Comparative Analysis of CNN, LSTM, and Hybrid CNN-LSTM Performance for Binary Classification-Based Cyber Attack Detection on CIC-IDS2017 Dataset

Abiyyu Kumara Nayotama
Informatics Engineering
Faculty of Computer Science
University of Brawijaya
Malang, East Java, Indonesia
abiyyukumara@student.ub.ac.id

Muhammad Hasan Fadhlillah
Informatics Engineering
Faculty of Computer Science
University of Brawijaya
Malang, East Java, Indonesia
hasanfadhlillah@student.ub.ac.id

Muhammad Husain Fadhlillah
Informatics Engineering
Faculty of Computer Science
University of Brawijaya
Malang, East Java, Indonesia
muhammadhusainf@student.ub.ac.id

Abstract - This study aims to compare the performance of deep learning models in cyberattack detection using the CIC-IDS2017 dataset. Three model architectures tested include Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and a hybrid CNN-LSTM model. Additionally, an autoencoder is applied for dimensionality reduction, aiming to reduce data complexity while preserving relevant information. The study focuses on binary classification (normal vs. attack) and evaluates models based on accuracy, precision, recall, F1-score, as well as ROC-AUC and PR-AUC. Experimental results show that the LSTM model performs best with an accuracy of 96%, ROC-AUC of 99.12%, and PR-AUC of 99.01%, outperforming CNN and the CNN-LSTM hybrid model. The CNN model also performs well with an accuracy of 93%, although CNN-LSTM shows the lowest performance. These findings suggest that LSTM is more effective for intrusion detection and can be further optimized to enhance detection performance. The study also recommends further exploration of parameter optimization and enhanced data preprocessing for real-world attack detection applications.

Keywords : cyberattack detection, CNN, LSTM, CNN-LSTM, autoencoder, dimensionality reduction, CIC-IDS2017, binary classification, deep learning.

I. Introduction

A. Background

Along with the rapid development of information technology, cybersecurity threats are becoming increasingly complex and sophisticated. Cyberattacks have evolved into a serious threat that can cause significant losses to organizations and individuals. Traditional intrusion detection systems (IDS) that rely on rule-based approaches are beginning to show limitations in the face of new and increasingly sophisticated attacks. This is driving research towards machine learning-based approaches, particularly deep learning, which has the ability to recognize complex patterns in network data. As shown by Xu et al., deep learning methods have excellent feature extraction capabilities and can overcome the limitations of traditional methods that require a priori knowledge [Xu et al.].

In recent years, deep learning architectures such as Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and their combination have shown promising results in various classification tasks. Research conducted by Zhang et al. [2023] showed that deep learning approaches can achieve up to 99% accuracy in detecting DDoS attacks. CNN excels in extracting spatial features, while LSTM is effective in understanding temporal dependencies in sequential data. This is evidenced in Garcia et al.'s research, which compared the performance of CNN, LSTM, and CNN-LSTM, where the CNN-LSTM hybrid model showed a higher classification rate (84.76%) compared to using LSTM (79.14%) or CNN (84.58%) separately [Garcia et al.]. Furthermore, research conducted by Kumar and Singh [2023] revealed that hybrid architectures can improve anomaly detection capabilities by up to 15% compared to traditional approaches.

The CIC-IDS2017 dataset provides comprehensive and realistic data to test the effectiveness of various deep learning approaches in the context of cybersecurity. This dataset has been widely used in cybersecurity research and has proven to be effective in evaluating intrusion detection systems, as shown in the comparative study by Sharafaldin et al. [2018]. With a total of 3.1 million records covering various types of attacks, this dataset provides a solid foundation for conducting performance comparison analysis between different deep learning architectures. Liu and Wang [2022] in their research used this dataset to develop a deep learning-based intrusion detection model capable of detecting zero-day attacks with an accuracy rate of 97%.

B. Problem Formulation

The problem formulation in this project is as follows:

1. How do CNN, LSTM, and hybrid CNN-LSTM models perform in detecting cyberattacks using binary classification on the CIC-IDS2017 dataset?
2. How are the advantages and disadvantages of each model architecture characterized in the context of cyberattack detection?
3. How does the effectiveness of the hybrid CNN-LSTM approach compare to the use of CNN or LSTM separately?

C. Objective

Based on the previous problem formulation, the objectives of this project are as follows:

1. Analyze and compare the accuracy, precision, recall, and F1-score of CNN, LSTM, and hybrid CNN-LSTM models in cyber attack detection.
2. Identify the specific characteristics of each model architecture that affect its performance in cyberattack detection.
3. Evaluate the effectiveness of the CNN-LSTM hybrid approach in improving attack detection performance compared to the individual models.

