# An Enhanced Intrusion Detection Model with FeedForward Neural Network Classifier

Asadov Amirjon*, Mrityunjoy Gain**, Keon Oh Kim*, Choong Seon Hong*

*Department of Computer Science and Engineering, **Department of Artificial Intelligence,
Kyung Hee University, Yongin, South Korea
**{amirjon7, gain, keonoh, cshong}@khu.ac.kr**

***Abstract*—— A major obstacle in the face of increasingly complex cyberattacks is network security. Proactive security measures require effective intrusion detection systems (IDS) that can precisely classify and categorize network threats. In order to improve network attack detection and classification, this paper proposes a reliable method utilizing a Feedforward Neural Network (FFNN) supplemented with Adaptive Synthetic (ADASYN) sampling. We created a model using the UNSW-NB15 dataset that efficiently handles high-dimensional datasets by preprocessing data using a combination of polynomial feature transformation and one-hot encoding. The FFNN model is optimized for binary and multi-class classification tasks. It consists of layers of dense units with dropout and batch normalization. Our method's efficacy is proven by rigorous training and validation procedures, where the model significantly increased its ability to handle class imbalances and improve classification accuracy. The synthesis of new training data by ADASYN was crucial in improving model performance, especially in underrepresented classes. Evaluation measures that highlight the potential of deep learning in network security applications are ROC-AUC scores and classification reports, which show a notable improvement in our IDS's detection capabilities. The results show that advanced machine learning techniques can be used to enhance conventional intrusion detection systems and provide a means to build stronger network security designs.**

***Index Terms*—Network Security, Feedforward Neural Network, Network Attack Categorization, Anomaly Detection in Network Security, Network Intrusion Detection System, Deep Learning in Intrusion Detection.**

## I. Introduction

Protecting network systems from cyberattacks is more important than ever in the current digital era. As the gatekeepers of network security, intrusion detection systems (IDS) are entrusted with promptly detecting and categorizing any threats in order to stop data breaches and safeguard system integrity.

But the sheer amount and variety of network traffic frequently overwhelms conventional IDS approaches, resulting in a high false alarm rate and constrained scalability. It becomes difficult as a result to stay up with more complex cyberthreats. Intrusion Detection Systems (IDS) are undergoing a revolution thanks to the advancement of machine learning and artificial intelligence. Because of their enormous improvements in IDS accuracy and processing speed, these cutting-edge solutions promise to not only keep up but also stay ahead of the cybercriminals in the arms race. We can make IDS systems that proactively anticipate and neutralize threats, rather than just reacting to them, by utilizing these cutting-edge technologies. This will make everyone's digital environment safer. The development of artificial intelligence and machine learning has brought about a revolution in intrusion detection systems, or IDS. IDS can now use anomaly detection techniques in addition to conventional rule-based approaches thanks to these cutting-edge technology. Machine learning algorithms can enhance the detection capabilities of intrusion detection systems (IDS) by identifying aberrant activity that may be indicative of possible threats by examining patterns and variations in network data.

Many studies have looked into how deep learning models might improve attack detection and classification. A noteworthy study developed a multi-modal deep transfer learning framework specifically for software-defined networks [1] to demonstrate how deep learning techniques may be applied to a range of network designs. In a similar vein, another study showed how well Graph Neural Networks detect network anomalies by using the relational data found in network traffic [2].

Moreover, there is a rising interest in creating sophisticated intrusion detection systems that reduce false positives and improve detection accuracy in dynamic network environments through the use of ensemble learning. These methods improve anomaly detection's effectiveness and dependability by combining the advantages of several algorithms [3]. Furthermore, hybrid deep learning approaches have been studied in current SDN security research, showing the benefit of combining several neural network models to effectively discriminate between legitimate and malicious network traffic [4].

Test the accuracy, efficiency, and scalability of the suggested framework against the most recent techniques. Drawing from the model and existing studies, the present study offers the following major insights:

1. Integration of ADASYN Sampling into FFNN: To improve the model's capacity to identify and categorize network attacks, the proposal suggests incorporating ADASYN sampling into a FFNN architecture.

