Hybrid Multi Layer Perceptron And XGBoost Framework For Network Intrusion Detection System

1st Talha Haneef
Department of Information and Communication Engineering
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
talhahaneef4037@gmail.com

2nd Muhammad Saif Khan
Department of Information and Communication Engineering
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
muhammadsaifkhan502@gmail.com

3rd Aoun Muhammad
Department of Information and Communication Engineering
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
aoun.muhammad@iub.edu.pk

4th Sana Tariq
Department of Information and Communication Engineering
The Islamia University of Bahawalpur
Bahawalpur, Pakistan
sana.tariq@iub.edu.pk

***Abstract*—Intrusion Detection System (IDS) has become one of the keys to keeping the existing network infrastructures secure against more developed cyber threats. Nevertheless, the difficulty to properly identify different types of attacks in mass network traffic is associated with the large dimensional feature space and the large class imbalance of cybersecurity data. To address these issues, this study will propose a hybrid intrusion detection model, which combines deep feature learning and ensemble classification. To be specific, Multilayer Perceptron (MLP) which is a subdivision of TensorFlow Keras is used to automatically acquire high level representations on network traffic data by using fully connected layer with batch normalization and drop out regularization. These deep features obtained are then used with the original normalized features to compose an enriched feature set that are then used to train an Extreme Gradient Boosting (XGBoost) classifier to eventually give the classification of attacks. The proposed model is evaluated on CIC-UNSW-NB15 augmented dataset, consisting of ten classes of normal and malicious traffic. Accuracy and F1-score provided by the experimental results are 92.32% and 0.927 with weights that shows strong performance under imbalanced classes. These results suggest that a hybrid approach comprising the MLP-based deep features extraction and the XGBoost can be viewed as an effective and scalable tool to be used in the modern network intrusion detection systems.**

***Index Terms*—Intrusion Detection System (IDS), Multilayer Perceptron (MLP), Deep Feature Learning, XGBoost, Network Security, CIC-UNSW-NB15 Dataset.**

I. Introduction

Some of the roles in which computer networks are useful in the modern digitally integrated world are cloud computing, e-banking, communications infrastructures and large-scale data exchange. As the use of internet based systems continues to increase,network traffic has been greatly exposing the systems to many cyber threats. Services can be affected and sensitive data can be stolen by denial-of-service (DoS) attacks, worms, backdoors, exploits, and other advanced attacks that cost organizations large amounts of money. As a result, Intrusion Detection Systems (IDS) have come to be a key element of contemporary cybersecurity environments. [1]

This study suggests a composite intrusion detection system, which combines a Multilayer Perceptron (MLP), developed with the help of the TensorFlow Keras library and an Extreme Gradient Boosting (XGBoost) classifier [2].Conventional IDS systems are mostly signature-based, which is based on predefined attack pattern to detect malicious actions [3]. High level feature representations of network traffic data are obtained by the MLP, which is a sequence of fully connected layers, regularized with batch normalization and dropout. These learned deep features are then added to the original normalized features to create a rich feature set. The resulting feature representation is then trained with XGBoost classifier to make ultimate attack detection. [4]

The proposed model is evaluated using the CIC-UNSW-NB15 augmented dataset, an enhanced version of the UNSW-NB15 dataset that combines original network traffic features with additional flow-based features generated using CICFlowMeter. The dataset contains ten classes of normal and malicious network activities suitable for multi-class intrusion detection. [5]. The model has been proven to have an accuracy of 92.32% and a weighted F1-score of 0.927, which highlights the success of the MLP-based deep feature extraction and ensemble learning combination in network intrusion detection.

II. Literature Review

Network Intrusion Detection System (NIDS) have undergone a phenomenal revolution compared to their early versions which were rule-based with expert systems that analyzed network traffic patterns using predefined attack signatures. As computational capacities increase and larger data sets become accessible, machine learning techniques including Random Forests, Decision Trees and Support Vector Machines have been shown to effectively detect malicious behavior in network traffic by extracting complex connections between variables

DOI: https://doi.org/10.67535/tsp.000003.013
Published under the ICACNC 2026 Proceedings

within network datasets [6]. Nonetheless, these techniques are often characterized by high false positive rate and lack of extrapolation to novel patterns of attack.

