International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 2, 2025

A Convolutional Neural Network-Based Deep Learning Framework for Automated Cyberattack Detection in IoT Applications

Sivananda Hanumanthu[1], G. Anil Kumar[2],

[1] Research Scholar, Department of CSE, Bharatiya Engineering Science and Technology Innovation University (BESTIU), Andhra Pradesh & Director of Enterprise Architecture at Rubrik, Bangalore, India.

* Corresponding Author Email: siva.phd1984@gmail.com ORCID: 0009-0003-3763-3952

[2] Principal & Professor Dept of CSE, Scient Institute of Technology, Hyderabad, TS, India.

Email: anil_deva@yahoo.com

Abstract

With the combination of diverse IoT technologies and no global standards imposed, the IoT has further opened up revolutionary, innovative applications yet has also presented new and complex security challenges. Restrictions are needed to safeguard IoT applications since cyber threats are increasing daily. Artificial Intelligence (AI) has made it possible to solve many real-world issues. This study introduces the Learning-based Cyberattack Detection system (LbCADF). This deep learning-based system employs a CNN-based model with enhanced sensitivity for autonomously detecting cyberattacks in IoT settings. The framework successfully distinguishes between benign and malevolent traffic flows. We add feature selection and hyperparameter tweaking to improve training quality to our proposed system, Enhanced CNN for Attack Detection and Classification (ECNN-ADC). To prevent overfitting, an early stopping criterion is applied. This work has been evaluated on the benchmark dataset UNSW-NB15. At the same time, the empirical results indicate that ECNN-ADC achieves the highest detection accuracy (95%) compared to various of the latest models (MLP, baseline CNN).

Keywords – Network Security, IoT Security, Artificial Intelligence, CNN-Based Threat Detection, Intrusion Detection System

1. INTRODUCTION

Security has become imperative, with computers and networks becoming indispensable resources in cyberspace, and the Internet of Things (IoT) applications are said to be multiplying. This connectivity has simultaneously brought a wave of successful cyberattacks thanks to serious security holes in the physical ecosystem. These vulnerabilities stem from naturally utilizing resource-constrained devices and lacking a globally recognized set of security protocols for IoT integration. Learning-Based Techniques for IoT Network Security: Deep Learning Algorithms The advent of Big Data with Artificial Intelligence (AI) and its methodologies have enabled security solutions to be provided for complex networks, such as IoT, through learning-based techniques [1]. As explained in [2] and [4], deep learning models, which are inspired by the structure of human brains, can be used to identify cybersecurity threats with great accuracy. Additionally, working on industrial cyber-physical systems shows deep learning approaches to mitigate cyberattacks [6], [7], [9]. Deep learning techniques detect various types of cyber threats, including distributed attacks. Deep learning, thereby, has a great potential to distinguish complicated patterns and features, which can contribute to improving cyberattack detection in the academic community based on a literature review.

Gani et al. [1] identified the security challenges driven by the proliferation of IoT and the need to combine big data and deep learning to reach practical solutions. For ICS, Parizi et al. [6] created a cyberattack detection algorithm based on deep learning, which achieved better detection performance than traditional classifiers in accuracy and score. Park et al. [7] introduced DeepBlockIoTNet. This secure deep learning-based model combines deep learning with blockchain technology and aims to overcome problems such as centralized control, security, privacy, and DOI improvements. Diro et al. [13] addressed the increasing concern of cyber-attacks by creating a distributed deep-learning framework for discovering breaches in IoT/Fog networks. Molanes et al. Deep learning was identified as a key technology for enabling knowledge extraction and usage in IoTs, as networks and IoT applications often generate large amounts of data ({[20]}). The literature review indicates a need to optimize CNN models for more effective performance in identifying cyberattacks.

The main contributions of this paper are as follows:

1. We devise an improved CNN-based framework named the Framework for Learning-based Cyberattack Detection (LbCADF).
2. To this end, we propose an algorithm called Attack Detection and Classification Enhanced CNN (ECNN-ADC) to implement the framework and enhance cyberattack classification efficiency.
3. We apply the proposed framework to the benchmark dataset UNSW-NB15 and validate that ECNN-ADC's performance is

DOI: https://doi.org/10.62647/IJITCE2025V13I2PP1453-1461

ISSN 2347–3657
Volume 13, Issue 2, 2025

better than that of some cutting-edge models, such as baseline CNN and MLP.

This is how this document is structured: Section 2 reviews the literature on current intrusion detection mechanisms for networks and IoT applications. Sec. 3 outlines the baseline CNN model from which the enhanced CNN model is derived. Materials and Methods: The proposed framework, the enhanced CNN model, the dataset, the algorithm, and the performance evaluation criterion are described in detail in Section 4. Our experimental results are in Section 5, while Section 6 discusses the study's contributions and limitations. Finally, Section 7 makes conclusions about our research and directions for future work.