2. Advanced Data Preprocessing Techniques: Shows how to handle high-dimensional datasets and extract complex patterns from network traffic using advanced data preprocessing

ISBN: 979-11-88428-13-7

techniques including polynomial feature transformation and one-hot encoding.

3. Handling Class Imbalance: To address class imbalance concerns and ensure a more fair representation of various attack categories in the training dataset, the ADASYN oversampling technique is employed.

4. Performance examination and Validation: Using the UNSW-NB15 dataset, a thorough examination of the suggested methodology is carried out, demonstrating notable gains in classification performance and detection accuracy over the state-of-the-art techniques.

5. Enhanced Model Performance Metrics: Offers assessment metrics including ROC-AUC scores and classification reports, giving users a thorough understanding of the model's functionality and its capacity to correctly categorize various types of network attacks.

## II. RELATED WORK

In the domain of network security, the accurate and timely detection of cyber threats is paramount. Machine learning (ML) offers a powerful suite of techniques that significantly enhance the capabilities of Intrusion Detection Systems (IDS) through both binary and multi-class classification methods. This section discusses the integration of ML techniques into network security, emphasizing their effectiveness in identifying and categorizing network attacks into distinct types, thereby facilitating more nuanced and effective security measures.

### A. *CLASSIFICATION*

**Binary Classification:** Binary classification is crucial in network security, distinguishing normal behavior from potential threats. Recent machine learning improvements enhance accuracy and efficiency.

In order to improve intrusion detection accuracy and speed, recent research has focused on hybrid models that combine machine learning with deep learning [5]. Further, strategies aimed at lowering false negatives in deep learning-based systems have shown promise in maximizing false positives and attaining high detection rates [6].

These advancements improve threat response in the face of expanding cyberthreats, which has practical implications. Continuous learning and adaptation through AI integration ensure robust defenses against evolving attacks.

**Multi-Class Classification:** Beyond binary classification, Multi-class classification in network security categorizes detected activity into distinct attack types like DoS, phishing, and malware, critical for targeted defenses and analyzing attacker behavior. Recent ML advances boost precision and efficiency.

A CNN-BiLSTM hybrid deep learning model for network intrusion detection in software-defined networking is presented in a recent paper [7]. To increase detection precision, the model makes use of a hybrid feature selection technique.

Comparing deep learning techniques for intrusion detection, a different study [8] also shows how well these algorithms can categorize different sorts of attacks. By increasing detection accuracy and swiftly adjusting to emerging threats, these techniques reinforce network security as a whole.

### B. *FEEDFORWARD NEURAL NETWORK IN NETWORK SECURITY*

Feedforward Neural Networks (FFNNs) are instrumental in enhancing the capabilities of Intrusion Detection Systems (IDS) within network security. Their one-directional flow allows for the efficient detection of abnormalities in network traffic, leveraging their pattern recognition prowess to quickly identify threats [9]. FFNNs are particularly adept at using labeled datasets during training, enabling them to accurately classify network data and swiftly detect various types of cyber assaults [10].

Moreover, the ability of FFNNs to classify detected anomalies into specific categories, such as malware infections or DoS attacks, is a significant advantage. This detailed classification capability allows security teams to tailor their response strategies more effectively, thereby improving the overall defense capabilities of network systems [11]. Modern Intrusion Detection Systems (IDS) depend on feedforward deep neural networks (FFNNs) to improve cyber threat detection and fortify network security [12].

## III. FEEDFORWARD NEURAL NETWORK

In this study using the UNSW-NB15 dataset [13], our proposed intrusion detection algorithm classifies network traffic and detects cyberattacks using a neural network. Both numerical and categorical features are present in the dataset; one-hot encoding is used for categorical data, while placeholders are used for missing values. Polynomial characteristics are added to numerical data to standardize and capture interactions. A ColumnTransformer [14] is used to merge these preprocessing stages.

