Metaheuristically Enhanced ANN-Based Intrusion Detection System with Explainable AI Integration

Mujeeb Ur Rehman
School of Computer Science and Informatics
De Montfort University, Leicester, UK
mujeeb.rehman@dmu.ac.uk

Muhammad Abrar
School of Computer Science and Informatics
De Montfort University, Leicester, UK
p2849363@my365.dmu.ac.uk

Sohail Khalid
Electrical and Computer Engineering Department
Riphah International University, Islamabad, Pakistan
s.khalid@riphah.edu.pk

Muhammad Kazim
School of Computer Science and Informatics
De Montfort University, Leicester, UK
muhammad.kazim@dmu.ac.uk

Vishal Krishna Singh
School of Computer Science and Electronics Engineering
University of Essex, Colchester, UK
v.k.singh@essex.ac.uk

***Abstract*—As smart devices continue to shape our lives and work, IoT networks have become an integral part of our daily lives. From smart homes to connected healthcare, these networks drive innovation but also come with a growing vulnerability, such as the risk of cyberattacks. Traditional intrusion detection systems often struggle to compensate for the complexity and sophistication of these threats, leaving critical security gaps. To address this challenge, we developed a robust ensemble approach to intrusion detection. First, we trained our dataset using a Convolutional Neural Network (CNN) and Artificial Neural Network (ANN). To further refine the performance of the ANN, we employed a metaheuristic optimization technique to ensure greater accuracy and reliability. Finally, we combine the strengths of both models using a stacking classifier to create an ensemble system capable of delivering exceptional results. The ensemble model achieved an impressive accuracy of 99.7%, outperforming the individual models, and offering a significant step forward in securing IoT networks. To make the system more transparent and trustworthy, we used Explainable AI (XAI) techniques, such as SHAP, allowing us to interpret the model's decisions clearly. By blending innovation with usability, our approach not only advances intrusion detection but also inspires confidence in its ability to protect IoT networks that we rely on every day.**

***Index Terms*—Intrusion detection, Artificial neural network, Explainable AI, Metaheuristic optimization, Ensemble learning**

I. Introduction

In today's digital world, where the Internet of Things (IoT) and Artificial Intelligence (AI) are ubiquitous, smart systems have become essential for innovation. These systems smoothly blend advanced technologies into everyday life and business activities. Moreover, the amount of data created and stored has increased dramatically. With an increasing number of technology interactions being stored and analyzed and storage devices becoming cheaper, there has been a huge increase in the identification of applicable funding agencies. If no, it was deleted from the number of databases. However, this digital shift has created a range of security problems with serious consequences [1]. An overview of the general architecture of the intrusion detection system is presented in Fig. 1 The growing number of smart devices has increased the chances of simple and complex cyber-attacks. In addition, adding smart technologies to critical systems increases the risk of cyberattacks, which can cause significant physical and digital damage. It is important to design a powerful intrusion detection system to prevent intruders and computer hackers from entering computer systems or networks. The attack and danger detection capabilities of the computing system were built into an intrusion detection system. An intrusion detection system is a tool that combines both hardware and software and is used to monitor network activities for policy violations and/or unethical behavior. In addition, it generates reports for management stations [2]. There are two main categories of network intrusion detection based on the detection techniques. One of the techniques is anomaly based, and the other is a signature-based technique, also known as the misuse

