An Efficient Hybridization of K-Means and Genetic Algorithm Based on Support Vector Machine for Cyber Intrusion Detection System

Yakub Kayode Saheed[1], Micheal Olaolu Arowolo[2,] and Abdulrauf U. Tosho[3]

[1]School of Information Technology & Computing, American University of Nigeria, Nigeria
[1]Kaptain A. I & Innovation Research Group
[2]Department of Computer Science, Landmark University, Omu-Aran, Nigeria
[2]Industry, Innovation, and Infrastructure Research Group, Landmark University SDG9
[1,3]Department of Computer Science, Al-Hikmah University, Ilorin, Nigeria
[1]yakubu.saheed@aun.edu.ng, [2]arowolo.olaolu@lmu.edu.ng, [3]autosho@alhikmah.edu.ng

Abstract: Intrusion Detection System (IDS) is a challenging cyberspace security technology to safeguard against a malicious threat. Although many soft computing approaches have been utilized to increment the effectiveness of IDS, it is a significant challenge for present-day intrusion detection classification algorithms to give and achieve high performance. The first significant challenge is that lots of needless, dispensable, superfluous, and meaningless data in high-dimensional datasets affect the IDS classification process. Secondly, attack patterns are also dynamic, requiring efficient classification and cyber-attacks prediction. Thirdly, a single classifier cannot work well to detect any form of attack. Lastly, the accuracy, detection rate (DR), and false alarm rate (FAR) are still significant issues to contend with. Thus, we propose an efficient hybridization technique in this paper to address these significant challenges. This paper proposes supervised and unsupervised learning techniques for detecting both known and unknown attacks. In the first line of this research, k-means clustering was applied to the normalized data to classify the data into normal and attack classes to resolve the dynamic nature of the attack patterns. Then, wrapper feature selection with a genetic algorithm (GA) was employed to address the needless and redundant dataset. Lastly, the classification of the inputted data from GA predictors was performed with a support vector machine (SVM). The analysis of the computational time needed for training and testing for use in time-critical applications was also carried out. The experimental results revealed a promising high accuracy of 99% with low FAR. The appealing benefits of the proposed model are its robustness, low computational cost, and also its impressive success in generalization by reducing possible overfitting.

Keywords: Intrusion Detection System; k-means; Support Vector Machine; Genetic Algorithm; Cyber Attacks, NSL-KDD, Wrapper feature Selection

1. Introduction

The need for network security and protection from cyber-attacks is growing because of the pervasive proliferation of network access. An Intrusion Detection System (IDS) is one of the frequently used methods for safeguarding the confidentiality, integrity, availability, and authentication of delicate assets and resources in protected systems in a network topology. Computer networks specifically the internet have played an essential role in different areas of human life, such as electronic commerce and communication since the last few decades [1]. It has entered, penetrated the lives of people from different communities, and connected more than three billion internet users in about 160 countries around the World enriching the experience of the people[2]. However, the resultant cyber-safety problem is becoming the main undermining factor disturbing the adoption of the network. With the rising growth of network infrastructure, especially with the internet popularity, a growing number of people have been focused on the issue of cyber security[3],[4]. The IDS is a security technology that can protect, monitor the network from illegal events, external attacks and generate alerts [5],[6],[7]. There are two detection methods that IDS can be categorized, known popularly as anomaly-detection and

Received: February 5th, 2021. Accepted: February 27th, 2022
DOI: 10.15676/ijeei.2022.14.2.11

misuse-detection models. Anomaly detection can spot known, proven, and novel attacks nonetheless with high false-positive rates. Whereas, misuse-detection can only locate and detect the known attacks[8],[9],[10].

Conventional IDS suffers from weaknesses such as high false-alarm rate, reduced efficiency in noticing novel and new forms of attacks, and reduce accuracy rate in detection[11]. Thus, it is essential to implement a stable IDS that can enhance the accuracy of the detection, significantly reduce the false-alarm rate and increase the effectiveness of discovering new types of attacks[12].

Machine Learning (ML) approaches have been thoroughly studied to satisfy network protection criteria to develop IDS that is capable of operating optimally. ML techniques that have been used successfully in IDS are K-Nearest Neighbor (KNN) [13], C4.5 [14], Artificial Neural Network (ANN) [15] [16], XGBoost[17], Convolution Neural Network (CNN)[18],[19],[20], Naïve Bayes [21], Random Forest [22], Firefly algorithm[23], Decision Tree (DT) [24] and bio-inspired approaches[25]. This research focuses both on a mixture of both supervised and unsupervised ML techniques. In the first step, the k-means clustering technique was adopted on the normalized data to classify the data into Normal (1) and Attack (2) classes. For an increased performance of the algorithm over the NSL-KDD multi-dimensional dataset, it is sacrosanct to perform Feature Engineering (FE) [26],[27]. FE is a technique of changing raw data into attributes that best reproduce the fundamental issue for the predictive models, resulting in increased model performance on data that is not seen [17]. The FE stage was performed with Feature Selection (FS) dimensionality reduction technique to hand-picked the most significant attributes removing irrelevant and needless ones, which do not affect the classifier accuracy. The FS serves a key role in the design of ML techniques and is equally an important stage in IDS. It also has the advantage of reducing the operating capacity because it lessens the number of instances in the dataset and creates new instances[28]. Irrelevant data features impact model consistency and enhance the training time required to construct the model[29],[30],[31]. It is a significant step in creating a reliable IDS to remove extracted attributes that raise false alarms and increase system accuracy. The FS is of three classes: filter methods, wrapper methods, and embedded methods[32]. The method based on the filter applies a statistical test to give a scoring for each feature. Authors in [33],[34],[35],[36] adopted the filter based for FS in IDS. Often, the technique is univariate and treats the feature independently, or about the dependent feature [37]. The filter methods example is a method of information gain[38], Chi-square test [39], and the method of scoring the correlation coefficient [40].

In the wrapper method, the selection of a group of features is known by wrapper approaches as a search problem, where various groupings are selected, validated, and associated with others. The GA and Recursive Feature Elimination (RFE) algorithm is both an example of wrapper methods. We used the GA wrapper method in this paper in which the search method is heuristic. Lastly, the embedded methods learn which characteristics better improve the model accuracy when constructing the model.

The rest of this paper is structured as follows: we present the related work in Section 2 and report the proposed technique in section 3. The results and discussion are highlighted in section 4, and section 5 concludes the paper.

2. Related Work

The work in [41] proposed a ranker-based feature selection method to decrease the number of features and estimates the implemented model with an ensemble of Instant Base Learning (IBK), K-Nearest Neighbor, REP Tree, Random Tree, J48graft, and Random Forest classifiers. The proposed model utilized the NSLKDD dataset with experimental results of 99.72% accuracy and 99.68% accuracy. The DR and training time were silenced on and not reported.

Authors in [29] suggested a wrapper feature selection algorithm. The algorithm used pigeon inspired optimizer and was evaluated on three widely studied datasets: NSLKDD, UNSW-NB15, and KDD'99. The true-positive rate, accuracy, false-positive rate, and F-score were the performance metrics used to ascertain the performance.