In recent years, there have been some developments focusing on neural network-based methods, and specifically Multilayer Perceptrons (MLPs) which are most appropriate when dealing with structured and tabular intrusion detection systems.Unlike convolutional or recurrent neural networks, MLPs rely on fully connected layers to learn non-linear relationships between input features in structured datasets such as UNSW-NB15 and CIC- IDS-2017.While these complex architectural designs are not required for MLPs, they are still capable of achieving high level feature interactions.Research has shown that MLPs can perform competitively , especially in hybrid frameworks with other machine learning techniques optimized as an MLP model [7], [8].

The important aspect of preprocessing involves feature selection, which is both a vital step towards enhancing model efficiency and performance. Such hybrid feature selection algorithms are able to combine filter methods like Information Gain and correlation analysis, with wrapper methods like Recursive Feature Elimination, to minimize dimensionality and maintain discriminative information [9]. Empirical experiments show that feature selection can be effectively used to cut feature sets by 40-50 percent and preserve or potentially even enhance classification accuracy, which is especially advantageous with models based on MLP.

A considerable amount of research has been undertaken on hybrid and ensemble methods for improving the detection performance and generalization.Specifically, MLP can be combined with gradient boosting models like XGBoost, which have the benefits of both the MLP and XGBoost as it handles complex non-linear feature interactions, whereas XGBoost can serve as a strong decision boundary and process structured data efficiently. The hybrid architectures have shown better generalization and detection performance than one model approaches in complex and high-dimensional intrusion detection scenarios.

Class imbalance is still a significant issue in intrusion detection as the proportion of benign traffic is typically high in actual network traffic.Traditional machine learning and neural network models work best with the majority class and therefore have low detection rates of the minority attack classes. [10] Synthetic oversampling, cost-sensitive learning decomposed into class weights, and ensemble methods are techniques suggested to deal with this challenge. Also, recent studies have stressed the susceptibility of learning-based NIDS to adversarial attacks, which underlines the necessity of strong training mechanisms that can effectively defend against attacks and provide mitigation strategies when attacks occur [11] , [12].

In recent years, promising methods for the architectures of NIDS have relied on Transformer-based models which are capable of modeling long-range dependence with the use of multi-head self-attention mechanisms. The self-attention mechanisms, however, have an $\mathcal{O}(n^2)$ complexity which makes the modelling of the temporal dependency too expensive for real-time deployment.The hybrid models of Autoencoder and XGBoost are trained to learn latent representations and reconstruction based anomaly scores for detection of intrusions. However, they frequently have restrictive performance capabilities in multi-class classification problems. Likewise, CNN-LSTM architectures are effective at learning spatio-temporal patterns in packet sequences, not ideal for flow-level tabular data as they have recurrent bottlenecks. Graph Neural Networks (GNNs) are used to model the topologies of host interactions for botnet and lateral-movement detection. Computational overhead and latency are however an issue for constructing graphs online, creating real-time deployment difficult.The proposed MLP-XGBoost framework is able to achieve comparable detection accuracy with significantly lower the computational requirements than these alternative paradigms [13]. A quantitative comparison of the computational overhead is given in Table II.

III. Methodology

This study proposes a hybrid intrusion detection system that combines a Multilayer Perceptron (MLP) as a deep feature extractor with an XGBoost classifier. The proposed framework is designed to effectively address the challenges associated with high-dimensional and imbalanced network traffic data. Furthermore, deep learning and ensemble-based methods demonstrated strong performance on structured cybersecurity datasets, combining the strengths of representation learning with powerful classification [14], [15].Hybrid models have emerged as strong alternatives to conventional intrusion detection techniques because of their ability to extract deep latent features, along with their capacity to efficiently handle ensemble learning.

