DOI: 10.36108/ujees/5202.70.0173
UNIOSUN Journal of Engineering and Environmental Sciences. Vol. 7 No. 1. March. 2025

Development of an Intrusion Detection System Using Mayfly Feature Selection and Support Vector Machine Algorithms

Abdulsalam, S. O., Adewale, T., Saka, K. K., Abdulrauf, U. T.

Abstract Numerous security strategies have been used to control the threats associated with computer and network security. Methods such as access control, software and hardware firewall restrictions, and the encryption of private information. Nevertheless, these methods are insufficient because they all have serious drawbacks. As a result, using additional defense mechanisms, such as intrusion detection systems (IDS), becomes crucial. This research developed an effective intrusion detection system using mayfly feature selection and support vector machine algorithm. The SVM classifier achieved an accuracy of approximately 99.9% the precision is 99.75% sensitivity 100%, F-score 99.87% While the training time display an average time of 1.4630sec. The results of this study suggest that security professionals and researchers should consider adopting ensemble methods like AdaBoost, especially when combined with robust base learners such as SVM, in the development of intrusion detection systems for IoT networks

Keywords: Intrusion detection system, Support vector machine, Mayfly optimization algorithm, NSL KDD dataset.

I. Introduction

Numerous security strategies have been used to control the threats associated with computer and network security. Methods such as software and hardware firewall regulations, access control, and encryption of private information. However, these techniques are not enough as each one of the techniques possess significant limitations. Therefore, it becomes important to use other additional defense mechanism like intrusion detection system (IDS) [1].

Performance is a major issue with intrusion detection systems (IDSs), despite the fact that they are an established technology. Performance in this context refers to the rate at which real risks are identified while prospective threats are reported accurately. False positives are the kinds of errors where the system incorrectly reports an attack. Reducing the percentage of false positives and raising the true detection rate would enhance IDS performance [2].

The functions of IDSs can be summed up as follows: monitoring, analyzing, detecting, and stirring alerts. IDS are divided into two categories: host-based IDS (HIDS), which identifies threats on specific computers or hosts within the network, and network-based IDS (NIDS), which analyzes network traffic to identify cyber threats at the network level. IDSs employ two detection techniques: (1) anomaly detection, which is predicated on the idea that the attacker's behavior differs from that of the typical user, and (2) misuse detection, which finds assaults using signature databases that include signatures of previous attacks [3].

To be able to defend against emerging threats, an IDS needs to implement an anomaly-detection approach. The concept behind this method is that hostile behavior differs from

Abdulsalam, S. O., Adewale, T.
(Department of Computer Science, Kwara State University Malete, Nigeria)
Saka, K. K., Abdulrauf, U. T.
(Department of Computer Science, Al-Hikmah University Ilorin, Nigeria)

Corresponding Author::
sulaiman.abdulsalam@kwasu.edu.ng

typical user behavior, and by identifying anomalous activity, one can identify even new dangers [4]. (GuhThe training of a classifier model that uses several features to distinguish between two or more classes in a given collection of observations constitutes the essence of this task, which is a classification issue. Support vector machines (SVM) are among the effective classifiers that have been widely employed for intrusion detection [5].

As a result, feature selection is a crucial step in an IDS design. Optimized feature sets lower the false alarm rate, increase classification accuracy, and save computational time and expense [6]. In essence, choosing an optimal feature set and training classifiers on a set of optimal parameters are optimization issues, hence metaheuristics are obvious potential answers. This work is established on the idea that by combining two or more bio-inspired metaheuristic-based optimization techniques, the shortcomings of current feature selection and classification approaches can be mitigated and their performance enhanced.

