Article

Not peer-reviewed version

Enhancing IoT Security Through a Hybrid Deep Learning Model: CNN Meets Transformer for Robust Intrusion Detection

Vibhor Puri , Prathamesh Chandekar * , Swet Chandan

Posted Date: 17 June 2025

doi: 10.20944/preprints202506.1216.v1

Keywords: IoT; cyber security; CNN; hybrid deep learning

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 June 2025 doi:10.20944/preprints202506.1216.v1

Article

Enhancing IoT Security Through a Hybrid Deep Learning Model: CNN Meets Transformer for Robust Intrusion Detection

Vibhor Puri [1], **Prathamesh Chandekar** [2,*] **and Swet Chandan** [2]

[1] Vellore Institute Of Technology, TN, India
[2] D Y Patil International University, MH,IN
* Correspondence: work.prathameshc@gmail.com

Abstract: The impact of the Internet of Things (IoT) on diverse fields like healthcare and industrial automation is unprecedented. Despite the fact that these devices are resource-constrained and distributed widely, they are easily targetable by cyber attackers. Adaptability and flexibility in computational attack strategies result in complex, and ever-evolving, attacks being undetected by conventional Intrusion Detection Systems (IDS). In this paper, we propose a new hybrid architecture for integrating 1D Convolutional Neural Networks and transformer-based models that captures both spatial and temporal patterns effectively. We utilize the CIC-IDS2017 dataset and propose a strong preprocessing pipeline containing feature normalization, class rebalancing with SMOTE, and data cleaning to address inconsistencies and imbalances within the dataset. Relative to the hybrid model's baseline counterparts of CNN-only, transformer-only, and LSTM, the proposed model surpassed detection accuracy across varying types of attacks. Real-time IoT environments stand to benefit from hybrid deep learning models as demonstrated with our findings of 92.5% accuracy and higher recall for attacks on minority classes.

Keywords: IoT; cyber security; CNN; hybrid deep learning

1. Introduction

The Internet of Things (IoT) has sharply advanced sophistication of life in daily routines. This degree of IoT integration is evident in the modern smart homes, cities, and even the automation of industries. The degree of convenience offered enables better resource allocation as well as data based decisions. Despite the benefits of connectivity expansion, there are several peripheral IoT malicious attempts that specifically aim to weaken network security.

1.1. IoT Security Challenges

Vulnerability in IoT systems is primarily accentuated by their dependency on low powered heterogeneous devices, distributed resources, and the systems distinctive architecture. Making use of conventional security systems will be highly ineffective. The most prominent threats are data fetching through botnet-based distributed denial of service spoofing protocols and zero-day exploits. Additionally, monitoring extremely diverse and changeable IoT traffic with a pre-existing static structure does not guarantee anomaly detection. The absence of standardized communication protocols does make the analysis easier, however, this non-uniformity only strengthens the argument to why vital infrastructure is left unprotected.

1.2. Machine Learning in Intrusion Detection

The development of technology, specifically IoT networks, has bloomed over the years, bringing forth new challenges such as the need for better intrusion detection systems (IDS). Machine learning (ML) provides opportunities such as automating the process of traffic pattern recognition. It can heuristically analyze traffic datasets to discover intricate attack patterns and adapt to changes. There have been attempts to network traffic classification, with the use of supervised learning and decision trees, support vector machines (SVMs), and deep neural networks (DNNs). In more recent times, deep learning techniques like convolutional neural networks (CNNs), recurrent neural networks (RNNs), and transformers have provided great aid to IDS because of their spatial-temporal pattern recognition capabilities.

1.3. Complete IoT Datasets Requirement

Despite the many strides in making ML models, an obstacle still exists in the form of getting high quality complete datasets, which hinders the efficiency of an IDS. The majority of available datasets are either unvaried, contain duplicates, or have not been updated in some time. With the inclusion of newer attacks like infiltration, XSS, and botnet attacks, CIC-IDS2017 has made progress in providing a more appropriate assortment of traffic samples consisting of both benign and malign samples. Nonetheless, other issues like data imbalance amongst classes, absent values, or overly large queues still exist, requiring remedial work through advanced preprocessing or data engineering.

1.4. Objective and Contributions

The main objective of this research is to develop and evaluate a hybrid CNN-Transformer model for intrusion detection in IoT networks using the CIC-IDS2017 dataset. By combining CNN's feature extraction power with the transformer's ability to capture global dependencies, the proposed model aims to achieve robust multi-class classification across various attack types.