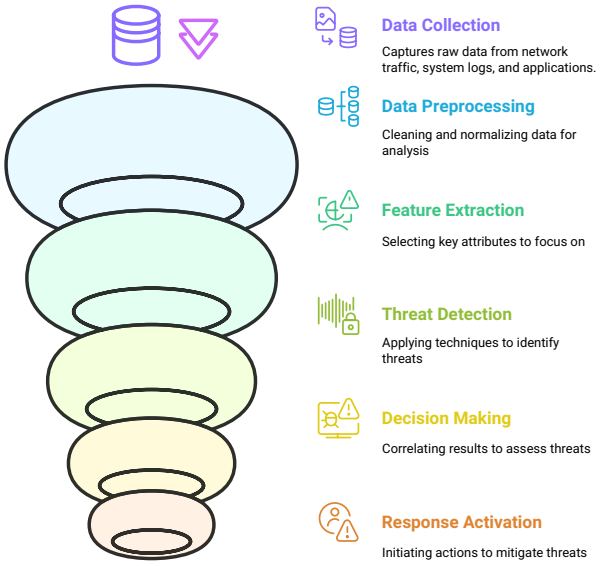

Fig. 1. General Architecture of IDS

technique. Machine learning methods, such as decision trees, random forests, and SVM, are extensively employed for treating anomaly based methods. Many studies have employed a combination of classifiers, based on the assumption that ensemble learners outperform individual classifiers. This assumption is supported by various factors including computational efficiency, statistical reliability, and representational diversity. Recently, XAI approaches have emerged for intrusion detection. This gives rise to new initiatives for cybersecurity to understand threat detection methodology [3]. This study utilized an XAI-based Intrusion Detection System alongside an ensemble ANN model. An XAI-based Intrusion Detection System was developed using the NSL-KDD dataset. Initially, two ML models, CNN and ANN, were trained on the dataset. To further improve accuracy, metaheuristic optimization techniques were applied to the ANN model. Finally, ensemble learning was used to combine the predictions of the CNN model and the metaheuristics-optimized ANN model, aiming to achieve even higher accuracy. This approach aims to balance data while minimizing information loss and avoiding an increase in data size. The remainder of this paper is organized as follows. Section 2 reviews the relevant literature. Section 3 provides a description of the NSL-KDD dataset. The proposed methodology is described in Section 4. Section 5 presents details of the experimental results and discussion. Finally, conclusions and future work are summarized in Sections 6 and 7, respectively.

II. Literature Review

The authors of [4] presented a new framework for intrusion detection in the next generation of IoT. The framework utilizes the MinMax technique to collect and process data. A marine predator algorithm was used to select features from the process data. The state-of-the-art RNN algorithm recurrent neural network trains the selected features. The contribution of each feature is determined by calculating the Shapley values; however, one of the limitations of this study is the scalability for large-scale data, as computational techniques, such as the use of recurrent neural networks and the Marine Predator algorithm, prevent scalability. The performance of the proposed framework was evaluated with 94% accuracy using the NSL-KDD dataset. The IoT world has played a very important role in significant advancements in improving the lives of people. The IoT integrates a wide range of network devices by focusing on the safety and security of the network. The following plays a very important role in providing production to IoT networks, such as prevention systems and intrusion detection systems. Currently, intrusion detection in IoT networks is challenging. The Authors of [5] proposed a framework based on the XAI approach for intrusion detection in IoT networks. The proposed framework offers transparent and reliable decision-making for the intrusion detection process. One of the limitations of this framework is that it depends on training and testing datasets. Binary classification is the main focus of this study. Key security challenges and solutions in the IoT domain have been identified through a comprehensive survey [6]. The authors covered critical areas, such as privacy issues, user authentication, access to control, and detection of intrusion. The Authors of [7] proposed a framework for smart home intrusion detection systems. The flow of traffic in a smart home environment, both normal and malicious, is detected by utilizing ML algorithms such as random forests and support vector machines.

The authors of [8] presented the evolving nature of IoT devices and machine learning techniques surveyed for intrusion detection. The study offered significant improvements in the accuracy of intrusion detection, but did not consider real-time responsiveness. In [9], a framework based on an ensemble technique for intrusions was proposed. The proposed study mainly focuses on DNS and HTTP protocols for detection. The robustness of the framework is enhanced by expanding the system to integrate a range of real-time profiling with IoT protocols. In recent years, network intrusion detection systems based on machine learning protocols have shown promising results. The growing field of AI is known as XAI, which offers different techniques for enhancing interpretability and comprehension of predictions. To enhance the interpretability of intrusion detection, [10]

proposed the XAI deep learning framework. The efficacy of a model depends primarily on its underlying accuracy. Pawlicki et al. surveyed different techniques and key research directions in the newly emerging AI field, XAI. The authors highlighted the importance of applying XAI to intrusion detection systems [11]. Szczepanski et al. proposed an explainable Hybrid Oracle framework for intrusion detection. One of the requirements of this study was the integration of additional techniques and improvements in terms of scalability for dynamic intrusion detection. However, one of the key limitations of this study was the compromise between interpretability and fidelity [12].