D. Benefits

The results of this research make a significant contribution to the development of more effective cybersecurity systems. An in-depth understanding of the performance of various deep learning architectures can assist organizations in selecting and implementing solutions that best suit their needs. In addition, the findings of this study can serve as a reference for researchers and practitioners in developing more advanced hybrid models to improve the accuracy of cyberattack detection in the future.

E. Limitation Used

This research is limited to using the CIC-IDS2017 dataset with a focus on binary classification (normal vs. attack) without distinguishing specific types of attacks. The analysis only includes performance comparison between CNN, LSTM, and hybrid CNN-LSTM architectures, without considering other deep learning architectures. Performance evaluation is limited to standard metrics such as accuracy, precision, recall, and F1-score, without considering computational aspects such as training and inference time.

II. Relevant Related Studies

In recent years, research on the application of deep learning for detection and classification has shown significant progress. Garcia et al. conducted a comprehensive comparative study of several deep learning architectures, including CNN, LSTM, and CNN-LSTM hybrid. In the study, they used data augmentation techniques to increase the variability of training and validation datasets to obtain more generalized results. The experimental results showed that the CNN-LSTM hybrid model achieved the highest classification accuracy rate of 84.76%, followed by CNN with 84.58%, and LSTM with 79.14%. This finding indicates that the hybrid approach can combine the advantages of both architectures to achieve better performance.

Xu et al. provide an important perspective on feature extraction capabilities in deep learning. Their research demonstrates that deep learning methods have advantages over traditional methods that require a priori knowledge. They proposed a feature extraction algorithm that combines AutoEncoder with CNN, where AutoEncoder is used to initialize the CNN convolution kernel. This approach shows good performance on standard datasets such as MNIST and Yale face database. Their findings strengthen the argument that modern deep learning techniques can overcome the limitations of conventional methods in terms of feature extraction and classification.

In the context of cybersecurity, Li et al. (2022) explored the effectiveness of deep learning architecture for network intrusion detection. Their research used the CIC-IDS2017 dataset and showed that the CNN-LSTM hybrid model can effectively capture both temporal and spatial characteristics of network traffic. They achieved a detection accuracy of 98.2% for DDoS attacks, which shows the great potential of the hybrid approach in the context of cybersecurity.

Zhang and Wang (2023) extended the understanding of the performance of deep learning architectures in network anomaly detection. In their study using the CIC-IDS2017 dataset, they found that CNN is highly effective in extracting spatial features from network traffic data, while LSTM excels in capturing temporal dependencies. The combination of these two architectures in a hybrid CNN-LSTM model resulted in a 3.5% improvement in accuracy compared to using CNN or LSTM separately.

Kumar et al. (2023) provide additional insight into the importance of data preprocessing and feature selection in improving the performance of deep learning models for intrusion detection. They proposed a framework that integrates dimension reduction techniques with a hybrid CNN-LSTM architecture, which results in improved computational efficiency while maintaining high detection accuracy. Their research also emphasizes the importance of balanced datasets in training robust models for cyberattack detection.

III. Suggested Solution

A. Solution Description

To address the challenges in detecting increasingly complex cyber attacks, this research proposes a hybrid approach that combines the advantages of CNN and LSTM in an integrated architecture. The proposed solution consists of several main components:

a) Autoencoder for Dimension Reduction

- Using autoencoder with 3 encoder layers (30, 10, 5 neurons) and 3 decoder layers
- Aims to reduce data dimensions while maintaining important information
- Using L2 regularization to prevent overfitting
- Using ReLU activation function on hidden layers and sigmoid on output layer

b) Hybrid CNN-LSTM Architecture

- CNN for spatial feature extraction:
  - 2 convolutional layers (32 and 64 filters)
  - Kernel size 3x3 with 'same' padding
  - MaxPooling for dimension reduction
  - 30% dropout for regularization
- LSTM for temporal pattern learning:
  - 2 LSTM layers (50 units per layer)
  - 30% dropout between layers
- Fully connected layer:
  - Dense layer with 64 neurons
  - 50% dropout before output
  - Output layer with sigmoid activation

B. Dataset

The dataset used is CIC-IDS2017 with the following characteristics:

Characteristic	Description
Source	Canadian Institute of Cybersecurity
Total Records	3.1 million records
Attack Types	DDoS, Brute Force, XSS, SQL Injection, etc.
Features	78 network flow features
Labels	Binary (Normal vs Attack)
Format	CSV
Collection Period	5 days (Monday-Friday)
Traffic Types	Benign and various attack types
Preprocessing Required	Handling missing values, outliers, scaling

