RESEARCH & INNOVATION DIVISION
IJIST

Hybrid Intrusion Detection System Based on Optimal Feature Selection and Evolutionary Algorithm for Wired Networks

Husnain Babar [1], Muhammad Imran[1], Anees Tariq[1*]
[1]Department of Robotics and Artificial Intelligence, SZABIST University, Islamabad, Pakistan
*** Correspondence:** Anees Tariq and anees.tariq@szabist-isb.edu.pk

Citation | Babar. H, Imran. M., Tariq. A, "Hybrid Intrusion Detection System Based on Optimal Feature Selection and Evolutionary Algorithm for Wired Networks", IJIST, Vol. 07 Issue.02 pp 916-925, May 2025.
Received | April 17, 2025 **Revised |** May 19, 2025 **Accepted |** May 21, 202 **Published |** May 23, 2025.

The field of cybersecurity encounters ongoing difficulties in identifying and preventing attacks in networks, and the pervasive threat of cyberattacks demands continual advancements in intrusion detection systems (IDS) to safeguard network integrity. Traditional intrusion detection systems face the challenge of class imbalance. Addressing the formidable challenges posed by class imbalance and high-dimensional data, this research proposes a novel hybrid IDS approach. Leveraging (ACO), the algorithm navigates complex datasets to identify salient features, effectively mitigating the complexities associated with high-dimensional data. Subsequently, a Weighted Stacking Classifier amalgamates the strengths of Random Forest, AdaBoost, and Gradient Boosting classifiers, fortifying the system's ability to handle class imbalance robustly. By strategically enhancing the importance of base classifiers with favorable training outcomes and diminishing the influence of those yielding inferior results, the hybrid IDS endeavors to optimize classification efficacy. The experimentation, conducted exclusively on the dataset named NSL-KDD, demonstrates the efficacy of the proposed model, yielding remarkable results. With a 90.13% Accuracy, 88.87% precision, 91.23% Recall, and 87.33% F1-score, the hybrid IDS exhibits superior performance in detecting malicious activity. The findings underscore the viability of the proposed hybrid IDS as a potent tool in the ongoing battle against cyber threats, positioning it for real-world deployment across diverse networks.

Keywords: Intrusion Detection System; Ant Colony Optimization; Feature Selection; High-Dimensional Data; Weighted Stacking Classifier.

Introduction:

As network communication has evolved, particularly with the rise of cloud computing and the Internet of Things (IoT), these technologies have become an integral part of contemporary life [1]. As cyber threats grow more sophisticated, organizations face increasing risks to their digital assets, necessitating robust Intrusion Detection Systems (IDS) to safeguard networks and systems. Recent reports, including the "2021 Bad Bot Report," reveal that just 59.2% of network traffic comes from human users, while a significant 25.6% is attributed to malicious automated bots [2]. These statistics underscore the importance of advanced IDS solutions to combat evolving threats effectively. Traditional IDS approaches, such as port-based and Deep Packet Inspection (DPI) methods, often fall short in dynamic environments where port usage is randomized or packet contents are inaccessible [3].

Consequently, researchers have shifted their concern towards machine learning techniques that analyze data stream features without relying on packet content. Among these, stacking algorithms have shown promise by leveraging multiple base classifiers to enhance model diversity and generalization. However, high-dimensional datasets and the presence of irrelevant features pose significant challenges to IDS efficiency and accuracy [4]. Y. Zhou et al. [5] proposed a HID framework combining Ant Colony Optimization (ACO) for feature selection and a weighted stacking algorithm for classification. The ACO algorithm optimizes feature subsets by simulating pheromone-based decision-making, effectively addressing the high-dimensional data problems. The weighted stacking algorithm assigns dynamic weights to base classifiers based on their performance, improving classification accuracy. This research demonstrates the effectiveness of the proposed system through experiments on the dataset named NSL-KDD.

