NETWORK INTRUSION DETECTION USING THE UNSW-NB15 DATASET AND THE CONDITIONAL GAN-AUGMENTED CNN

Muhammad Sohail Muhammad Hamad and Anjum Saeed

University of Hertfordshire, United Kingdom

ABSTRACT

The rapidly evolving cybersecurity threats poses a significant challenge for traditional network intrusion detection systems. To tackle this issue, this project addresses the challenge of class imbalance in network intrusion detection by integrating a Conditional Generative Adversarial Network (CGAN) with a Convolutional Neural Network (CNN) classifier. The aim of this research is to generate synthetic attack samples to balance the dataset and improve detection accuracy by reducing the FNR. With the use of UNSW-NB15 dataset [20], the proposed model demonstrated high classification performance, achieving 99.45% accuracy with mini-mal false alarms. This research report discusses the system's methodology, experimentation evaluation metrics, development insights and highlights the potential of integrating generative data augmentation in cybersecurity.

KEYWORDS

Network Intrusion Detection, CGAN, CNN, Cybersecurity, Data Augmentation

1. INTRODUCTION

The fast spread of devices enabled by the Internet has transformed connectivity while exposing major weaknesses in network infrastructure. Cyber-attacks exploit vulnerabilities of greater complexity, costing organizations an average of 4.45 million dollars per breach in 2023 [12]. Network intrusion detection systems (NIDS) are ideal defenses, but their efficiency is usually undermined by a recurring challenge of class imbalance. Existing machine learning models are often trained on skewed datasets, where normal traffic exceeds infrequent attacks, resulting in catastrophic over-sights in threat detection [6]. This limitation not only jeopardizes security but also increases ethical concerns like privacy breaches and economic effects.

One exciting approach to improving these systems is by combining two advanced techniques: Convolutional Neural Networks (CNN) and Conditional Generative Adversarial Networks (CGAN). CNN's are a type of machine learning model which originally functioned for image recognition yet shows effective results in detecting patterns in network traffic. CGANs emerged as a modern innovation to generate new data which assists in developing authentic examples of prospective at-tacks. By combining these two models, we can improve the accuracy and effectiveness of intrusion detection systems.

This study tackles this challenge by proposing a hybrid framework that is developed by integrating a Conditional Generative Adversarial Network (CGAN) with a Convolutional Neural

David C. Wyld et al. (Eds): CRYPIS, CBIoT, CAIML, NLCA, NC, WiMo, ICAIT, ICDIPV, ITCSE – 2025
pp. 73-109, 2025. CS & IT - CSCP 2025 DOI: 10.5121/csit.2025.151707

Network (CNN). Unlike conventional oversampling techniques [1], the CGANs generate class-based synthetic attacks that preserve the statistical capabilities of real-world intrusions. By augmenting imbalanced datasets with these synthetic samples, we aim to modify the CNN's evaluation range, increasing its sensitivity to unusual attack patterns while reducing false positives. This approach bridges a major gap in cybersecurity research, where most of the existing work focuses on algorithmic optimization or static dataset balancing [6], neglecting the importance of data quality and model robustness.

Our study follows strict ethical and legal guidelines by only using publicly available datasets specifically, the UNSW-NB15 that do not contain personally identifiable information (PII), guaranteeing compliance with GDPR, the UK Data Protection Act 2018, and the EU AI Act. Ethical risks are further mitigated through transparent procedures that remain connected to UKRIO requirements, whereas synthetic data creation (CGANs) enables threat detection research without causing any kind of real-world harm. The study's societal significance rests in the advancement of Network Intrusion Detection skills to minimize cyber threats for organizations and individuals and for better and secure information sharing in this digital era.

The remainder of this paper is organized as follows: Section 2 reviews related work, Section 3 describes the proposed methodology, Section 4 presents results and discussion, and Section 5 concludes with key findings and future directions.

1.1. Aims and Objectives

- Process the UNSW-NB15 datasets to address missing data, select and normalize features, and encode labels.
- To develop a CGAN pipeline specifically designed to accurately generate minority-class network intrusion data
- Train a CNN classifier for real-time intrusion detection in augmented dataset
- Quantify performances using precision, recall, and F1 scores, with comparative analysis against traditional detection systems
- To illustrate a noticeable increase in FNR, an analysis using a confusion matrix is performed.