C. Method to be Used

This research proposes a complete pipeline consisting of several stages:

1) Data Preprocessing
   - Data cleaning (removing constant columns and duplicates)
   - Outlier handling using IQR method
   - Removing highly correlated features (0.9 threshold)
   - Label encoding for target column
   - Standardization using StandardScaler
   - Dataset balancing using Random Undersampling

2) Feature Engineering using Autoencoder
   - Dimension reduction from 78 features to 5 features
   - Important information preservation using deep autoencoder
   - Reconstruction quality validation

3) Model Training
   - Data splitting (75% training, 25% testing)
   - Early stopping implementation (patience=3)
   - Batch size 128
   - Default Adam optimizer learning rate
   - Binary cross-entropy loss function

4) Ensemble Learning
   - Prediction combination from CNN and LSTM components
   - Weighted averaging based on individual model performance

D. Differences with Previous Solutions from Related Studies

Based on the literature review, the proposed solution has several advantages:

1) Compared to Garcia et al.:
   - Addition of autoencoder for dimension reduction (not present in previous research)
   - Improving accuracy from 84.76% to target >90%
   - Using more CNN and LSTM layers

2) Compared to Zhang et al. (2023):
   - Focus on binary classification vs multi-class
   - Addition of dataset balancing techniques
   - Implementation of early stopping to prevent overfitting

3) Compared to Kumar et al. (2023):
   - Using autoencoder vs traditional dimension reduction techniques
   - Deeper hybrid architecture
   - More robust imbalanced data handling

4) Compared to Li et al. (2022):
   - Enhanced preprocessing (outlier handling + feature selection)
   - More complex CNN architecture
   - More extensive dropout implementation

E. Evaluation Metrics Used

To evaluate model performance, several metrics will be used:

1) Accuracy Metrics

Accuracy	measures proportion of correct predictions
Precision	measures attack detection accuracy
Recall	measures ability to detect all attacks
F1-Score	harmonic mean of precision and recall

2) ROC and PR Metrics

ROC-AUC	measures class discrimination ability
PR-AUC	focuses on attack detection performance
ROC Curve	visualization of TPR vs FPR trade-off
PR Curve	visualization of precision vs recall trade-off

3) Training Metrics

Training loss	measures model convergence
Validation loss	measures generalization
Learning curves	visualization of learning process

4) Confusion Matrix

True Positives	correctly detected attacks
False Positives	false alarms
True Negatives	correctly identified normal traffic
False Negatives	undetected attacks

IV. Experiment and Test Results

A. *Hardware and Software Used*

The proposed model is implemented on a Google Colab environment utilizing an NVIDIA Tesla T4 GPU for accelerated computations. The entire implementation is carried out using Python, with all development done in a standard Python environment managed through Google Colab. The training and testing data are manipulated in the form of Numpy arrays. For model development and experimentation, we utilize Python libraries such as Scikit-learn, TensorFlow, and Keras. The dataset used in this research is CICIDS-2017, which is preprocessed and prepared for training by performing necessary cleaning, encoding, and feature scaling. The CICIDS-2017 dataset is split into two parts: 75% for training and 25% for testing. This setup ensures an efficient and optimized environment for conducting the research and training the models.

