Machine Learning for Traffic Prediction

Niki Esmaeili
School of Engineering
University of Guelph
Ontario, Canada
esmaeiln@uoguelph.ca

Janice Austin
School of Engineering
University of Guelph
Ontario, Canada
jausti02@uoguelph.ca

Declarations regarding the use of Generative AI- *ChatGPT and Google Gemini were used to aid in fixing a few bugs in the program and to help rewrite the future scope and a few other paragraphs in this report.*

***Abstract*— The increasing demand for network capacity, driven by the pervasive use of digital devices and automated systems, necessitates the development of novel strategies for efficient resource optimisation and management. The innovative framework known as Software-Defined Networking (SDN) incorporates machine learning (ML) to improve the intelligence and flexibility of networks. Large amounts of network telemetry data may be collected and analysed because to SDN's centralised management and programmability, which are made possible by the separation of the control plane from the data plane. In order to increase network efficiency and security, this study looks at how machine learning (ML) techniques can be used to decipher network traffic patterns and automate configuration procedures inside an SDN framework.**

I. INTRODUCTION

The effectiveness of machine learning (ML) approaches in categorising network data has been demonstrated by recent studies. However, challenges like guaranteeing data quality, improving model generalizability, and addressing computing efficiency arise when putting these models into practice.

Conventional methods, like as payload-based and port-based classification, are limited by their dependence on static packet payloads or port numbers, which leaves them open to assault. On the other hand, ML-based techniques offer a more adaptable and durable substitute. These techniques use advanced machine learning algorithms to identify complex patterns in network traffic, allowing for precise categorisation even when working with encrypted data or new attack techniques.

Because they usually rely on manually created characteristics, classical machine learning techniques like K-Nearest Neighbours (KNN) are not as flexible in different network contexts. On the other hand, by automatically extracting significant characteristics straight from raw network data, deep learning techniques like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) provide a more adaptable solution. This feature enables deep learning models to function well in intricate and changing network environments, overcoming the drawbacks of conventional techniques.

TABLE I. OVERVIEW OF TRAFFIC CLASSIFICATION METHODS

Methods	**Description**	**Benefit**	**Limitation**
Port-Based	Categorizes packets based on their port numbers.	Efficient, resource-friendly, and highly accurate.	Does not utilize application layer payloads and is ineffective for hidden ports.
Payload-Based	uses Deep Packet Inspection (DPI) to examine packet data content.	Capable of managing services that use dynamic ports.	Requires high computational resources, is unsuitable for encrypted data, and raises privacy concerns.
ML-Based	Utilizes machine learning models to extract features from packet payloads or statistical traffic characteristics.	Effectively manages dynamic ports and encrypted data, offering a faster alternative to deep packet inspection classification.	Takes longer to classify compared to port-based methods.

This research investigates the application of machine learning, specifically the K-Nearest Neighbor (KNN) algorithm, for network traffic classification. By evaluating the performance of ML algorithms using various metrics, the

research seeks to provide valuable insights to communication organizations for network optimization and security.

II. METHODOLOGY

A. DATA COLLECTION

The dataset, which is a CSV file with a 500 KB size limit and was obtained via Kaggle, records historical network traffic statistics. It summarizes the actions of ten local workstations over a three-month period and has about 21,000 entries.
DATASET OVERVIEW:

- ***Local IP (l_ipn)****: Represented as integers ranging from 0 to 9, identifying local workstations.*
- *Flows (f): The number of network connections recorded for a given day.*
- *Date: Entries span from 2006-07-01 to 2006-09-30, formatted as yyyy-mm-dd.*
- *Remote ASN (r_asn): An integer indicating the Autonomous System Number (ASN) of the remote ISP.*

B. DATA EXPLORATION AND ANALYSIS

By using graphical representations to show patterns and trends, network traffic data visualization aids in the discovery of new information.

Scatter Plot
The network traffic dataset's variables "Packets" and "Flow Duration" are used to create a scatter graphic. For every data point, this visualization provides a better understanding of the association between the number of packets and the flow length.