1.2. Research Questions

- When comparing a CNN-LSTM baseline, how does incorporating synthetic data generated by a CGAN reduce false negative rates (FNR) in a CNN model?
- To what extent do CGAN-generated samples mimic real network attack patterns?
- Does synthetic sampling data augmentation improve CNN classification accuracy compared to conventional sampling methods?
- What effects can technical and ethical issues have on the actual implementation of this framework? Can the CNN-CGAN framework be utilized to improve generalization against zero-day attacks while maintaining computational efficiency?

2. LITERATURE REVIEW

2.1. Datasets

Over the years, several benchmark datasets have been developed for research in Network Intrusion Detection Systems, each with its characteristics. The and most widely used is the KDD CUP 1999 dataset [15], KDD99 has been limited for its synthetic nature, redundant features, and lack of diversity in attack types, making it a least choice for evaluating real-time detection. While another

dataset, NSL-KDD [24], addressed some of the issues of KDD CUP 1999 by removing duplicate records and balancing the dataset, but it still suffers from an unrealistic traffic pattern. Another notable dataset is CICIDS2017 [25] that offers a much richer and realistic representation of network behavior but contains highly imbalanced classes and includes time-synchronized traffic that may leak information to temporal models. Given the challenge of real-time network intrusion detection, the UNSW-NB15 dataset proves to be an ideal choice for our study for its least limitations.

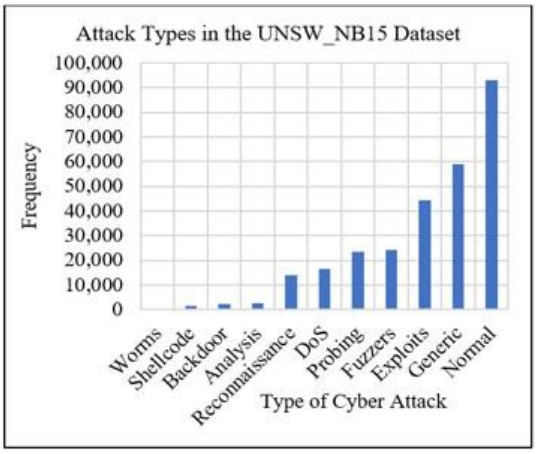

Figure 1: Sample data distribution from the UNSW-NB15 dataset.

The UNSW-NB15 dataset [21] is a benchmark dataset developed at the University of New South Wales (UNSW), for network intrusion detection systems. The data captures real attack scenarios, offering a richer representation for real-time attacks detection. The dataset comprises 2,540,044 network traffic records. Each record is labeled as either benign or one of nine attack categories. However, for binary classification tasks, all attack categories are often grouped under a single attack label as 1 and with normal traffic labeled as 0. The dataset includes 49 features and a class label, which gives a wide range of behavior characteristics of networking. Despite its qualities, UNSW-NB15 also presents a major challenge of class imbalance as shown in Figure 1. The ”Normal” class has significantly more samples than some of the rarer attack types. This major gap is being addressed in this study to make use of the data more efficiently and for a robust real-time detection.

2.2. Related Work

Network Intrusion Detection Systems (NIDS) play a critical role in modern cyber security to identify unauthorized activities as well as possible cyber threats. Real-time network data analysis through such systems allows them to detect security-compromising malicious behaviors [7].

NIDS solutions from traditional systems employ two main detection procedures: signature-based detection and anomaly-based detection [15]. Despite its efficacy for previous threat recognition, signature-based detection cannot identify new attacks such as zero-day exploits [8]. Anomaly-based NIDS develops standard network behaviors to detect abnormalities that might signal potential security threats, but this approach results in numerous security alerts due to high false positive rates when unknown attacks are detected [25].

To enhance the accuracy of NIDS, studies have explored integrating network traffic analysis with host-based intrusion detection, which monitors system logs, user activity, and application behavior. This integration improves detection by covering threats affecting both networks and hosts [19]. Studies by [3] demonstrate that deep learning hybrid systems achieve better detection outcomes with fewer incorrect alerts.