2. LITERATURE & RELATED WORK

This section reviews the literature on deep learning models used in existing research to detect cyberattacks. According to Gani et al. [1], the expansion of IoT creates security issues. Big data and deep learning can potentially address vulnerabilities, necessitating innovative solutions. Ji et al. [2] proposed an IPSO-LSSVM method for identifying green peppers amidst challenging foliage, yielding 89.04% accuracy. Shu et al. [3] analyzed federated deep learning in IoT cybersecurity, assessing vulnerabilities and performance compared to centralized learning. Pecori et al. [4] investigated deep learning applications in IoT security, presenting a systematic review and future research directions. Scotti et al. [5] examined recent deep learning-based intrusion detection methods in cybersecurity, emphasizing benchmark datasets' significance. The deep learning-based ICS cyber-attack detection model put forth by Parizi et al. [6] outperformed traditional classifiers regarding the accuracy and F1 score. To overcome centralized management, security, and privacy issues and improve accuracy, Park et al. [7] suggested DeepBlockIoTNet, which combines safe DL with blockchain for IoT. Driss et al. [8] investigated DL and IoT, revolutionizing smart cities and enhancing various sectors. Their survey highlights applications, challenges, and open issues. Khalil et al. [9] stated that DL enhances IIoT with innovative manufacturing and accident prevention applications. This review addresses potential challenges and directions. Sarker [10] explored deep learning's application in cybersecurity, emphasizing neural network techniques and possible research directions. Diro et al. [11] observed that the rise in IoT devices calls for distributed deep learning for effective cyber-attack detection, proving its superiority over shallow models. Koroniotis et al. [12] focused on a Particle Deep Framework (PDF) that enhances IoT network security via effective cyber-attack detection and tracing, outperforming other models. Diro et al. [13] suggested a distributed deep learning system for attack detection in IoT/Fog networks in response to growing cybersecurity risks. Demonstrated superiority over centralized models and traditional machine learning approaches highlights its potential for real-time threat detection. Further comparative analysis and exploration of network payload data are planned for future investigations. Shahraki et al. [14] explored the application of deep learning in Network Traffic Monitoring and Analysis (NTMA) for modern communication systems. It discusses the benefits and challenges of deep learning techniques, highlighting their role in traffic classification, fault management, and network security, along with future research directions.

Rahman et al. [15] tested DL-based COVID-19 diagnostic applications against adversarial attacks, emphasizing the need for improved defense mechanisms. They plan to enhance defense strategies against attacks, including AE and malware. They intend to explore additional attack vectors and implement industry-standard tools for evaluation and defense. Dixit et al. [16] highlighted deep learning's role in enhancing cybersecurity applications, discussing various algorithms and their benefits. Analysis reveals significant accuracy, scalability, and reliability improvement in real-time implementations. Gui et al. [17] proposed that the LAE-BLSTM hybrid deep learning method efficiently detects botnets in IoT networks by reducing memory usage and achieving robust classification performance. Kondaka et al. [18] Introduced the iCAIDL algorithm, integrating IoT, Cloud, and deep learning for healthcare enhancement. Further suggestions include multi-cloud adoption for improved efficiency. Ahmad et al. [19] reviewed recent trends in IoT security, emphasizing the need for advanced machine-learning techniques. It discusses key research questions and highlights the integration of IoT, cybersecurity, and big data. According to Molanes et al. [20], the increasing impact of the Internet of Things encourages research into knowledge usage and data extraction. The transformational potential of deep learning in handling large IoT datasets makes it stand out.

3. PRELIMINARIES

This section lays the groundwork for understanding our proposed methodology in Section 4. We have further developed this model into our deep learning framework. CNN is a deep learning model widely used in different domains, including intrusion detection research. According to its design, CNN, which comprises convolutional and pooling layers, can learn complex correlations from high-dimensional data. This is why it is excellent to work with network traffic data. In this study, the UNSW-NB15 dataset has 78 features, and CNN has shown an ability to process such complex, high-dimensional data via URL swiftly.