Performance comparison of IDS between State Preserving Extreme Learning Machine (SPELM) and Deep Belief Network was presented in [42]. The SPELM algorithm has been used for facial recognition, pedestrian detection, and IDS successfully. SPELM was used as a classifier and analyzed on the NSLKDD dataset. The splitting strategy employed showed that 40% of the data was used for training, and 60% of the data for testing. The findings revealed that the SPELM with 93.20% accuracy outperformed DBN with 52.8% accuracy and computational time of 90.8 seconds utilized by SPELM as contrary to 102 seconds of DBN.

A hybrid layered approach was proposed in [43] based on different machine learning methods. In the paper, data preprocessing based on transformation and normalization was first performed on the NSLKDD dataset. Then, the dataset was split with 20% of the data for testing the model. The feature selection stage used two methods. The performance metrics of the model were on the accuracy, detection rate, and time was taken are used.

The scholars in [38] presented a hybrid approach utilizing the NSL-KDD dataset. The training dataset was 80% whereas the remaining 20% was used for testing on both binary and multi-class problems. The feature selection strategy used was vote algorithm and information gain. The detection accuracy of the proposed model was 99.81% and 98.56% respectively.

The authors [44] proposed a hybrid method for IDS using GA and SVM. Their proposed method reduced the attributes from forty-one (41) to ten (10). The reduced attributes were now divided into three (3) priorities by using GA, where the main importance is assigned in the (1^{st}) priority and the least important in the (3^{rd}) third priority. Feature distribution was performed as four (4) attributes placed in priority one, four (4) in priority two, and two (2) in priority three. The method was analyzed on the KDD'99 dataset. The results findings gave a 97.3% detection rate while the FAR was 0.017.

The work of [45] presented a novel scheme built on feature selection with filtering and discretization to improve the classification. They evaluated the proposed method on the KDD Cup 99 with the aid of binary and multi-class classification. The results obtained gave a small false-positive rate and fast performance. The only problem is that the detection rate is less than the average detection rate.

3.1. Proposed Technique

The major stages of the proposed technique which follows a stepwise procedure include dataset collection, data filtering, normalization, data clustering with k-means, feature selection based on GA, classification of selected features utilizing SVM, and performance evaluation. Figure 1 demonstrates the flowchart of the proposed IDS. In the first line of this research, the data filtering and normalization are performed to ensure that the dataset is scaled in the range. The preprocessing phase helps to eliminate the outliers and standardized the dataset to take a form that is meaningful to the SVM model. This is done since NSLKDD contains mixed features, only continuous features are required due to classifier computing limits. There is no need for conversion if a categorical feature has only two different values, such as (Yes or No), and it can be handled as continuous. If there are more than two different values, conversion is required. Using 1 of k coding, all categorical features are converted to continuous. 'k' separate features are constructed to represent distinct 'k' values of category characteristics in this conversion procedure. The K-means step helps to clustered and grouped the class into normal and attack class. The GA step helps to perform feature selection by selecting eighteen (18) significant attributes. The SVM phase classified the attributes received from the GA as anomaly network traffic and normal traffic. This method is different from previous work by the hybridization of K-means and GA for feature dimensionality reduction.

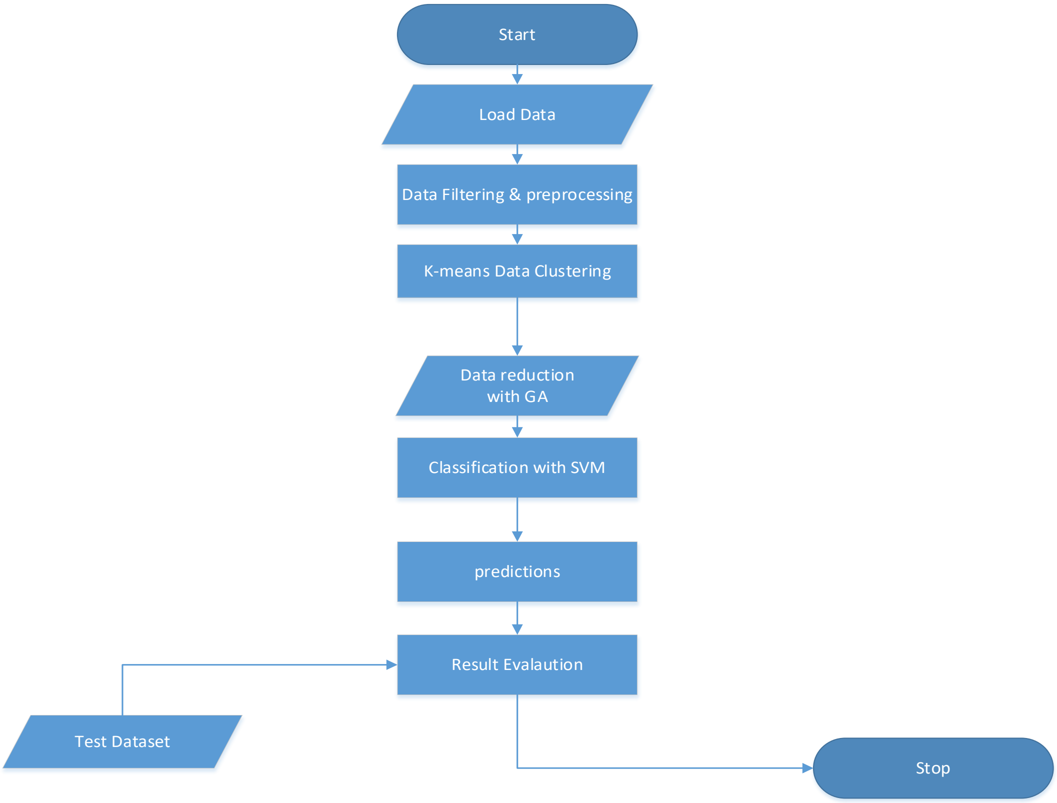

Figure 1. Flowchart of the proposed IDS

3.2. Dataset Description

Dataset selection stage for analysis is an important task, since system performance is dependent upon dataset correctness. The more reliable the data the better the system's performance. The dataset can be compiled and collected by several ways, like 1) sanitized data-set, 2) testbed dataset, 3) simulated data-set, and 4) standard data [8]. Nevertheless, problems arise in the usage of the number one to three procedures.

A real traffic scheme is costly, while the sanitized technique is insecure. Simulation systems are also dynamic and difficult to build. In addition, various traffic types are needed to model different network attacks which are costly and complex. In order to resolve these problems, the NSL–KDD dataset is utilized to authenticate the proposed model for intrusion detection. A total instance of 25192 was extracted from the NSL KDD Cup Dataset with four major class of attacks and the non-attack class which is normal was taken into consideration for the system experimental set up of this paper as well as a total of 41 attributes[46]. This NSLKDD dataset was compiled by simulating numerous attacks on the Unix and Window platforms, including probe, user to root, remote to local, and denial of service. Five million connection records are created from four gigabytes of raw compressed TCP dump data. A connection is made up of a series of TCP packets that are sent between two timestamps. Data goes from the source to the target IP address in a specific connection. This dataset has officially been designated as a benchmark KDDCup dataset for IDS research.