The proposed framework is trained and evaluated using the CIC-UNSW-NB15 augmented dataset [15].The MLP learns meaningful latent representations of network traffic from the input features [16].These learned representations are then concatenated with the original normalized features and supplied to the XGBoost classifier, which classifies the attacks.

A. Dataset Acquisition

The experiments were conducted using the CIC-UNSW-NB15 augmented dataset, an enhanced version of UNSW-NB15 that integrates original traffic features with additional flow-based features extracted using CICFlowMeter. The dataset consists of benign traffic and the different categories of attack, such as Fuzzers, DoS, Exploits, Generic, Reconnaissance, and Worms, that can benefit multi-class intrusion detection tasks.

B. Data Preprocessing

Several preprocessing steps were applied to ensure data consistency and improve model performance:

- Numerical features were selected only to fit into machine learning and deep learning models as per compatibility.

- Missing and invalid values were properly managed to prevent data from becoming corrupt.
- To enhance the convergence and training stability of the neural network, the distributions of the features were normalized using the StandardScaler algorithm.
- To maintain class distribution, stratified sampling with an 80/20 ratio was used to divide the dataset into a training and testing set.

The aim of these preprocessing steps is to ensure the input data is clean, normalized, and ready for effective model training.

C. Feature Extraction

The preprocessed network traffic data is used to learn high-level representations by using a Multi-Layer Perceptron (MLP) as a feature extractor [17]. MLPs are suitable for tabular intrusion detection data, since they can capture complex non-linear relationships among traffic attributes that are hard to model using conventional machine learning techniques [18].The MLP architecture proposed has a total of 512, 256 and 128 neurons in the fully connected dense layers, respectively. Non-linearity and capability to learn features through different layers is achieved using ReLU as the activation function for each activation hidden layer. The hidden layers are followed by a batch normalization to stabilize and accelerate the training process, and dropout is added as a regularization term to prevent overfitting and boost the generalization ability. The final hidden layer called features, produces a 128 dimensional latent feature embedding that represent abstract patterns of traffic behaviour.

After training, the extracted deep features are concatenated with the original normalized input features using horizontal feature stacking to form an enriched feature representation. As the original input has 155 features, the final feature vector is 283 dimensions (155 + 128). This representation combines both of them to use the original statistical traffic features and automatically learned deep latent features to boost the intrusion detection performance of the XGBoost classifier. The choice of embedding size, n = 128, was determined to ensure that the embedding space was adequate in size to capture meaningful features of the signal, yet not too large to cause overfitting or computational inefficiency.

D. Model Architecture

The proposed hybrid architecture is combination of MLP with deeply feature-extracting and XGBoost with ensemble-based feature-classification

- **Deep Feature Extraction using MLP:**

The MLP architecture has connected 512, 256 and 128 layer of neurons in full. ReLU activation is applied to each of the layers with the use of a dropout regularization and a batch normalization [19].The network is trained using the Adam optimizer that is a good stochastic optimization of the deep learning models that are constituted by the network under training on the network evidences. It ends with the last hidden layer to give rise to a concise feature embedding, indicative of network traffic behavior.

- **XGBoost Classification:**

The deep features obtained are concatenated with their input normalized features and supplied into XGBoost classifier [14], [16]. XGBoost is an ideal implementation of gradient boosting, which complies with the structured data and has proven to be of great value in the classification. To improve generalization, the classifier is set with a variety of estimators, regulated tree depth, learning rate and regularization parameters. Gradient boosting enables the model to continually bring down the error rate of classification and improve strength.

As a solution to class imbalance, the use of class weighting methods can systematically allocate more attention to minority ones, leading to better detection results on rare attack types. The proposed system has a hybrid intrusion detection pipeline which follows as shown in Fig. 1.