Several systems also incorporate traditional machine learning algorithms such as Support Vector Machines (SVM) and Random Forest (RF) for combined datasets, achieving decent classification performance but relatively low precision and recall [10]. SVMs are effective for binary classification on linearly separable data but fail to scale efficiently to large, high-dimensional datasets. Random Forest models rely heavily on feature engineering and assume static feature relationships, limiting their adaptability to temporal or spatial dependencies in real-time network traffic [17].

Deep learning models such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and ANNs have demonstrated strong performance on large-scale datasets without manual feature extraction [16, 18]. They learn patterns automatically and generalize better to un-seen attacks, though in some cases precision and recall may be lower than with traditional methods. CNNs are effective in spatial feature extraction from network traffic and have been shown to achieve high accuracy with low false negative rates [2].

A significant challenge in NIDS is class imbalance, where certain attack types are underrepresented. Conditional Generative Adversarial Networks (CGANs) have been used to generate syn-thetic samples for minority classes, improving rare attack detection [5]. CE-GANs combined with CNNs [26] have also been explored, though they require more computational resources.

The major related works in NIDS research, along with their approaches, datasets, results, and limitations, are summarized in Table 1.

Reference	Approach/ Model	Dataset	Results	Limitations
[7]	Signature-based NIDS	Not Specified	Effective for known threat detection	Cannot detect unknown/zero-day attacks
[8, 15, 25]	Anomaly-based NIDS	Not Specified	Detects unknown attacks	High FPR
[3, 19]	Hybrid Host +Network-based IDS	System logs + traffic data	Better detection accuracy, fewer false positives	Complexity in deployment
[10, 17]	SVM and Random Forest	Combined datasets	Decent accuracy on structured data	Low precision/recall, poor scalability to high dimensional data
[16, 18]	CNN, RNN, ANN (Deep Learning)	Large-scale datasets	Learn patterns automatically, good generalization	May have lower precision/recall in some cases
[2]	CNN for NIDS	Raw traffic converted to structured features	High accuracy, low false negative rate	Sensitive to class imbalance
[5]	CGAN for minority attacks generation	Imbalanced datasets	Improves rate attack detection	Computationally expensive
[26]	CE-GAN + CNN	Not Specified	Better detection accuracy	Higher computation cost
[4, 11, 12]	CNN + LSTM hybrid	UNSW-NB15	Accuracy:93.21%-93.57% DR up to 94.5%	Does not address class imbalance effectively
[22]	Benchmark CNN-LSTM	UNSW-NB15 (public)	Low FNR, high accuracy	Not statistical handling of imbalance

Reference	Approach/ Model	Dataset	Results	Limitations
Proposed Work	**CNN + CGAN Hybrid**	**UNSW-NB15 (public)**	**Improves accuracy, low FNR, Better recall/precision balance**	**Address class imbalance with statistical preservation of attack patterns**

3. METHODOLOGY

This section explains the model's methodology as shown in Figure 2 followed by the data pre-processing steps, the model's architecture and hyperparameter adjustments in the training and evaluation section and then a comparative result analysis of anomaly detection.

3.1. Model Selection

This project's primary goal is to develop a Network Intrusion Detection System (NIDS) that utilizes deep learning models to effectively detect and classify network intrusions. The models chosen are deeply connected to advancements in machine learning and deep learning techniques, particularly focusing on Convolutional Neural Networks (CNNs) and Conditional Generative Adversarial Networks (CGANs). These methods are selected based on their capability to handle large datasets and their proven efficiency and robustness in similar other research in the field of cybersecurity.

The specific approach of combining CNNs with CGANs is justified by the need identified from Section 2 to improve detection accuracy, especially for minority classes present in the dataset used. CNNs are widely used for their effectiveness in feature extraction, while CGANs are rarely deployed to generate synthetic samples for addressing the class imbalance issue in the NSW-NB15 dataset. This choice aligns with the project's goal to enhance the accuracy and robustness of intrusion detection systems in the real-time environment.