Both the training and testing sets of this dataset have a total of 148517 connection records. TCP, UDP, and ICMP protocol records account for 121569, 17614, and 9334 of the total records. There are 77054 regular connections and 71463 abnormal connections in it. It has 41 characteristics and a single class label.

3.2.1 Service Protocol Imbalances

The proportion of links of various protocols in the network traffic dataset is not even. Some protocols have a large number of connections, whereas others have a small number. Pre-processing is harmed by this imbalance. Because all types of protocol and service connections are assured to be considered because connections are divided according to standardization, the influence of protocol service imbalance in the dataset is reduced.

3.3. Dataset Attacks

The dataset is grouped under the following sub-attacks:

Table 1. Attack Labelling

Attacks	Data Labelling
Normal	1
Denial-of-Service (DoS)	2
Probe	3
User-to-Route(U2R)	4
Remote-to-local (R2l)	5

3.4. Data Pre-processing method

The algorithm is incapable to analyze the raw NSL-KDD dataset since there are presence of symbolic features. Hence, data pre-processing is vital in which symbolic features are removed, since they do not specify essential contribution in detecting intrusion. In this paper, we adopted the dataset creation and feature construction as our data preprocessing methods as explained below;

3.4.1 Data Creation

For training and testing, we identified representative network traffic. These datasets should be tagged with whether or not the connection is normal. Identifying network traffic is a time-consuming and challenging operation.

3.4.2 Feature Construction

In this phase, we created additional features that have a higher discriminative ability than the basic feature set. The proposed SVM method, which is used to detect abnormal and normal connections, could benefit greatly from this.

3.5. Data Filtering and Normalization

The filtered data helps to present a well formatted data into the system. The data was filtered by converting string variable to numeric variable and removing inconsistent factor. An inconsistent factor is also eliminated during the normalization phase. We adopted the standard scalar technique for the normalization. Our goal is to have all the values of the features in the same range. Hence, we make the NSLKDD data standard deviation =1, and mean = 0.

$$X_{\text{standard deviation}} = \frac{x - mean(x)}{Standard\ deviation\ (x)} \tag{1}$$

3.6. K-means clustering

K-means clustering is a simple, popularly used supervised machine learning algorithm [47], [48]. In K-means, a group of points with a representative entity known as centroid is a collection [49], [50].

The k-means algorithm was able to clustered the dataset into two (2) major groups comprising of the normal and the attacks, the k-means algorithm grouped the groups with associative characteristic in the data pattern into the same class. The k-means group all normal class to a single group and all attacks form DoS, U2R, Probe and R2L to a single class group labeled as attacks as shown in Table 2.

Table 2. K-means clustered results

S/N	Clustered class
1	2
2	2
3	1
4	2
5	2
6	1
7	1
8	1
9	1
10	1
11	1
12	1
13	2
14	2
15	1
16	1
17	2
18	2
19	2
20	2
21	1
22	1
23	2
24	2
25	1
26	2
27	1

The figure 2 chart shows the silhouette value which scales from 0 to 1 within the two classes fell into.

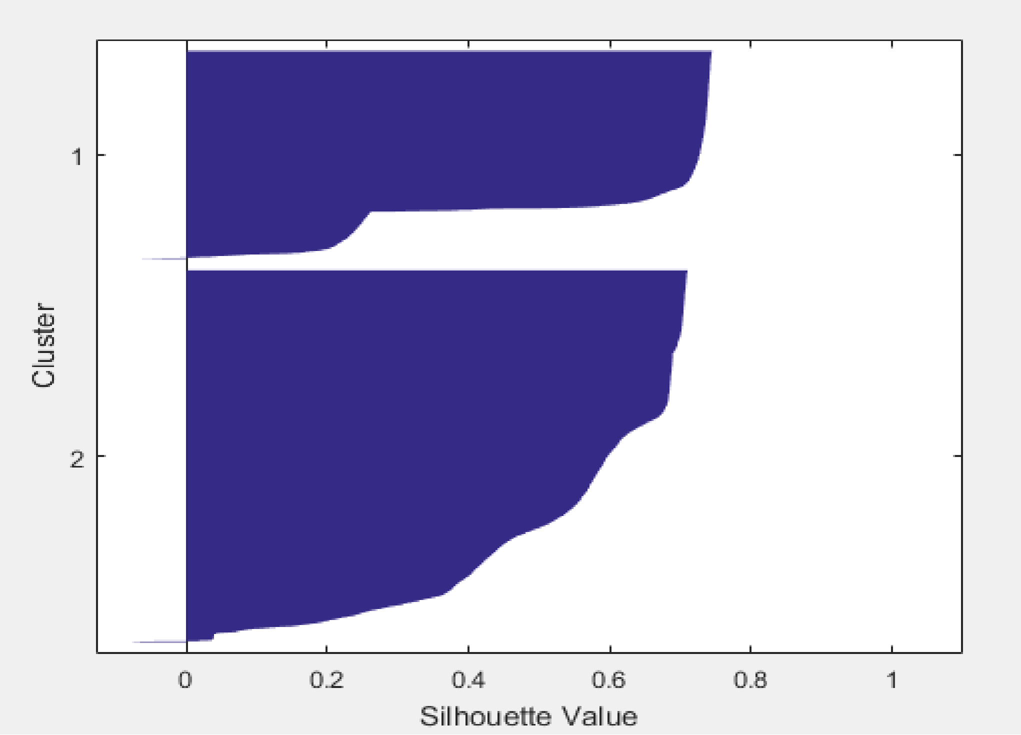

Figure 2. Sihoutte Value

3.7. Feature Selection

The datasets for intrusion detection certainly comprise many irrelevant and redundant features attributes which reduce the effectiveness of machine learning algorithms and trigger construe performance[51]. Thus, this stage is non-negotiable in IDS design as it is critical both in efforts and time. We introduced GA for feature selection to select the most relevant features in the dataset. We used GA to reduce the time complexity and space complexity. The proposed GA find the optimal subset of feature. The GA then present lesser number of values to the SVM model.

3.8. Genetic Algorithm

GA is a widely known technique in evolutionary computation research that mimic the process of natural selection [52][53]. It can be effectively utilized in many problem areas such as business, engineering and ideal approach for getting optimal solution to problem[54][55]. The three important operators in GA are crossover, selection, and mutation. Crossover combines second half of the primary record with first half of the second record. Selection differentiates the most suitable individuals in the population size available using the fitness function[56]. Mutation exchanges the 0 to 1 bits at random and the other way around. The figure 3 depict the GA flow diagram.