The ADASYN technique oversamples underrepresented classes in the training dataset to promote model generalization and rectify class imbalances commonly found in cybersecurity data [15]. The architecture of the neural network includes dropout layers to avoid overfitting, batch normalization for stable learning, and dense layers with ReLU activation. The last layer outputs probabilities for each class using softmax activation [16].

In order to improve training efficiency, actions such as early stopping, model checkpointing, and learning rate reduction on plateau are used in conjunction with the Adam optimizer and categorical crossentropy loss to construct the model [17].

Our proposed intrusion detection system model as shown in figure 1 is designed to handle both binary and multi-class classifications.

After performing preprocessing and data balancing using ADASYN, the model classifies network traffic as either malicious or benign (binary classification) or identifies specific types of attacks (multi-class classification).

For binary classification, the output layer consists of a single neuron with a "Sigmoid" activation function that outputs a value of 0 or 1 to indicate benign or malicious traffic.

ISBN: 979-11-88428-13-7

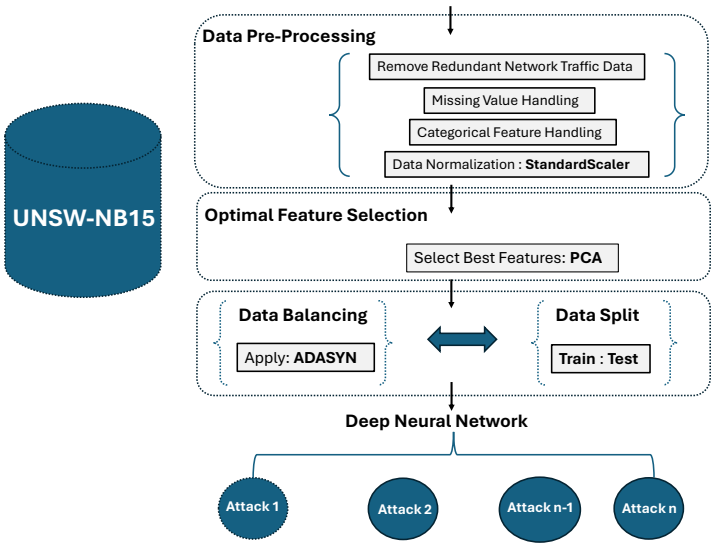


Fig. 1. Proposed Network Intrusion Detection System Model.

For multi-class classification, the output layer uses multiple neurons with a "Softmax" activation function to classify traffic into one of 10 different attack types.

In this study our proposed model FFNN was applied with 5 layers, 2 layers as an input and output layers and 3 hidden layers were applied with 128, 64, 32 neurons and as an activation function we used "Relu".

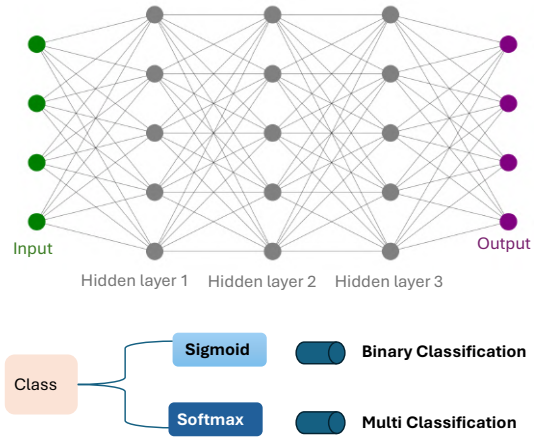


Fig. 2. Proposed Model Architecture.

We used "binary-crossentropy" loss function for binary-classification and "categorical-crossentropy" loss function for multi-class classification with epochs 100 and batch-size 32. The dataset after all pre-processing steps and handling class imbalance by applying ADASYN was split with ratio 0.2. This approach ensures accurate and comprehensive detection of network threats as shown in figure 2

## IV. ANALYSIS OF FFNN

### A. EXPERIMENTAL SETUP

**Dataset Description:** The UNSW-NB15 dataset, created by Australian Centre for Cyber Security, blends realistic activities with modern attack behaviors to evaluate intrusion detection systems. It contains 175,341 training records and 82,332 testing records, featuring 49 numerical and categorical attributes relevant to network intrusion detection.