The creation of a distributed denial of service detection model using ensemble machine learning techniques [7]. Ensemble machine learning (ML) models that integrate bagging, boosting, and stacking techniques are used in this work. The implementation was carried out using the agile software development process to allow for modifications at every level. The user interface was developed using the HTML, CSS, and JavaScript frameworks. The ensemble models were assessed using a number of assessment metrics. Compared to the other models, the Bagging Ensemble approach performed better, with an approximate F1-score of 95.61%, 97% precision, 94.88% recall, and 99.5% accuracy. The experimental findings demonstrated that while building a DDoS attack detection model, the bagging ensemble strategy is recommended. The problem of huge dimensionality features in this work should be addressed in future research, which should focus on lowering the feature by applying machine learning techniques for feature selection.

[8], examined the application of deep learning methods to the detection of distributed denial of service attacks in network traffic. The DIO4 dataset is used in this study to detect DDoS assaults using deep learning models, including recurrent neural networks (RNN), gradient recurrent units (GRU), and long short-term memory (LSTM). According to the experimental results, models perform similarly on the CICIDS2019 dataset, with an accuracy score of 0.99; however, there is a difference in execution time, with GRU demonstrating a shorter execution time than RNN and LSTM.

[9], proposed two new models for feature selection and intrusion detection, Particle Swam optimization and PSO-Artificial Neural Network, were proposed. The study used the UNSW-NB15 dataset for evaluation purposes. The evaluation criteria include recall, precision, false positive, false negative, and true positive. According to the findings, PSO and GWO are excellent choices for intrusion detection feature selection. Lastly, experiments demonstrate that the PSO-GWO-NB classifier performs better than the PSO-GWO-ANN classifier in terms of intrusion detection and feature selection. The results, which had an accuracy of 99.9% and 99.97%, were competitive when compared to previous studies. To strengthen the system, future research should add more attributes to the dataset.

[10], identified attacks on Internet of Things networks using machine learning in an intrusion

detection system (ML-IDS). In the first stage of this investigation, the UNSW-NB15 dataset was treated to feature scaling utilizing the Minimum-maximum (min–max) concept of normalization to reduce information leakage on the test data. Principal Component Analysis (PCA) was then used to lower dimensionality. The experimental outcomes were assessed using the following metrics: Mathew correlation coefficient (MCC), recall, F1, precision, accuracy, area under the curve, kappa, and validation data-set. The results were competitive with an accuracy of 99.9% and MCC of 99.97% when compared to previous research. To strengthen the system in the future, new features should be added to the dataset.

[11], suggested two distinct intrusion detection (ID) classification methods that use the Particle Swarm Optimization (PSO) algorithm for feature selection. The authors used PSO + Decision Tree (PSO+DT) and PSO + K-Nearest Neighbor (PSO+KNN) as classification approaches to classify the network abnormalities. This study used the KDD-CUP 99 dataset to validate the detection methods' results. The results showed that PSO+KNN display an accuracy of 96.71% which outperformed the (PSO+DT) classifier algorithm in terms of identifying network anomalies.

II. Materials and Method

This study offers a comprehensive framework for choosing the ideal collection of NSL-KDD dataset features that effectively describe typical traffic and differentiate it from anomalous traffic using support vector machines. Network intrusion detection patterns were identified using the training data set from the NSL-KDD Cup dataset, and the model was assessed using the test data set created from same dataset. The Mayfly algorithm is used in the suggested method to pick a subset of data in order to improve the accuracy of model classification, taking into account the variety and quantity of features of user behavior and network traffic. In order to identify the features that are significant and associated with the class label, the mayfly feature selection strategy has been applied in the suggested method. Figure 1 presents the system framework