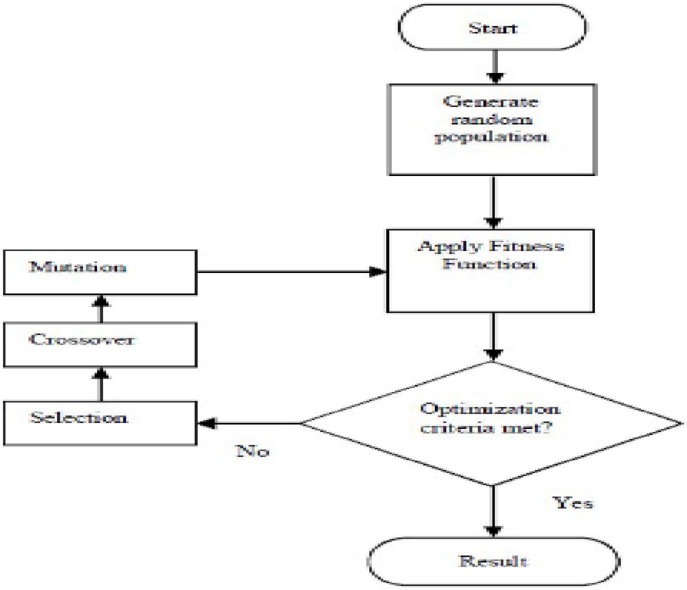

Figure 3. Flow diagram of GA

GA was used for feature selection as demonstrated in Table 3 out of which eighteen (18) attributes were selected from forty-one (41) attributes. The Table 2 gives the ranking order of the factors in relation to the class label (attacks), a total number of eighteen (18) attributes was selected from forty (40) including one (1) class label. These are factor the GA considered as factors that are optimum to predicting normal and attacks in intrusion dataset. The GA helps the classifier in disabling the problem of getting struck at local minima.

Table 3. Selected features

Selected Attributes	Index
'num_root'	16
'src_bytes'	5
'wrong_fragment'	8
'duration'	1
'hot'	10
'srv_count'	22
'is_guest_login'	20
'srv_serror_rate'	24
'num_access_files'	19
'su_attempted'	15
'num_shells'	18
'dst_host_rerror_rate'	38
'root_shell'	14
'num_compromised'	13
'flag'	4
'num_failed_logins'	11
'diff_srv_rate'	28
'dst_host_srv_diff_host_rate'	35

3.9. Support Vector Machine

SVM is a supervised machine learning classifier used for both classification and regression problem introduced by Vapnik in 1990s. SVM model maps from input space to a higher dimensional space to solve nonlinear classification problems where maximum hyperplane is established[57]. Hyper-plane is a linear method whose maximum margin allows for the maximum separation among the decision classes. In recent times, various good applications have been proposed by scholars because of the growing attention in SVM[58].

Suppose a training set T $\{(x_1, y_1),..,(x_1,y_1)\} \Sigma R^n$ x $\{-1,1\})^l$ where xi $\epsilon([x_i]_{1,...,}[x_i]_n)$ is the feature input vectors, $y_i \epsilon \{+1,-1\}$ is the equivalent output of x_i, the sample number is n, in the feature vector space. Through changing the b and w, we can limit the position of the separating hyper-plane. In respect to maximizing the margin, the optimization problem can be defined as follows.

$$\text{Min}_{w,b,\varepsilon_i} \frac{1}{2} | w ||^2 + C \sum_{i=1}^{l} \varepsilon_i$$
$$\text{s.t.} y_i((W^T.x_i)+b) \geq 1-\varepsilon_i, i=1,2,3,\ldots,l \quad (2)$$
$$\varepsilon_i \geq 0, i=1,2,3,\ldots,l$$

provided the constant C > 0; limits the trade-off amongst training-error minimization, margin maximization, and the slack-variable ε_i defined as some noises that cause the intersection of the classes [59]. SVM is a state-of-the-art, sophisticated classifier with high generalization ability [60]. SVM is a powerful classifier for the following reasons [61]:

Their need on moderately few support vectors means that they are very compact models, thus utilize very little memory.

Provided the model is trained; the SVM prediction stage is high-speed.

It performed well with high dimensional data and data that has more dimensions than samples, which is a severe problem for other algorithms.

They are versatile because of their incorporation with kernel methods and adaptability to different types of data.

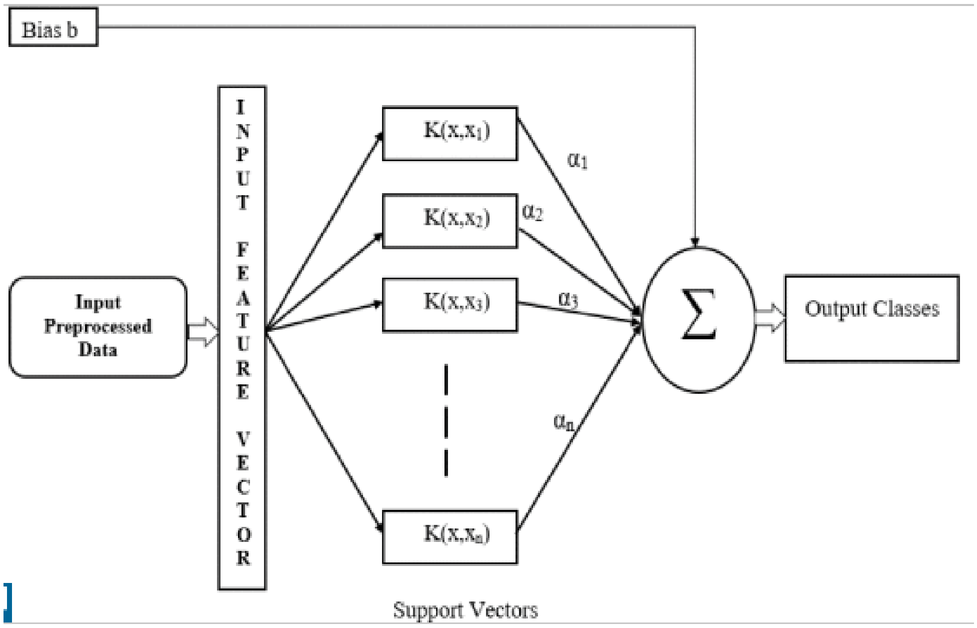

Figure 4. The framework of SVM for intrusion detection system

3.9.1 Experimental Setup

The data was separated into training and testing sets at this point, and the data was projected into training and testing sets. The system trained the Support Vector Machine classification algorithms with 75% of the data. The predictor's button loads the selected data using the GA wrapper feature selection technique, the response loads the class label, and the split rate was set at 0.25, indicating a 25% holdout from the data for testing the classification algorithm's efficiency and performance.

3.9.2 Tools and Software Used

The matrix laboratory environment widely known as Matlab was used for the experimental analysis. MATLAB is a high-performance technical computing language. It combines visualization, programming, and computing in a user-friendly environment where problems and solutions are written in mathematical notation.

3.9.3 Hardware Requirement

The tests were performed on a 64-bit Windows 10 Professional computer with an x64-based processor, 8.00 GB of RAM, and an Intel (R) Core (TM)i5-8250U CPU running at 1.60 GHz 1.80 GHz.

4.1. Results and Discussion

The performance of our proposed model is evaluated using 75% of the data for training the SVM classifier, and 25% was used for testing.

4.2. Experimental Results Evaluation

The experimental findings are observed based on the classification algorithm. The testing (probing) evaluation was achieved using the Accuracy, True Positive rate (TP) also known as Detection Rate (DR), False Positive also known as False Alarm Rate (FAR) and error rate as shown in Table 5[62]. The evaluation parameters for performance measurement of the proposed method were done using Accuracy, DR, FAR and Error rate as the performance metrics. To assess the efficiency and effectiveness of IDSs, various performance metrics has been proposed. DR and FAR is the most widely accepted metrics. A well-performed IDSs must have a high DR and low FAR.