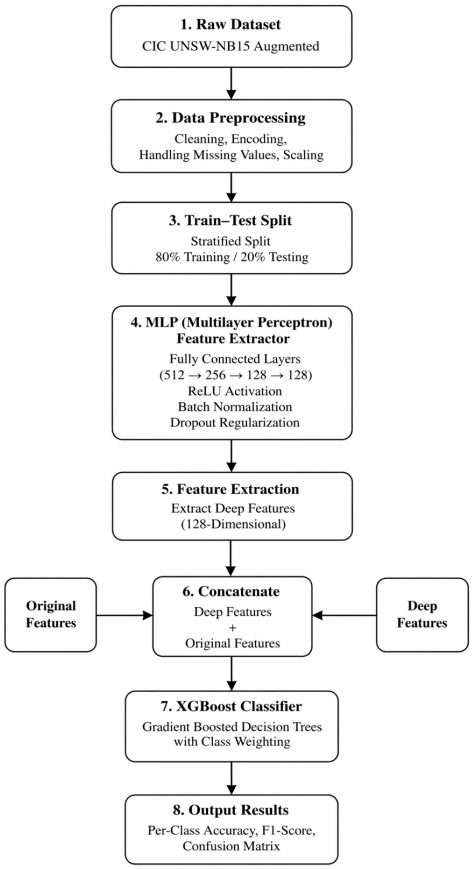

Fig. 1. Proposed Hybrid MLP-XGBoost Architecture Flowchart

E. Class Imbalance Handling

The inverse of the frequency class weight is calculated as $w_k = N/(K \cdot N_k)$, where N is the total training sample set is the set of samples and N_k is the number of samples in each class k.These weights rescale the gradient contribution of each sample during XGBoost training, equivalent to the

weighted loss: $\mathcal{L} = -\sum_i w_{y_i} \log \hat{p}(y_i \mid \mathbf{x}_i)$. SMOTE generates synthetic minority samples via linear interpolation but introduces noise in the high-dimensional sparse feature space of the NIDS data. Focal loss uses an exponential modulating factor via $(1 - \hat{p}_t)^\gamma$ but does not resolve feature-space overlap between minority attacks and benign traffic.Cost-sensitive misclassification cost matrices need to be specified by experts to allow learning, and which are hard to adjust across the 10 attack classes.The class weighting preserves the original data and the architecture is simple.

IV. Experiments and Results

This section presents a comprehensive evaluation of the proposed hybrid intrusion detection system [20], which uses a feature extractor based on a Multilayer Perceptron (MLP) and a classifier based on the XGBoost algorithm. We used the CIC-UNSW-NB15 dataset for evaluation, not only to assess the overall classification accuracy of the model but also to measure its effectiveness in detecting different attack categories and under class imbalances.

A. Experimental Setup

The CIC-UNSW-NB15 dataset is chosen as it offers a realistic representation of contemporary network traffic, with a mix of different attack categories (Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, Worms) and normal traffic.The data is preprocessed before training to ensure consistency and enhance learning. The numerical features are chosen for analysis, and missing, undefined and infinite values are set to zero [21]. The features are normalized using Z-score normalization via StandardScaler to stabilize training in the MLP feature extraction layer [22]. In order to maintain class balance during evaluation, the dataset is split into 80% training and 20% testing sets using a stratified split. Class weights are calculated to compensate for class imbalance and are applied during training to achieve better training for under-represented attack classes.

The proposed system is a hybrid approach. Normalized network traffic data are initially processed by a Multilayer Perceptron (MLP) to extract deep features. The MLP is composed of several dense layers with batch normalization and dropout regularization [23]. These deep features are then joined with the original normalized features and fed into the XGBoost classifier for detecting the attacks. The XGBoost classifier uses 150 boosting iterations with a learning rate of 0.06, the maximum tree depth of 12 and the sub sampling of 0.8 on both the data and features. These values are chosen to balance between accuracy and efficiency.

B. Overall Performance

The accuracy of the hybrid model on the test set is 92.32% with the weighted F1-score equals to 0.9270 and macro F1-score equals to 0.561.The high weight in the F1-score suggest that the overall accuracy of most of the majority classes is high compared to the overall accuracy which is much lower in the macro F1-score which is indicative of the low accuracy of the model in responding to most of the minority attacks. This difference indicates the current imbalance between the classes and suggests the difficulty to balance the performance of the detection model to all kinds of attacks.