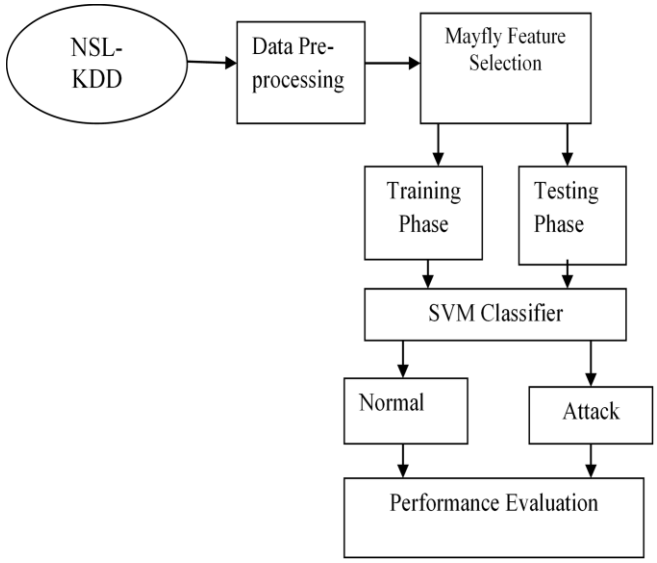

Figure 1: System Framework

A. NSL-KDD Dataset

The dataset used in this research is the NSL-KDD dataset which is a new dataset for the evaluation of researches in network intrusion detection system. It consists of selected records of the complete KDD 99 dataset. NSL-KDD dataset solve the issues of KDD 99 benchmark and connection record contains 41 features. Among the 41 features, 34 features are numeric and 7 features are symbolic or discrete. The NSL-KDD training set contains a total of 22 training attack types; with an additional 17 types in the testing set only.

B. Dataset Preprocessing

In the data preprocessing stage, raw data is cleaned, formatted, normalized, and transformed into an orderly, clean format that may be used for modeling or analysis. It involves feature scaling, normalization, standardization, resolving missing values, and eliminating duplicates. In essence, it ensures data consistency and quality, which lays the groundwork for efficient data analysis and modeling.

C. Mayfly Optimization Algorithm for Feature Selection

The open-source Anaconda Python environment was used for the experiment, and the Py_FS library was used for feature selection using the Mayfly Algorithm. 10,000 dataset samples were used to conduct the experiment.

To identify features that have a significant impact on our prediction and those that feature selection did not performe on the 41 features in the NSL KDD dataset.16 features were selected with Leader agent of fitness value 0.956 and execution time is 1907.4 seconds. The NSL-KDD dataset and user interface are depicted in the Table 1 below. The Mayfly algorithm was employed as a feature selection method to extract pertinent data from the dataset and feed it into the SVM classifier. The chosen data were sent to the training and testing sets, where they were divided into training and testing sets, respectively. For SVM classification algorithm, the system used 70% of the data for training and 30% for testing. Table 1 displays the list of features selected using Mayfly algorithm.

Table 1: List of Features Selected using Mayfly Algorithm

Feature Number	Feature Name	Feature Number	Feature Name
1	Duration	21	Is_host_login
3	Service	25	Serror_rate
9	Urgent	28	Srv_rerror_rate
11	Num_failed_logins	30	Diff_srv_rate
14	Root_shell	34	Dsv_host_same_srv_rate
17	Num_file_creations	35	Dsv_host_diff_srv_rate
18	Num_shells	37	Dsv_host_srv_diff_host_rate
20	Num_outbound_cmds	42	Class

D. Classification based on SVM

One common machine learning technique that can separate data into two classes is the SVM algorithm. The SVM method has strong usability in intrusion detection because it can differentiate between intrusion activity and the network's typical behavior. SVM is a discriminative classifier using a splitting hyperplane as its definition. SVM maps the training data into a higher-dimensional space using a kernel function, allowing for the linear classification of intrusions. SVMs are renowned for their capacity for generalization and are most useful when there are many attributes and few data points. Several kernels, including linear, polynomial,

Gaussian Radial Basis Function (RBF), and hyperbolic tangent, can be used to create different kinds of separating hyperplanes. In IDS datasets, a lot of attributes are either unnecessary or have little bearing on classifying data pieces. The SVM algorithm's performance is primarily determined by two parameters: penalty factor C and kernel parameter ϱ. Finding the ideal parameter value and optimizing the method's performance are significant challenges for the SVM algorithm. Therefore, features selection should be considered during SVM training. Training a SVM can be illustrated with the following pseudo code