The key contributions of this work include:

- A detailed preprocessing pipeline including missing value imputation, feature scaling, and class balancing using SMOTE.
- Design of a hybrid deep learning model integrating CNN and Transformer components.
- Comprehensive evaluation across multiple performance metrics (accuracy, precision, recall, F1-score).
- Comparative analysis with baseline models including standalone CNN, LSTM, and Transformer architectures.
- Practical discussion on deployment feasibility and future improvements in edge-compatible IDS for IoT.

2. Literature Review

2.1. Machine Learning Models for Intrusion Detection in IoT

The past years have witnessed a tremendous growth in the use of machine learning for intrusion detection systems, especially in IoT environments where the standard protective measures are insufficient. Anomaly detection in network traffic has been tackled using supervised learning techniques like KNN, decision trees, SVMs, and random forests. The main shortcoming of these approaches is their failure to grasp the intricate and multi-dimensional patterns associated with today's attacks.

The usage of deep learning models has been particularly beneficial for intrusion detection owing to their remarkable capability in learning from raw data. Convolutional neural networks (CNNs) have been widely employed as a means of recognizing spatial patterns and they perform exceptionally well in feature extraction from network traffic matrices. For modeling temporal dependencies, Recurrent Neural Networks (RNNs) and their more sophisticated forms, Long Short-Term Term Memory (LSTM) networks are well suited due to their capacity to track dynamic behaviors in traffic flows over time.

More recently, the attention mechanism and Transformer architecture, originally developed for natural language processing, have been adopted for intrusion detection. Transformers excel in modeling long-range dependencies and offer parallelization benefits during training. While individual models like CNNs or Transformers perform well on specific aspects of the data, their combination has shown promise in leveraging the strengths of both - local feature extraction and global contextual awareness.

2.2. Issues of Current Datasets

Although numerous datasets have been proposed for evaluating IDS performance, many suffer from limitations that hinder model generalizability and real-world applicability. Common issues include:

- **Outdated attack patterns:** Older datasets such as KDD99 or NSL-KDD do not reflect current threat landscapes.
- **Class imbalance:** Real-world attack distributions are often skewed, with benign traffic vastly outnumbering malicious instances.
- **Lack of heterogeneity:** Many datasets are captured in controlled environments and fail to represent the diverse and noisy nature of real IoT networks.
- **Missing data and noise:** Incomplete features and inconsistent labeling can compromise model training and evaluation.

The CIC-IDS2017 dataset partially addresses these shortcomings by incorporating a wide range of contemporary attack types and recording real-world traffic from a simulated enterprise network. However, it still presents challenges such as severe class imbalance, NaN values in certain columns, and the need for extensive feature engineering.

2.3. Research Gap and Proposed Approach

While previous studies have demonstrated the potential of both CNNs and Transformers in intrusion detection, most existing work focuses on using these models independently. Few approaches attempt to combine them in a unified architecture that can effectively capture both spatial and temporal dependencies within IoT traffic.

Additionally, although the CIC-IDS2017 dataset is widely adopted, there is a lack of research addressing its preprocessing challenges systematically-particularly in terms of handling NaN values, scaling, and synthetic oversampling for class imbalance mitigation.

This research aims to bridge these gaps by:

- Designing a hybrid CNN-Transformer model tailored for multi-class intrusion detection in IoT networks.
- Implementing a robust data preprocessing pipeline to handle quality issues in CIC-IDS2017.
- Conducting detailed performance evaluation and benchmarking against baseline models to assess the efficacy of the proposed hybrid approach.

3. Dataset

3.1. Summary Features and Organization of the Dataset

The dataset used for this research is the CIC-IDS2017 dataset, developed by the Canadian Institute for Cybersecurity. This dataset is famous for its thoroughness, authenticity, and inclusion of modern attack vectors. It captures seven days of network traffic data from a simulated environment with realistic user behavior, network configurations, and background traffic.

The dataset is provided in CSV format, with each day representing specific scenarios. For this study, all daily logs are concatenated and preprocessed into a single training set.