TABLE I
Execution Summary under Experimental Setup

Metric	Value
Test Accuracy	0.9232
Weighted F1 Score	0.9270
Macro F1 Score	0.5617
MLP Training Epochs	20
XGBoost Rounds	150
Best Test Loss	0.2475
Execution Time	29.9 min

C. Per-Class Analysis

The individual class performance reveals that the model performs well for the attack classes that are highly frequent, Eg: Denial of Service and Generic attack, and it is very accurate since it has the most distinct and consistent patterns. However, average performance is observed in classes such as Fuzzers and Backdoors, whose patterns are less defined and consistent. The lowest accuracy is seen in classes like Analysis, Exploits, Reconnaissance and Shellcode. The attacks in these classes have distinct but subtle features and are more likely to be mixed with benign traffic.

There is no direct relationship between class sizes and their performance. For example, the accuracy of the Worms class is not too low despite the low number of samples, showing that the discrimination strength of the features extracted is more important than the number of samples. These findings confirm that while this model can be used to capture the main pattern of attacks, it is not strong with respect to rare and stealthy attacks.

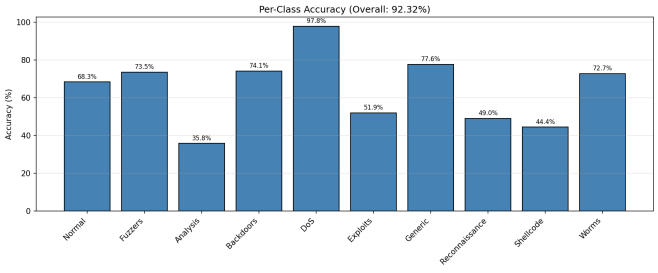

Fig. 2. Per-class accuracy of the proposed model

D. Confusion Matrix Insights

The confusion matrix in Fig. 3 illustrates that the proposed model has a good performance for majority classes (e.g., Denial of Service and Generic attacks) due to their higher representation and clearly different traffic characteristics. In contrast, there is some considerable misclassification for minority attack types such as Analysis, Reconnaissance, Shellcode, and Exploits, which are often mistaken for Normal and

Generic traffic. This means subtle attack behaviors are hard to detect in an imbalanced network traffic data.

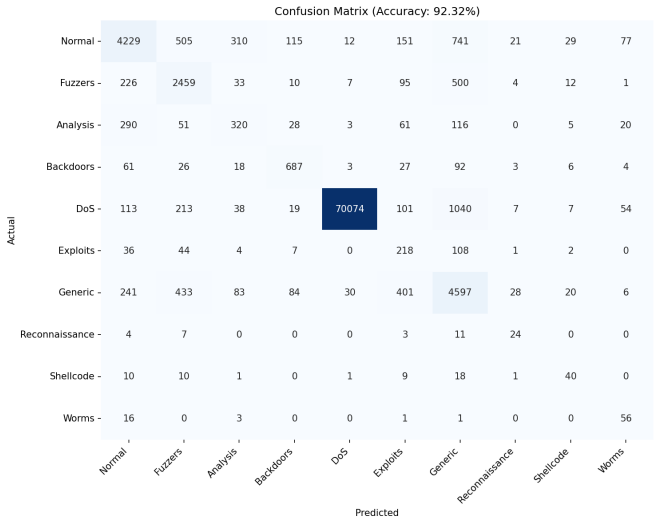

Fig. 3. Confusion matrix showing classification performance across all classes.

The confusion on the minority classes is similar to the lower score obtained by macro F1-score (0.562) than the weighted F1-score (0.927). Class weighting increases the overall classification accuracy, but the accuracy of minority classes is still relatively low. These findings suggest that additional imbalance handling strategies and more discriminative feature learning techniques may further improve the detection of rare and stealthy attacks.