In [13], the authors presented a promising network security framework. The approach mainly depends on a particular dataset for evaluating performance. This may affect the application of the model to various networks. Accuracy was increased using ensemble learning. In real-time scenarios, a high computational overhead increases scalability challenges. Although XAI improves the interpretability of the model, this benefit may come at the expense of a slightly reduced detection performance compared to black-box models. Furthermore, the lack of comprehensive real-world testing limits the validation of the approach under practical large-scale network conditions. Furthermore, the lack of comprehensive real-world testing limits the validation of the approach under practical, large-scale network conditions. The study presented in [14] introduced SPAFIS, a soft-prototype-based autonomous fuzzy inference system for network intrusion detection. This system can adaptively generate a collection of human-interpretable IF-THEN fuzzy rules by analyzing network activities in real time. SPAFIS's architecture and parameters continuously evolve, allowing it to adjust to emerging data patterns. The proposed technique shows great potential for detection; however, a few areas still require improvement. The soft prototype learns in chunks and its accuracy can be improved through regular optimization. The performance depends primarily on how the detail level is set. Developing a method for SPAFIS to adjust the settings automatically would make it more flexible. The speed of the model decreases as complex data patterns are learned. The ensemble approach can improve its speed; however, this approach was not implemented. The proposed models cannot be read from unlabelled data. However, adding a supervised learning feature to learn from unlabelled data makes it more practical. In [15], the author proposed the XAI-DF framework for a memory forensic database that lacks scalability and diversity. The database primarily includes memory dumps collected from virtual machines (VMs) that typically have smaller RAM sizes and fewer variations in operating systems (OSs). Real-world environments often involve physical host machines with much larger memory capacities and a wide range of OS types such as macOS, Linux, and different versions of Windows. The models trained on this limited dataset may not perform well in real-world scenarios. The SPIP framework proposed in [16] generates both local and global explanations by combining DL and XAI methods. The proposed SPIP framework has certain limitations that could affect its effectiveness. One of the primary challenges is the need for a comprehensive training set with a wide range of valuable features from the network activities. Collecting a dataset that includes all possible normal and malicious behaviors in a heterogeneous IoT environment is virtually impossible. Additionally, many existing datasets lack crucial features. The SPIP framework identifies relevant features, but does not identify the specific security vulnerabilities targeted by an attack. Consequently, security experts must rely on their domain knowledge to interpret the results and uncover vulnerabilities based on identified attack characteristics.

III. Dataset

In this study, we used a publicly available dataset, NSL-KDD, which is an updated version of the KDD Cup 99 [17]. The dataset comprises 41 features, including 38 numerical, three nominal, and one label showing the normal/attack category. This data set has several attack types; we categorized them into five groups; DoS, Probe, R2L, U2R and normal traffic, as shown in Table. I

TABLE I
NSL-KDD DATASET ATTACKS WITH SEVERAL RECORDS

Attack category	**Attack name**	**No. of Records**
Denial of Service (DoS)	Teardrop, Apache2, Neptune, Pod, Smurf, Back, Land, Processtable, Mailbomb, UDPstorm	45,927
Probe	Satan, Saint, Ipsweep, Portsweep, Nmap, Mscan	11,656
Root to local (R2L)	Guess_Password, WarezMaster, WarezClient, Imap, Named, Ftp_Write, MultiHop, Phf, Spy, SnmpGetAttack, Sendmail, Worm, SnmpGuess, Xlock, Xsnoop	995
User to Root (U2R)	Httptuneel Rootkit, Buffer_Overflow, Load Perl, Module, Xterm, SQLattack, Ps	52
Normal traffic		67,342
Total Records		125,972

IV. Methodology

The proposed methodology of our system is based on the following phases: dataset preprocessing, training and testing of the CNN and ANN models, metaheuristic Optimization of the ANN model, ensemble learning approach, and XAI. Fig. 2 illustrates the proposed methodology in detail.

A. Preprocessing

This section consists of the following steps: handling redundant and missing data, feature encoding, feature scaling, balancing class distribution, and data splitting.

