Comparative Analysis of Deep Learning Models for Network Traffic Classification

Jinsong Liu[1, *]

[1]Huazhong University of Science and Technology, Wuhan, China
*Corresponding author: U202013903@hust.edu.cn

Abstract. In many sectors, network traffic categorization is a crucial duty, including network security, quality of service, and traffic engineering. Deep learning models have demonstrated potential in this area. This study used a comparative analysis method to evaluate and compare how well various models performed at categorizing network traffic. Convolutional neural networks (CNNs) excel at capturing local patterns and spatial dependencies, which are prevalent in network traffic data. On the other hand, recurrent neural networks (RNNs) are better suited for tasks that require modeling sequential dependencies over time, but they may struggle to capture the spatial characteristics of network traffic effectively. While deep learning models like CNNs hold promise, their effectiveness can vary depending on the specific characteristics of the data. It is crucial to consider the nature of the task, the available data, and the strengths and weaknesses of different models when making decisions. The results revealed the superiority of the CNN model over RNN models. The CNN achieved 77.41% accuracy, while the RNN with gate recurrent unit (RNN-GRU) model reached 45.43% accuracy and the RNN with long short-term memory (RNN-LSTM) model achieved 45.94% accuracy. In terms of precision, CNN achieved a score of 76.88%, while RNN-GRU scored 20.05% and RNN-LSTM scored 27.14%. Overall, this research underscores the importance of selecting appropriate models for categorizing network traffic.

Keywords: network traffic classification, deep learning model, neural network.

1 Introduction

Network traffic categorization is crucial for managing networks and security against the backdrop of soaring network traffic and an increase in cyber threats. It involves categorizing and analyzing data packets based on their protocols, applications, or other distinguishing characteristics [1,2]. Understanding network traffic enables effective resource management, enhanced security measures, optimized performance, and appropriate quality of service (QoS) policies. Intrusion detection systems (IDS) complement traffic classification techniques by monitoring network traffic and identifying unauthorized or malicious activities [3]. IDS systems use two different strategies. Signature-based systems compare network traffic against predefined patterns or signatures of known attacks [4], while anomaly-based systems establish a

© The Author(s) 2023
P. Kar et al. (eds.), *Proceedings of the 2023 International Conference on Image, Algorithms and Artificial Intelligence (ICIAAI 2023)*, Advances in Computer Science Research 108,
https://doi.org/10.2991/978-94-6463-300-9_11

baseline of normal behavior and detect deviations from it [3]. The classification of network traffic using a variety of deep learning techniques, especially CNNs, and RNNs, is the focus of this study. By evaluating the strengths and weaknesses of these models, network administrators, security professionals, and researchers can determine the most suitable approach. The study is divided into five sections: Section 2, relevant work discussion, Section 3, technique description, Section 4, experimental data and analysis, and Section 5, conclusion.

2 Related Work

Traditional and contemporary neural network models have both been used to classify network traffic. Traditional models include linear regression, naive Bayes, support vector machines, etc [5]. They often rely on handcrafted features extracted from network traffic data, such as statistical features, flow-based features, or payload-based features. While these models have achieved moderate success, they may struggle to capture complex patterns and adapt to evolving network traffic dynamics.

Deep learning models have emerged as a powerful approach for network traffic classification. CNNs have been widely adopted for their ability to automatically learn hierarchical representations of network traffic data [6, 7]. RNNs have been employed to capture temporal dependencies in sequential network data [8]. These models prove encouraging outcomes in the area of enhancing network traffic classification precision and robustness.

Other techniques and methods, such as ensemble learning, transfer learning, and attention mechanisms, have also been explored in the context of network traffic classification. Ensemble learning combines multiple models to improve classification performance [9]. Transfer learning improves the generalization ability of network traffic classifiers by using models that have already been trained on massive datasets [10]. Attention mechanisms focus on important features or parts of the input data, allowing the model to attend to relevant information for classification [11].

While prior research has proposed traffic classification algorithms or provided overviews of the field, the present study stands out by conducting a thorough evaluation and comparison of multiple machine learning models. Specifically, this work moves beyond devising new techniques or summarizing previous studies by conducting a comprehensive empirical analysis focused on accuracy, precision, and recall.

3 Methodology

To meet the goals mentioned in the preceding section, this study provides a methodology for network data categorization that combines the strengths of machine learning algorithms with domain expertise. This section provides a full discussion of the methodology's steps, highlighting the essential components and strategies used.