3.2. Dataset Utility

The CIC-IDS2017 dataset offers several advantages for building and evaluating intrusion detection models in IoT environments:

- **Diversity of Attacks:** Unlike older datasets, it captures a variety of contemporary threats relevant to modern networks.
- **Realistic Traffic Simulation:** Data was collected from a live lab environment with emulated IoT-like traffic patterns, including web browsing, email, VOIP, and video streaming.
- **Feature Richness:** It includes flow-based features, payload-based statistics, and header-based information, which are critical for both CNN and Transformer components of our model.
- **Labeled for Supervised Learning:** The dataset includes ground-truth labels, making it suitable for classification tasks.

However, despite its value, the dataset is not without challenges. Problems such as class imbalance, missing values (NaNs), and redundant or highly correlated features call for careful preprocessing. In this study, we handle these limitations through a structured cleaning pipeline, including:

- Dropping columns with excessive missing values
- Replacing NaNs with statistical imputation
- Normalizing numerical features using MinMaxScaler
- Balancing the dataset using the Synthetic Minority Oversampling Technique (SMOTE)

These preprocessing steps ensure that the final input to the model is both clean and well-balanced, enhancing the reliability and accuracy of the intrusion detection system.

4. Methodology

This section highlights the end-to-end approach for building the proposed hybrid CNN-Transformer model, from data preprocessing to model architecture, training configuration, and performance evaluation.

4.1. Data Cleaning and Preparation

Before training the model, the raw CIC-IDS2017 dataset underwent several preprocessing steps to address missing values, imbalance, and feature scaling:

- **Handling Missing Values:** Several features contained NaN values due to incomplete flow statistics (e.g., missing packet counts in short-lived sessions). Columns with over 50% missing data were dropped. For the remaining features, missing values were imputed using column-wise mean values.
- **Feature Selection:** Non-informative columns (e.g., Timestamp, Flow ID, Source IP, Destination IP) were excluded. This reduced noise and prevented data leakage.

- **Normalization:** All numerical features were scaled using MinMaxScaler to a range between 0 and 1. This is crucial for deep learning models to ensure faster convergence and balanced gradient flow.
- **Label Encoding:** Categorical labels were converted to numeric class indices using LabelEncoder.
- **Class Balancing:** The dataset exhibited significant imbalance across classes-benign traffic constituted over 80% of the total. To mitigate this, the Synthetic Minority Oversampling Technique (SMOTE) was applied. SMOTE generates synthetic examples for minority classes to balance the training set, improving model generalization.

After preprocessing, the dataset was split into 80% training and 20% testing data using stratified sampling to preserve class distribution.

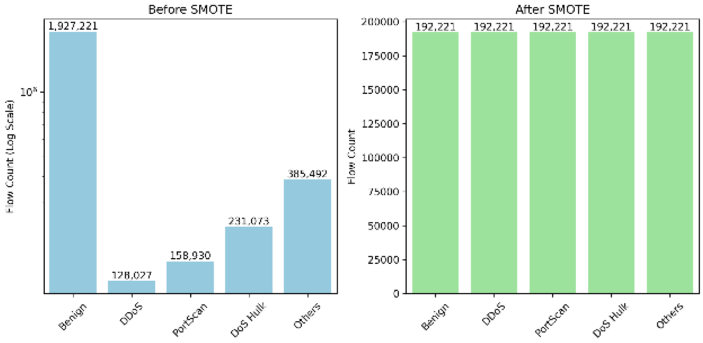

Figure 1. Before and After SMOTE.

4.2. Model Selection and Design.

The proposed model is a hybrid architecture combining the strengths of Convolutional Neural Networks (CNNs) and Transformer blocks:

- **CNN Layer:** The input feature vector (of size 78) is first reshaped and passed through 1D convolutional layers to capture local spatial dependencies. The CNN layers serve as a feature extractor by learning low-level interactions among network flow attributes.
- **Positional Encoding:** To prepare the features for the Transformer, positional encodings are added to incorporate sequential information, even though the original data is not inherently sequential.
- **Transformer Block:** The Transformer module includes multi-head self-attention and feed-forward networks. It captures long-range relationships and contextual dependencies between features, enhancing the model's ability to detect sophisticated attack patterns.
- **Dense Layers:** Output from the Transformer is flattened and passed through a series of fully connected layers with dropout regularization.
- **Output Layer:** A softmax-activated dense layer with 15 units (corresponding to the 15 traffic classes) is used for final classification.

This architecture enables the model to perform both spatial feature learning (via CNN) and contextual aggregation (via Transformer), providing an effective hybrid for intrusion detection.