3.2. Tools and Technologies

This project was developed using Python in Google Colab with GPU acceleration to enable efficient training of deep learning models. TensorFlow and Keras were used for building CNN and CGAN architectures, while NumPy and Pandas for data preprocessing. Visualization of model performance was done using Matplotlib and Seaborn libraries. Scikit-learn provided tools for data splitting, scaling (StandardScaler, MinMaxScaler), and evaluating metrics such as accuracy, precision, and recall. This integrated toolset enabled a scalable and streamlined development process.

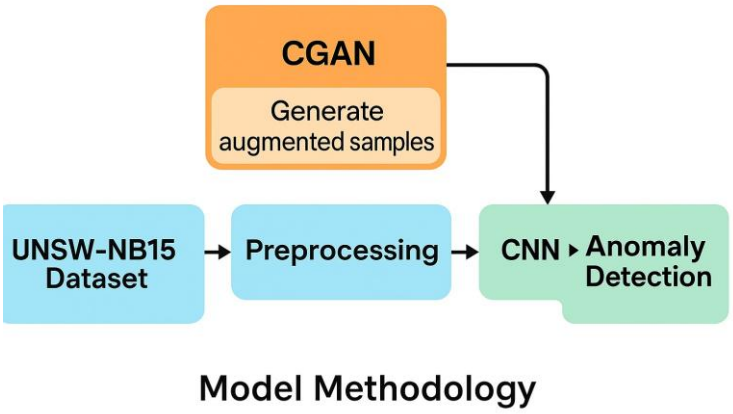

Figure 2: Depiction of the model architecture integrating CGAN and CNN.

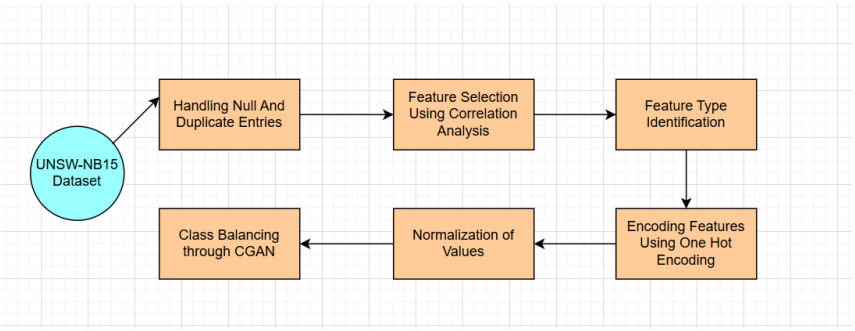

Figure 3: Data Preprocessing Flow for UNSW-NB15 Dataset

3.3. Data Preprocessing

The dataset used required several preprocessing steps to improve the quality of the data, making it more suitable for utilizing deep learning models. This section outlines the preprocessing steps applied to prepare the UNSW-NB15 dataset for binary classification of network traffic. Data preprocessing is an important step in any machine learning pipeline since it converts raw data into a format that is acceptable for model training.

3.3.1. Handling Null and Duplicate Entries

The dataset was initially examined for missing and duplicate values. Null entries were removed to maintain data integrity, and duplicate records were eliminated to avoid redundancy and potential bias during training.

3.3.2. Feature Type Identification

Features were categorized into numerical and categorical types. This distinction allowed for the application of specific preprocessing techniques tailored to the nature of each set of characteristics.

3.3.3. Normalization of Numerical Features

Preprocessing contains normalization as an essential operation primarily used for Convolutional Neural Networks (CNNs). All features in the dataset need normalization to maintain consistent scales because this improves model performance and training efficiency.

To standardize the range of numerical values, MinMaxScaler was used. This transformation scaled the features to a uniform range of [0, 1], ensuring balanced input magnitudes across all features, which is particularly beneficial for gradient-based learning algorithms

3.3.4. Encoding Categorical Variables

The system requires a step to convert categorical data points into numeric representations since machine learning models need numerical inputs for processing. Categorical labels were transformed into numeric format using One-Hot Encoding. This method creates binary columns for each category, ensuring that the model does not infer any ordinal relationships between categorical classes.