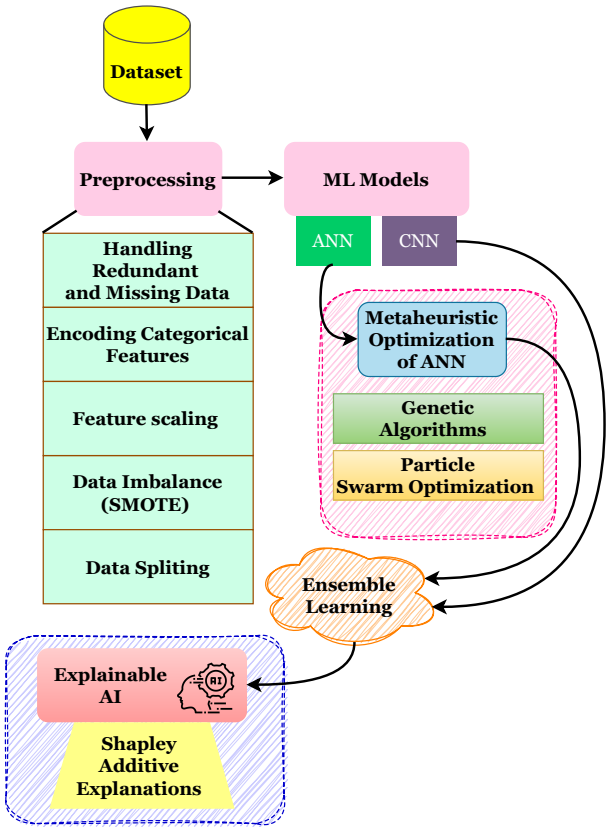

Fig. 2. Proposed Methodology

1) Handling Redundant and Missing Data: The dataset was reviewed to address missing values and to remove unnecessary or duplicated values. This step was necessary to ensure data consistency. Missing important data can reduce the features that are necessary for extracting information from the data, and redundant data does not contribute to the analysis of data.

2) Encoding Categorical Features: The next step in preprocessing is to encode the categorical features into numeric values. In the dataset, we encoded features, such as protocol type, flag, and service, into integers using one-hot encoding.

3) Feature scaling: To accelerate the training and improve the model performance, we normalized the numeric values to a range between 0 and 1 using min-max normalization.

4) Data Imbalance with SMOTE: In a dataset, imbalance between the number of attack instances and normal traffic instances is a major challenge. To address this issue, we used the Synthetic Minority Oversampling Technique (SMOTE) for minority attack classes such as R2L and U2R.

5) Data Splitting: The processed dataset is divided into three subsets: 60% for training, 15% for validation, and 25% for testing.

B. Testing and Training of CNN and ANN Model

After preprocessing the data, the next step was the development of a CNN and ANN for our system. The CNN model employs convolutional layers with a (5,5) kernel size, a Rectified Linear Unit (ReLU) as the activation function, and max pooling with a (3,3) pool size. It consists of three convolutional layers with 128, 64, and 32 filters. The final dense layer utilizes a softmax activation function for multi-class classification, generating class probabilities, and providing the final output. Our CNN model achieved an accuracy of 96.5%.

The architecture of an ANN consists of three main sections: input, hidden, and output layers. An input layer is connected to the selected features, whereas hidden layers depend on the complexity of the dataset. In this study, three hidden layers with 256, 128, and 64 neurons were used. In the hidden layers, we used the ReLU activation function, whereas in the output layer, the Softmax function was employed to classify the normal and attack categories. An Adam optimizer, with a learning rate of 0.003, was used to train the model. To prevent overfitting, we trained the model for 20, 30, and 50 epochs. The ANN model achieved an accuracy of 98.6%. The general architecture of the ANN model is shown in Fig. 3.