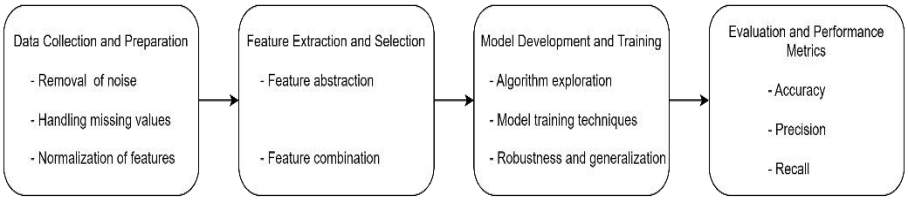

Fig. 1. Network Traffic Classification Process Flow (Photo/Picture credit: Original)

Figure 1 presents the network traffic classification pipeline. The first step involves the collection of a comprehensive dataset of network traffic, encompassing various protocols, applications, and scenarios. This dataset is carefully curated and preprocessed to ensure data quality and consistency. The preprocessing includes the removal of noise, handling missing values, and normalization of features to establish a standardized representation.

Next, feature extraction techniques are applied to transform the raw network traffic data into meaningful and informative features. These features capture important characteristics of the traffic, such as packet statistics, payload content, or flow behavior. Feature selection methods are then employed to identify the most relevant features that contribute to accurate classification.

The selected models are trained on the preprocessed dataset using appropriate training techniques, such as cross-validation or data augmentation, to ensure robustness and generalization. The trained models are evaluated using various performance metrics, including accuracy, precision, recall, and F1-score.

3.1 Model Selection

CNNs have proven to be quite effective in traffic through network classification. These specialized deep learning models are adept at analyzing the complex patterns and structures present in network traffic data. The convolutional layers within CNNs apply filters to the network traffic data, extracting local patterns and feature maps that are crucial for classification. The pooling layers further enhance the process by downsampling the feature maps while preserving relevant information. Fully connected layers enable the extraction of higher-level representations and accurate predictions based on the learned features. Figure 2 depicts the standard CNN architecture.

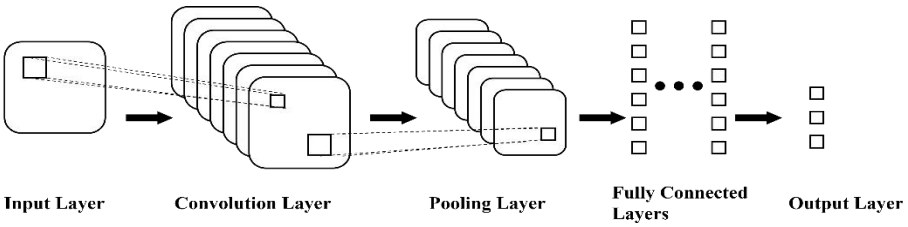

Fig. 2. Standard CNN Architecture (Photo/Picture credit: Original)

RNNs, unlike standard feedforward neural networks, can store information about past inputs via recurrent connections, making them extremely successful at capturing temporal dependencies.

An RNN takes input at each time step and modifies its hidden state. This hidden state serves as a memory that encapsulates the information from previous inputs. With their recurrent connections and hidden state, RNNs offer a powerful approach to modeling and understanding sequential patterns in network traffic data. The typical RNN architecture is illustrated in Figure 3.

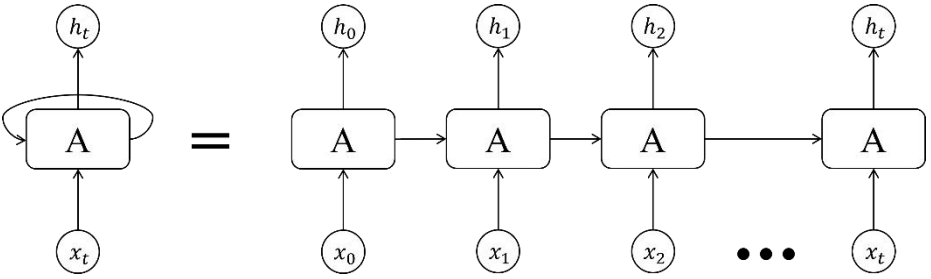

Fig. 3. Typical RNN Architecture (Photo/Picture credit: Original)

3.2 Confusion Matrix

When assessing how well a classification model will perform on untested data, it is critical to look at the classifier's accuracy. In the case of a binary classification scenario, the classification outcomes derived from the model can be succinctly summarized through a matrix, as depicted in Table 1.

Table 1. Model Evaluation Matrix

Matrix		Authentic	
		+	-
Anticipated	+	TP	FP
	-	FN	TN

From the confusion matrix, the Accuracy (represented by Ac) (All samples represented the total number of the four situations), Precision (represented by Pr), Recall (represented by Re), and F1-score can be calculated:

$$Ac = \frac{TP + TN}{All\ samples} \tag{1}$$

$$Pr = \frac{TP}{TP + FP} \tag{2}$$

$$Re = \frac{TP}{FN + TP} \tag{3}$$

$$F1 = \frac{2 * Re * Pr}{Re + Pr} \tag{4}$$

The usefulness and efficiency of the suggested methods for precisely classifying network traffic are evaluated through extensive testing.