Background Study:

Contemporary research in intrusion detection mainly focuses on two key aspects: selecting relevant features and accurately classifying various types of cyberattacks. Author suggested a feature selection algorithm for IoT intrusion detection that relies on Information Gain (IG) and Gain Ratio (GR) [6]. The algorithms LSTM and CNN merged the top 50% of IG and GR features, achieving improved performance on IoT-BoT and KDD CUP 1999 datasets [7]. Experiments conducted on the NSL-KDD and CIC-IDS-2017 datasets demonstrated the method's effectiveness in both feature reduction and enhancing detection accuracy. Author proposed a hybrid feature selection technique that combines Particle Swarm Optimization (PSO) with the chi-square method, achieving impressive performance on the NSL-KDD dataset [8]. This proposed study builds on these advancements by employing Ant Colony Optimization (ACO), a bio-inspired algorithm that dynamically evaluates and optimizes feature subsets to improve IDS classification accuracy and computational efficiency.

The ensemble method of stacking has been widely utilized in the field of intrusion detection due to its ability to enhance classification performance. One notable application of this approach was demonstrated by author who developed a stacking-based classification model specifically designed to improve the accuracy of intrusion detection in surveillance systems. Their ensemble model integrated multiple base classifiers, including Random Forest, LightGBM, and XGBoost, each contributing distinct predictive strengths to improve overall performance. To further refine the final classification output, they employed a Multi-Layer Perceptron (MLP) as the secondary or meta-classifier. This combination of diverse machine learning techniques aimed to optimize detection accuracy and robustness, making the system more effective in identifying potential intrusions. [9]. In another study, author focused on meta-classifiers in stacking and found that Meta Decision Tree (MDT) performed best among multiple alternatives [10]. Author combined secondary classifiers using grid search optimization to improve stacking performance [11]. Furthermore, author developed an ML model using DT, RF, KNN, and DNN, achieving 85.2% accuracy on the KDDTest+ dataset.

However, stacking methods are often affected by the poor performance of certain base classifiers, which can degrade overall results [12][13]. Meanwhile, author enhanced stacking algorithms by employing grid search to identify the most effective meta-classifiers. In parallel, author introduced adaptive ensemble models that integrated decision trees, random forests, and neural networks, achieving an accuracy of 85.2% on the KDDTest+ dataset [14][15]. However, the presence of underperforming base classifiers remains a challenge, as they can negatively impact the overall effectiveness of the ensemble model. Hybrid approaches have emerged as a promising direction, combining feature selection with ensemble methods. Author introduced a multi-measure feature selection algorithm, integrating chi-square, Information Gain, and ReliefF with decision tree classifiers, leading to improved results [16]. Author enhanced stacking performance using artificial bee colony algorithms, while author integrated information gain-based feature selection with stacking techniques, demonstrating significant accuracy improvements on datasets like UNSW-NB15 [17][18].

Objectives:

- Our proposed model introduces a hybrid IDS framework integrating ACO-based weighted stacking classification and feature selection to enhance accuracy and efficiency.
- ACO is employed to optimize feature subsets, reduce dimensionality and retaining critical attributes for effective intrusion detection.
- The weighted stacking classifier improves classification performance on unbalanced datasets by emphasizing high-performing base classifiers and down-weighting underperforming ones.
- Experiments conducted on the NSL-KDD dataset validate the proposed framework, showing improvements in performance metrics with existing methods.
- To develop a hybrid intrusion detection system using Ant Colony Optimization and a Weighted Stacking Classifier to improve detection accuracy and address class imbalance in high-dimensional network data.

Materials and Methods:

This section provides a detailed overview of all the methodologies employed in our proposed Intrusion Detection System (IDS). The hybrid intrusion detection system follows a structured methodology, as illustrated in Fig. 1. The overall process primarily consists of the following key stages:

The data loading phase supplies the model with both training and test sets, ensuring a structured learning process. During data pre-processing, several techniques are employed to enhance data quality and optimize model performance. One-hot encoding is applied to expand categorical features, allowing the model to interpret them effectively. Additionally, variance filtering is used to eliminate low-variance features that contribute little to classification accuracy. This imbalance in data distribution can impact the accuracy of classification results, particularly for less common samples like R2L and U2R. Data Preparation, The dataset underwent comprehensive preprocessing steps, commencing with the loading of both the "training set" and "testing set." Categorical features were transformed through one-hot encoding, enhancing model compatibility. Numerical features underwent scaling via Standard Scaler to standardize their magnitudes. To streamline the dataset, a Variance Threshold technique was employed, systematically reducing the number of features. These preprocessing measures collectively optimize the data for machine learning algorithms, fostering improved model performance and generalization capabilities.