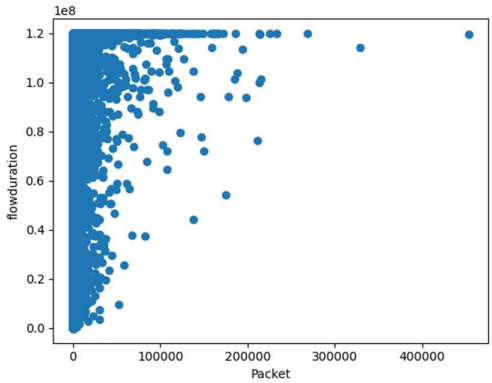

Fig 1: Scatter plot of the dataset's data frame

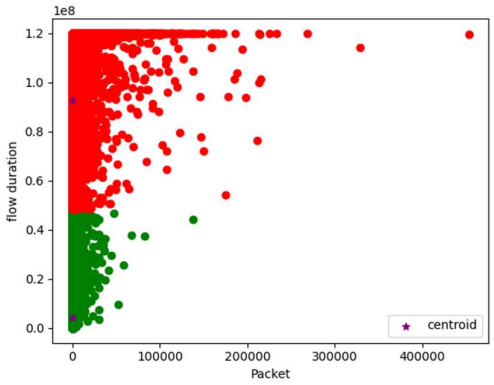

Fig 2 : Scatter plot of the dataset's data frame after clustering

Figure 1 shows the dataset's scatter plot prior to grouping. The KMeans algorithm, an unsupervised machine learning technique, is then used for clustering, dividing the dataset into K unique, non-overlapping groups. Two clusters are created from the data once KMeans clustering is applied:
Cluster 0: Represented as dataframe1, marked in green.
Cluster 1: Represented as dataframe2, marked in red.

1. Line Plot

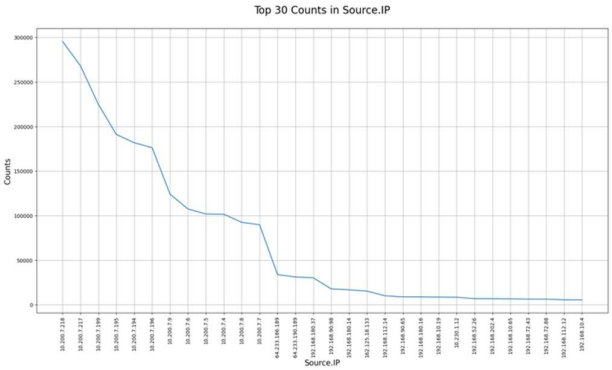

Fig3: Destination IP

The distribution of the most common "Destination IP" values in the dataset is shown in this image, which shows a line plot of the top 30 counts in the "Destination IP" column.

2. Bar graph

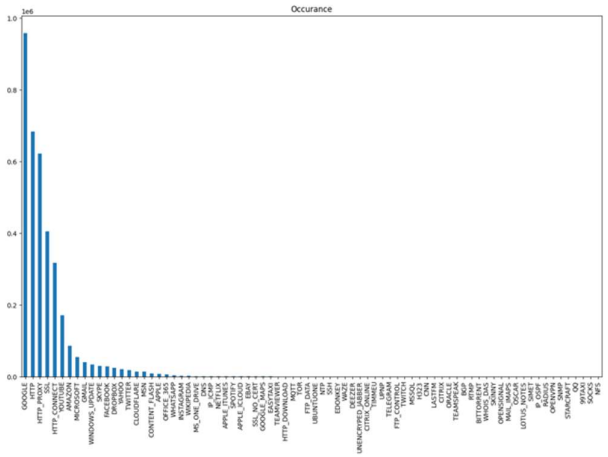

Fig 4: Count of records for each application

A bar plot displaying the quantity of records for every single application according to their occurrence counts in the dataset is depicted in the figure. It calculates the frequency of every distinct protocol name and uses a bar chart to display these statistics.

3. Heatmap

Heatmaps graphically represent data using color to indicate raster values, making them particularly useful for visualizing patterns and relationships. In the context of confusion matrices, heatmaps offer an intuitive way to interpret and analyze the performance of classification models.

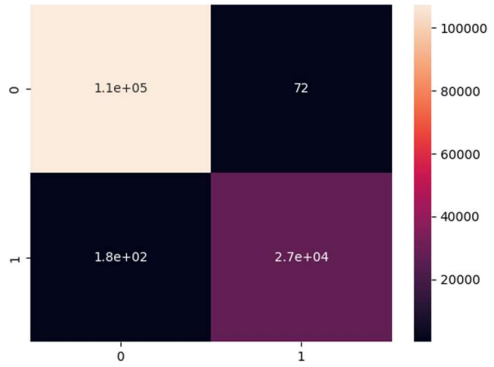

Fig 5 : Heatmap of confusion matrix

- Comprehensive data on true positives, true negatives, false positives, and false negatives for various network traffic categories can be found in the confusion matrix (CM).
- Cell annotations display exact numerical values for classification outcomes, making it easier to understand the distribution of accurate and inaccurate predictions.
- A better understanding of the model's performance is offered by the heatmap depiction of the normalized confusion matrix, which shows normalized values for true positives, false positives, true negatives, and false negatives across different network traffic classes.

