Deep RNN-based traffic analysis scheme for detecting Target Applications

Dr Maivizhi R Dibyajyoti Ganguly Tarun Negi

*Assistant Professor, School School of Computing School of Computing*

*of Computing,Srm Insititute Srm Institute of Science and Technology Srm Institute of Science and Technology*

*of Science and Technology* Tamil Nadu, India Tamil Nadu, India

*Tamil Nadu. India* dg7255@srmist.edu.in tn9791@srmist.edu.in

***Abstract*—**

**The growing complexity of network traffic, driven by the proliferation of diverse applications and evolving attack vectors, presents significant challenges for network security and performance optimization. A novel classification learning method is proposed, where input features and output labels are represented as two-dimensional images, with traffic packets and target applications, respectively. The proposed scheme leverages the power of deep learning and recurrent neural networks to analyze network packets in real-time, classifying them into predefined application categories. The scheme leverages a commercial deep long short-term memory system to provide fast and accurate traffic analysis. Simulation-based experiments demonstrate the scheme's high accuracy of 99.82% with low complexity. This research has the potential to enhance tools like Wireshark used for packet analysis and monitoring in network environments**.

***Keywords— Website traffic, Time series, Forecasting, Historical data, CNN, LSTM***

I. INTRODUCTION

In a time when the internet is a vital part of our everyday existence, network management, cybersecurity, and application optimization all depend on being able to comprehend and track network traffic. In order to guarantee that networks operate effectively, identify security risks, and improve service quality, it is now crucial to identify and analyze network traffic due to the quick growth of internet-based services and applications.

Traditionally, network traffic analysis has involved analyzing data packets to determine the type, destination, and source of traffic. Although helpful, this method has limitations when it comes to identifying the types of applications and services that are driving the traffic. The increase in encrypted and obfuscated traffic makes traffic analysis an even more difficult task. In addition, new services and applications are constantly being added to contemporary networks, which are dynamic and ever-evolving. It becomes extremely difficult to identify and categorize these applications, especially given their ongoing updates and shifting communication patterns.

An ever-growing dependence on digital applications and data-driven decision-making have made precise and efficient network traffic monitoring crucial. Numerous applications, such as network management, security, and quality of service optimization, depend on an understanding of the traffic composition of the network.

This necessitates the use of cutting-edge methods, and the Deep Recurrent Neural Network (RNN)-based traffic analysis technique is one such novel way. Recognized for their capacity to represent sequential data and grasp temporal connections, RNNs have become more important in the artificial intelligence community. We are going to take a tour through the intriguing realm of deep RNN-based traffic analysis, which enables us to better understand network behavior, pinpoint specific applications, and improve our network security and management tactics. This method offers a significant advance in the field of traffic analysis since it provides precise identification as well as adaptability to the constantly changing environment of network applications.

The objective of the proposed methodology is to develop and deploy a deep neural network architecture that can identify target applications by analyzing network traffic sequences. A wide range of network traffic, including both well-known and recently developed applications, will be used to train the model.By assessing the Deep RNN-based approach's accuracy, precision, recall, and F1-score in identifying target applications across a range of network scenarios, its efficacy will be thoroughly assessed. There will be comparison studies done with conventional traffic analysis methods.

The proposed methodology will investigate scenarios of real-world applications, such as network management, quality of service optimization, and security. Network administrators can prioritize network resources, better understand and regulate application use, and spot possible security risks by identifying target application

The methodology focuses on applying deep learning methods to analyze network traffic data in order to detect applications. The necessity for effective traffic analysis and management has grown due to the growth of various traffic data and the rising demand for network services. This paper presents a novel approach to classification learning that uses deep learning—more particularly, Long Short-Term Memory (LSTM) networks—to discover target applications and categorize network traffic with high accuracy.

II. RELATED WORK:

[1] The authors of the paper "Traffic Prediction-Enabled Energy-Efficient Dynamic Computing Resource Allocation in CRAN Based on Deep Learning" are Yongqin Fu and Xianbin Wang. The paper was published in 2022. The proposed methodology offers several advantages. Firstly, it utilizes a novel two-dimensional CNN LSTM model with temporal aggregation for wireless traffic prediction.