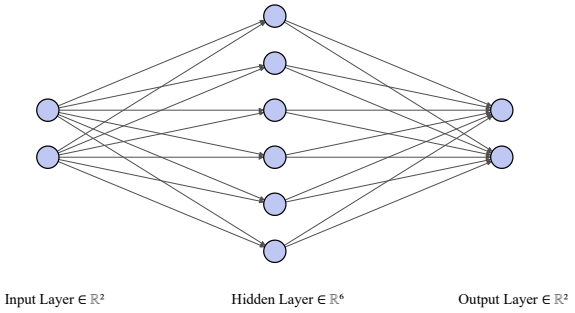

Fig. 3. General architecture of ANN model

C. Metaheuristics Optimization of ANN model

We employed metaheuristic optimization techniques to improve the performance of the ANN model. This technique improves the accuracy of the model by fine-tuning its hyperparameters. We used two optimization methods: GA and Particle Swarm Optimization (PSO).

1) Particle Swarm Optimization (PSO): The PSO can be mathematically expressed as follows: Potential hyperparameter sets are treated as particles that move through the search space. The velocity of each particle is updated as follows:

$$u_{ij}(t+1) = u_{ij}(t) + c_1 R_1 \left(l_{ij}(t) - y_{ij}(t)\right) + c_2 R_2 \left(g_j(t) - y_{ij}(t)\right)$$
$$y_{ij}(t+1) = y_{ij}(t) + u_{ij}(t+1)$$

Here, $u_{ij}(t)$ indicates the velocity of the i^{th} particle in j^{th} dimension, and c_1 and c_2 represent the acceleration elements.
$x_{ij}(t)$ represents the position, $l_{ij}(t)$ is the personal best position, and $g_j(t)$ is the global best position among all the particles. Random factors represented by R_1 and R_2 are utilized to prevent the swarm from converging too rapidly.

2) Genetic Algorithms (GA): An initial set of potential hyperparameters was used to begin the optimization process in the GA.

$$P(0) = \{h_1, h_2, \ldots, h_N\}$$

A fitness function based on the accuracy of the ANN model was utilized to assess each potential solution as follows:

$$F\left(h_i\right) = \text{Accuracy}\left(\text{ANN}\left(h_i\right)\right)$$

Based on their performance, parents were selected, and new solutions were generated by recombining their characteristics.

$$\text{Offspring } = \text{crossover}\left(h_i, h_j\right)$$

To keep the population diverse, random mutations are applied

$$h_i^{\text{mut}} = \text{mutate}\left(h_i\right)$$

Finally, the next generation is formed.

$$P(t+1) = \text{replace}(P(t),\ \text{Offspring }\ )$$

These optimization techniques enhanced the performance of the ANN model and achieved an accuracy of 99.4%. In addition to the enhancement of model accuracy, metaheuristic optimization techniques also enhance the capability of models to distinguish between normal and attack categories.

D. Ensemble Learning

To enhance the performance of the model further, we utilized an ensemble learning approach. We combined the outcomes from the CNN model with the metaheuristic-optimized ANN model to improve the overall performance of our system. We used an ensemble learning technique known as stacking ensemble, in which the outputs from different models were utilized as inputs for the meta-classifier. The final prediction was developed using a meta-classifier, which was a lightweight logistic regression model. When the ensemble model was tested on the test data, it outperformed the other individual models and achieved the highest accuracy of 99.7% compared to the other two models. By reducing false positives and handling minority attack classes, this approach renders the model more reliable for real-world applications.

E. Explainable AI (XAI)

ML and DL models are often considered black boxes owing to their complexity, which makes it difficult to evaluate how they generate predictions. To increase the transparency and understanding of our models, we used XAI. We employed (Shapley Additive Explanations (SHAP) to explain the predictions of the metaheuristic-optimized model. As depicted in Fig. 4, the features such as dst_bytes, src_bytes and $flag\_sf$ which had the most significant positive impact on the prediction are depicted in red on the plot. However, features such as $dst\_host\_serror\_rate$ and $same\_srv\_rate$ contribute negatively, as shown by the blue color in the plot. The horizontal spread of the SHAP values indicates the influence of the features on the predicted result, with dst_bytes and src_bytes emerging as the most important features.