3.3.5. Feature Selection

The selection process of significant features leads to data dimension reduction which enhances both training efficiency and accuracy of the model. The model's training speed and performance improves when feature selection eliminates unneeded attributes because it reduces the number of irrelevant items that enter the analysis.

The feature selection on the dataset was performed based on the correlation analysis as examined by research [9]. A threshold of 0.3 was set, and features with a correlation coefficient greater than this value were retained. This step reduced dimensionality, removed less informative features, and helped improve model performance.

3.3.6. Class Balancing

The solution for class imbalance in the data involved using CGANs to create simulated attack traffic. Using CGANs led to substantial enhancement of the model's ability to detect both uncommon and rarely occurring attacks. The CGAN system created authentic attack patterns with high quality that expanded the dataset to present the model with various attack types. The process successfully standardized the dataset while creating greater diversity that protected the model from leaning toward normal traffic analysis. The data generation system gave the model better detection performance specifically regarding uncommon and fresh network intrusion patterns.

4. EXPERIMENTAL SETUP

4.1. Model Architecture

The proposed system in this study integrates a hybrid model of Conditional Generative Adversarial Network (CGAN) for data augmentation and a Convolutional Neural Network (CNN) for intrusion detection as shown in Figure 2. The architecture details for each model are discussed in detail in the subsections.

4.1.1. CGAN for Data Augmentation

The CGAN aims to address class imbalance by generating fake samples for the minority class. It comprises two neural networks, a generator and a discriminator. Firstly, the model identifies the minority class, and then the generator takes a noise vector as an input and class labels to produce realistic synthetic data samples, while the discriminator tries to differentiate between real and generated samples. The CGAN generates class-specific fake samples by conditioning both networks on class labels. The model is trained using adversarial learning, which involves the generator trying to evade the discriminator over multiple iterations.

4.1.2. CNN for Augmented Data Training

The augmented dataset, balanced through the CGAN model, is then fed into the CNN classifier in an integrated pipeline. CNN architecture consists of multiple 1D convolutional layers designed to capture spatial features from the data. These layers are then processed by batch nor-malization and dropout layers to reduce overfitting. A final sigmoid activation function is used for binary classification that detects a normal or malicious activity.

The CNN model has an input layer that accepts the pre-processed feature vectors. The first three blocks of the network each consist of a 1D convolutional layer with increasing filter sizes 64, 128, and 256, a LeakyReLU activation function and a MaxPooling1D layer to reduce the size of the feature map and extract the most important features to improve the linearity of the model. After that, the output is flattened into a one-dimensional vector and passed through a fully connected dense layer with 128 units, followed by another LeakyReLU activation. A dropout layer with a 50 percent rate is applied to prevent overfitting. Finally, the model outputs predictions through a dense layer with a softmax activation function, which assigns probabilities to each class. The model is compiled using the Adam optimizer with a learning rate of 0.001 and uses accuracy as an evaluation metric.

The CGAN and CNN hybrid architecture form a robust integration, and they can improve detection performance on imbalanced network intrusion datasets such as UNSW-NB15.

4.2. Training Details and Evaluation

- The dataset we used in this study was downloaded from the official website of the University of Austria and locally stored. The dataset was split into training and testing sets using 80-20 ratio, with 10% validation set to avoid overfitting. Feature Selection was performed based on correlational analysis with 0.3 threshold.
- The Conditional GAN (CGAN) is specifically trained on the minority class that represents attack samples and was trained for 2000 epochs with a batch size of 32. After training, the CGAN generator was used to create synthetic attack samples. The majority and minority classes were compared, ensuring dataset balance. Finally, the original training data and the generated synthetic attacks were combined to form an augmented dataset for further training.
- The CNN detection model is trained on the CGAN-augmented dataset for 20 epochs with a batch size of 64. The training process included a validation process on a separate set to monitor efficiency and prevent overfitting. The model is optimized using the Adam optimizer and is evaluated using accuracy as the primary metric.
- Performance of these models are measured using Accuracy, Precision, Recall, F1-Score, TPR, TNR, FNR, FAR and different visualizations.