Require: X and y loaded with training labeled data, α<= 0 or α<= partially trained SVM
1. C<= some value (10 for example)
2. repeat
3. for all {xi, yi}, {xj, yj} do
4. Optimize αi and αj
5. end for
6. until no changes in α or other resource constraint criteria met
Ensure: Retain only the support vectors (αi> 0)

E. Evaluation Criteria

This study executes the model prediction based on the evaluation measures that are established based on the confusion matrix in order to assess the performance of the produced model. (True Positive TP, False Positive FP, True Negative TN and False Negative FN). The metrics listed below can be used to evaluate the evaluation measures.

Classification rate or Accuracy: one of a classification algorithm's most crucial performance metrics, demonstrating the algorithm's capacity to precisely forecast both positive and negative instances, as shown in Eqs 1-4.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Precision or the positive predictive value**:** is used to describe the proportion of accurately anticipated positive observations to all predicted positive observations.

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

Recall known as sensitivity: refers to the accurately calculated real positive rate.

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

F-score: is the harmonic mean of the precision and recall.

$$F - score = \frac{2\ X\ Precision * Recall}{Precision+Recall} \quad (4)$$

III. Results and Discussion

The dataset are labeled accordingly. During the feature selection procedures, the dataset was separated into training and testing sets in order to evaluate the performance of the trained model. The system performance was assessed using a different test set that wasn't used for training after the model had been trained and verified. A thorough analysis of the model's performance in terms of training and testing

times on the Mayfly Optimization algorithm using the NSL-KDD dataset is given in Table 2.

Table 2: Observed Training and Testing Time of Mayfly Optimization Algorithm using NSL-KDD Dataset

Feature Selection Algorithm	Training Time (s)	Testing Time (s)
Mayfly Optimization Algorithm	1.4630	1.1742

Table 2 discusses the Mayfly Optimization algorithm's training and testing times using the NSL-KDD dataset. However, the experimental results from the feature selection algorithm perform better.

Table 3: Result of SVM Classifier on NSL-KDD Dataset for DDoS

Evaluation Measure (%)	SVM Classifier
Accuracy	99.87
Sensitivity	100
Precision	99.75
F1-Score	99.87

From Table 3, the SVM classifier achieved an accuracy of approximately 99.87%, the precision is 99.75%, Sensitivity 100%, F1score 99.87%. Figure 2 display the graphical result of SVM classifier on NSL-KDD Dataset.