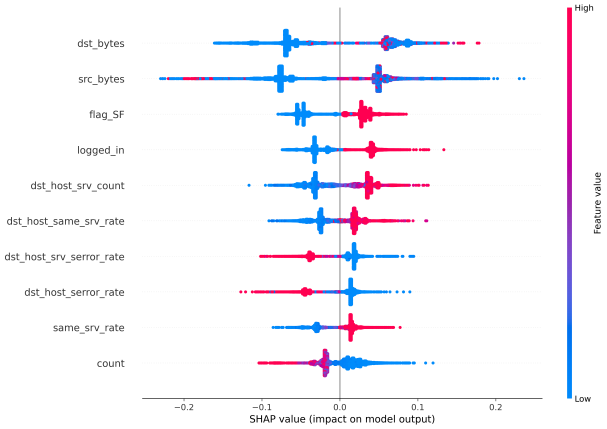

Fig. 4. Contribution of the most relevant features

V. Results and Discussion

The performance of the proposed IDS System is evaluated in this section using ANN and CNN models, a metaheuristic-optimized approach, and an ensemble learning technique.

1) Model Performance: The CNN model achieved an accuracy of 96.5%, whereas the ANN achieved a slightly higher accuracy of 98.6%, as shown in Fig. 5 ANN achieves higher accuracy owing to its deeper architecture compared to the CNN model which is very effective for capturing nonlinear relationships within a network. Table. II list the detailed analysis of the evaluation metrics including precision, recall, F1-score, and accuracies of all models.

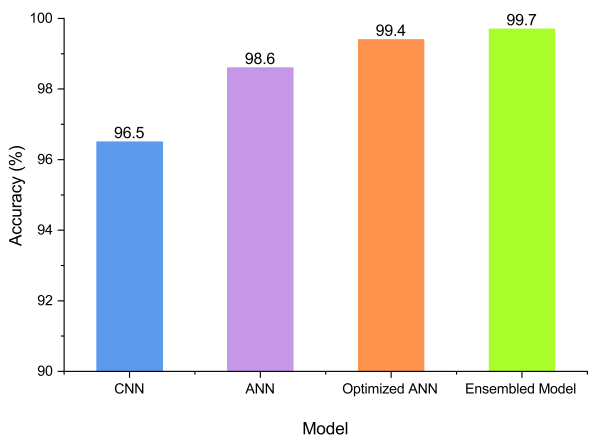

Fig. 5. Comparison of accuracies

2) Metaheuristic Optimization: The performance of the ANN model was enhanced by utilizing metaheuristic optimization techniques. We employed two optimization methods, PSO and GA, to fine-tune the hyperparameters of the model. As illustrated in Fig. 6 and 7 the training and validation curves of the optimized ANN model showed significantly improved performance, achieving an accuracy of 99.4%.

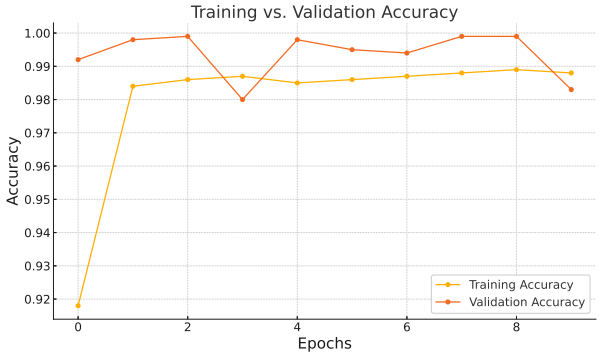

Fig. 6. Training and Validation Accuracy of Optimized ANN model

3) Ensemble learning: To further improve the accuracy of the model, we used an ensemble learning technique that combined the outcomes of the CNN model with the metaheuristic-optimized ANN model, which surpassed other models and achieved the highest accuracy of 99.7%, as depicted in Fig. 5 and Table. II respectively. The performance comparison between the proposed system and different studies for intrusion detection is listed in Table. III highlights the best performance of the proposed model over existing techniques.