**Experiential Setup:** All trials ran on Windows 10 with a 10th Gen Intel processor (3.70 GHz, 64 GB RAM) and an NVIDIA RTX 2080 Ti GPU. The Anaconda Python distribution was used, including libraries like Keras, TensorFlow, PyTorch, NumPy, Pandas, and Scikit-learn. Detailed specs are in Table I.

TABLE I
COMPUTATIONAL ENVIRONMENT SPECIFICATIONS

| Component | Specification |
|---|---|
| CPU | Intel64 Family 6 Model 165 Stepping 3, GenuineIntel |
| RAM | 16GB |
| Language | Python |
| Software | NumPy, TensorFlow, Scikit-Learn, Pandas, PyTorch |

### B. PRE-PROCESSING

**1. Imputation Step (Handling Missing Values: )** To handle missing values, the `SimpleImputer` replaces missing entries with a constant value ('missing'). Let $X = [x_{ij}]$ represent the dataset where $x_{ij}$ is the value of the feature $j$ for the sample $i$. If $x_{ij}$ is missing, it is replaced by:

$$x_{ij} = \text{'missing'} \quad \text{if } x_{ij} \text{ is NaN} \tag{1}$$

Equation (1) ensures that no data points are lost due to missing values.

**2. Standardization Step (Scaling Numerical Features: )** The `StandardScaler` standardizes each numerical feature $x_{ij}$ by removing the mean $\mu_j$ and scaling it to unit variance $\sigma_j$. The standardized value $\tilde{x}_{ij}$ is given by:

$$\tilde{x}_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j} \tag{2}$$

where:

- $\mu_j$ is the mean of the feature $j$,
- $\sigma_j$ is the standard deviation of the feature $j$.

Equation (2) helps ensure that each numerical feature contributes equally to the model, preventing dominance by features with larger scales.

**3. One-Hot Encoding(Encoding Categorical Features:)** The `OneHotEncoder` transforms each categorical feature into one or more binary columns. If $C_j$ represents the set of unique categories in the feature $j$, the encoded form $\tilde{x}_{ik}$ for category $c \in C_j$ is:

$$\tilde{x}_{ik} = \begin{cases} 1, & \text{if } x_{ij} = c \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

ISBN: 979-11-88428-13-7

Equation (3) ensures that categorical features are represented in a format suitable for model training.

**4. Principal Component Analysis (PCA):** PCA reduces the dimensionality of the data by projecting it onto a set of principal components that capture the most variance. The transformation can be described as:

$$Z = XW \tag{4}$$

where:

- $X$ is the standardized data matrix,
- $W$ is the matrix of principal component weights (eigenvectors),
- $Z$ is the transformed matrix with fewer dimensions (i.e., principal components).

Given the configuration $n_{\text{components}} = 0.97$, PCA selects the minimum number of components such that:

$$\sum_{k=1}^{m} \lambda_k \geq 0.97 \sum_{k=1}^{d} \lambda_k \tag{5}$$

where:

- $\lambda_k$ represents the eigenvalues (explained variance) of each principal component,
- $m$ is the number of selected principal components,
- $d$ is the total number of original features.

Equations (4) and (5) ensure that the dimensionality reduction retains significant information while discarding noise.

**Full Preprocessing Result:** The final data matrix $Z$ contains the most relevant features, reduced from 49 original features to 25 principal components while maintaining 97% of the explained variance. This step ensures that the data retains the most significant information for further analysis or model training, as shown in Figure 3.