[2] The paper, authored by Prajwal Kaushal, was published in 2021. The paper proposes a deep learning-based traffic analysis scheme using Long Short Term Memory (LSTM) networks. It preprocesses network traffic data, converting it into a format suitable for LSTM learning, and achieves high accuracy in classifying traffic into target applications..

[3] The paper “Design and Development of RNN Anomaly Detection Model for IoT Networks" by Imtiaz Ullah and Qusay H. Mahmoud was received on March 30, 2022, accepted on April 12, 2022, and published on May 18, 2022, with a current version date of June 17, 2022. The paper presents several methodologies for anomaly detection in IoT networks using recurrent neural networks (RNNs), specifically Long Short Term Memory (LSTM).

[4] The paper "Machine Learning Based Web-Traffic Analysis for Detection of Fraudulent Resource Consumption Attack in Cloud" by Rishabh Rustogi, Abhishek Agarwal, Ayush Prasad, and Samant Saurabh was published in 2019. The authors proposed a novel approach for detecting Fraudulent Resource Consumption (FRC) attacks by dividing web-pages into quantiles.

[5] The paper “A Hybrid Approach for Web Traffic Prediction Using Deep Learning Algorithms" by Anupama Prasanth, Priyanka Surendran, Densy John, and Bindhya Thomas was published in 2022. The authors propose a hybrid model for web traffic prediction using Long Short Term Memory (LSTM) and Radial Basis Functional Networks (RBFNs) combined through ensemble learning.

[6] The authors of the article titled "DA-Transfer: A Transfer Method for Malicious Network Traffic Classification with Small Sample Problem" are Ruonan Wang, Jinlong Fei, Min Zhao, RongKai Zhang, MaoHua Guo, Xue Li, and Zan Qi. The year of publication is 2022. The article proposes a method called DA-Transfer for small-sample malicious network traffic classification based on deep transfer learning.

III. DATASET DESCRIPTION

Source of the Dataset:

The dataset used in the code is obtained from Kaggle, a popular platform for data science and machine learning datasets.

Data Structure:

The dataset contains two columns: "Hour Index" and "Sessions."

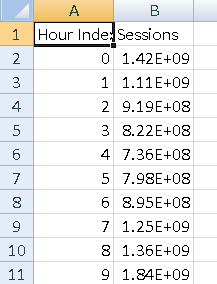
"Hour Index" represents the time or hour, and it appears to be a sequential index or identifier for each hour. For example, "This is the first hour," "This is the second hour," and so on. "Sessions" is the second column in the dataset and represents the volume of web traffic at an hourly level. It is a numeric value, likely representing the number of web sessions or users' interactions with a website during each hour.

Dataset Size:

The dataset consists of 4,896 rows, which means it contains 4,896 data points or records.

Time Series Data:

The dataset is described as a "six-month series," indicating that it covers a period of six months. This suggests that the dataset contains web traffic data collected over a continuous six-month timeframe.

****

**Fig. 4. Dataset with two features**

IV. PROPOSED WORK:

**Require**: webtraffic.csv file containing session data

**Ensure**: Comparison plot of true values and predicted values using LSTM, Simple RNN, GRU, and Conv1D models

**for** model\_type in model\_types:

# **Build model**

model = build\_model(model\_type)

# **Train the model**

train\_model(model, X\_train, y\_train, X\_test, y\_test)

**# Load the best weights for the model**

model.load\_weights('best\_model.hdf5')

**# Evaluate the model on the training data**

mse = evaluate\_model(model, X\_train, y\_train)

print(f"{model\_type} - Mean Square Error: {mse}")