To standardize the dataset, we employed min-max normalization, which scales all feature values to a uniform range between 0 and 1, ensuring consistency and comparability across features. Furthermore, the log1p function was used to smooth the data, transforming

it into a distribution that more closely resembled a Gaussian curve. This step enhanced the model's ability to detect subtle patterns within the dataset. Feature dimensionality reduction is performed by evaluating the variance of each feature. Features with variance below a predefined threshold are removed, ensuring that only the most informative attributes are retained. To further refine feature selection, we proposed an Ant Colony Optimization (ACO), based approach that efficiently searches for the optimal feature subset, enhancing both model efficiency and accuracy.

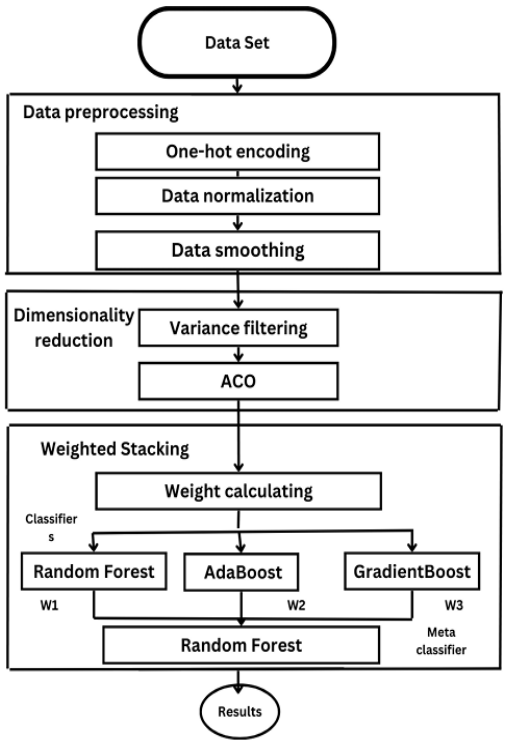

Figure 1. Flow Diagram of Proposed Model

To improve intrusion detection classification, we introduced a weighted stacking algorithm designed to enhance the precision of IDS predictions. This method assigns adaptive weights to the predictions of base classifiers, based on their performance. Highly accurate classifiers receive increased weight, while less accurate ones are assigned lower weights. By dynamically adjusting these weights, the algorithm strengthens overall classification performance, making the IDS more robust and reliable.

ACO For Feature Selection:

The ACO (Ant Colony Optimization) algorithm serves as a pivotal component in assessing feature importance and determining a subset crucial for attack detection [19]. By utilizing a designated fitness function, it directs the search process toward identifying the optimal subset of features. Through iterative evaluation and pheromone trail updates, ACO effectively identifies the most pertinent features, thereby proficiently reducing data dimensionality [15]. This strategic feature selection process enhances the efficiency of the overall intrusion detection system, ensuring that the selected features play a pivotal role in accurately identifying and classifying potential attacks within the dataset. Combining variance filtering with the Ant Colony Optimization (ACO) algorithm streamlines the dimensionality reduction process by retaining only the most relevant and discriminative features, thus boosting the efficiency and overall performance of the later stages in the intrusion detection system. By leveraging Ant Colony Optimization (ACO), the intrusion detection system uses a metaheuristic algorithm inspired by the cooperative foraging behavior of ants [20]. This approach replicates the efficient communication and collaboration seen in ants as they

collectively identify the shortest path between their colony and food sources. Feature selection using ACO involves:

Initialization:

Initialize the pheromone trails:

$$T_{i,j}^{(0)} t_0, \ \forall i,j \in features \quad (1)$$

where $T_{i,j}^{(0)}$ is the initial pheromone level between feature i and feature j, and τ0 is the initial pheromone value.