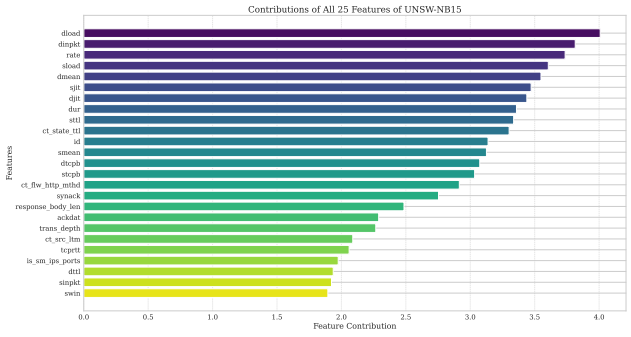


Fig. 3. Contributions of All 25 Features of UNSW-NB15. This bar plot highlights the relative contributions of each feature after preprocessing, emphasizing their importance in the analysis.

## C. DATA BALANCING

Data balancing is crucial for our intrusion detection model to prevent bias towards more common classes. In cybersecurity, where some network attacks are rare, we use the ADASYN (Adaptive Synthetic Sampling) technique to address this imbalance.

**Binary Classification:** We used an abridged method for the binary classification problem, classifying all forms of network attacks into a single 'attack' class and immediately contrasting it with 'normal' traffic, which stood for benign activity. This binary configuration improved our first filtering of network activity by helping our model distinguish between normal operations and possible security risks as shown in figure 4.

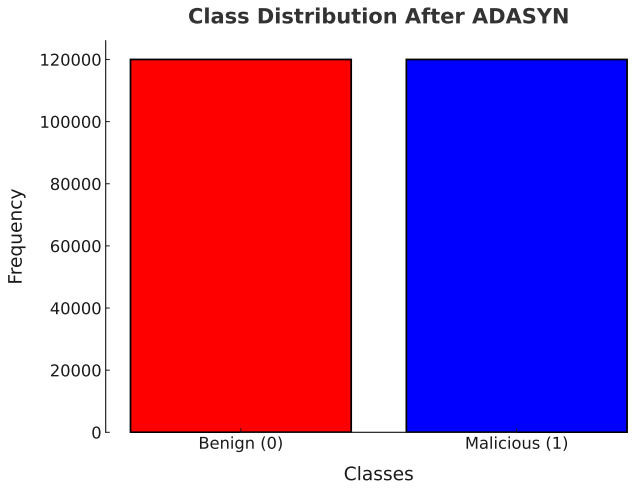


Fig. 4. Binary-Class Classification

**Multi classification:** Our approach tackled the multi-class classification problem in addition to binary classification. Here, we treated each distinct assault type as a separate class by utilizing the descriptive labels included in the UNSW-NB15 dataset. This method not only assists in detecting the existence of an attack but also classifies the particular sort of attack, which is essential for implementing suitable countermeasures as shown in figure 5.

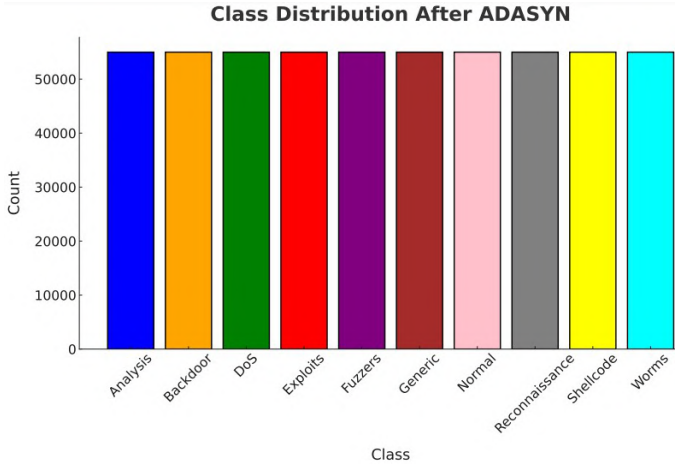


Fig. 5. MULTI-CLASS CLASSIFICATION

## D. EVALUATION METRICS

The performance of the trained FFNN model was evaluated using a comprehensive set of metrics, including accuracy, pre-

ISBN: 979-11-88428-13-7

cision, recall, F1-score and AUC-ROC values. These metrics provided a detailed understanding of the proposed model's efficiency in classifying data points and its effectiveness in distinguishing between anomalies and normal network behavior:

- **Accuracy**, the ratio of correct to total predictions, is a key metric for evaluating model performance.

$$\text{Accuracy} = \frac{TP + TN}{TP + FN + FP + TN} \tag{6}$$

- **Precision** measures the proportion of correctly predicted positive values.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{7}$$

- **Recall** quantifies the proportion of actual positive cases that the algorithm correctly identifies.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{8}$$

- The **F1-Score** represents the harmonic mean of precision and recall.