4 Results and Analysis

4.1 Setup

Table 2 displays the hardware setup for this experiment.

Table 2. Hardware Configuration

Component	Specification
Operating System	Windows 10
Memory	16GB
Processor	Intel Core i7-9750H

The software configuration for this experiment utilized the following major Python libraries (version 3.10.2), as summarized in Table 3.

Table 3. Major Libraries

Library	Version
TensorFlow	2.12.0
NumPy	1.23.0
Keras	2.12.0
Pandas	1.5.3

4.2 Dataset

The Advanced Security Network Metrics & Tunneling Obfuscations (ASNM_TUN) dataset has been used in this paper [12]. The dataset consists of legitimate, direct attacks and obfuscated attacks. Table 4 displays the dataset's distribution.

Table 4. ASNM_TUN Dataset Distribution

Service Usage	Legitimate	Direct Attacks	Indirect Attacks	Total
HTTP Server	38	102	61	201
RPC Protocol	4	4	8	16
Proxy Server	95	4	10	109
Other traffic	40	20	8	68
Total	177	130	87	394

And there are two important types of labels in the dataset. Label_2's dichotomous value denotes if a real record is a network assault. As well as separating legitimate traffic and direct and disguised network attacks, the three-class label, represented by label_3, has three classes. Label_3 is utilized for the experiment in this paper.

4.3 Experiment and Comparison

To find the ideal subset of features, the baseline models used cross-validation and stepwise feature selection. The CNN model employed convolutional layers with ReLU activation functions and max pooling for dimensionality reduction. This was followed by dense layers incorporating dropout for regularization, leading to a softmax output layer for final classification. The RNN_GRU model consists of multiple GRU layers with Dropout regularization to prevent overfitting. The model takes in sequential input data and processes it to capture temporal dependencies and extract relevant patterns. Dense layers with the ReLU activation function are employed to enhance the model's ability to understand complex relationships within the data. The final Dense layer with softmax activation produces class probabilities for the traffic network multi-class classification task. In addition to these models, the RNN_LSTM model was employed, which is similar to the RNN_GRU model. Unlike the GRU, the LSTM model retains its memory cell state and modularizes its interactions via input, forget, and output gates. The effectiveness of these three baseline models on the dataset is shown in Figure 4.

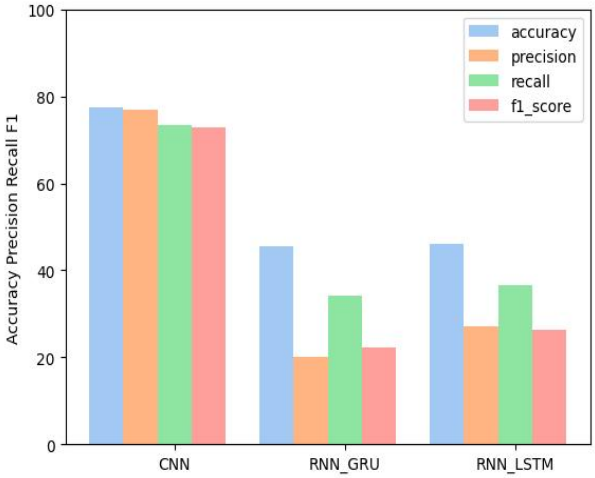

Fig. 4. Deep Learning Model Performance Comparison (Photo/Picture credit: Original)

The outcome demonstrates that on the prediction dataset, the CNN model beats the RNN_GRU and RNN_LSTM models. The CNN model achieves relatively high scores indicating its ability to make accurate predictions. The RNN_GRU and RNN_LSTM models perform similarly, but LSTM shows a slight advantage in performance.

Some basic machine learning algorithms were also applied in this experiment, including Decision Tree (DT) and Random Forest (RF). As shown in Figure 5, the outcomes showed that they exceeded the three baseline deep learning models with respect to accuracy.