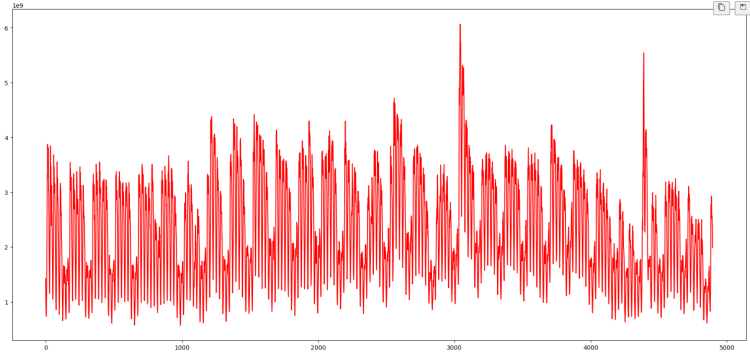
**# Forecast future values using the trained model**

y\_pred = forecast(model, X\_test, no\_of\_pred, ind)

**# Plot true vs predicted value**s

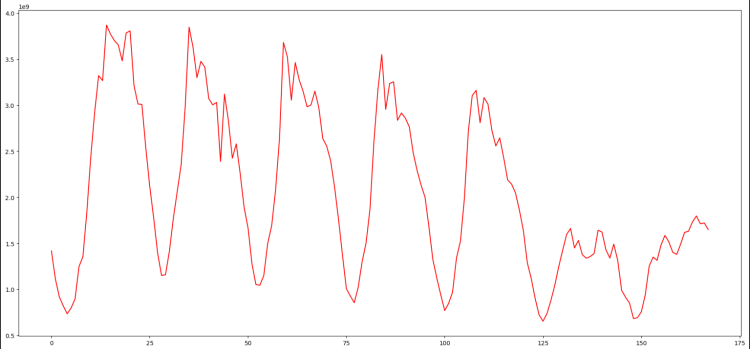
plot\_comparison(y\_test[ind:ind+no\_of\_pred],y\_pred, model\_type=model\_type)

In “Fig.1” we have plotted the complete time series while exploring the data. At each point of this curve, which represents an early session count, you can see that there are some reoccurring patterns throughout the time series. The amount of traffic drops off after roughly equal periods of time. In addition, there are a few traffic peaks in this plot.



**Fig. 1. Plot of sessions and hours on the whole dataset**

Let's take a closer look at this information. We might only use a section of the time series rather than the entire one. In “Fig.2” we have just presented the data from the first week, so the repeating trend can now be seen more clearly. Once every 24 hours, these drops in web traffic are possible. It follows that there are two times during the day when we encounter both high traffic volumes (such as occasionally) and low traffic volumes.

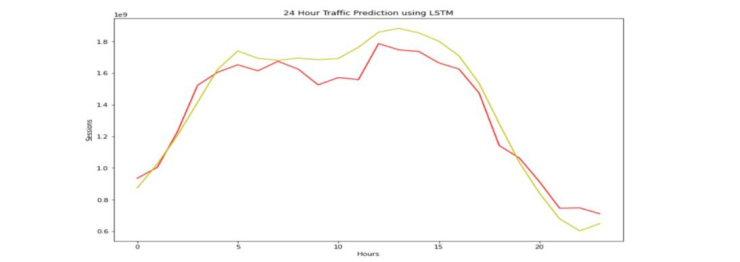
****

**Fig. 2. Plot of sessions and hours of a week’s data**

V. EXPERIMENTAL RESULTS

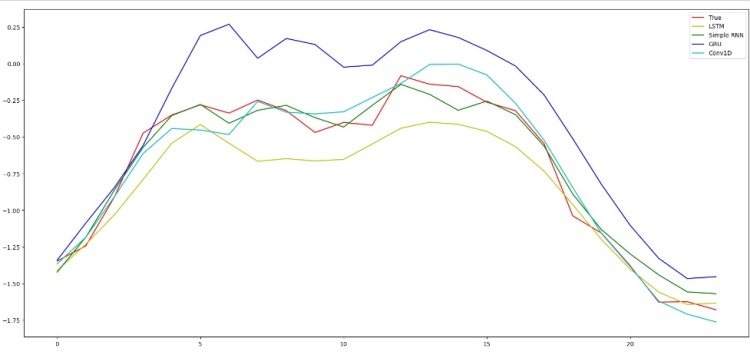
*Result Analysis Results of the LSTM model*:

The mean squared error for the validation data is only 0.014, for the LSTM model. We use this mean squared error to measure the model's performance.