E. XGBoost Training Accuracy Analysis

The training accuracy and test accuracy of the XGBoost classifier on each boosting round are plotted in Fig 4. The curves were directly extracted from the evaluation metrics measured during the training of the model, without any smoothing or post processing that can be done, and on the same pipeline used to produce the final experiments. The improvement is gradual for both curves as incrementing increases suggesting the presence of effective learning from the more informative feature representation created by the MLP feature extractor.

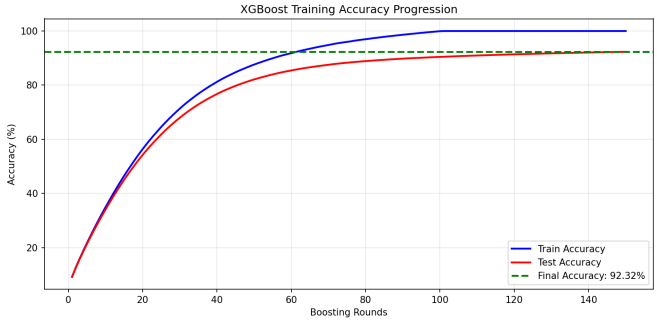

Fig. 4. XGBoost training and testing accuracy across boosting rounds

The training accuracy gradually increases and reaches saturation at more boosting rounds, and the testing accuracy nears 92.32% at higher boosting rounds. The difference between training and testing errors is relatively small, indicating that the proposed hybrid framework has a reasonable generalization with minimal overfitting under the tested hyper-parameters setting.

F. XGBoost Training and Validation Loss Analysis

The training and testing log-loss values during training of XGBoost for boosting rounds can be seen in Fig 5. The curves were directly derived through the training process via the evaluation metrics calculated in each iteration. The training loss and testing loss continue to drop over iterations as boosting progresses, and the training loss and testing loss appear stable and separation between the benign and malicious traffic classes improves with every iteration.

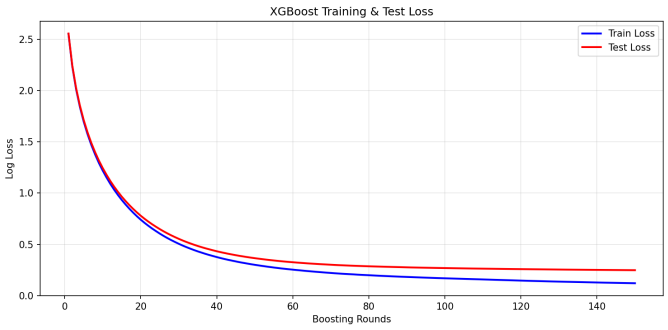

Fig. 5. XGBoost training and testing loss across boosting rounds

Although the training loss remains lower than the testing loss throughout training, the difference between the curves remains moderate, indicating controlled overfitting behavior. As the loss curves converge gradually, the proposed hybrid MLP-XGBoost framework is shown to have stable learning dynamics for the chosen learning rate and boosting configuration.

G. Comparison with Advanced Architectures

The proposed framework is compared with four advanced intrusion detection paradigms which are summarized in Table II. The computation intensity of transformer-based IDS models is as high as $\mathcal{O}(n^2)$ because of self-attention mechanisms, and it takes more than 20 ms for an inference to be performed for a single sample, which makes it difficult to use this approach in real-world applications.AE-XGBoost invokes a similar hybrid learning strategy and focuses on the reconstruction-based anomaly detection.However, it has not been studied extensively for multi-class classification, due to the lack of discriminating loss optimization. In the same way, CNN-LSTM architectures successfully model the spatio-temporal relationships among packets, but fail to appropriately handle aggregated tabular features, like those in the CIC-UNSW-NB15 dataset, at the flow-level.Graph Neural Network (GNN)-based methods can model topology level anomalies or