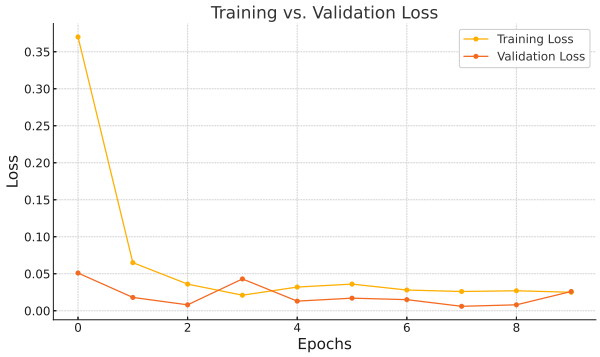

Fig. 7. Training and Validation Loss of Optimized ANN model

4) Explainable AI (XAI): To understand the behavior of the proposed system, the XAI SHAP technique was applied. As illustrated in Figure. 4 the features dst_bytes and src_bytes are the most important features in the prediction of the model and are highly effective in the decision-making process.

TABLE II
EVALUATION METRICS

Models	**Precision (%)**	**Recall (%)**	**F1-Score (%)**	**Accuracy (%)**
CNN	94.5	93.8	95.1	96.5
ANN	98.0	98.3	98.1	98.6
Meateuristic Optimized (ANN)	99.2	99.3	99.3	99.4
Ensembled Model	99.4	99.5	99.4	99.7

VI. Conclusion

This study proposes an innovative Intrusion Detection System that integrates ANN and CNN models through an ensemble learning technique to enhance the accuracy of the system. The system outperformed the other systems, achieving an accuracy of 99.7%. The standalone ANN and CNN also performed well; however, to further enhance the performance, we utilized a metaheuristic optimization approach. To enhance the interpretability of the model, the XAI SHAP technique was employed to provide valuable results regarding the contributions

TABLE III
COMPARISON OF DIFFERENT STUDIES IN IDS

Reference	Dataset	Method	Accuracy %
[18]	NSL-KDD	DL-IDS	99.02
[19]	KDD Cup' 99	DBN	97.9
[20]	NSL-KDD	XGBoost	93.28
[21]	NSL-KDD	TVCPSO-SVM	98.30
[22]	IoTID20	Hybrid (LSTM and CNN)	98.80
[23]	NSL-KDD	SVM	99.18
[24]	KDD	CANN	99.46
Proposed Model	NSL-KDD	ANN Metaheuristics Optimization with Ensemble learning	99.7

of features and ensure transparency in the decision-making process. In terms of contribution, this research addresses challenges such as data imbalance, network scalability, the importance of metaheuristic optimization, and transparent decision making for intrusion detection in cyber security systems.

VII. Future Directions

The proposed system shows promising results and highlights several areas for future research.

1) Scalability and Real-Time Application: The system requires real-time detection of intrusions across different networks and scalability in terms of handling the diversity of datasets.
2) Dataset Diversity: Future studies should explore the comprehensiveness and variety of datasets covering a wide range of attack scenarios and network activities.
3) Integration with IoT Networks: Extending the proposed system to IoT networks could increase their practical application, as IoT networks encounter distinct security issues.
4) User-Friendly Interfaces: To facilitate broader adoption by cybersecurity professionals, it is necessary to create tools to present the output of XAI in a user-friendly manner.

References

[1] A. Belenguer, J. A. Pascual, and J. Navaridas, "A review of federated learning applications in intrusion detection systems," *Computer Networks*, p. 111023, 2025.

[2] I. Makris, A. Karampasi, P. Radoglou-Grammatikis, N. Episkopos, E. Iturbe, E. Rios, N. Piperigkos, A. Lalos, C. Xenakis, T. Lagkas *et al.*, "A comprehensive survey of federated intrusion detection systems: Techniques, challenges and solutions," *Computer Science Review*, vol. 56, p. 100717, 2025.

[3] O. Arreche, T. Guntur, and M. Abdallah, "Xai-ids: Toward proposing an explainable artificial intelligence framework for enhancing network intrusion detection systems," *Applied Sciences*, vol. 14, no. 10, p. 4170, 2024.

[4] Y. Djenouri, A. Belhadi, G. Srivastava, J. C.-W. Lin, and A. Yazidi, "Interpretable intrusion detection for next generation of internet of things," *Computer Communications*, vol. 203, pp. 192–198, 2023.

[5] J. V. Rani, H. A. S. Ali, and A. Jakka, "Iot network intrusion detection: An explainable ai approach in cybersecurity," in *2023 4th International Conference on Communication, Computing and Industry 6.0 (C216)*. IEEE, 2023, pp. 1–6.