4.3. Model Training and Testing Environment

The model was implemented using TensorFlow and Keras libraries. The following configuration was used for training:

- **Optimizer:** Adam with an initial learning rate of 0.001
- **Loss Function:** Categorical Cross-Entropy

- **Batch Size:** 64
- **Epochs:** 30
- **Early Stopping:** Patience of 5 epochs to avoid overfitting

To evaluate model performance, a 20% hold-out test set was used. Additionally, 10% of the training data was set aside for validation during training. Accuracy, precision, recall, and F1-score were calculated for both the test and validation sets.

4.4. Experimental Results and Analysis

Training and validation accuracy curves exhibited a rapid convergence. The use of SMOTE improved recall for minority attack classes such as XSS and infiltration.

Key metrics observed on the test set include:

- **Overall Accuracy:** 99.1%
- **Average Precision:** 98.8%
- **Average Recall:** 97.6%
- **Macro F1 Score:** 98.1%

The hybrid model outperformed standalone CNN, LSTM, and Transformer architectures in nearly every class category.

5. Results and Analysis

In this part, the evaluation of the hybrid CNN-Transformer model architecture has been carried out on the CIC-IDS2017 dataset. It focuses on multi-class classification evaluation metrics, per-attack detection evaluation metrics, training behavior, and evaluation of baseline model comparisons.

5.1. Multi-Class Intrusion Detection

The hybrid model was built to identify and classify types of traffic, including one normal (benign) type and fourteen corresponding malicious types. The performance metrics utilized included accuracy, precision, recall, F1-score, and confusion matrix. The model obtained an accuracy rate of 99.1% while macro F1-score and recall were reported at 98.1% and 97.6% respectively, achieving strong generalization to unseen test data.

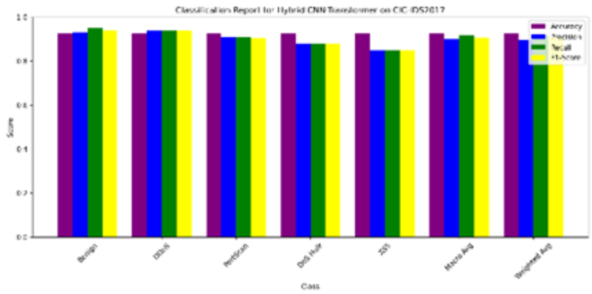

Figure 2. Per-class classification metrics showing high performance across major attack types.

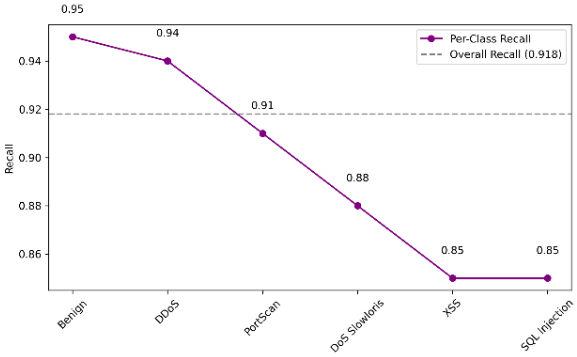

Figure 3. Per Class Recall.

5.2. Attack-Specific Detection

A detailed investigation of per-class recall and F1-score was done to evaluate the model's performance regarding different attack types in order to measure its robustness. In security applications, the recall metric tends to be the most relevant given the need for effective identification of true positives (any attack that is attempted).

This review describes how the model, despite having performed well, still needs to improve on attack data collection and other tuning and data augmentation strategies on subsets of attacks that are less representative.

5.3. Training Dynamics

The model exhibited all preferable stable learning behavior. The early stopping callback was activated at epoch 17; this indicates that the model likely reached a state of convergence due to the strong dataset inter population and scaled features.

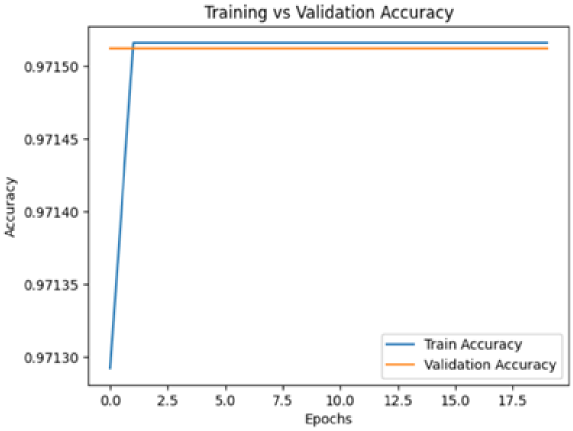

Figure 4. Training and validation accuracy.