DOI: https://doi.org/10.62647/IJITCE2025V13I2PP1453-1461

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 13, Issue 2, 2025

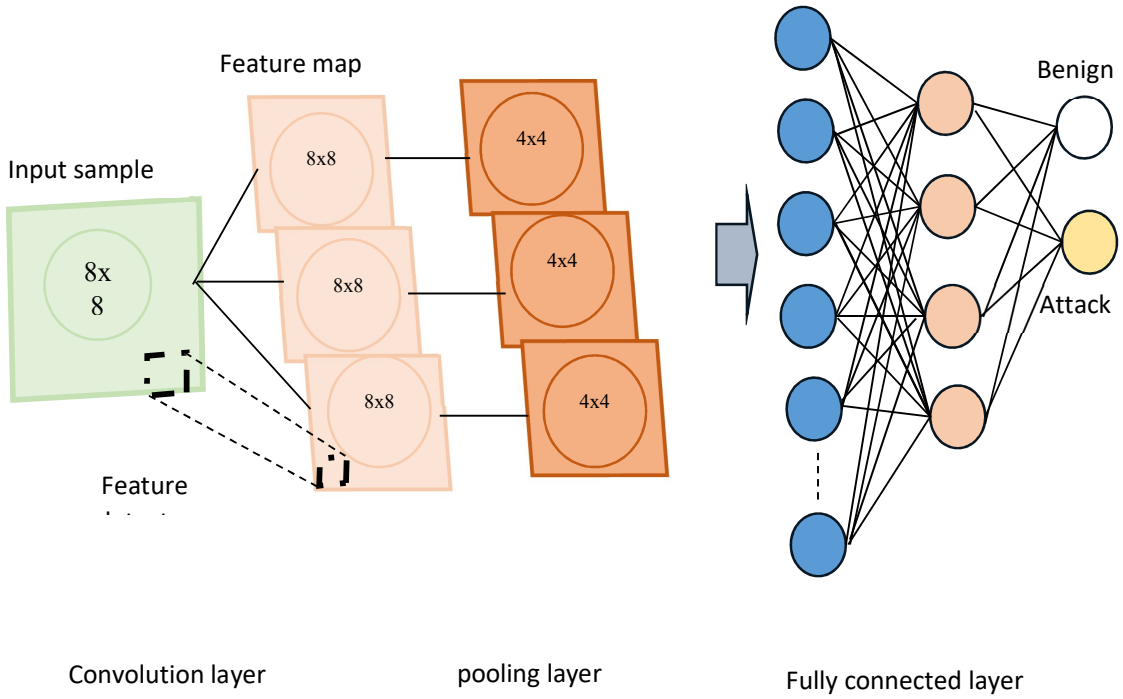

Figure 1: Architecture of a typical CNN

Figure 1 demonstrates the architecture of a typical CNN model, which is created with an input size of an 8×8 matrix. The convolution layer, pooling layer, and fully linked layer are the three layers that make up this model. The feature maps from the input data are extracted for the convolutional layer and optimized in the pooling layer. The fully connected layer performs classification. Equation 1 represents how these feature maps are generated in the convolutional layer:

$$feature\ map\{i\} = input \otimes feature\ detector\ \{i\}; i \epsilon K \quad (1)$$

After generating feature maps, we apply an activation function to add non-linearity to our model. Pooling layer: The feature is optimized in the pooling layer, lowering dimensionality. Then, after pooling, a flattening process changes it into a simple one-dimensional format for the fully connected layer. This last level comprises input, optional hidden, and output neurons that supply the target class labels. In the context of this research, the output classes are related to different courses of cyberattacks.

CNNs have shared weights, making them highly efficient and resulting in fewer parameters that need to be optimized. This weight-sharing converges faster for the optimizer and prevents overfitting as well. Moreover, the constraints imposed on the weights of the model help in generalization, and hence, CNNs can perform well in detecting cyberattacks when deployed for Intrusion Detection and Analysis (IDA). CNNs are also well-suited for learning complex patterns from training data. This paper contributed a data-balancing pre-processing step for classifying the UNSW-NB15 dataset using deep learning.

4. EXPERIMENTS AND METHODS

This section presents our methodology for efficient detection and classification of cyberattacks.

4.1 Problem Definition

This study aims to tackle the issue of devising a deep learning framework capable of automatically identifying and categorizing cyberattacks from unlabeled network flows. Regarding attack detection and classification, our prototype system should be more effective than current models.

4.2 Proposed Framework

As shown in Figure 2, we propose a deep learning-based framework that automatically classifies the different types of cyberattacks based on training data. To enhance performance, the framework implements strong pre-processing techniques. In this section, the given dataset goes through an elaborate pre-processing stage, including Feature selection techniques, hyperparameter tuning with XGBoost, and PCA and t-SNE for dimensionality reduction.