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{9}$$

## V. EXPERIMENTAL RESULTS

In this research, an improved intrusion detection system (IDS) has been proposed to provide robust security for networks against intricate cyber threats. The main model architecture is Feedforward Neural Network (FNN), powered by Adaptive Synthetic (ADASYN) sampling and highly tailored for binary, multi-class classification tasks. The model is trained using the UNSW-NB15 dataset and for computational evaluation, it improves class imbalance issues and its classification accuracy is increased to 0.999% in train set and 0.9995% in test set. Importantly, additional synthetic-maker data were generated from the application of ADASYN, which in turn proved to be beneficial for under-represented classes.

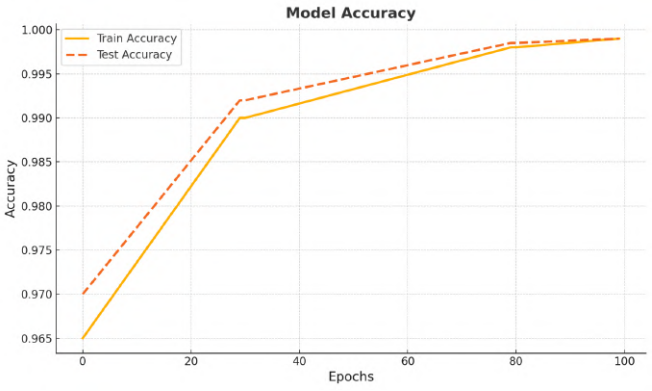


Fig. 6. Training and Testing Accuracy Over Epochs

The outcomes shown in Figure 6 support our hypothesis that, in situations where class distributions are not uniform, adaptive synthetic sampling greatly increases the efficacy of neural network models.

A model's training and testing losses over several epochs are shown in Figure 7, which indicates efficient learning and generalization with a strong initial fall in training losses and a more gradual decline in testing losses.

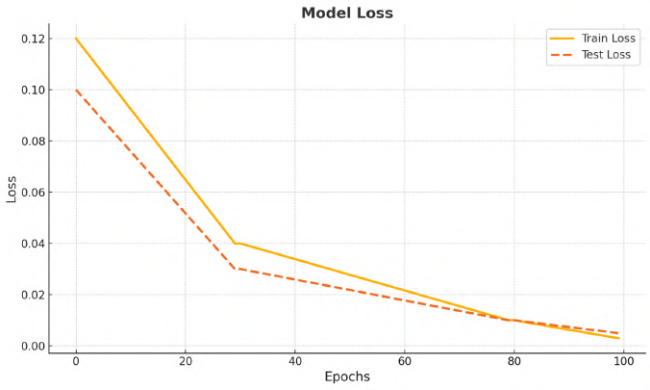


Fig. 7. Training and Testing Loss Over Epochs

After training, the model is evaluated on a test set focusing on accuracy, AUC-ROC, and a detailed classification report for each attack type. The ROC curve illustrates the sensitivity-specificity trade-off across thresholds [18].

Figure 8 shows the ROC curves for 10 attack types in the UNSW-NB15 dataset. All classes achieve an AUC of 1.00, indicating no overlap between true positive and false positive rates.