Table 4 shows the analysis per each class based on the class label from the Normal, and attacks. The table highlights the TP value, the TN value, FP value and FN value of each of the class groups.

Table 4. Analysis per class

Analysis per class.	True Positive	True Negative	False Positive	False Negative
Class 1	2222	4023	19	34
Class 2	4023	2222	34	19

4.3. Confusion Matrix

Confusion matrix is a summary of prediction results on a classification problem[63],[64],[65].The confusion matrix is important, as it exactly shows correctly classified records and incorrectly classified records [66]. The number of correct and incorrect predictions are summarized with count values and broken down by each class. The class 1 represents the normal class which gives a total of 2256 from the test observation set, a total of 2222 was classified correctly and 34 was misclassified, the attack class is represented by label 2 gives a total of 4023 from the test observation set, a total of 4023 was classified correctly and 19 was misclassified.

Table 5. Confusion matrix

Confusion Matrix		
	1	2
1	2222	34
2	19	4023

In Table 6, the evaluation results of the SVM classifier based on the accuracy, detection-rate, false-alarm rate, and error rate is presented. The proposed model gave an accuracy of 99.16%, DR of 98.49, FAR of 0.4 and error rate of 0.0000841537.

Table 6. Evaluation Performance results of the proposed K-means+GA+SVM

Proposed Technique	Accuracy	Detection Rate	False Alarm Rate	Error Rate
Kmeans+GA+SVM	99.16	98.49	0.4	0.000841537

4.4. Comparison with other IDS Methods

In order to substantiate the results of our findings, we compared the results obtained with other methods in Figure 5. The results of our findings outperformed other works in terms of DR and FAR except for Parsaei et al.[68] and Akashdeep et al[70] that gave a slight higher DR. However, our FAR is better.

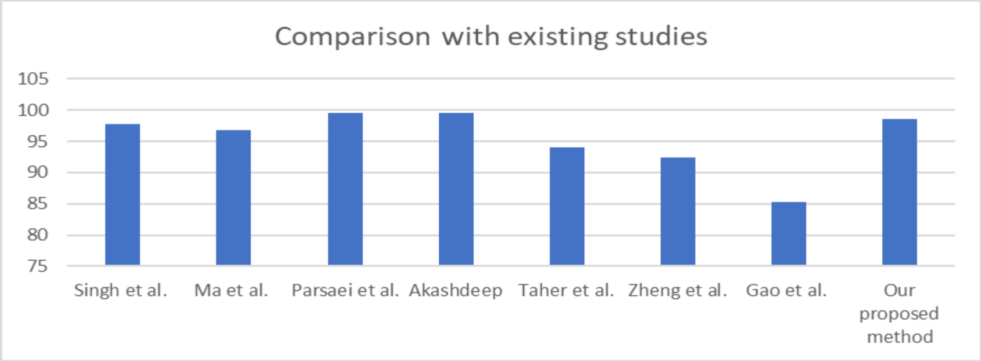

Figure 5. Comparison with other studies

4.5. Results comparison of proposed method training time

The actual computational time used in processing the proposed method in a time-critical application for training the dataset is taken, which is measured in terms of the total seconds taken for executing the training phase. The training time for the proposed model is demonstrated in Table 6. The training time shows the time taken by the model to create knowledge retention of the data supplied to the classifier. We noticed that many existing works did not pay attention to the training time. However, we are able to compare our proposed method with the work of [67] that proposed several classifiers. Our training time results in findings, as shown in Table 6 stands out when compared with other work training time.

4.6. Comparison of FAR with other state-of-art methods

We compared our experimental results obtained in terms of FAR with the works of Kuang et al.[68] which gave 1.03, Bamakan et al.[58] obtained 2.41, Lin et al. [69] gave 2.95 and Wang et al.[4] revealed 0.60. We noticed our results required less FAR as a result of the inherent generalization capability of SVM as shown in Figure 6.

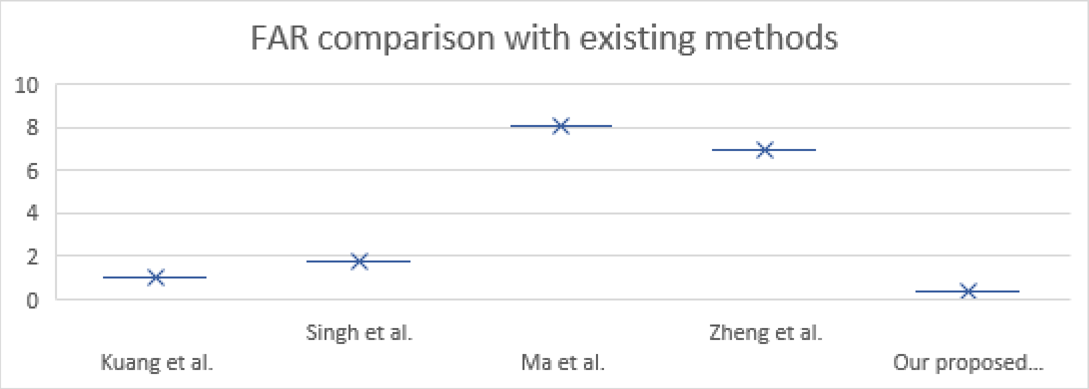

Figure 6. The FAR of the proposed model against other works

4.7. Result Analysis of the error rate

The error rate indicates the possible lowest error for the classifier in any random outcome in the classification phase. As shown in figure 7, the SVM algorithm gave the lowest error rate of 0.000841537, which is pointing to the fact that the proposed K-means-GA-SVM shows a very high detection rate.

Table 7. Comparison of training time of the proposed methods with other works

Authors	Algorithms	Training Time
Lopez-Martin et al.[67]	SVM	65.06
Lopez-Martin et al.[67]	KNN	91.53
Lopez-Martin et al.[67]	Reinforcement learning	507.01
Mehmood et al.[70]	SVM-ACO	4540
Our proposed model	SVM	51.98