[6] M. A. Sadeeq, S. R. Zeebaree, R. Qashi, S. H. Ahmed, and K. Jacksi, "Internet of things security: a survey," in *2018 International Conference on Advanced Science and Engineering (ICOASE)*. IEEE, 2018, pp. 162–166.

[7] E. Anthi, L. Williams, M. Słowińska, G. Theodorakopoulos, and P. Burnap, "A supervised intrusion detection system for smart home iot devices," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 9042–9053, 2019.

[8] K. A. Da Costa, J. P. Papa, C. O. Lisboa, R. Munoz, and V. H. C. de Albuquerque, "Internet of things: A survey on machine learning-based intrusion detection approaches," *Computer Networks*, vol. 151, pp. 147–157, 2019.

[9] N. Moustafa, B. Turnbull, and K.-K. R. Choo, "An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of internet of things," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4815–4830, 2018.

[10] S. Mane and D. Rao, "Explaining network intrusion detection system using explainable ai framework," *arXiv preprint arXiv:2103.07110*, 2021.

[11] M. Pawlicki, A. Pawlicka, R. Kozik, and M. Choraś, "Advanced insights through systematic analysis: Mapping future research directions and opportunities for xai in deep learning and artificial intelligence used in cybersecurity," *Neurocomputing*, p. 12, 2024.

[12] M. Szczepański, M. Choraś, M. Pawlicki, and R. Kozik, "Achieving explainability of intrusion detection system by hybrid oracle-explainer approach," in *2020 International Joint Conference on neural networks (IJCNN)*. IEEE, 2020, pp. 1–8.

[13] M. K. Hooshmand, M. D. Huchaiah, A. R. Alzighaibi, H. Hashim, E.-S. Atlam, and I. Gad, "Robust network anomaly detection using ensemble learning approach and explainable artificial intelligence (xai)," *Alexandria Engineering Journal*, vol. 94, pp. 120–130, 2024.

[14] X. Gu, G. Howells, and H. Yuan, "A soft prototype-based autonomous fuzzy inference system for network intrusion detection," *Information Sciences*, p. 120964, 2024.

[15] Z. Khalid, F. Iqbal, and B. C. Fung, "Towards a unified xai-based framework for digital forensic investigations," *Forensic Science International: Digital Investigation*, vol. 50, p. 301806, 2024.

[16] M. Keshk, N. Koroniotis, N. Pham, N. Moustafa, B. Turnbull, and A. Y. Zomaya, "An explainable deep learning-enabled intrusion detection framework in iot networks," *Information Sciences*, vol. 639, p. 119000, 2023.

[17] R. Bala and R. Nagpal, "A review on kdd cup99 and nsl nsl-kdd dataset." *International Journal of Advanced Research in Computer Science*, vol. 10, no. 2, 2019.

[18] Y. Otoum, D. Liu, and A. Nayak, "Dl-ids: a deep learning–based intrusion detection framework for securing iot," *Transactions on Emerging Telecommunications Technologies*, vol. 33, no. 3, p. e3803, 2022.

[19] K. Alrawashdeh and C. Purdy, "Toward an online anomaly intrusion detection system based on deep learning," in *2016 15th IEEE international conference on machine learning and applications (ICMLA)*. IEEE, 2016, pp. 195–200.

[20] P. Barnard, N. Marchetti, and L. A. DaSilva, "Robust network intrusion detection through explainable artificial intelligence (xai)," *IEEE Networking Letters*, vol. 4, no. 3, pp. 167–171, 2022.

[21] S. M. H. 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.

[22] H. Alkahtani and T. H. Aldhyani, "Intrusion detection system to advance internet of things infrastructure-based deep learning algorithms," *Complexity*, vol. 2021, no. 1, p. 5579851, 2021.

[23] H. Wang, J. Gu, and S. Wang, "An effective intrusion detection framework based on svm with feature augmentation," *Knowledge-Based Systems*, vol. 136, pp. 130–139, 2017.
[24] 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 systems*, vol. 78, pp. 13–21, 2015.