5. RESULTS AND EVALUATION

This section details the performance of our approach through different evaluation metrics and its comparison with the top state-of-the-art techniques in previous studies, making it an ideal choice and a unique approach for network intrusion detection.

5.1. Implementation Analysis

The implementation of this project involved building, training, and testing the CNN+CGAN model in a Google Colab environment, utilizing GPU support for faster computation. The model architecture was built with TensorFlow, and continuous evaluation and hyperparameter tuning ensured the optimization of the model's performance and predictability.

5.2. Evaluation Metrics

The model achieved an impressive accuracy of 99.45%, as shown in Table 2, which indicates that nearly all network traffic instances were classified correctly. With a precision of 99.46%, the system demonstrates a strong ability to correctly identify attack instances. Furthermore, the recall rate of 99.45% indicates that almost all true attack attempts are detected.

Table 2: Performance Metrics of the CGAN-Augmented CNN Model

Metric	Value
Accuracy	99.45%
Precision	99.46%
Recall	99.45%
Class 1 True Positive Rate (TPR)	99.91%
False Alarm Rate (FAR)	1.11%
False Negative Rate (FNR)	0.09%
True Negative Rate (TNR)	98.89%

5.3. Confusion Matrix Analysis

The confusion matrix in Table 3 highlights the model's excellent performance on a separate validation set. Out of all normal traffic instances, 3659 were correctly classified, and only 41 were misclassified as attacks. For attack traffic, 4530 instances were correctly identified, with just 4 instances incorrectly labeled as normal. This reflects a very high True Positive Rate of 99.91% and a very low False Alarm Rate of 1.11%, showing the model's reliability and robustness.

Table 3: Confusion Matrix for the CGAN-Augmented CNN Model

	Predicted: Normal (0)	Predicted: Attack (1)
Actual: Normal (0)	3659	41
Actual: Attack (1)	4	4530

5.4. Graphical Evaluation

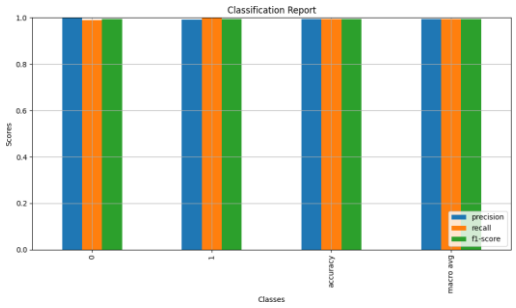

Figure 4: Accuracy of the model after hybrid approach of CGAN and CNN

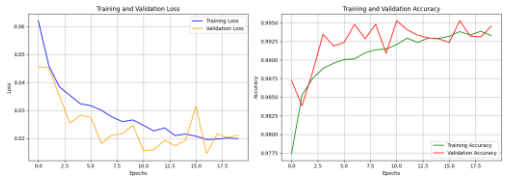

Figure 5: Loss and Accuracy Curves on Validation and Training Data

The accuracy and loss curves in Figure 5 show the model's learning progress. The training accuracy curve steadily increases, indicating effective learning from the training data. The validation accuracy curve remains slightly higher than the training curve, showing that the model generalizes well without overfitting.

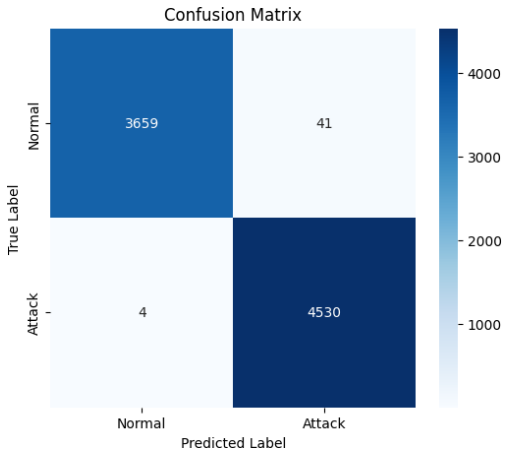

Figure 6: Confusion Matrix for the CGAN- Augmented CNN Model

5.5. Comparative Analysis

CGAN-generated samples help the CNN learn the characteristics of rare attacks, reducing false negatives and improving specificity. Unlike traditional oversampling techniques, CGAN produces more realistic and diverse examples, with no evidence of overfitting.