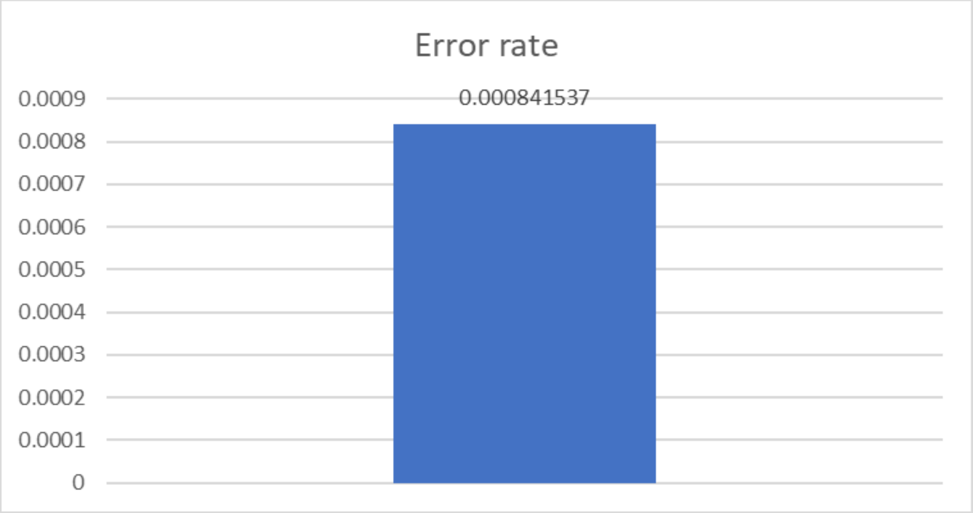

Figure 7. Error rate of the proposed model

5. Conclusion and future work

In recent time, IDS have become an essential component of network architecture because many of our critical national infrastructures now depend on the network. The significant issues in the performance of IDS based on the machine learning approach are DR and FAR. We proposed an efficient method in this paper using K-means-GA-SVM method. The K-means-GA-SVM method first uses k-means to differentiate benign and normal networks packet. Then, GA was employed to reduce the dimensionality of the data as the feature selection stage. Finally, SVM was used as the classifier, and the analysis was on NSL-KDD dataset. We noted the time taken for training the classifier in respect of time-critical application and error rate was also considered. The experimental results of our findings were compared with the state-of-the-art methods, and our results conclusions are superior in terms of DR, FAR, training time and error rate. In future work, we planned to introduce PSO for feature selection to replace GA and SVM for classification then compared the results with our proposed model.

6. References

[1]. Y. Zhou, G. Cheng, S. Jiang, and M. Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier," *Comput. Networks*, vol. 174, 2020, doi: 10.1016/j.comnet.2020.107247.

[2]. W. Fang, X. Tan, and D. Wilbur, "Application of intrusion detection technology in network safety based on machine learning," *Saf. Sci.*, vol. 124, no. December 2019, p. 104604, 2020, doi: 10.1016/j.ssci.2020.104604.

[3]. L. Lv, W. Wang, Z. Zhang, and X. Liu, "A novel intrusion detection system based on an optimal hybrid kernel extreme learning machine," *Knowledge-Based Syst.*, vol. 195, no. xxxx, 2020, doi: 10.1016/j.knosys.2020.105648.

[4]. H. Wang, J. Gu, and S. Wang, "An effective intrusion detection framework based on SVM with feature augmentation," *Knowledge-Based Syst.*, vol. 136, pp. 130–139, 2017, doi: 10.1016/j.knosys.2017.09.014.

[5]. W. L. Al-Yaseen, "Improving intrusion detection system by developing feature selection model based on firefly algorithm and support vector machine," *IAENG Int. J. Comput. Sci.*, vol. 46, no. 4, pp. 1–7, 2019.

[6]. R. Yahalom, A. Steren, Y. Nameri, M. Roytman, A. Porgador, and Y. Elovici, "Improving the effectiveness of intrusion detection systems for hierarchical data," *Knowledge-Based Syst.*, vol. 168, pp. 59–69, 2019, doi: 10.1016/j.knosys.2019.01.002.

[7]. A. Khraisat, I. Gondal, P. Vamplew, J. Kamruzzaman, and A. Alazab, "Hybrid intrusion detection system based on the stacking ensemble of C5 decision tree classifier and one class support vector machine," *Electron.*, vol. 9, no. 1, 2020, doi: 10.3390/electronics9010173.

[8]. I. Dutt, S. Borah, and I. K. Maitra, "Immune System Based Intrusion Detection System (IS-IDS): A Proposed," *IEEE Access*, vol. 8, pp. 34929–34941, 2020, doi: 10.1109/ACCESS.2020.2973608.

[9]. H. J. Liao, C. H. Richard Lin, Y. C. Lin, and K. Y. Tung, "Intrusion detection system: A comprehensive review," *J. Netw. Comput. Appl.*, vol. 36, no. 1, pp. 16–24, 2013, doi: 10.1016/j.jnca.2012.09.004.

[10]. L. S. Chen and J. S. Syu, "Feature extraction based approaches for improving the performance of intrusion detection systems," *Lect. Notes Eng. Comput. Sci.*, vol. 1, no. November, pp. 286–291, 2015.

[11]. S. M. Kasongo and Y. Sun, "A deep learning method with wrapper based feature extraction for wireless intrusion detection system," *Comput. Secur.*, vol. 92, 2020, doi: 10.1016/j.cose.2020.101752.

[12]. S. N. Mighan and M. Kahani, "A novel scalable intrusion detection system based on deep learning," *Int. J. Inf. Secur.*, no. 0123456789, 2020, doi: 10.1007/s10207-020-00508-5.

[13]. A. A. Aburomman, M. Bin, and I. Reaz, "A novel SVM-kNN-PSO ensemble method for intrusion detection system," vol. 38, pp. 360–372, 2016.

[14]. W. L. Al-yaseen, Z. A. Othman, M. Zakree, and A. Nazri, "Hybrid Modified ? -Means with C4 . 5 for Intrusion Detection Systems in Multiagent Systems," vol. 2015, 2015.

[15]. A. Shenfield, D. Day, and A. Ayesh, “Intelligent intrusion detection systems using artificial neural networks,” *ICT Express*, vol. 4, no. 2, pp. 95–99, 2018, doi: 10.1016/j.icte.2018.04.003.

[16]. B. Hssina, B. Bouikhalene, and A. Merbouha, “Europe and MENA Cooperation Advances in Information and Communication Technologies,” vol. 520, pp. 103–112, 2017, doi: 10.1007/978-3-319-46568-5.

[17]. B. Sweta *et al.*, “A Novel PCA-Firefly Based XGBoost Classification Model for Intrusion Detection in Networks,” *Electron.*, vol. 9, no. 2, p. 219, 2020.

[18]. Z. Wu, J. Wang, L. Hu, Z. Zhang, and H. Wu, “A network intrusion detection method based on semantic Re-encoding and deep learning,” *J. Netw. Comput. Appl.*, vol. 164, no. April, 2020, doi: 10.1016/j.jnca.2020.102688.

[19]. P. Borgen, K. Fuglseth, and R. Skarsten, “Md - Mn,” *Complet. Work. Philo Alexandria A Key-Word-In-Context Concord.*, pp. 3956–4200, 2019, doi: 10.31826/9781463210939-004.

[20]. W. Tao, W. Zhang, C. Hu, and C. Hu, “A Network Intrusion Detection Model Based on Convolutional Neural Network,” *Adv. Intell. Syst. Comput.*, vol. 895, no. 4, pp. 771–783, 2020, doi: 10.1007/978-3-030-16946-6_63.

[21]. K. Wang, “Network data management model based on Naïve Bayes classifier and deep neural networks in heterogeneous wireless networks R,” *Comput. Electr. Eng.*, vol. 75, pp. 135–145, 2019, doi: 10.1016/j.compeleceng.2019.02.015.

[22]. N. Farnaaz and M. A. Jabbar, “Random Forest Modeling for Network Intrusion Detection System,” *Procedia Comput. Sci.*, vol. 89, pp. 213–217, 2016, doi: 10.1016/j.procs.2016.06.047.

[23]. B. Selvakumar and K. Muneeswaran, “Firefly algorithm based Feature Selection for Network Intrusion Detection,” *Comput. Secur.*, 2018, doi: 10.1016/j.cose.2018.11.005.