DOI: https://doi.org/10.62647/IJITCE2025V13I2PP1453-1461

International Journal of Information Technology & Computer Engineering
ISSN 2347-3657
Volume 13, Issue 2, 2025

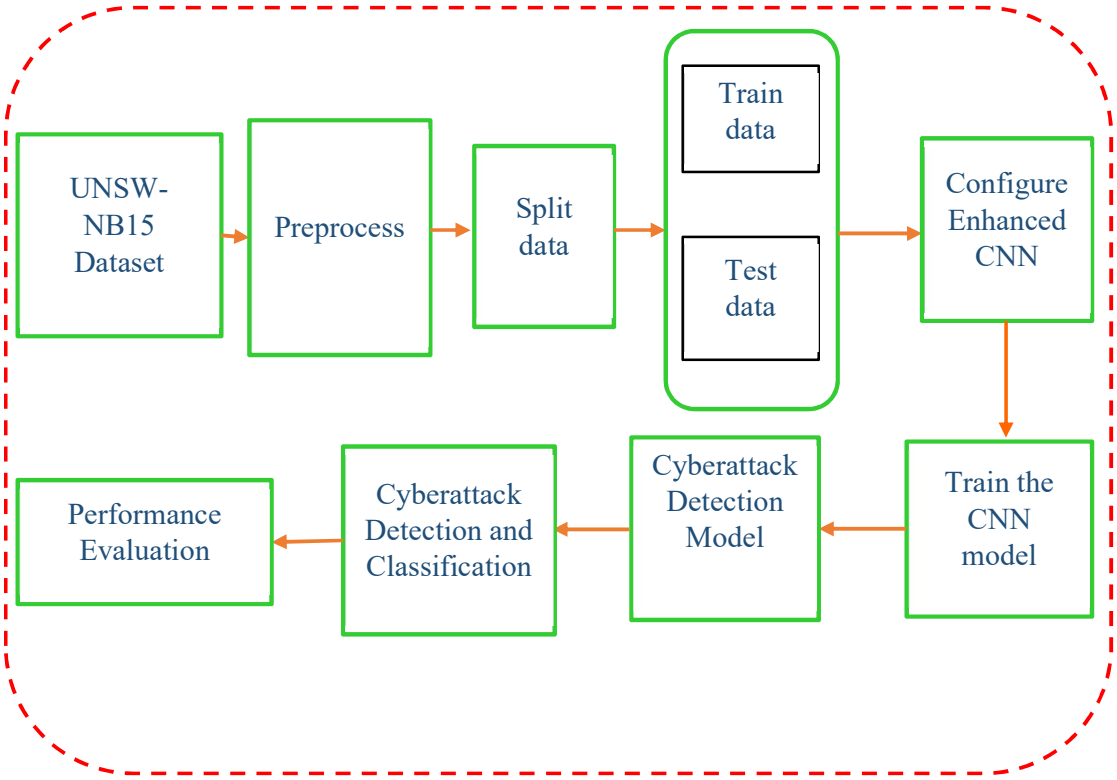

Figure 2: Deep learning-based framework for classifying and detecting cyberattacks

After evaluating feature importance, a threshold is placed for feature selection—a threshold (initially set to 0.1 but can be adjusted empirically in this framework). We computed by multiplying the average feature importance by 0.1. Then, a subset of 80% of the dataset is assigned to the training set (T1), and the remaining 20% is allocated to the test set (T2). A deep CNN Model (briefly described in 4.3) is configured and compiled. The training data is used to train this model, and the trained model is then saved for later use. The trained model is fed test data with no instances. Thus, an advanced CNN model is used to detect and classify cyberattacks.

4.3 Enhanced CNN Model

We empirically designed layers to enhance our baseline CNN model presented in Section 3, which improved cyberattack prediction and performance. Figure 3 illustrates the proposed model. The implementation utilizes Python 3 with PyTorch for the framework involved in the layer configuration of the improved CNN model. The model is constructed with the ideal number of layers per an empirical study, echoing previous work in [41], [43], and [44]. Due to the complexity of the task, deeper CNN architectures, such as the ResNet50 architecture, may be more suitable. However, increasing the number of layers does not always lead to better performance [44]. Therefore, in this work, we empirically defined the number of layers.

You are trained data until October 2023; the model receives batches of matrices as input—size 9×9, including 77 features and four zero-padded values. Our improved model comprises two fully connected and convolutional layers and two max-pooling layers. The size of the kernel, number of layers, hyperparameters, and neuron count were optimized through an exhaustive grid search. A set of values and one or more evaluation rounds are used to choose hyperparameters.