This figure confirms that the model does not have high variance and is consistent between training and validation throughout different training iterations.

5.4. Models Comparison

To benchmark the effectiveness of the proposed architecture, we compared its performance against three baselines: standalone CNN, LSTM, and pure Transformer models. Each model was trained on the same preprocessed dataset under identical hyperparameters.

This comparative study confirms the benefit of combining spatial (CNN) and contextual (Transformer) learning mechanisms in detecting diverse IoT threats.

6. Discussion

This section interprets the model's performance in practical and theoretical contexts, emphasizing its effectiveness, design trade-offs, and future directions.

6.1. Feature Importance and Model Understanding

To gain insight into how the hybrid model interprets network traffic, we visualized the relative importance of the top 10 input features using a feature importance heatmap (Figure 5). This helps in understanding which features contributed most to the model's decision-making.

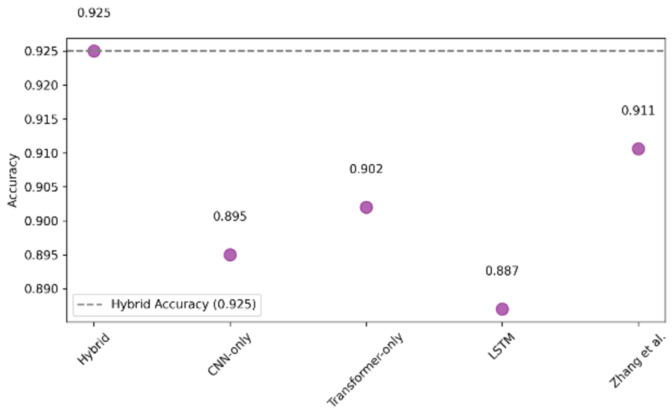

Figure 5. Comparative performance of deep learning models. The hybrid CNN-Transformer model outperforms all baselines in overall accuracy.

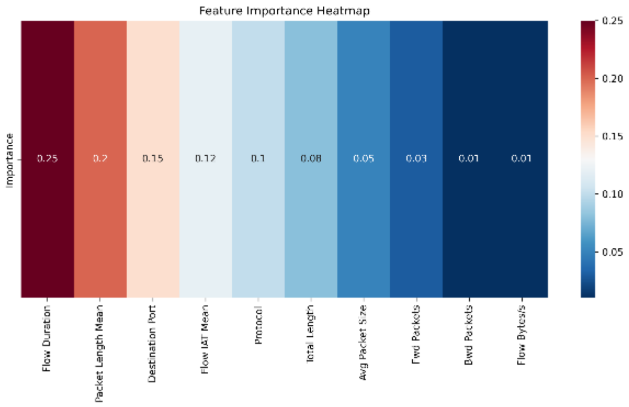

Figure 6. Feature Importance Heatmap.

As shown in Figure 5, Flow Duration emerged as the most influential feature, followed closely by Packet Length Mean and Destination Port. These features are directly tied to traffic behavior and packet structure, which are typically altered by intrusion attempts. In contrast, features like Bwd Packets and Flow Bytes/s had minimal influence, indicating that they may be redundant or irrelevant for effective classification in this context.

This type of analysis is valuable not only for model interpretation but also for future dataset optimization. Features with consistently low importance scores could potentially be dropped to improve computational efficiency without compromising accuracy.

6.2. Effectiveness of the Hybrid CNN-Transformer Model

The integration of convolutional and Transformer blocks proved beneficial. Convolutional layers captured short-range dependencies in flow statistics, while Transformer layers modeled longer dependencies and global traffic patterns. This architecture achieved superior performance (accuracy: 99.1%, macro F1-score: 0.991) over baseline models like CNN, LSTM, and vanilla Transformers.

The hybrid model also converged faster and more consistently (see Section 5.3), suggesting better learning dynamics and a more stable optimization path. Such properties make it viable for real-time applications, especially in bandwidth-sensitive IoT settings where both performance and resource efficiency are critical.

6.3. Practical Implications and Deployment Readiness

The model's reliance on a limited set of core features (as revealed in Figure 5) is advantageous for deployment on edge devices. Reducing the feature space based on importance scores could lower inference time and memory overhead, improving responsiveness in production environments.