[24]. M. A. Haque, A. Verma, J. S. R. Alex, and N. Venkatesan, *Experimental evaluation of cnn architecture for speech recognition*, vol. 1045. 2020.

[25]. M. S. Husain, “Nature Inspired Approach for Intrusion Detection Systems,” *Des. Anal. Secur. Protoc. Commun.*, pp. 171–182, 2020, doi: 10.1002/9781119555759.ch8.

[26]. V. R, M. Alazab, S. Srinivasan, Q.-V. Pham, S. Kotti Padannayil, and K. Simran, “A Visualized Botnet Detection System based Deep Learning for the Internet of Things Networks of Smart Cities,” *IEEE Trans. Ind. Appl.*, vol. 9994, no. c, pp. 1–1, 2020, doi: 10.1109/tia.2020.2971952.

[27]. R. Magán-Carrión, D. Urda, I. Díaz-Cano, and B. Dorronsoro, “Towards a reliable comparison and evaluation of network intrusion detection systems based on machine learning approaches,” *Appl. Sci.*, vol. 10, no. 5, pp. 1–21, 2020, doi: 10.3390/app10051775.

[28]. M. A. Hall and L. A. Smith, “Practical Feature Subset Selection for Machine Learning.”

[29]. H. Alazzam, A. Sharieh, and K. E. Sabri, “A feature selection algorithm for intrusion detection system based on Pigeon Inspired Optimizer,” *Expert Syst. Appl.*, vol. 148, p. 113249, 2020, doi: 10.1016/j.eswa.2020.113249.

[30]. J. Sharma, C. Giri, O. C. Granmo, and M. Goodwin, “Multi-layer intrusion detection system with ExtraTrees feature selection, extreme learning machine ensemble, and softmax aggregation,” *Eurasip J. Inf. Secur.*, vol. 2019, no. 1, 2019, doi: 10.1186/s13635-019-0098-y.

[31]. P. S. Muhuri, P. Chatterjee, X. Yuan, K. Roy, and A. Esterline, “Using a long short-term memory recurrent neural network (LSTM-RNN) to classify network attacks,” *Inf.*, vol. 11, no. 5, pp. 1–21, 2020, doi: 10.3390/INFO11050243.

[32]. Y. K. Saheed and F. E. Hamza-Usman, “Feature Selection with IG-R for Improving Performance of Intrusion Detection System,” *Int. J. Commun. Networks Inf. Secur*, vol. 12, no. 3, pp. 338–344, 2020.

[33]. A. Chandra, S. K. Khatri, and R. Simon, “Filter-based Attribute Selection Approach for Intrusion Detection using k-Means Clustering and Sequential Minimal Optimization Technique,” *2019 Amity Int. Conf. Artif. Intell.*, pp. 740–745, 2019.

[34]. S. M. Kasongo and Y. Sun, "A Deep Learning Method with Filter Based Feature Engineering for Wireless Intrusion Detection system," *IEEE Access*, vol. PP, no. DL, p. 1, 2019, doi: 10.1109/ACCESS.2019.2905633.

[35]. M. A. Ambusaidi, X. He, S. Member, P. Nanda, S. Member, and Z. Tan, "Building an intrusion detection system using a filter-based feature selection algorithm," vol. 9340, no. NOVEMBER 2014, pp. 1–13, 2016, doi: 10.1109/TC.2016.2519914.

[36]. M. S. Pervez and D. Farid, "Feature Selection and Intrusion classification in NSL-KDD Cup 99 Dataset Employing SVMs."

[37]. Y. K. Saheed, O. Longe, U. A. Baba, S. Rakshit, and N. R. Vajjhala, "An Ensemble Learning Approach for Software Defect Prediction in Developing Quality Software Product," in *Advances in Computing and Data Sciences. ICACDS 2021. Communications in Computer and Information Science*, 2021, doi: 10.1007/978-3-030-81462-5_29.

[38]. S. Aljawarneh, M. Aldwairi, and M. Bani, "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model," *J. Comput. Sci.*, vol. 25, pp. 152–160, 2018, doi: 10.1016/j.jocs.2017.03.006.

[39]. I. Sumaiya Thaseen and C. Aswani Kumar, "Intrusion detection model using fusion of chi-square feature selection and multi class SVM," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 29, no. 4, pp. 462–472, 2017, doi: 10.1016/j.jksuci.2015.12.004.

[40]. S. Mohammadi, H. Mirvaziri, M. Ghazizadeh-Ahsaee, and H. Karimipour, "Cyber intrusion detection by combined feature selection algorithm," *J. Inf. Secur. Appl.*, vol. 44, pp. 80–88, 2019, doi: 10.1016/j.jisa.2018.11.007.

[41]. Kunal and M. Dua, "Attribute Selection and Ensemble Classifier based Novel Approach to Intrusion Detection System," *Procedia Comput. Sci.*, vol. 167, no. 2019, pp. 2191–2199, 2020, doi: 10.1016/j.procs.2020.03.271.

[42]. K. Singh and K. J. Mathai, "Performance Comparison of Intrusion Detection System Between Deep Belief Network (DBN)Algorithm and State Preserving Extreme Learning Machine (SPELM) Algorithm," *Proc. 2019 3rd IEEE Int. Conf. Electr. Comput. Commun. Technol. ICECCT 2019*, pp. 1–7, 2019, doi: 10.1109/ICECCT.2019.8869492.

[43]. Ü. Çavuşoğlu, "A new hybrid approach for intrusion detection using machine learning methods," *Appl. Intell.*, vol. 49, no. 7, pp. 2735–2761, 2019, doi: 10.1007/s10489-018-01408-x.

[44]. B. M. Aslahi-Shahri *et al.*, "A hybrid method consisting of GA and SVM for intrusion detection system," *Neural Comput. Appl.*, vol. 27, no. 6, pp. 1669–1676, 2016, doi: 10.1007/s00521-015-1964-2.

[45]. V. Bolón-Canedo, N. Sánchez-Maroño, and A. Alonso-Betanzos, "Feature selection and classification in multiple class datasets: An application to KDD Cup 99 dataset," *Expert Syst. Appl.*, vol. 38, no. 5, pp. 5947–5957, 2011, doi: 10.1016/j.eswa.2010.11.028.

[46]. D. Zheng, Z. Hong, N. Wang, and P. Chen, "An improved LDA-based ELM classification for intrusion detection algorithm in IoT application," *Sensors (Switzerland)*, vol. 20, no. 6, pp. 1–19, 2020, doi: 10.3390/s20061706.

[47]. A. Meryem and B. EL Ouahidi, *Hybrid intrusion detection system using machine learning*, vol. 2020, no. 5. Springer Singapore, 2020.

[48]. I. Aljamal, A. Tekeoglu, K. Bekiroglu, and S. Sengupta, "Hybrid intrusion detection system using machine learning techniques in cloud computing environments," *Proc. - 2019 IEEE/ACIS 17th Int. Conf. Softw. Eng. Res. Manag. Appl. SERA 2019*, pp. 84–89, 2019, doi: 10.1109/SERA.2019.8886794.

[49]. U. Qamar and M. S. Raza, *Data Science Concepts and Techniques with Applications*. 2020.