Ant's Behavior:

The selection probability $p_{i,j}^{k}$ For feature j at iteration k is calculated using the pheromone trail. $\tau_{i,j}^{k}$ And the heuristic information ni,j.

$$p_{i,j}^{k} = \frac{(T_{i,j}^{k})^{\propto}.(Heuristic_{i,j})^{\beta}}{\sum l \in Candidate\ Features\ (T_{il}^{k})^{\alpha}.(Heuristic_{il})^{\beta}} \quad (2)$$

where α and β are the pheromone and heuristic information influence parameters, respectively.

Pheromone Update:

After each ant constructed a solution, the pheromone trails were updated based on the quality of the solutions.

$$T_{i,j}^{k} = (1-\rho).T_{i,j}^{k-1} + \sum_{m=1}^{M} \Delta T_{i,j}^{m} \quad (3)$$

where ϱ is the pheromone evaporation rate, M is the total number of solutions, and Quality(m) is the quality of the m-the solution.

Termination Criteria:

The ACO algorithm terminates once a predefined condition is met, such as reaching the maximum number of iterations or the convergence of solutions. Dataset Detail Acquiring a dependable dataset for intrusion detection proved challenging due to issues related to data diversity, balance, and its relevance to real-world applications. The NSL-KDD dataset was selected as it effectively addresses these challenges and is widely recognized within the research community. It was obtained from the online repository Kaggle [21] and represents a diverse set of network traffic data, including both normal and malicious activities.

Table 1. Distribution of Training and Testing Samples Across Different Attack Classes in the Dataset

Class	Training Set	Testing Set
Normal	67,343	9,711
DoS	45,927	7,456
Probe	11,656	2,421
R2L	995	275
U2R	52	52

Hardware/ Software Utilized:

Experiments were carried out on a laptop equipped with an "Intel(R) Core (TM) i5-7200U CPU @ 2.50GHz, 2.70 GHz processor" and 12 GB of RAM. The system ran a 64-bit operating system with an x64-based architecture. The experiments were conducted in a local Python environment using PyCharm.

Results and Discussion:

Dataset:

To assess the performance of our proposed model, we opted for the NSL-KDD in our experiments. The NSL-KDD dataset is a huge collection of real-world examples of network traffic. It shows a wide range of activities that can happen on a network, both good and bad.

NSL-KDD:

Enhancing testing efficacy, the dataset's inclusivity of diverse samples enhances its role as a robust evaluation tool for intrusion detection models. Comprising 42 attributes per record, 41 delineate intricate data characteristics, while the remaining one specifies the attack type. Attacks are categorized into five classes. Figure 2 shows how the data are distributed in the "training set" and "testing set". Figure 2 indicates that most of the data consists of Normal and DoS samples, with very few R2L and U2R samples, especially in the training set.

Table 2 Intro to NSL-KDD

Types	Features	Description
DoS	Neptune, Back, Smurf, Teardrop, Pod, land	Attempts to make a network unavailable.
Probe	Ipsweep, Nmap, Satan, Imad, Spy	Scanning and port monitoring
User to Root	buffer overflow, load module, Perl, rootkit	Using a device remote for unauthorized access.
Remote to Local	guess_passwd, ftp_write, multihop, phf, imap, warezmaster	unauthorized access from a remote machine to a local system unauthorized access.
Normal	Normal	Legitimate network activity.

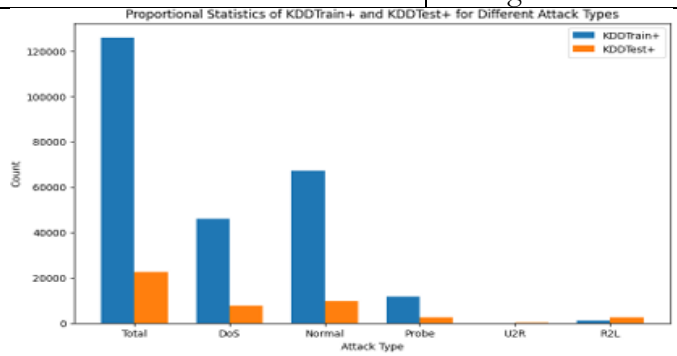

Figure 2 Features Importance in IDS Classification.