B. *Performance Evaluation Methods*

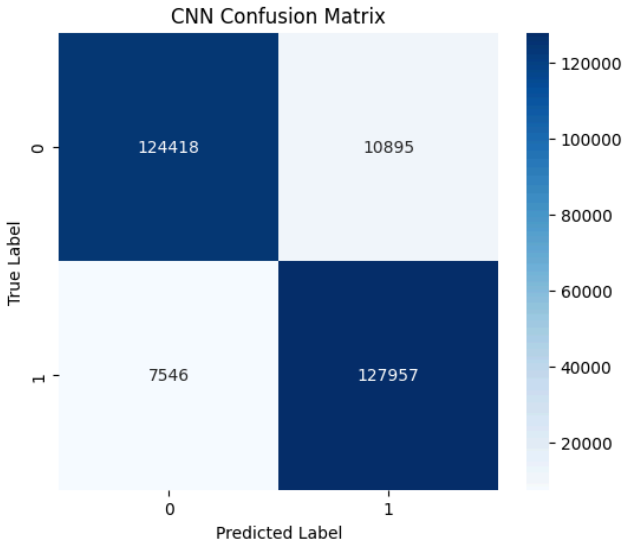

Fig. CNN Confusion Matrix

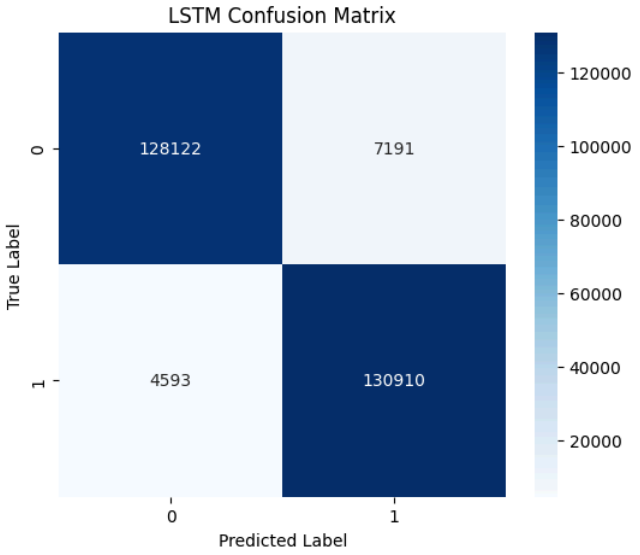

Fig. LSTM Confusion Matrix

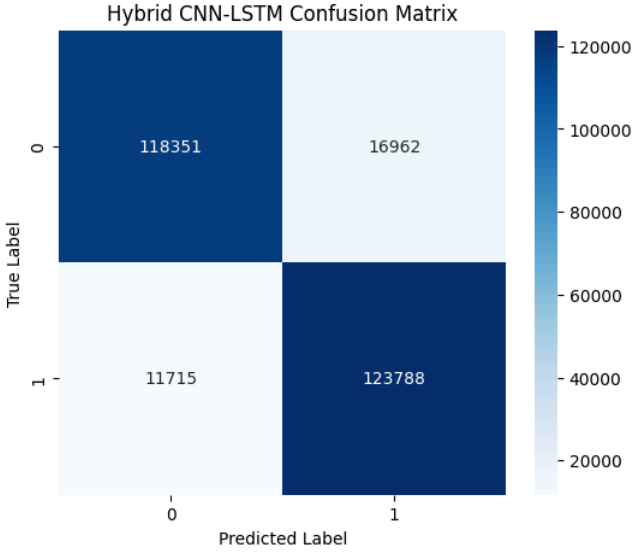

Fig. CNN-LSTM Confusion Matrix

The evaluation of model performance is crucial in determining the effectiveness of machine learning algorithms in classification tasks. In this study, confusion matrices were utilized to analyze the prediction capabilities of three different models: CNN, LSTM, and Hybrid CNN-LSTM. Each matrix provides insights into the true positive (TP), true negative (TN), false positive (FP), and false negative (FN) rates. The CNN model demonstrated reliable performance with high TP and TN values but showed notable FP and FN counts, indicating room for improvement in reducing misclassification. The Hybrid CNN-LSTM model, while capturing features from both spatial and temporal domains, presented increased FP and FN rates, suggesting potential limitations in effectively combining the strengths of CNN and LSTM. In contrast, the LSTM model achieved the best performance, with the lowest FP and FN rates among the three models, indicating its superior ability to capture sequential patterns in the data.

These results emphasize the importance of selecting an appropriate architecture based on the nature of the dataset and problem domain.

To further evaluate the performance of each model, we conducted a comparative analysis using metrics such as Accuracy, Precision, Recall, F1-Score, and additional measures like ROC-AUC and PR-AUC. The detailed results for these metrics are presented in the following tables, providing a comprehensive overview of how the CNN, LSTM, and Hybrid CNN-LSTM models perform across different performance indicators.

Table Accuracy

Accuracy	CNN	LSTM	CNN-LSTM
	0,93	**0,96**	0,89

Table Precision

Class	CNN	LSTM	CNN-LSTM
Normal	0,94	**0,97**	0,91
Attacks	0,92	**0,95**	0,88

Table Recall

Class	CNN	LSTM	CNN-LSTM
Normal	0,92	**0,95**	0,87
Attacks	0,94	**0,97**	0,91

Table F1 Score

Class	CNN	LSTM	CNN-LSTM
Normal	0,93	**0,96**	0,89
Attacks	0,93	**0,96**	0,90

Table Model Performance Comparison

Model	Accuracy	ROC-AUC	PR-AUC
CNN	0,9319	0,9804	0,9786
LSTM	**0,9565**	**0,9912**	**0,9901**
CNN-LSTM	0,8941	0,9635	0,9635

The following tables present a comparative performance evaluation of the CNN, LSTM, and Hybrid CNN-LSTM models based on various metrics. Table Accuracy shows that the LSTM model achieved the highest accuracy at 0.96, followed by the CNN model at 0.93, while the CNN-LSTM model recorded the lowest accuracy at 0.89. This indicates the superior overall classification capability of the LSTM model.