TABLE II
COMPARISON OF IDS ARCHITECTURES

Architecture	Acc. (%)	Complexity	Latency
Transformer-IDS	93–96	$\mathcal{O}(n^2 d)$	$>$20 ms
AE-XGBoost	89–93	$\mathcal{O}(e \cdot d^2)$	$\sim$10 ms
CNN-LSTM	91–95	$\mathcal{O}(Lkd{+}Th^2)$	$>$15 ms
GNN-IDS	92–96	$\mathcal{O}(\lvert\mathcal{E}\rvert d)$	$>$50 ms
MLP-XGBoost (Ours)	**92.32**	$\mathcal{O}(Ld^2)$	**$<$5 ms**

interactions between hosts, which some of the traditional per-flow classifiers are unable to capture. The computational cost of online graph construction is however higher (typically ¿ 50ms) and poses a constraint for real-time IDS deployment in practice.

In contrast, the proposed MLP-XGBoost framework provides a classification accuracy of 92.32% with less inference latency (under 5ms per sample). Thus, the suggested method offers a Pareto optimal trade-off between detection capability and computation efficiency in tabular multi-class intrusion detection.

V. CONCLUSION AND FUTURE WORK

In this paper, a hybrid approach for intrusion detection was proposed, which combines a Multilayer Perceptron (MLP) to learn deep features with XGBoost to classify network attacks. The use of both learned latent features and engineered features allowed the model to effectively learn complex non-linear patterns in high-dimensional network data. The results on the CIC-UNSW-NB15 dataset show that the proposed method performs well overall (accuracy of 92.32% and weighted F1-score of 0.927), which suggests the effectiveness of the method in learning the major classes of traffic and its ability to generalize.However, the macro-averaged evaluation metrics and class-based metrics show a weakness for detecting minority and stealthy attack classes, which can be explained by the imbalanced distribution of traffic classes and the overlapping feature representations. However, the hybrid framework demonstrates enhanced performance over conventional single-model approaches through the combination of deep feature learning learning and gradient boosting-based classification.

Future research will address minority class detection by exploring novel imbalance handling methods such as focal loss, data augmentation and cost-sensitive learning, and more effective feature learning frameworks to enhance detection performance in practice.

REFERENCES

[1] P. S. Lal and A. Ashok, "Deep learning-based network intrusion detection systems: A comprehensive survey of methods, evolution, and research perspectives," in *2026 7th International Conference on Mobile Computing and Sustainable Informatics (ICMCSI)*, pp. 379–385, 2026.

[2] Y. Yin, J. Jang-Jaccard, W. Xu, A. Singh, J. Zhu, F. Sabrina, and J. Kwak, "Igrf-rfe: a hybrid feature selection method for mlp-based network intrusion detection on unsw-nb15 dataset," *Journal of Big data*, vol. 10, no. 1, p. 15, 2023.

[3] M. Agoramoorthy, A. Ali, D. Sujatha, M. R. TF, and G. Ramesh, "An analysis of signature-based components in hybrid intrusion detection systems," in *2023 Intelligent Computing and Control for Engineering and Business Systems (ICCEBS)*, pp. 1–5, IEEE, 2023.

[4] K. Subramanian, F. Hajamohideen, V. Vimbi, N. Shaffi, and S. Khan Shaik, "Artificial intelligence driven analysis of intrusion detection systems using xgboost," in *Journal of Physics: Conference Series*, vol. 3191, p. 012005, IOP Publishing, 2026.

[5] A. M. Mohammed and H. K. Hoomod, "A two-stage hybrid intrusion detection framework based on hierarchical attack mapping and pruned cnn-gru models," *Engineering, Technology & Applied Science Research*, vol. 16, no. 1, pp. 32342–32347, 2026.

[6] A. Pinto, L.-C. Herrera, Y. Donoso, and J. A. Gutierrez, "Survey on intrusion detection systems based on machine learning techniques for the protection of critical infrastructure," *Sensors*, vol. 23, no. 5, p. 2415, 2023.

[7] W. Alayash, M. Rahrouh, A. A. Ibrahim, M. H. Mohamed, S. T. Ahmed, M. H. Albarri, and M. H. Ahmed, "Assessing lstm and gru for multi-dataset intrusion detection in iot environments," *Statistics, Optimization & Information Computing*, 2026.