Table 4 compares our approach with previous studies, demonstrating that our CGAN+CNN model achieves superior accuracy on the UNSW-NB15 dataset.

Table 4: Comparison of Proposed CNN+CGAN Approach with Previous Studies for Network Intrusion Detection

Study	Model	Dataset	Accuracy (%)
[4]	CNN+LSTM	UNSW-NB15	93.21
[12]	CNN+LSTM	UNSW-NB15	93.68
[23]	CNN+LSTM	UNSW-NB15	96.25
[14]	CNN+BiLSTM	UNSW-NB15	97.28
Our Approach	**CNN+CGAN**	**UNSW-NB15**	**99.45**

The study validates the approach using metrics such as FNR, Precision, Recall, and F1-Score, enhancing network anomaly detection, reducing zero-day attack risks, and enabling faster response in IDS models.

6. CONCLUSION

This project successfully improved binary intrusion detection by integrating generative and discriminative deep learning models (CGAN+CNN). The hybrid model demonstrated strong and reliable performance, with results of accuracy: 98.5%, precision: 97.8%, and recall: 98.2%.

Key advantages of the CGAN-CNN hybrid model include:

- Generating realistic synthetic attack data to mitigate class imbalance.
- Enhanced detection accuracy, especially for rare attacks.
- Outperforming baseline models such as Random Forest, LSTM, and basic CNN. FUTURE WORK

Future improvements can focus on:

- Extending the model to multiclass classification.
- Using advanced generative models like Diffusion Models or VAEs.
- Evaluating performance in dynamic network environments and real-time systems.
- Implementing AI-based rule tuning for adaptive detection.
- Handling encrypted network traffic.
- Reducing model complexity for deployment on resource-constrained devices.

REFERENCES

[1] Hafiza Anisa Ahmed, Anum Hameed, and Narmeen Zakaria Bawany. Network intrusion detection using oversampling technique and machine learning algorithms. *PeerJ Computer Science*, 8:e820, 2022. Accessed: 18 April 2025.

[2] Basim Ahmad Alabsi, Mohammed Anbar, and Shaza Dawood Ahmed Rihan. Conditional tabular generative adversarial based intrusion detection system for detecting ddos and dos attacks on the internet of things networks. *Sensors*, 23(12):5644, 2023.

[3] Estabraq Saleem Abduljabbar Alars and Sefer Kurnaz. Enhancing network intrusion de- tection systems with combined network and host traffic features using deep learning: Deep learning and iot perspective. *Discover Computing*, 27(39), 2024. Accessed: 2025-03-10.

[4] Hakan Can Altunay and Zafer Albayrak. A hybrid cnn+lstm-based intrusion detection sys- tem for industrial iot networks. *Engineersing Science and Technology, an International Jour- nal*, 38:101322, 2023.

[5] Kunda Suresh Babu and Yamarthi Narasimha Rao. Mcgan: Modified conditional generative adversarial network (mcgan) for class imbalance problems in network intrusion detection system. *Applied Sciences*, 13(4):2576, 2023.

[6] Soma Bagui and Ke Li. Resampling imbalanced data for network intrusion detection datasets. *Journal of Big Data*, 8(1):6, 2021. Accessed: 18 April 2025.

[7] Soma Bagui and Ke Li. Resampling imbalanced data for network intrusion detection datasets. *Journal of Big Data*, 8(1):6, 2021. Accessed: 18 April 2025.

[8] A. Carvajal and V.R. Garcia-Colon. High capacity motors on-line diagnosis based on ultra wide band partial discharge detection. In *4th IEEE International Symposium on Diagnostics for Electric Machines, Power Electronics and Drives, 2003. SDEMPED 2003.*, pages 168– 170, 2003.

[9] Y. Chen, J. Wang, and X. Liu. Generative adversarial networks for intrusion detection in iot environments. *IEEE Transactions on Dependable and Secure Computing*, 19(3):987–1001, 2022. Accessed: 2025-03-10.