Use a kernel size within the architecture's first convolutional layer (Figure 3). 3×3, while maintaining padding and strides of 1 and using 16 maps for the input feature map set, with ReLU as the activation function. The model then follows it with ReLU activation on the generated 16 feature maps of 9 × 9 for each matrix in the training batch. A max pooling layer with a sliding window of size 2×2, padding 1, and stride two is applied to the feature maps, resulting in 16 feature maps size 5×5. Each of the second convolutional and max pooling layers follows the same kernel size, padding, and stride parameters as their respective preceding layers.

DOI: https://doi.org/10.62647/IJITCE2025V13I2PP1453-1461

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 13, Issue 2, 2025

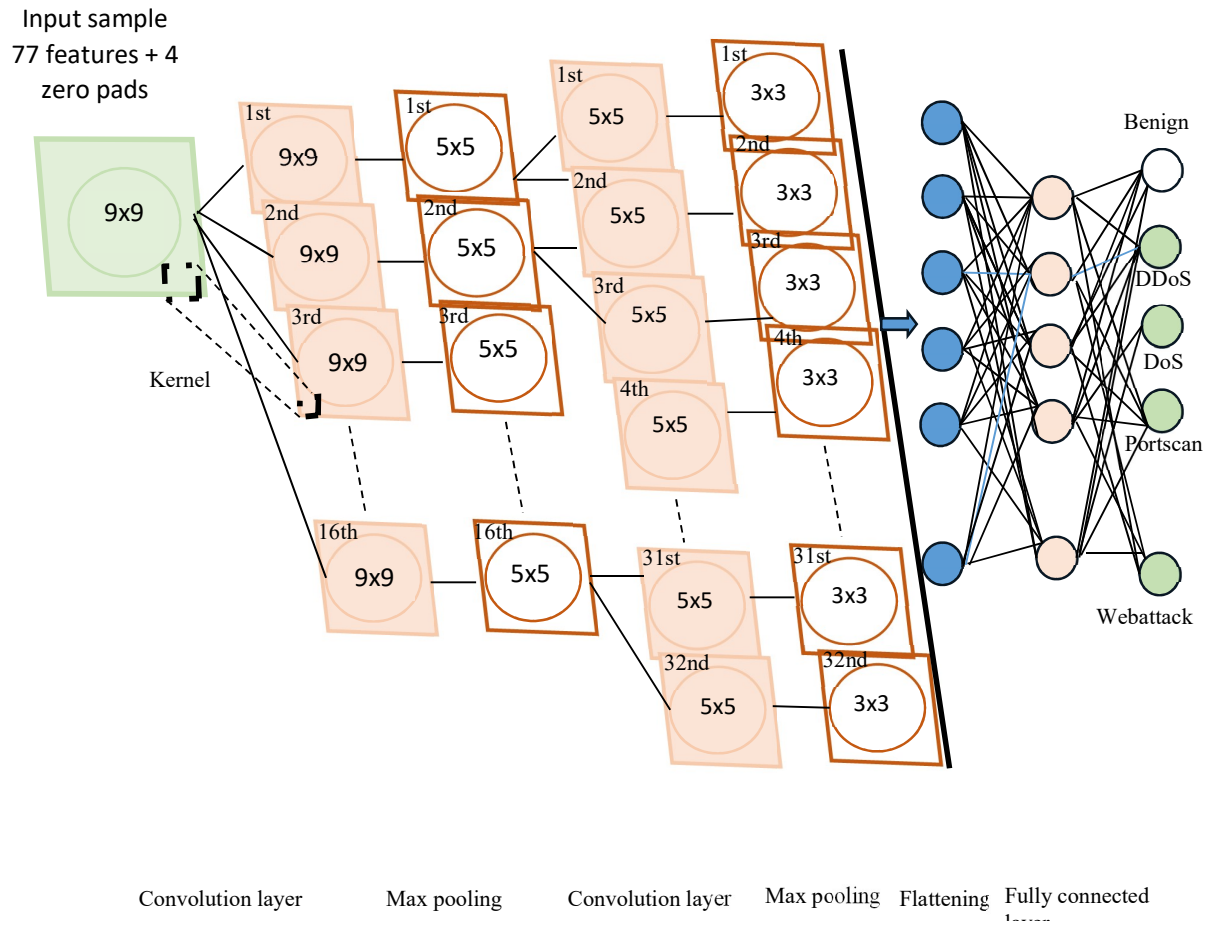

Figure 3: Architectural overview of the proposed system for automatic detection of cyberattacks

This gives us 32 feature maps from the second convolutional layer, which goes through ReLU. The pooling layers lead to 32 3x3 feature maps pooled for each input matrix. Flatten Layer The flattening layer takes pooled feature map(s) and converts it into a 1D array, which requires layers for fully connected networks. The actual number is 288 elements for any input matrix. So, the input to a fully connected layer shall have 288 neurons and one hidden layer followed by nine output neurons. For multi-class classification, the proposed model is trained with the training set. The model is made to be tuned to several class labels. At this point, the model is set to assign nine classes for each network flow.