Data Splitting: The dataset underwent standard division into two sets: a training set for model training and a testing set for performance assessment. The model was trained on the training set, and its efficacy was subsequently evaluated using the distinct testing set.

Classifier/Regressor Selection: Three base classifiers were selected for the stacking ensemble.

- Random Forest Classifier (RF)
- AdaBoost Classifier
- Gradient Boosting Classifier

The stacking ensemble used a final Random Forest Classifier as a meta-classifier.

Evaluation Metrics:

In gauging the effectiveness of our proposed model, we employed various performance metrics. Among these metrics is the Confusion Matrix, a structured table that provides insight into the model's proficiency in attack classification. It enumerates the counts of "True Positive" (correctly identified normal traffic), "False Positive" (misclassification of normal instances as attacks), "True Negative" (correctly identified attack instances), and "False Negative" (misclassification of attack instances as normal). To break it down further:

True Positive (TP): Instances of normal traffic that were accurately classified.

False Positive (FP): Instances of attack traffic that were incorrectly classified.

True Negative (TN): Instances of normal traffic that were accurately classified.

False Negative (FN): Instances of normal traffic that were incorrectly classified as attack traffic.

To evaluate the model's effectiveness in attack classification, we used four key performance metrics. **Accuracy**, the accuracy represents the proportion of correctly predicted samples to the total number of samples, expressed as a percentage.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

Precision: Denotes the ratio of accurately classified instances to the overall instances predicted as positive.

$$Precision = \frac{TP}{TP+FP} \quad (5)$$

Recall: It is the percentage of correctly predicted instances of a certain type to the total number of instances of that type.

$$Recall = \frac{TP}{TP+FN} \quad (6)$$

F1-score: It is a balanced measure that considers both precision and recall. It is the harmonic mean of precision and recall.

$$F1 - Score = \frac{2.(Precision.Recall)}{Precision+Recall} \quad (7)$$

Results:

The proposed model was evaluated to assess its performance across various attack categories. It demonstrated outstanding performance, particularly in detecting DoS attacks. The classification results based on five intrusion categories are shown in Table 3. Figure 3 shows a breakdown of performance metrics for each intrusion category as well as normal network activity.

Table 3: Classification Results for Five Intrusion Categories.

Category	Accuracy	Precision	Recall	F1-Score
DoS	99.63	99.25	99.63	99.43
Probe	98.44	90.83	98.44	94.48
U2R	11.81	99.99	11.81	21.12
R2L	12.26	98.29	12.26	21.8
Normal	98.65	85.10	98.65	91.37

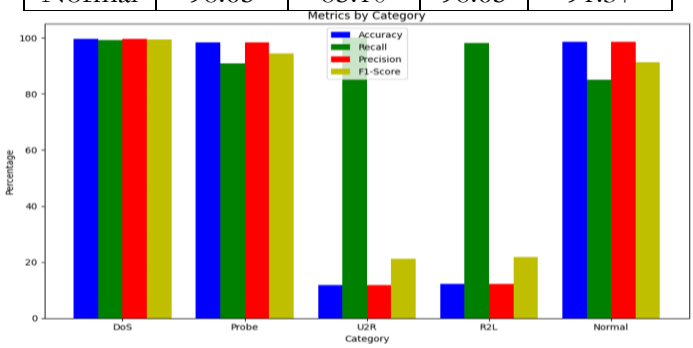

Figure 3 Graphical Representation of Results for Five Intrusion Categories.