****

**Fig. 6. Traffic Prediction Using LSTM for 24 hours**

“Fig.6” shows the prediction of 24 hours of data using LSTM, with the Hour Index as the X-axis and Sessions as the Y-axis. The yellow curve displays the predicted values, which are very close to the red curve's display of the actual value. LSTM's mean squared error is 0.14



**Fig. 7. Comparison of LSTM with other models**

LSTM can capture both short-term and long-term dependencies in time series data effectively. It is less prone to vanishing gradient problems compared to Simple RNN. Suitable for tasks with complex temporal patterns and long-range dependencies. Simple RNNs are limited by their ability to capture only short-term dependencies.They are more likely to suffer from vanishing gradient problems when dealing with long sequences.Suitable for tasks where short-term patterns dominate, and computational resources are limited.

LSTM is likely to outperform Simple RNN when the time series data has significant long-term dependencies or complex temporal patterns. It can capture both short and long-term trends effectively.

LSTM is suitable for tasks with complex temporal dependencies and long-range dependencies. It can capture both short-term and long-term patterns. However, it may require more parameters and careful tuning.

GRU is a balance between complexity and efficiency.

It performs well on tasks with moderate-length dependencies.

It is computationally more efficient and often easier to train than LSTM.

The choice between LSTM and GRU depends on the specific characteristics of the data and available computational resources. LSTM is more powerful for capturing long-range dependencies, but GRU may be preferred when a balance between accuracy and efficiency is required.

LSTM models are capable of capturing both local and global patterns in time series data. They are effective at handling sequences with varying lengths and complex temporal relationships. Conv1D models are focused on capturing local patterns within a sequence. They excel at recognizing short-term, local dependencies in data.

LSTM can outperform Conv1D in scenarios where the time series data contains significant long-range dependencies and complex temporal relationships. Conv1D may be more suitable for tasks where the emphasis is on local, short-term patterns.

In summary, LSTM is a powerful choice for time series forecasting tasks, especially when the data exhibits both short-term and long-term dependencies. It excels at capturing complex temporal patterns. However, the choice of model should always be based on the specific characteristics of the data and the requirements of the task, and it may vary depending on the dataset and the computational resources available.

VI. CONCLUSION

The research demonstrates the effectiveness of deep learning, particularly LSTM, in traffic analysis and application detection. DR-TAS achieves high accuracy in classifying network traffic, making it a promising approach for real-world deployment. Future research may focus on implementing this approach in actual networks and addressing challenges related to classifying new, previously unseen packets

Our study's major objective is to develop a forecasting algorithm that can accurately predict how much traffic Wikipedia pages will receive in the future.

LSTM Recurrent Neural Networks perform more effectively and precisely for time series of online traffic. We've trained the model using this data using metrics like the hours and the number of views, or sessions, to anticipate future online traffic for pages over the period of a year. It is possible to predict how many visitors the website will get in the future.

As more user data is fed into the system, it will continue to get better. To improve business analysis and load control of online traffic, all websites can make use of our service. LSTM RNN increases the effectiveness of our system. One of the least studied areas of forecasting is time series, and a number of models are tested to increase forecast accuracy. The proposal's main goal is to forecast future web traffic so that better congestion control can be achieved through decision-making. When predicting future values, past values are taken into account.

Aside from that, we'll look into multivariate time series and make recommendations for how to make instantaneous decision-making easier. To get the best prediction results in the future, prepare to investigate Seq to Seq LSTM, RNN topologies, and Google page trends.