4.4 Proposed Algorithm

We proposed Attack Detection and Classification Enhanced CNN (ECNN-ADC), a Hybrid attack detection and classification algorithm.

DOI: https://doi.org/10.62647/IJITCE2025V13I2PP1453-1461

International Journal of Information Technology & Computer Engineering
ISSN 2347–3657
Volume 13, Issue 2, 2025

Algorithm: Improved CNN for Classification and Attack Detection

Input:
Cybersecurity dataset: UNSW-NB15 dataset D
Detection threshold: th
Early stopping criterion: c

Output:
Classification results: R
Performance metrics: P

1. **Begin**
2. **Data Preprocessing:**
   2.1. Apply necessary preprocessing techniques on D (e.g., normalization, encoding categorical features).
   2.2. Obtain the preprocessed dataset D'.
3. **Dataset Splitting:**
   3.1. Split D' into training set T_train and testing set T_test.
4. **Enhanced CNN Model Configuration:**
   4.1. Initialize and configure the enhanced CNN architecture, integrating advanced layers (as depicted in Figure 3).
5. **Model Compilation:**
   5.1. Assemble the model using the proper optimizer, evaluation metrics, and loss function.
6. **Model Training:**
   6.1. Train model m on T_train using early stopping criterion c to avoid overfitting.
7. **Model Saving:**
   7.1. Save the trained model m for future inference.
8. **Model Evaluation:**
   8.1. Use the trained model m to predict on T_test and obtain classification results R.
9. **Performance Assessment:**
   9.1. Evaluate R against ground truth labels to compute performance statistics P (e.g., accuracy, precision, recall, F1-score).
10. **Result Presentation:**
    10.1. Display classification results R.
    10.2. Display performance metrics P.
11. **End**

Algorithm 1: Enhanced CNN for Attack Detection and Classification

It has as input dataset D, threshold Th, and early stopping criterion c of UNSW-NB15, and classification results and performance statistics as output, as indicated in Algorithm 1. The algorithm allows for a robust pre-process methodology, as detailed in Section 4.2. After pre-processing, the dataset is divided into the testing set (T2) in 20% and a training set (T1) in 80% of cases, respectively. Then, the improved CNN model is set, as shown in Figure 3. T1 contains the compiled mod and train with T1 and the early stopping. Once training is complete, we will persist with the model for future reuse. Next, the model is restored for multiclass classification on test data T2. Performance statistics are determined by comparing the algorithm's predictions against the ground truth labels.