[8] H. C. Altunay and Z. Albayrak, "A hybrid cnn+ lstm-based intrusion detection system for industrial iot networks," *Engineering Science and Technology, an International Journal*, vol. 38, p. 101322, 2023.

[9] M. Awad and S. Fraihat, "Recursive feature elimination with cross-validation with decision tree: Feature selection method for machine learning-based intrusion detection systems," *Journal of Sensor and Actuator Networks*, vol. 12, no. 5, p. 67, 2023.

[10] H. M. Dinh, W. Zong, and Y.-W. Chow, "Investigating oversampling techniques to mitigate class imbalance in network intrusion detection datasets," *Pragmatic Cybersecurity*, vol. 1, no. 1, p. 4, 2026.

[11] A. Abomakhelb, K. A. Jalil, A. G. Buja, A. Alhammadi, and A. M. Alenezi, "A comprehensive review of adversarial attacks and defense strategies in deep neural networks," *Technologies*, vol. 13, no. 5, p. 202, 2025.

[12] K. A. Hashim, Y. B. M. Yussoff, and S. B. Shahbudin, "Mitigating zero-day vulnerabilities in iiot systems: Challenges and advances in ai-powered intrusion detection systems," *Mesopotamian Journal of Cyber-Security*, vol. 5, no. 3, pp. 1184–1198, 2025.

[13] Q. Zhao, F. Wang, W. Wang, T. Zhang, H. Wu, and W. Ning, "Research on intrusion detection model based on improved mlp algorithm," *Scientific reports*, vol. 15, no. 1, p. 5159, 2025.

[14] M. Janati and F. Messaoudi, "Network behavior–driven intrusion detection: A hybrid deep learning and xgboost approach," in *AI-Driven Security for Next-Generation IoT Systems*, pp. 107–118, Springer, 2026.

[15] S. A. Hussein and S. R. Répás, "A hybrid intrusion detection framework using deep autoencoder and machine learning models," *AI*, vol. 7, no. 2, p. 39, 2026.

[16] Y. Chen, X. Zheng, and N. Wang, "Construction of vae-gru-xgboost intrusion detection model for network security," *PloS one*, vol. 20, no. 6, p. e0326205, 2025.

[17] H. Taud and J.-F. Mas, "Multilayer perceptron (mlp)," in *Geomatic approaches for modeling land change scenarios*, pp. 451–455, Springer, 2017.

[18] M. Przybyła-Kasperek and K. F. Marfo, "A multi-layer perceptron neural network for varied conditional attributes in tabular dispersed data," *PloS one*, vol. 19, no. 12, p. e0311041, 2024.

[19] L. S. Kumar, S. R. Nethi, R. Uyyala, P. Vurubindi, S. C. Narahari, A. K. Das, V. B. K, and M. J. Alenazi, "Anomaly-based intrusion detection on benchmark datasets for network security: a comprehensive evaluation," *Scientific Reports*, vol. 16, no. 1, p. 8507, 2026.

[20] A. Villafranca and M.-D. Cano, "A hybrid inference pipeline for ids: Combining dnns and xgboost through stacking for real-world intrusion detection," *Ad Hoc Networks*, p. 104227, 2026.

[21] A. Salim, O. Salim, O. M. Khudhur, S. M. Al-Barzinji, and F. M. Jasem, "Feature selection techniques in intrusion detection systems: A review," *Journal of Cybersecurity and Information Management (JCIM) Vol*, vol. 17, no. 02, pp. 97–112, 2026.

[22] P. Laxkar *et al.*, "Intelligent cyber threat categorization in cloud environments using scalable machine learning algorithms," *Journal of Artificial Intelligence in Healthcare and FinTech Systems*, vol. 1, no. 1, pp. 8–15, 2026.

[23] J. Chandroth and J. Ali, "Lightweight mlp-based feature extraction with linear classifier for intrusion detection system in internet of things," *Electronics*, vol. 15, no. 8, p. 1604, 2026.