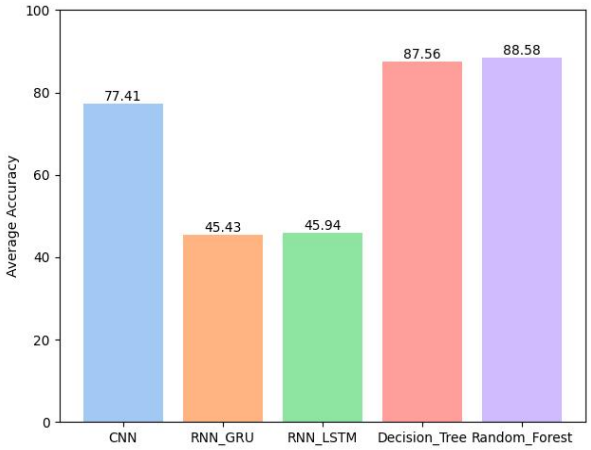

Fig. 5. Average Accuracy Comparison (Photo/Picture credit: Original)

Although CNN and RNN lag behind traditional DT and RF algorithms in terms of accuracy, other factors need to be considered. Deep learning models typically have higher model complexity and computational resource requirements. This means that deploying deep learning models requires more careful model design and resource allocation to ensure optimal performance.

5 Conclusion

In this study, a multi-class classification problem in a traffic network was analyzed using various deep learning models and fundamental machine learning methods. The models examined included CNN, RNN_GRU, RNN_LSTM, as well as DT and RF. The experimental results consistently favored the CNN model over RNN_GRU and RNN_LSTM models. Furthermore, the basic machine learning algorithms such as DT and RF outperformed the deep learning models in terms of accuracy. This suggests that for this specific task, simpler algorithms can yield satisfactory results, surpassing the performance of the baseline deep learning models. The strength of CNNs lies in their ability to effectively capture spatial relationships in data through convolutional filters. This capability likely allowed the CNN model to better learn distinctive features within the traffic network data. On the other hand, RNNs specialize in modeling temporal sequences, which did not provide a significant advantage in this particular task. Further analysis of the deep learning models could involve techniques like visualization of activations and occlusion experiments to gain insights into the learned features. Additionally, conducting hyperparameter tuning and testing different architectures can help explore the limits of deep learning for this problem. In future work, optimizing the baseline CNN and RNN models will be a key focus. Strategies such as parameter tuning, architectural refinement, and advanced regularization methods will be explored to enhance their performance in terms of accuracy and efficiency. Moreover, experimenting with ensemble techniques, such as combining

multiple models, may hold the potential for further improving performance in this context.

References

1. Y. D. Goli and R. Ambika. Network Traffic Classification Techniques-A Review. 2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS), pp. 219-222. Belgaum, India (2018).
2. A. Borkar, A. Donode and A. Kumari. A survey on Intrusion Detection System (IDS) and Internal Intrusion Detection and protection system (IIDPS), 2017 International Conference on Inventive Computing and Informatics (ICICI), pp. 949-953. Coimbatore, India (2017).
3. R. Samrin and D. Vasumathi. Review on anomaly based network intrusion detection system, 2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT), pp. 141-147. Mysuru, India (2017).
4. A. H. Almutairi and N. T. Abdelmajeed. Innovative signature based intrusion detection system: Parallel processing and minimized database. 2017 International Conference on the Frontiers and Advances in Data Science (FADS), pp. 114-119. Xi'an, China (2017).
5. M. Ramires, A. S. Gomes, S. Rito Lima and P. Carvalho. Network Traffic Classification using ML: A Comparative Analysis. 2022 17th Iberian Conference on Information Systems and Technologies (CISTI), pp. 1-6. Madrid, Spain (2022).
6. X. Wang, Y. Liu and W. Su. Real-Time Classification Method of Network Traffic Based on Parallelized CNN. 2019 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS), pp. 92-97. Shenyang, China (2019).
7. R. Xin, J. Zhang, and Y. Shao. Complex network classification with convolutional neural network. Tsinghua Science and technology 25.4, 447-457 (2020).
8. P. Kaushal. Deep RNN-based Traffic Analysis Scheme for Detecting Target Applications. 2021 IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI), pp. 1-5. Singapore (2021).
9. C. Q. Qiang, L. J. Ping, S. Gang and W. Z. Hui. Ensemble Method For Net Traffic Classification Based On Deep Learning. 2021 18th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), pp. 466-470. Chengdu, China (2021).
10. X. Hu, C. Gu, Y. Chen and F. Wei. tCLD-Net: A Transfer Learning Internet Encrypted Traffic Classification Scheme Based on Convolution Neural Network and Long Short-Term Memory Network. 2021 International Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI), pp. 1-5. Beijing, China (2021).
11. J. Zhang, J. Zhou and N. Zhou. Network Traffic Classification Method Based on Subspace Triple Attention Mechanism. 2022 3rd International Conference on Information Science, Parallel and Distributed Systems (ISPDS), pp. 312-316. Guangzhou, China (2022).
12. Homoliak I, Barabas M, Chmelar P, et al. ASNM: Advanced security network metrics for attack vector description[C]//Proceedings of the International Conference on Security and Management (SAM). pp. 350-358. Las Vegas, USA (2013).

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (http://creativecommons.org/licenses/by-nc/4.0/), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.