DOI: https://doi.org/10.62647/IJITCE2025V13I2PP1453-1461

International Journal of Information Technology & Computer Engineering

ISSN 2347–3657

Volume 13, Issue 2, 2025

5. RESULTS AND DISCUSSION

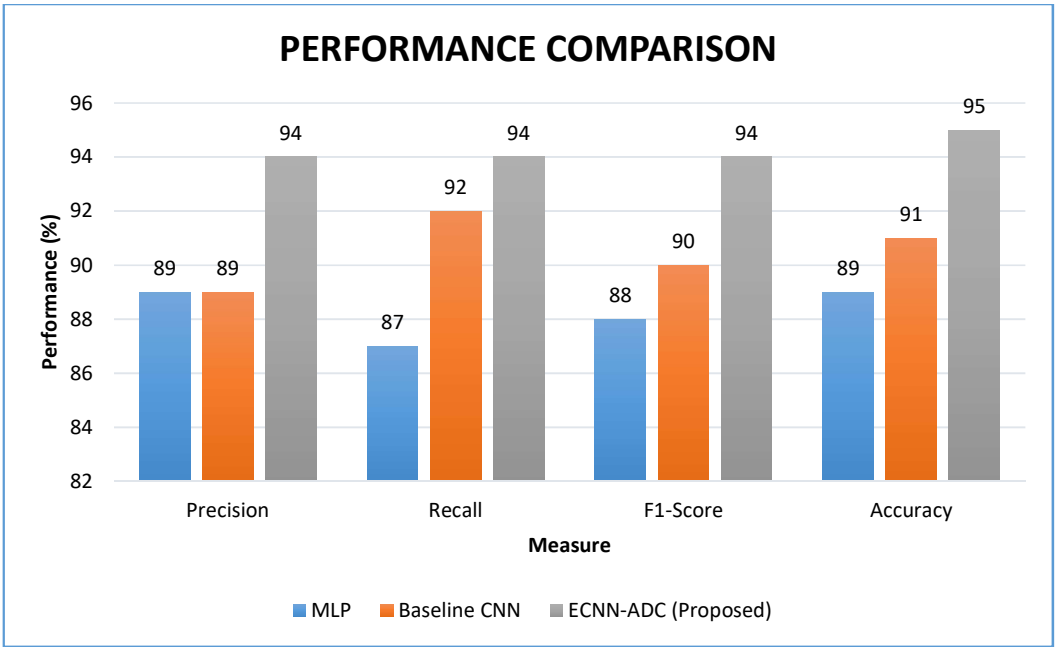

Figure 4: Attack detection performance evaluation

The results are illustrated in Figure 4, showing the performance comparisons of the proposed ECNN-ADC algorithm with state-of-the-art models. Compared to MLP and baseline CNN, the suggested algorithm has the highest precision (89%, 89%, and 94%, respectively). Similarly, MLP and baseline-CNN obtained 87% and 92% recall, respectively, whereas ECNN-ADC produced the highest with 94%. The harmonic mean of precision and recall is the F1-score, so its trend follows these indicators. MLP F1-score = 88%, baseline CNN F1-score = 90% and the proposed algorithm F1-score = 94%. Baseline and MLP CNN's accuracy rates were 89% and 91%, respectively, while the accuracy achieved by the proposed algorithm was 95%.

6 . CONCLUSION

We presented the Learning-based Cyberattack Detection Framework (LbCADF), a DL-based framework, in this research to automate the process of cyberattack detection in the context of IoT based on an improved form of CNNs. The framework's objective is to identify traffic flow and classify traffic flow into malicious and benign. Solid pre-processing contains dimensionality reduction via feature selection and hyperparameter tweaking using XGBoost, as well as PCA and t-SNE of the dataset. The number of features is decreased through dimensionality reduction, whereas hyperparameter tuning improves the model performance according to the dataset. Feature selection is performed to keep the features that contribute most to predicting the class label while dropping the features that do not significantly influence the prediction of the class label. We suggested the Enhanced CNN for Attack Detection and Classification (ECNN-ADC) algorithm, which combines hyperparameter adjustment and feature selection to enhance training quality. To avoid overfitting, an early stopping criterion was applied. The benchmark was implemented successfully using the UNSW-NB15 dataset. The empirical study results validated our hypothesis that ECNN-ADC achieves a 95% accuracy of cyberattack detection while outperforming various impressive models like baseline CNN. We will extend this work further by applying feature engineering and scaling the model; this should improve the model's ability to detect new attacks.

References

[1] Amanullah, Mohamed Ahzam; Habeeb, Riyaz Ahamed Ariyaluran; Nasaruddin, Fariza Hanum; Gani, Abdullah; Ahmed, Ejaz; Nainar, Abdul Salam Mohamed; Akim, Nazihah Md; Imran, Muhammad (2020). *Deep learning and big data technologies for IoT security. Computer Communications, S0140366419315361–. http://*doi:10.1016/j.comcom.2020.01.016

[2] Ullah, Farhan; Naeem, Hamad; Jabbar, Sohail; Khalid, Shehzad; Latif, Muhammad Ahsan; Al-Turjman, Fadi and Mostarda, Leonardo (2019). Cyber Security Threats detection in

DOI: https://doi.org/10.62647/IJITCE2025V13I2PP1453-1461

Internet of Things using Deep Learning approach. IEEE Access, 1–1. http://doi:10.1109/ACCESS.2019.2937326

[3] MOHAMED AMINE FERRAG, OTHMANE FRIHA, LEANDROS MAGLARAS, HELGE JANICKE AND LEI SHU. (2021). Federated Deep Learning for Cyber Security in the Internet of Things: Concepts, Applications, and Experimental Analysis. *IEEE*. 9, pp.138509-138542. http://doi:10.1109/ACCESS.2021.3118642

[4] Lerina Aversano; Mario Luca Bernardi; Marta Cimitile and Riccardo Pecori; (2021). A systematic review on Deep Learning approaches for IoT security. Computer Science Review. http://doi:10.1016/j.cosrev.2021.100389

[5] Gumusbas, Dilara; Yldrm, Tulay; Genovese, Angelo and Scotti, Fabio (2020). A Comprehensive Survey of Databases and Deep Learning Methods for Cybersecurity and Intrusion Detection Systems. IEEE Systems Journal, 1–15. http://doi:10.1109/JSYST.2020.2992966

[6] Al-Abassi, Abdulrahman; Karimipour, Hadis; Dehghantanha, Ali and Parizi, Reza M. (2020). An Ensemble Deep Learning-based Cyber-Attack Detection in Industrial Control System. IEEE Access, 1–1. http://doi:10.1109/ACCESS.2020.2992249