The feature importances were calculated using a method that assigns weights to each feature based on its contribution to the overall classification accuracy. To understand the proposed model's performance in deep, a confusion matrix was constructed based on the classification results. The confusion matrix, in Figure 4, provides an overview of the actual and predicted labels across different attacks and normal network operations. The values in the matrix represent the count of instances where the predicted class matches the actual class. For instance, the value at row 1, column 2 indicates the number of instances where the actual class was "normal," but it was predicted as "DoS." The diagonal elements represent correctly classified instances, while off-diagonal elements represent misclassifications. This matrix provides a comprehensive view of the classifier's performance across different classes, aiding

in the assessment of its accuracy and effectiveness in distinguishing between various types of network activities.

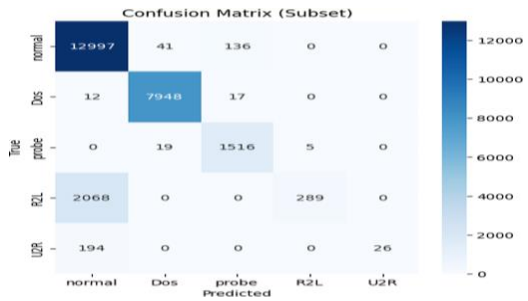

Figure 4 Confusion Matrix for IDS Classification

Performance Analysis of Meta-Classifiers in an Ensemble Model:

Table 4 presents the evaluation metrics for three different meta-classifiers in an ensemble model. Notably, the results indicate varying levels of accuracy, precision, F1-score, and recall across the different meta-classifiers. Gradient Boost achieved an accuracy of 85.20%, ADA Boost demonstrated 87.0% accuracy, and Random Forest outperformed with 90.13% accuracy. These metrics offer insights into the nuanced impact of each meta-classifier on the overall model performance.

Table 4: Performance Analysis of Meta-Classifier

Meta Classifier	Accuracy	Precision	Recall	F1-Score
Gradient Boost	85.20	86.50	85.20	84.90
ADA Boost	87.09	85.31	87.11	85.78
Random Forest	90.13	88.87	91.23	90.03

Conduct a comprehensive evaluation of our proposed Hybrid Intrusion Detection System (HIDS) by comparing its performance against several state-of-the-art techniques [6][19][15][22], as outlined in Table 5.

Table 5*:* Proposed Model with Existing Model

References	Accuracy	Precision	Recall	F1-Score
[6]	82.16	88.22	71.00	81.20
[19]	83.29	86.02	85.50	85.78
[15]	85.50	86.07	85.50	85.78
[22]	87.44	89.09	87.44	88.25
Proposed	90.13	88.87	91.23	90.03

The results demonstrated that our model achieved an impressive accuracy rate of 90.13%, highlighting its superior performance in comparison to other prominent existing models. Moreover, our model demonstrated high precision and recall rates of 88.87% and 91.23%, respectively, further solidifying its efficacy in accurately detecting intrusions. These metrics not only outperformed those of the compared models but also closely matched them in certain instances, underscoring the reliability and robustness of our proposed approach. Additionally, the F1-score, a composite metric that balances precision and recall, stood at a commendable 87.33%, indicating the model's effectiveness in mitigating both false positives and false negatives. Overall, these findings highlighted the significant potential of our HIDS model in enhancing cybersecurity measures and detecting malicious activities with high accuracy and efficiency. Figure 5 illustrates the graphical representation of the comparative analysis.

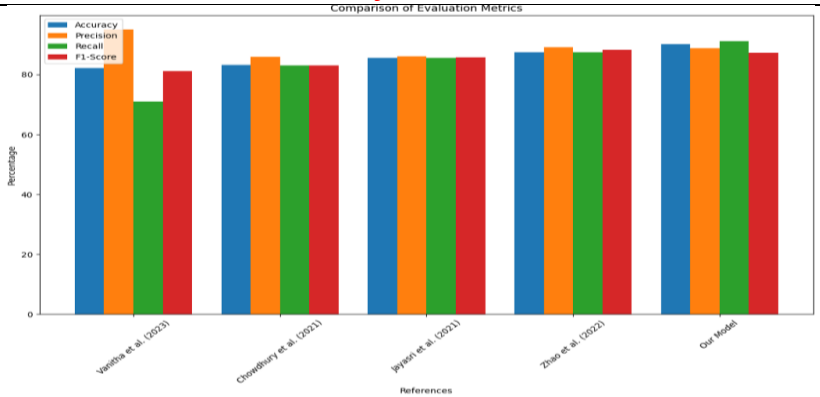

Figure 5. Comparative Analysis

Conclusion:

This study proposed a hybrid intrusion detection system utilizing Ant Colony Optimization (ACO) for feature selection and a weighted stacking classifier for improved classification accuracy and computational efficiency. The ACO algorithm effectively reduced the dimensionality of the NSL-KDD dataset, selecting an optimal subset of 18 features while maintaining critical discriminatory power. The proposed system achieved notable performance, with an accuracy of 89.56%, precision of 90.87%, recall of 89.34%, and an F1-score of 90.03%. Compared to the other benchmark approaches, the ACO-based method demonstrated improvement in accuracy and an increase in F1-score. These results highlight the potential of combining ACO and ensemble learning methods to enhance the effectiveness and efficiency of intrusion detection systems.

Future recommendations:

Future work will involve testing the proposed framework on additional datasets and optimizing its scalability for larger, real-world network environments.

References:

[1] R. Fotohi and S. Firoozi Bari, "A novel countermeasure technique to protect WSN against denial-of-sleep attacks using firefly and Hopfield neural network (HNN) algorithms," *J. Supercomput.*, vol. 76, no. 9, pp. 6860–6886, Sep. 2020, doi: 10.1007/S11227-019-03131-X/METRICS.

[2] Hasson and Timothy, "Bad bot report 2021," *Imperva*, 2021, [Online]. Available: https://www.imperva.com/blog/bad-bot-report-20 21-the-pandemic-of-the-internet/

[3] İ. N. A. Ahmet Efe, "Comparison of the Host Based Intrusion Detection Systems and Network Based Intrusion Detection Systems," *Celal Bayar Univ. J. Sci.*, vol. 18, no. 1, 2022, doi: https://doi.org/10.18466/cbayarfbe.832533.

[4] A. A. Alqarni, "Toward support-vector machine-based ant colony optimization algorithms for intrusion detection," *Soft Comput.*, vol. 27, no. 10, pp. 6297–6305, May 2023, doi: 10.1007/S00500-023-07906-6/METRICS.

[5] M. Yuyang Zhou, Guang Cheng , Shanqing Jiang, Dai, "Building an efficient intrusion detection system based on feature selection and ensemble classifier," *Comput. Networks*, vol. 174, p. 107247, 2020, doi: https://doi.org/10.1016/j.comnet.2020.107247.

[6] P. B. S. Vanitha, "Improved Ant Colony Optimization and Machine Learning Based Ensemble Intrusion Detection Model," *Intell. Autom. Soft Comput.*, vol. 36, no. 1, pp. 849–864, 2023, doi: https://doi.org/10.32604/iasc.2023.032324.

[7] Z. G. and J. W. C. Liu, "A Hybrid Intrusion Detection System Based on Scalable K-Means+ Random Forest and Deep Learning," *IEEE Access*, vol. 9, pp. 75729–75740, 2021, doi: 10.1109/ACCESS.2021.3082147.

[8] M. T. I. R. Md. Rayhan Ahmed , salekul Islam ,Swakkhar Shatabda ,A. K. M. Muzahidul Islam, "Intrusion Detection System in Software-Defined Networks Using Machine Learning and Deep Learning Techniques --A Comprehensive Survey," *Commun. Netw.*

Broadcast Technol., 2021, doi: 10.36227/techrxiv.17153213.v1.

[9] J. Snehi, A. Bhandari, V. Baggan, and M. Snehi, "Diverse Methods for Signature based Intrusion Detection Schemes Adopted," *Int. J. Recent Technol. Eng.*, vol. 9, no. 2, pp. 44–49, Jul. 2020, doi: 10.35940/IJRTE.A2791.079220.

[10] D. H. W. Smys Smys, Dr. Abul Basar, "Hybrid Intrusion Detection System for Internet of Things (IoT)," *J. ISMAC*, vol. 2, no. 4, pp. 190–199, 2020, doi: 10.36548/jismac.2020.4.002.