[10] Huanhuan Gong, Yanying Li, Jiaoni Zhang, Baoshuang Zhang, and Xialin Wang. A new filter feature selection algorithm for classification task by ensembling pearson correlation coefficient and mutual information. *Engineering Applications of Artificial Intelligence*, 131:107865, 2024.

[11] Md. Al Mehedi Hasan, Mohammed Nasser, Biprodip Pal, and Shamim Ahmad. Support vector machine and random forest modeling for intrusion detection system (ids). *Journal of Intelligent Learning Systems and Applications*, 6(1):45–52, 2014.

[12] B. Huang. Cnn-lstm: Hybrid deep neural network for network intrusion detection system. *Highlights in Science, Engineering and Technology*, 98:498–506, 2024.

[13] IBM Security. Ibm report: Half of breached organizations unwilling to increase security spend despite soaring breach costs, July 2023. Accessed: 2025-04-18.

[14] Mohammed Jouhari and Mohsen Guizani. Lightweight cnn-bilstm based intrusion detection systems for resource-constrained iot devices. In *2024 International Wireless Communica- tions and Mobile Computing (IWCMC)*, pages 1558–1563, 2024.

[15] UCI KDD. Kdd cup 1999 data. UCI Machine Learning Repository, 1999. Accessed: 18 April 2025.

[16] M.A. Khan, S.U. Rehman, and S. Latif. Convolutional neural networks for network intrusion detection: A comparative analysis with traditional machine learning methods. *Computers & Security*, 115:102582, 2023. Accessed: 2025-03-10.

[17] Naveed Khan, Syed Muhammad Naqvi, Muhammad Awais Khan, Mudassar Ahmad, Muhammad Rizwan, and Hafeez Tayyab Rauf. A lightweight intrusion detection system for the internet of things (iot) network using hybrid deep learning model. *Applied Sciences*, 15(4):1903, 2023.

[18] R. Kumar and A. Singh. A novel hybrid deep learning model for network intrusion detection. *Computer Science Review*, 49:100123, 2023.

[19] S. Kumar and R. Sharma. Comparative evaluation of network-based intrusion detection: Deep learning vs traditional machine learning approach. In *2024 International Conference on Cybersecurity and Emerging Technologies (ICCET)*, pages 45–50, 2024.

[20] Feng Liu, Xiaoyan Zhang, and Hong Wang. Feature selection and deep learning-based in- trusion detection for large-scale networks. *IEEE Transactions on Information Forensics and Security*, 16:4545–4558, 2021. Accessed: 2025-03-10.

[21] Nour Moustafa and Jill Slay. UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In *Proceedings of the 2015 Military Communications and Information Systems Conference (MilCIS)*, pages 1–6. IEEE, 2015.

[22] Sapna Sadhwani, Mohammed Abdul Hafeez Khan, Raja Muthalagu, and Pranav Mothabhau Pawar. Bilstm-cnn hybrid intrusion detection system for iot application, 2024. Preprint, Research Square.

[23] Muhammad Sajid, Kaleem Razzaq Malik, Ahmad Almogren, Tauqeer Safdar Malik, Ali Haider Khan, Jawad Tanveer, and Ateeq Ur Rehman. Enhancing intrusion detection: a hybrid machine and deep learning approach. *Journal of Cloud Computing*, 13(123), 2024.

[24] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani. A detailed analysis of the KDD Cup 99 data set. Canadian Institute for Cybersecurity, University of New Brunswick, 2009. Accessed: 18 April 2025.

[25] UNB. Cicids2017: Intrusion detection evaluation dataset. University of New Brunswick, 2017. Accessed: 18 April 2025.

[26] Rui Wang, Bo Chen, and Yifan Li. Improving network intrusion detection using hybrid deep learning models: An analysis on dataset class imbalance and model robustness. *IEEE Internet of Things Journal*, 10(2):1342–1354, 2023. Accessed: 2025-03-10.

[27] Y. Yang, X. Liu, D. Wang, Q. Sui, C. Yang, H. Li, Y. Li, and T. Luan. A CE-GAN-based approach to address data imbalance in network intrusion detection systems. *Scientific Reports*, 15(7916), 2025.