[7] Shailendra Rathore and Jong Hyuk Park; (2021). A Blockchain-Based Deep Learning Approach for Cyber Security in Next Generation Industrial Cyber-Physical Systems. IEEE Transactions on Industrial Informatics. http://doi:10.1109/TII.2020.3040968

[8] Atitallah, Safa Ben; Driss, Maha; Boulila, Wadii and GhÃ©zala, Henda Ben (2020). Leveraging Deep Learning and IoT big data analytics to support the smart cities development: Review and future directions. Computer Science Review, 38, 100303–. http://doi:10.1016/j.cosrev.2020.100303

[9] Ruhul Amin Khalil; Nasir Saeed; Mudassir Masood; Yasaman Moradi Fard; Mohamed-Slim Alouini and Tareq Y. Al-Naffouri; (2021). Deep Learning in the Industrial Internet of Things: Potentials, Challenges, and Emerging Applications . IEEE Internet of Things Journal, –. http://doi:10.1109/jiot.2021.3051414

[10] Iqbal H. Sarker; (2021). Deep Cybersecurity: A Comprehensive Overview from Neural Network and Deep Learning Perspective . SN Computer Science. http://doi:10.1007/s42979-021-00535-6

[11] Abeshu, Abebe and Chilamkurti, Naveen (2018). Deep Learning: The Frontier for Distributed Attack Detection in Fog-to-Things Computing. IEEE Communications Magazine, 56(2), 169–175. http://doi:10.1109/MCOM.2018.1700332

[12] Koroniotis, Nickolaos; Moustafa, Nour and Sitnikova, Elena (2020). A particle deep framework is a new network forensic framework based on deep learning for Internet of Things networks. Future Generation Computer Systems, 110, 91–106. http://doi:10.1016/j.future.2020.03.042

[13] Diro, Abebe Abeshu and Chilamkurti, Naveen (2017). Distributed attack detection scheme using deep learning approach for Internet of Things. Future Generation Computer Systems, S0167739X17308488–. http://doi:10.1016/j.future.2017.08.043

[14] Mahmoud Abbasi; Amin Shahraki and Amir Taherkordi; (2021). Deep Learning for Network Traffic Monitoring and Analysis (NTMA): A Survey . Computer Communications. http://doi:10.1016/j.comcom.2021.01.021

[15] Rahman, Abdur; Hossain, M. Shamim; Alrajeh, Nabil A. and Alsolami, Fawaz (2020). Adversarial Examples â“ Security Threats to COVID-19 Deep Learning Systems in Medical IoT Devices. IEEE Internet of Things Journal, 1–1. http://doi:10.1109/JIOT.2020.3013710

[16] Dixit, Priyanka and Silakari, Sanjay (2021). Deep Learning Algorithms for Cybersecurity Applications: A Technological and Status Review. Computer Science Review, 39, 100317–. http://doi:10.1016/j.cosrev.2020.100317

[17] Popoola, Segun I.; Adebisi, Bamidele; Hammoudeh, Mohammad; Gui, Guan and Gacanin, Haris (2020). Hybrid Deep Learning for Botnet Attack Detection in the Internet of Things Networks. IEEE Internet of Things Journal, 1–1. http://doi:10.1109/jiot.2020.3034156

[18] Lakshmi Sudha Kondaka; M. Thenmozhi; K. Vijayakumar and Rashi Kohli; (2021). An intensive healthcare monitoring paradigm by using IoT based machine learning strategies . Multimedia Tools and Applications. http://doi:10.1007/s11042-021-11111-8

[19] Rasheed Ahmad and Izzat Alsmadi; (2021). Machine learning approaches to IoT security: A systematic literature review . Internet of Things. http://doi:10.1016/j.iot.2021.100365

[20] Fernandez Molanes, Roberto; Amarasinghe, Kasun; Rodriguez-Andina, Juan and Manic, Milos (2018). Deep Learning and Reconfigurable Platforms in the Internet of Things: Challenges and Opportunities in Algorithms and Hardware. IEEE Industrial

DOI: https://doi.org/10.62647/IJITCE2025V13I2PP1453-1461

ISSN 2347–3657

Volume 13, Issue 2, 2025

Electronics Magazine, 12(2), 36–49. http://doi:10.1109/MIE.2018.2824843

[21] The UNSW-NB15 Dataset. Retrieved from https://research.unsw.edu.au/projects/unsw-nb15-dataset

DOI: https://doi.org/10.62647/IJITCE2025V13I2PP1453-1461