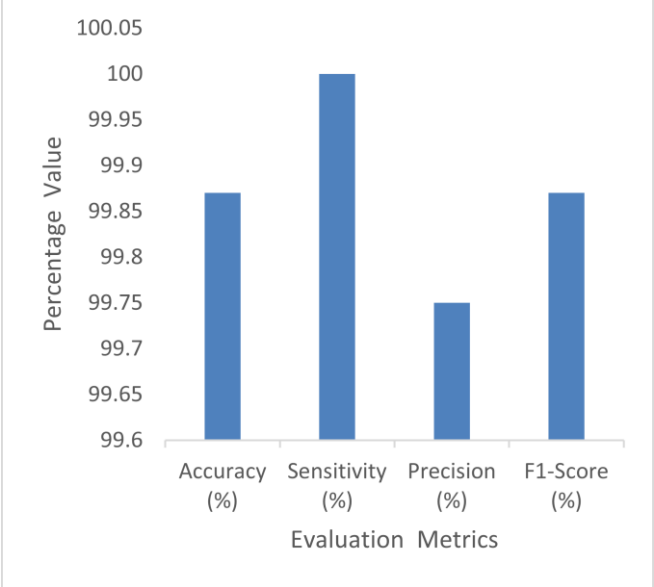

Figure 2: Result of SVM Classifier on NSL-KDD Dataset for DDoS

IV. Conclusion

This study sought to improve the effectiveness and precision of intrusion detection systems (IDS), by combining the Support Vector Machine (SVM) algorithm with the Mayfly feature selection technique. The specific goals included using SVM for improved classification, maximizing system efficiency through intelligent feature selection, and assessing IDS performance with an emphasis on accuracy and computing economy. The research employed a thorough framework that was derived from the NSL-KDD dataset. The NSL-KDD Cup dataset was used for training, and the Mayfly algorithm was used to pick features in order to increase the model's classification accuracy. The study is important because it tackles the ever-changing cyber threat landscape and overcomes the difficulties caused by duplicated or unnecessary components in conventional IDS. Further study should explore the application of Mayfly feature selection in different cyber threat scenarios.

References

[1] Abolarinwa, M. O., Adegoke, E. A., Ojo, O. E., Adewuya, A. M., Bakare, O. S. and Adigun, O.I. "Development of a Distributed Denial of Service Detection Model Using Ensemble Machine Learning Techniques". Adeleke University Journal of Science (AUJS), Vol. 3 No. 9, 2024, PP.27-32

[2] Balasaraswathi, V., Sugumaran, M. and Hamid, Y. "Feature Selection Techniques for Intrusion Detection using Non-Bio-Inspired and Bio-Inspired Optimization Algorithms". Journal of Communications and Information Networks, Vol. 10 No. 4, 2017, PP. 107–119

[3] Farahnakian, F., and Heikkonen, J., A. "Deep Auto-encoder Based Approach for Intrusion Detection System". In International Conference on Advanced Communication Technology, IEEE, Vol. 12 No 24, 2018, PP. 39-45

[4] Guha, V.N., Chatterjee, B.T. Hassan, S. K. Ahmed, S. Y. Bhattacharyya, T.N. and Sarkar, R. A. "Python Package for Feature Selection using Meta-heuristic Optimization Algorithms". Journal of Computational Intelligence in Pattern Recognition, Vol. 2 No. 9, 2022, PP. 56-59

[5] Hamdan, A., Alwadan, T. Almomani, O. Smadi, S. and Elomari, N. "Bio-inspired Hybrid Feature Selection Model for Intrusion Detection". Journal of Computers Communication, Vol. 5 No. 4, 2022, PP. 133–150

[6] Idris, S., Oyefolahan, O. and Ndunagu, N. "Intrusion Detection System Based on Support Vector Machine Optimised with Cat Swarm Optimization Algorithm". 2nd International Co nference of the IEEE Nigeria Computer Chapter, Vol. 16 No. 15, 2019, PP. 1–8

[7] Kuang, F. N., Xu, W. and Zhang, S. A. "Novel Hybrid KPCA and SVM with GA Model for Intrusion Detection. Elsevier Journal of Applied Soft Computing, Vol. 18 No. 8, 2023, pp. 178 – 184

[8] Kirsal, Y., Sekeroglu, B. and Dimililer, K. "Classification Analysis of Intrusion Detection on NSL-KDD Using Machine Learning Algorithms". In Proceedings of the International Conference on Mobile Web and Intelligent Information Systems, Vol. 7 No. 24, 2019, PP.111–122

[9] Mahrukh, R., Muhammad, S. Shazia, A. Faiza, I. Ángel, K.C. and Imran A. "Distributed Denial of Service Attack Detection in Network Traffic Using Deep Learning Algorithm". Journal of Advance Technology, Vol. 9 No. 13, 2023, PP 45-55

[10] Yakub, J. S., Aremu, I. A. Sanjay, M. Monica, K. H. and Ricardo, C. P. "A Machine Learning-based Intrusion Detection for Detecting Internet of Things Network Attacks". Journal of Alexandria Engineering, vol. 7 No. 10, 2022, PP. 67-72