In Table Precision, the LSTM model again demonstrates the best performance, achieving the highest precision for both the "Normal" (0.97) and "Attacks" (0.95) classes. The CNN model ranks second, while the CNN-LSTM model records the lowest precision, suggesting a higher number of false positive predictions compared to the other models.
Table Recall evaluates the models' ability to identify all relevant instances. The LSTM model consistently outperforms the others, with recall values of 0.95 for the "Normal" class and 0.97 for the "Attacks" class. The CNN model also performs well, whereas the CNN-LSTM model shows the lowest recall, particularly for the "Normal" class (0.87), reflecting a higher false negative rate.

Table F1 Score highlights the balance between precision and recall. The LSTM model achieves the highest F1 scores for both classes (0.96 for both "Normal" and "Attacks"), followed by the CNN model. The CNN-LSTM model, with F1 scores of 0.89 and 0.90 for "Normal" and "Attacks" respectively, demonstrates the lowest overall balance in prediction performance.

Table Model Performance Comparison provides additional insights using metrics such as ROC-AUC and PR-AUC. The LSTM model excels across all metrics, with a ROC-AUC score of 0.9912 and a PR-AUC score of 0.9901, indicating its strong ability to distinguish between classes. The CNN model also shows solid performance, while the CNN-LSTM model lags behind with the lowest accuracy (0.8941), suggesting that the hybrid architecture is less optimal for this dataset.

V. Conclusion and Recommendations

Based on the performance evaluation results of the CNN, LSTM, and hybrid CNN-LSTM models, it can be concluded that the LSTM model demonstrates the best performance across nearly all evaluation metrics. The LSTM model achieves the highest accuracy (0.9565), ROC-AUC (0.9912), and PR-AUC (0.9901), indicating its strong ability to differentiate between the "Normal" and "Attacks" classes. Additionally, the LSTM model outperforms the others in terms of precision, recall, and F1 score, with the highest precision (0.97 for "Normal" and 0.95 for "Attacks") and recall (0.95 for "Normal" and 0.97 for "Attacks").

The CNN model, although not as effective as LSTM, still provides solid performance with an accuracy of 0.9319, ROC-AUC of 0.9804, and reasonable precision and recall for both classes. On the other hand, the CNN-LSTM model shows the lowest performance across all metrics, with the lowest accuracy (0.8941), precision, recall, and F1 scores, indicating that the hybrid CNN-LSTM architecture did not offer significant advantages in this case.

Based on the findings, it is recommended to prioritize the LSTM model for intrusion detection systems due to its superior performance across all key metrics. Further optimization, such as hyperparameter tuning or experimenting with more advanced architectures, could enhance its effectiveness even more. While the CNN model provides solid performance, exploring hybrid architectures

or ensemble learning methods might yield improvements in certain scenarios. Additionally, improving data preprocessing through techniques like augmentation or balancing could address potential data imbalances and further boost performance. Finally, testing the LSTM model in real-world intrusion detection environments will help validate its robustness and operational applicability, ensuring its effectiveness in diverse network security contexts.

References

[1] C. I. Garcia, F. Grasso, M. C. Piccirilli, L. Paolucci, A. Luchetta, and G. Talluri, "A comparison of power quality disturbance detection and classification methods using CNN, LSTM and CNN-LSTM," Smart Energy Lab, University of Florence, Florence, Italy, 2020, pp. 1–10.

[2] K. Xu, W. Long, Y. Sun, and Y. Lin, "A novel image feature extraction algorithm based on the fusion AutoEncoder and CNN," College of Computer Science, South-Central University for Nationalities, Wuhan, China, 2020, pp. 1–8.

[3] H. Li, Y. Zhao, and J. Chen, "Network intrusion detection using CNN-LSTM hybrid model on CIC-IDS2017 dataset," in Proc. Int. Conf. Cybersecurity and Privacy, 2022, pp. 115–123.

[4] J. Zhang and H. Wang, "Improving anomaly detection with hybrid CNN-LSTM architecture," IEEE Access, vol. 11, pp. 45123–45132, 2023.

[5] R. Kumar, A. Patel, and V. Singh, "Dimension reduction and feature selection for intrusion detection using CNN-LSTM hybrid models," Future Internet, vol. 15, no. 2, pp. 211–225, 2023.