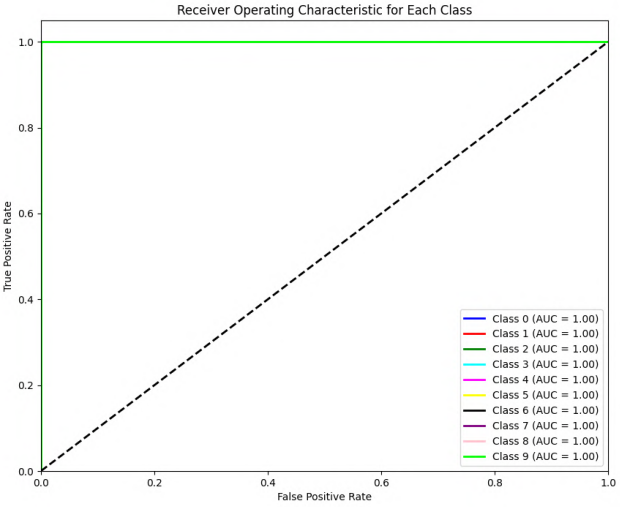


Fig. 8. ROC Curves for Different Attack Types

Performance evaluation of the model on the test set, showcasing precision, recall, F1-score, and support for each attack type. Overall metrics indicate excellent classification performance across all categories as shown in II.

ISBN: 979-11-88428-13-7

TABLE II
CLASSIFICATION REPORT ON TEST SET

| Class | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Analysis | 1.00 | 1.00 | 1.00 | 677 |
| Backdoor | 1.00 | 1.00 | 1.00 | 583 |
| DoS | 1.00 | 1.00 | 1.00 | 4089 |
| Exploits | 1.00 | 1.00 | 1.00 | 11132 |
| Fuzzers | 1.00 | 1.00 | 1.00 | 6062 |
| Generic | 1.00 | 1.00 | 1.00 | 18871 |
| Normal | 1.00 | 1.00 | 1.00 | 37080 |
| Reconnaissance | 1.00 | 1.00 | 1.00 | 3496 |
| Shellcode | **0.99** | 1.00 | **0.99** | 378 |
| Worms | 1.00 | **0.93** | **0.96** | 44 |
| **Accuracy** | – | – | **0.99** | 82332 |
| **Macro Avg** | 1.00 | **0.99** | 1.00 | 82332 |
| **Weighted Avg** | 1.00 | 1.00 | 1.00 | 82332 |

Table III presents a comparison demonstrating the superior performance of our suggested model, a Feedforward Neural Network (FFNN), over alternative techniques for binary and multi-class classification on the UNSW-NB15 dataset. While our model maintains competitive performance across various classes, the MMDTL model also achieves good accuracy on the CIC-IDS2017 dataset. In both binary and multi-class classification tests, the CNN-BLSTM and MINDFUL models exhibit reasonable performance, suggesting possible areas for development.

TABLE III
COMPARISON OF METHODS ON DIFFERENT DATASETS

| Method | Authors | Dataset | Class | Accuracy |
|---|---|---|---|---|
| MMDTL | [1] | CIC-IDS2017 | Both | 99.97% / 99.99% |
| CNN-BLSTM | [7] | UNSW-NB15 | Both | 93.51% / 84.23% |
| MINDFUL | [16] | UNSW-NB15 | Binary | 93.40% |
| **FFNN** | **Our** | UNSW-NB15 | Both | 99.9% / 99.95% |

## VI. CONCLUSION

As part of an improved intrusion detection system, our study has effectively demonstrated the effects of a Feedforward Neural Network supplemented with Adaptive Synthetic (ADASYN) sampling. We used a thorough preprocessing approach that comprised imputation for missing data, uniformity-checking numerical features, one-hot encoding for categorical variables, and Principal Component Analysis (PCA) for efficient feature selection. By ensuring that the dataset only contained the most important properties, these procedures improved the accuracy and efficiency of the model. ADASYN was used to alleviate class imbalance, which allowed the model to continue performing well against different kinds of attacks while guaranteeing balanced learning. Advanced sampling and careful preprocessing together greatly increased detection accuracy, demonstrating the importance of these strategies in creating high-performing neural network models for cybersecurity applications. To further strengthen the flexibility and resilience of network security systems, future research will concentrate on applying these strategies to bigger, more diverse datasets and testing out different data balancing strategies.