[11] P. A. Osanaiye Opeyemi, Ogundile Olayinka, Aina Folayo, "Feature selection for intrusion detection system in a cluster-based heterogeneous wireless sensor network," *Facta Univ. - Ser. Electron. Energ.*, vol. 32, no. 2, pp. 315–330, 2019, doi: https://doi.org/10.2298/FUEE1902315O.

[12] B. Y. Neelu Khare, Preethi Devan, Chiranji Lal Chowdhary, Sweta Bhattacharya, Geeta Singh, Saurabh Singh, "SMO-DNN: Spider Monkey Optimization and Deep Neural Network Hybrid Classifier Model for Intrusion Detection," *Electronics*, vol. 9, no. 4, p. 692, 2020, doi: https://doi.org/10.3390/electronics9040692.

[13] M. Catillo, M. Rak, and U. Villano, "2L-ZED-IDS: A Two-Level Anomaly Detector for Multiple Attack Classes," *Adv. Intell. Syst. Comput.*, vol. 1150 AISC, pp. 687–696, 2020, doi: 10.1007/978-3-030-44038-1_63.

[14] Q. R. S. Fitni and K. Ramli, "Implementation of ensemble learning and feature selection for performance improvements in anomaly-based intrusion detection systems," *Proc. - 2020 IEEE Int. Conf. Ind. 4.0, Artif. Intell. Commun. Technol. IAICT 2020*, pp. 118–124, Jul. 2020, doi: 10.1109/IAICT50021.2020.9172014.

[15] P. Vaid, S. K. Bhadu, and R. M. Vaid, "Intrusion detection system in Software defined Network using machine learning approach - Survey," *Proc. 6th Int. Conf. Commun. Electron. Syst. ICCES 2021*, pp. 803–807, Jul. 2021, doi: 10.1109/ICCES51350.2021.9489141.

[16] C. Wang, X., Qiao, Y., Xiong, J., Zhao, Z., Zhang, N., Feng, M., & Jiang, "Advanced Network Intrusion Detection with TabTransformer," *J. Theory Pract. Eng. Sci.*, vol. 4, no. 3, pp. 191–198, 2024, doi: https://doi.org/10.53469/jtpes.2024.04(03).18.

[17] L. D'hooge, T. Wauters, B. Volckaert, and F. De Turck, "Inter-dataset generalization strength of supervised machine learning methods for intrusion detection," *J. Inf. Secur. Appl.*, vol. 54, p. 102564, 2020, doi: https://doi.org/10.1016/j.jisa.2020.102564.

[18] S. K. Yakubu Imrana, Yanping Xiang , Liaqat Ali, Zaharawu Abdul-Rauf, Yu-Chen Hu, "χ 2-BidLSTM: A Feature Driven Intrusion Detection System Based on χ 2 Statistical Model and Bidirectional LSTM," *Sensors (Basel)*, vol. 22, no. 5, 2022, [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/35271164/

[19] R. Chowdhury, A. Roy, B. Saha, and S. K. Bandyopadhyay, "A Step Forward to Revolutionize Intrusion Detection System Using Deep Convolutional Neural Network," *Data-driven Ind. Comput. Springer, Singapore*, pp. 337–352, 2021, doi: 10.1007/978-981-15-9873-9_27.

[20] Muhammad Ashfaq Khan, "HCRNNIDS: Hybrid Convolutional Recurrent Neural Network-Based Network Intrusion Detection System," *Processes*, vol. 9, no. 5, p. 834, 2021, doi: https://doi.org/10.3390/pr9050834.

[21] G. Candea and B. Plattner, "Nsl-kdd data set," *Kaggle*, 2003, [Online]. Available: https://www.kaggle.com/datasets/hassan06/nslkdd

[22] L. Z. and X. W. R. Zhao, Y. Mu, "A Hybrid Intrusion Detection System Based on Feature Selection and Weighted Stacking Classifier," *IEEE Access*, vol. 10, pp. 71414–71426, 2022, doi: 10.1109/ACCESS.2022.3186975.