature selection by using chaotic cuckoo optimization algorithm with levy flight, opposition-based learning and disruption operator. Soft Comput 2021;25:2911–2933. [CrossRef]
[41] Tubishat M, Alswaitti M, Mirjalili S, Al-Garadi M, Alrashdan M, Rana T. Dynamic butterfly optimization algorithm for feature selection. IEEE Access 2020;11:194303–194314. [CrossRef]

[42] Mendeley. KDD 99 Dataset. Available at: https://data.mendeley.com/datasets/zw7knrxpy5/1 Accesed May 14, 2025.
[43] Sultana N, Palaniappan S. A survey on online social network anomaly detection. Int J Innov Sci Technol 2018;3:243–257.
[44] George SCG, Sumathi. Grid search tuning of hyperparameters in random forest classifier for customer feedback sentiment prediction. Int J Adv Comput Sci 2020;11:173–178. [CrossRef]
[45] Yang L, Shami A. On hyperparameter optimization of machine learning algorithms: Theory and practice. Neurocomputing 2020;415:295–316. [CrossRef]
[46] Chen T, Guestrin C. XGBoost: A scalable tree boosting system. Proc ACM SIGKDD Int Conf Knowl Discov Data Min 2016. https://doi.org/10.48550/arXiv.1603.02754 [CrossRef]
[47] Lunawat S, Rao J, Patil P. GridBoost: A classifier with increased accuracy to detect anomaly in social media networks. J Eng Sci Technol Rev 2023;16:13–18. [CrossRef]
[48] Jamal T. KDD 99 Dataset. Available at: https://www.kaggle.com/datasets/toobajamal/kdd-99-dataset Accesed May 14, 2025.
[49] Yiğit Ö. Feature extraction for DNA capillary electrophesis signals based on discrete wavelet transform combined with multi-scale permutation entropy. Sigma J Eng Nat Sci 2022;40:475–490. [CrossRef]
[50] Maseer ZK, Yusof R, Bahaman N, Mostafa SA, Foozy CFM. Benchmarking of machine learning for anomaly-based intrusion detection systems in the CICIDS2017 dataset. IEEE Access 2021;9:22351–22370. [CrossRef]
[51] Chang Y, Tang H, Cheng Y, Zhao Q, Li B, Yuan X. Dynamic hierarchical energy-efficient method based on combinatorial optimization for wireless sensor networks. Sensors 2017;17:1665. [CrossRef]