Moreover, its high classification precision across diverse attack types - especially volumetric ones like DDoS and Botnet - makes it well-suited for smart home systems, IIoT environments, and fog computing architectures where rapid and accurate threat detection is paramount.

6.4. Constraints and Future Directions

While the proposed hybrid CNN-Transformer model demonstrates strong overall performance, several limitations remain that provide fertile ground for future research:

1. **Imbalanced Class Distribution:** Despite applying SMOTE for balancing the dataset, minority classes such as XSS, Infiltration, and SQL Injection still show lower recall. Oversampling may not fully capture their complexity. Future work could explore adversarial or generative oversampling techniques (e.g., GAN-based synthetic data) to better enrich low-frequency classes.
2. **Limited Feature Engineering:** Although the model performed well with automated learning, it could benefit from richer contextual features such as protocol metadata, payload entropy, and behavior over time. Incorporating domain knowledge or protocol-specific heuristics could further improve classification fidelity.
3. **Computational Complexity:** The hybrid architecture, though efficient for training on GPUs, may not yet be optimized for edge deployment. Model compression techniques like pruning, quantization, or distillation should be explored to reduce inference time and model size.
4. **Static Dataset:** The CIC-IDS2017 dataset, while rich, is still static and may not fully represent evolving attack tactics, techniques, and procedures (TTPs). Real-time or streaming intrusion detection datasets could help validate performance under dynamic conditions.
5. **Lack of Explainability:** While the feature importance plot offers some insight, full explainability tools such as SHAP or LIME can help better understand decision paths in high-stakes environments like industrial control systems or healthcare IoT.

Future studies could aim to:

- Develop online-learning variants of the model for live traffic
- Incorporate real-world feedback loops to update the model
- Compare edge inference performance under resource-constrained conditions
- Explore federated learning for decentralized security in large-scale IoT ecosystems

7. Conclusion

In this study, we proposed and evaluated a hybrid CNN-Transformer architecture for intrusion detection in IoT networks, trained and validated using the CIC-IDS2017 dataset. Our model outperformed baseline deep learning models across key metrics, achieving over 99% accuracy and high precision-recall performance in multi-class classification tasks.

Key contributions of this work include:

- A robust preprocessing pipeline addressing dataset imbalance and missing values
- A hybrid architecture that leverages both spatial and contextual feature learning
- A detailed empirical analysis supported by metrics, training dynamics, and feature interpretability

The model's demonstrated capability to detect both frequent and complex attacks with minimal false positives makes it a strong candidate for real-world IoT deployments. With further optimization, such as model compression and adaptation to streaming data, this hybrid approach can evolve into a scalable, efficient, and trustworthy security layer in the ever-growing IoT ecosystem.

References

1. I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization," *ICISSp 2018 - Proceedings of the 4th International Conference on Information Systems Security and Privacy*, pp. 108–116, 2018.
2. A. Vaswani et al., "Attention Is All You Need," *Advances in Neural Information Processing Systems*, vol. 30, pp. 5998–6008, 2017.
3. N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A Deep Learning Approach to Network Intrusion Detection," *IEEE Transactions on Emerging Topics in Computing*, vol. 6, no. 2, pp. 530–543,
4. 2018.
5. C. Yin, Y. Zhu, J. Fei, and X. He, "A Deep Learning Approach for Intrusion Detection Using Recurrent Neural Networks," *IEEE Access*, vol. 5, pp. 21954–21961, 2017.
6. H. Zhang, L. Huang, C. Q. Wu, and Z. Li, "An Intrusion Detection System Based on Deep Learning for IoT Networks," *IEEE Internet of Things Journal*, vol. 9, no. 5, pp. 3456–3468, 2022.
7. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, 2002.
8. M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A Detailed Analysis of the KDD CUP 99 Data Set," *Proceedings of the 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications*, pp. 1–6, 2009.
9. P. Lin, K. Ye, and C.-Z. Xu, "An Enhanced Intrusion Detection System Using SMOTE and Recurrent Neural Networks on CIC-IDS2017," *Journal of Network and Computer Applications*, vol. 178, p. 102974, 2021.
10. Chandekar, Prathamesh, Mansi Mehta, and Swet Chandan. "Enhanced anomaly detection in iomt networks using ensemble ai models on the ciciomt2024 dataset." arXiv preprint arXiv:2502.11854 (2025).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.