## ACKNOWLEDGMENT

This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the Convergence security core talent training business support program (IITP-2023(2023)-RS2023-00266615), (No.RS-2024-00398993) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation), (No.2019-0-01287, Evolvable Deep Learning Model Generation Platform for Edge Computing). *Dr. CS Hong is the corresponding author.

## REFERENCES

[1] "A Multi-Modal Deep Transfer Learning Framework for Attack Detection in Software-Defined Networks," IEEE Access, DOI: 10.1109/ACCESS.2023.3324878, 2023.
[2] "Application of a Dynamic Line Graph Neural Network for Intrusion Detection With Semisupervised Learning," IEEE Trans. Inf. Forensics Security, vol. 18, pp. 1556-6013, Dec. 2022.
[3] "Development of Intrusion Detection System using Residual Feedforward Neural Network Algorithm," 2021 4th Int. Seminar Res. Inf. Tech. Intell. Syst. (ISRITI), ISBN: 978-1-6654-0151-7.
[4] "A GAN-based Hybrid Deep Learning Approach for Enhancing Intrusion Detection in IoT Networks," Int. J. Adv. Comput. Sci. Appl. (IJACSA), vol. 15, no. 6, 2024.
[5] "Enhancing intrusion detection: a hybrid machine and deep learning approach," J. Cloud Comput., 13:123, 2024. DOI: 10.1186/s13677-024-00685-x.
[6] "Reducing the False Negative Rate in Deep Learning Based Network Intrusion Detection Systems," Algorithms, vol. 15, no. 8, p. 258, 2022. DOI: 10.3390/a15080258.
[7] "CNN-BiLSTM: A Hybrid Deep Learning Approach for Network Intrusion Detection in SDN With Hybrid Feature Selection," IEEE Access, DOI: 10.1109/ACCESS.2023.3340142, 2023.
[8] "A Comparative Study of Using Deep Learning Algorithms in Network Intrusion Detection," IEEE Access, DOI: 10.1109/ACCESS.2024.3389096, 2024.
[9] "Improving Intrusion Detection in IoT Networks with Feed-Forward Neural Networks Based on UNSW-NB15 Dataset," 2023 14th Int. Conf. Comput. Commun. Netw. Technol. (ICCCNT), ISBN: 2473-7674.
[10] "Deep Learning-based Intrusion Detection for IoT Networks," 2019 IEEE 24th Pacific Rim Int. Symp. Dependable Comput. (PRDC), ISBN: 2473-3105.
[11] "Improving Intrusion Detection in IoT Networks with Feed-Forward Neural Networks Based on UNSW-NB15 Dataset," IEEE Access, 14th Int. Conf. Comput. Commun. Netw. Technol. (ICCCNT), ISBN: 2473-7674, Nov. 2023.
[12] "Machine Learning and Deep Learning Approaches for CyberSecurity: A Review," IEEE Access, 2169-3536, Feb. 2022.
[13] "Available: https://research.unsw.edu.au/projects/unsw-nb15-dataset."
[14] "Design and Testing Novel One-Class Classifier Based on Polynomial Interpolation With Application to Networking Security," IEEE Access, 2169-3536, June 2022.
[15] "Addressing the Class Imbalance Problem in Network-Based Anomaly Detection," 2024 IEEE 14th Symp. Comput. Appl. Ind. Electron. (ISCAIE), 2024.
[16] "Multi-Channel Deep Feature Learning for Intrusion Detection," IEEE Access, 2169-3536, Mar. 2020.
[17] "Enhancing Intrusion Detection in IoT Environments: An Advanced Ensemble Approach Using Kolmogorov-Arnold Networks," ArXiv, 2024. DOI: 10.48550/arXiv.2408.15886.
[18] "Traffic Anomaly Detection in Wireless Sensor Networks Based on PCA and Deep CNN," IEEE Access, 2169-3536, Sept. 2022.

ISBN: 979-11-88428-13-7