[50]. P. L. Lekshmy and M. Abdul Rahiman, "Hybrid Approach to Speed-Up the Privacy Preserving Kernel K-means Clustering and its Application in Social Distributed Environment," *J. Netw. Syst. Manag.*, vol. 28, no. 2, pp. 398–422, 2020, doi: 10.1007/s10922-019-09511-1.

[51]. N. Acharya and S. Singh, “An IWD-based feature selection method for intrusion detection system,” *Soft Comput.*, vol. 22, no. 13, pp. 4407–4416, 2018, doi: 10.1007/s00500-017-2635-2.

[52]. Y. Saheed and A. Babatunde, “Genetic Algorithm Technique In Program Path Coverage For Improving Software Testing,” vol. 7, no. 5, pp. 151–158, 2014.

[53]. A. H. Hamamoto, L. F. Carvalho, L. D. H. Sampaio, T. Abrão, and M. L. Proença, “Network Anomaly Detection System using Genetic Algorithm and Fuzzy Logic,” *Expert Syst. Appl.*, vol. 92, pp. 390–402, 2018, doi: 10.1016/j.eswa.2017.09.013.

[54]. Y. Zhang, P. Li, and X. Wang, “Intrusion Detection for IoT Based on Improved Genetic Algorithm and Deep Belief Network,” *IEEE Access*, vol. 7, no. c, pp. 31711–31722, 2019, doi: 10.1109/ACCESS.2019.2903723.

[55]. A. Chaudhary and G. Shrimal, “Intrusion Detection System Based on Genetic Algorithm for Detection of Distribution Denial of Service Attacks in MANETs,” *SSRN Electron. J.*, pp. 370–377, 2019, doi: 10.2139/ssrn.3351807.

[56]. B. Chakrabarty, O. Chanda, and M. Saiful, “Anomaly based Intrusion Detection System using Genetic Algorithm and K-Centroid Clustering,” *Int. J. Comput. Appl.*, vol. 163, no. 11, pp. 13–17, 2017, doi: 10.5120/ijca2017913762.

[57]. I. M. Yulietha and S. A. Faraby, “Comparison between Support Vector Machine and Fuzzy C-Means as Classifier for Intrusion Detection System Comparison between Support Vector Machine and Fuzzy C- Means as Classifier for Intrusion Detection System,” 2018.

[58]. S. M. Hosseini Bamakan, H. Wang, T. Yingjie, and Y. Shi, “An effective intrusion detection framework based on MCLP/SVM optimized by time-varying chaos particle swarm optimization,” *Neurocomputing*, vol. 199, pp. 90–102, 2016, doi: 10.1016/j.neucom.2016.03.031.

[59]. Y. K. Saheed, “A Binary Firefly Algorithm Based Feature Selection Method on High Dimensional Intrusion Detection Data.,” in *Illumination of Artificial Intelligence in Cybersecurity and Forensics. Lecture Notes on Data Engineering and Communications Technologies*, S. Misra and C. Arumugam, Eds. Springer Cham, 2022.

[60]. L. Saitta, “Support-Vector Networks,” vol. 297, pp. 273–297, 1995.

[61]. J. Vanderplas, *Python Data Science Handbook*. .

[62]. Y. K. Saheed and M. A. Hambali, “Customer Churn Prediction in Telecom Sector with Machine Learning and Information Gain Filter Feature Selection Algorithms,” in *2021 International Conference on Data Analytics for Business and Industry (ICDABI)*, 2021, pp. 208–213, doi: 10.1109/ICDABI53623.2021.9655792.

[63]. F. Louati and F. B. Ktata, “A deep learning-based multi-agent system for intrusion detection,” *SN Appl. Sci.*, vol. 2, no. 4, pp. 1–13, 2020, doi: 10.1007/s42452-020-2414-z.

[64]. A. R. bhai Gupta and J. Agrawal, “A Comprehensive Survey on Various Machine Learning Methods used for Intrusion Detection System,” pp. 282–289, 2020, doi: 10.1109/csnt48778.2020.9115764.

[65]. M. Gao, H. Liu, Z. Zhang, Z. Ning, and J. Xu, “Malicious Network Tra ffi c Detection Based on Deep,” pp. 1–14, 2020.

[66]. Y. K. Saheed, “Performance Improvement of Intrusion Detection System for Detecting Attacks on Internet of Things and Edge of Things,” in *Artificial Intelligence for Cloud and Edge Computing. Internet of Things (Technology, Communications and Computing)*, S. Misra, T. K. A., V. Piuri, and L. Garg, Eds. Springer, Cham, 2022, pp. 321–339.

[67]. M. Lopez-Martin, B. Carro, and A. Sanchez-Esguevillas, “Application of deep reinforcement learning to intrusion detection for supervised problems,” *Expert Syst. Appl.*, vol. 141, p. 112963, 2020, doi: 10.1016/j.eswa.2019.112963.

[68]. F. Kuang, W. Xu, and S. Zhang, “A novel hybrid KPCA and SVM with GA model for intrusion detection,” *Appl. Soft Comput. J.*, vol. 18, pp. 178–184, 2014, doi: 10.1016/j.asoc.2014.01.028.

[69]. W. C. Lin, S. W. Ke, and C. F. Tsai, “CANN: An intrusion detection system based on combining cluster centers and nearest neighbors,” *Knowledge-Based Syst.*, vol. 78, no. 1, pp. 13–21, 2015, doi: 10.1016/j.knosys.2015.01.009.

[70]. T. Mehmood and H. B. M. Rais, “SVM for network anomaly detection using ACO feature subset,” *2015 Int. Symp. Math. Sci. Comput. Res. iSMSC 2015 - Proc.*, vol. 2015, pp. 121–126, 2016, doi: 10.1109/ISMSC.2015.7594039.

Yakub Kayode Saheed is currently an Assistant Professor with the American University of Nigeria. His research areas are intrusion detection, information security, bioinformatics, residue number system, machine learning, and Artificial Intelligence. He is a member of IEEE, the Internet Society, IAENG, SDWIC, and is a Certified Network Security Specialist. He has published in several local and international journals and conference proceedings.

Arowolo Micheal Olaolu, is a faculty of the Department of Computer Science at Landmark University, Omu-Aran Nigeria. He holds a Bachelor's Degree from Al-Hikmah University, Ilorin, Nigeria and a Master Degree from Kwara State University, Malete Nigeria, he completed his PhD from Landmark University, Omu-Aran Nigeria. His area of research interest includes machine learning, Bioinformatics, Datamining, Cyber Security and Computer Arithmetic. He has published widely in local and international reputable journals, he is a member of IEEE, IAENG, APISE, SDIWC, and an Oracle Certified Expert.

Abdulrauf Tosho is a senior lecturer at the Department of Computer Science, Faculty of Natural and Applied Sciences, Al-Hikmah University, Ilorin. He received the Bachelor MSc of Computer Science from the University of Ilorin, Nigeria, and a Ph.D of Multimedia Technology and Communication from Universiti Utara Malaysia. His current research interests include Artificial Intelligence, Mobile applications, Usability, Instructional Interface design, and Computers in teaching and learning. Among his research include interactive content for various groups of children including those with special needs.