VII. REFERENCES

[1] Tondale, S., Yadav, S., & Ahir, S. (2020). Web traffic time series forecasting using ARIMA and LSTM RNN [Review of Web traffic time series forecasting using ARIMA and LSTM RNN].

[2] Golait, A., Pardeshi, S., & Javeri, R. (n.d.). Web Traffic Time Series Forecasting Using ARIMA Model [Review of Web Traffic Time Series Forecasting Using ARIMA Model].

[3] Zheng, J., & Huang, M. (n.d.). Traffic Flow Forecast Through Time Series Analysis Based on Deep Learning [Review of Traffic Flow Forecast Through Time Series Analysis Based on Deep Learning].

[4] Zhuo, Q., Yan, H., & Qi, Y. (2019). A network traffic prediction method based on LSTM. [Review of A network traffic prediction method based on LSTM.].

[5] Liu, H., Xu, H., & Zhang, X. (2022). Network Traffic Prediction Based on LSTM and Transfer Learning. [Review of Network Traffic Prediction Based on LSTM and Transfer Learning.].

[6] Sharma, S., Parashar, L., & Singh, U. (2020). Network traffic prediction using long short-term memory. [Review of Network traffic prediction using long short-term memory.].

[7] Yang, F. (2018). Research on network traffic prediction based on long short-term memory neural network. [Review of Research on network traffic prediction based on long short-term memory neural network.].

[8] Petluri , N., & Al-Masri, E. (2018). Wikipedia Page Traffic Prediction [Review of Wikipedia Page Traffic Prediction].

[9] Aggelos, & Prasanna., V. (2019). An LSTM framework for modeling network traffic [Review of An LSTM framework for modeling network traffic].

[10] Zheng, J., & Huang, M. (2020). Traffic Flow Forecasting Using Deep Learning and Time Series Analysis [Review of Traffic Flow Forecasting Using Deep Learning and Time Series Analysis].

[11] Shuvo, M. A. R., Zubair, M., Purnota, A. T., & Hossain, S. (2021). Traffic Forecasting Using Time-Series Analysis [Review of Traffic Forecasting Using Time-Series Analysis].

[12] Wunnava, & Praveen, V. (2020). Exploration of Wikipedia traffic data to analyze the relationship between multiple pages [Review of Exploration of Wikipedia traffic data to analyze the relationship between multiple pages].

[13] Miyaguchi, A., Chakrabarti, S., & Garcia, N. (2019). Forecasting Wikipedia Page Views with Graph Embeddings [Review of Forecasting Wikipedia Page Views with Graph Embeddings].

[14] Chakraborty, A., J.S. Banerjee, and A. Chattopadhyay*.*Non-uniform quantized data fusion rule alleviating control channel overhead for cooperative spectrum sensing in cognitive radio networks*in*2017 IEEE 7th International Advance Computing Conference (IACC)2017. IEEE.

[15] Chattopadhyay, A. (n.d.). Non-uniform quantized data fusion rule for data rate saving and reducing control channel overhead for cooperative spectrum sensing in cognitive radio networks. [Review of Non-uniform quantized data fusion rule for data rate saving and reducing control channel overhead for cooperative spectrum sensing in cognitive radio networks.].

[16] Rueda, *A.*A survey of traffic characterization techniques in telecommunication networks*in*Proceedings of 1996 Canadian Conference on Electrical and Computer Engineering1996. IEEE

[17] Axelsson, S. (2000). Intrusion detection systems: A survey and taxonomy [Review of Intrusion detection systems: A survey and taxonomy].

[18] Mukkamala, S., Janoski, G., & Sung, A. (2002). Intrusion detection: support vector machines and neural networks [Review of Intrusion detection: support vector machines and neural networks].

[19] Shafiq, M. (2016). Network traffic classification techniques and comparative analysis using machine learning algorithms [Review of Network traffic classification techniques and comparative analysis using machine learning algorithms].

[20] Zamani, M. (2009). A DDoS-aware IDS model based on danger theory and mobile agents [Review of A DDoS-aware IDS model based on danger theory and mobile agents].