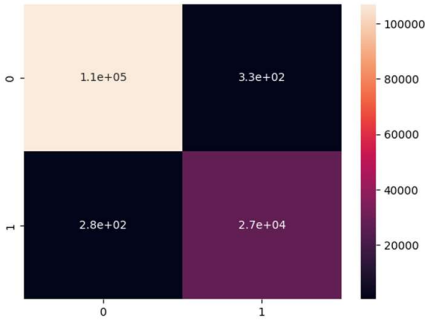

Fig 6: Heatmap of confusion matrix after GRU

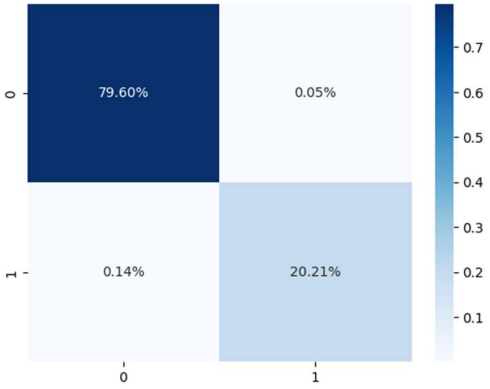

Fig 7 : normalized confusion matrix

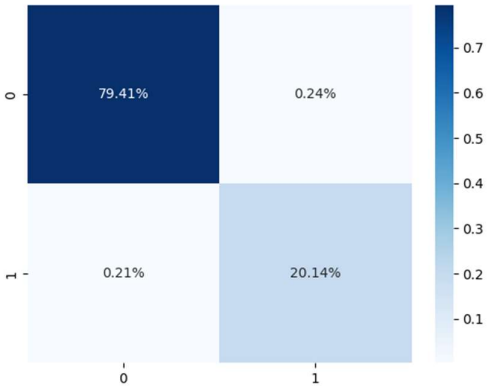

Fig 8 : normalized confusion matrix after GRU

The classification results of the GRU model are displayed in these heatmap visualizations (Figures X and Y), which demonstrate the distribution of false positives, true positives, true negatives, and false negatives across various network traffic categories.

*C.*DATA PREPROCESSING

Preprocessing procedures were used to improve model performance and preserve data quality. The mean of the corresponding columns was used to fill in the missing data. Label encoding was used to convert categorical information into numerical representations. In order to lessen the impact of different magnitudes, numerical features were standardised to a consistent scale. Finally, to facilitate efficient model training and assessment, the dataset was separated into training and testing sets. Outliers were identified and treated to minimize their influence on the model's performance.

D. APPLIED MACHINE LEARNING ALGORITHM

K Nearest Neighbors (KNN):

Because it is easy to use and can represent non-linear correlations, K-Nearest Neighbours (KNN) is a good technique for predicting network traffic. KNN and other classification models are frequently evaluated using measures including accuracy, precision, recall, and F1-score.

Key Performance Metrics for Binary Classification:

- Accuracy: Measures the ratio of correct predictions to the total number of predictions.
- Precision: Represents the proportion of correctly classified positive instances out of all predicted positives.
- Recall: Reflects the proportion of actual positive instances correctly identified.

We assessed precision, sensitivity, and accuracy using a Gated Recurrent Unit (GRU) model. The GRU architecture was fed sequences created from the preprocessed network traffic data. Using this sequential data, the model was trained to find trends and produce predictions.

IV. RESULTS

	Accuracy	**Recall**	**Precision**
KNN	0.97906	0.993609	0.98005
GRU	0.99658	0.998184	0.99753

The findings show that in terms of accuracy, sensitivity/recall, and precision, the Gated Recurrent Unit (GRU) performs better than the K-Nearest Neighbour (KNN) approach.

V. FUTURE SCOPE

Intelligence (AI) have enormous potential to advance network traffic analysis. With the help of these technologies, networks may become more effective, safe, and flexible in response to changing needs and threats. Sophisticated threat detection, intelligent network optimisation, smooth 5G and IoT integration, privacy-focused analytics, and explainable AI solutions are some of the main application cases. Nonetheless, it is still crucial to address ethical issues and provide solid frameworks for the appropriate application of AI in network security.

VI CONCLUSION

The classification of network traffic using Machine Learning (ML) techniques was examined in this study. We specifically focused on network traffic analysis utilising the K-Nearest Neighbours method, which offered valuable insights into identifying and deciphering traffic patterns. By categorising network traffic according to feature space proximity, our results showed that KNN was useful in particular situations despite its ease of use and intuitiveness. Future work could focus on exploring a wider range of algorithms and more sophisticated techniques to deepen understanding and achieve greater accuracy in network traffic analysis.

REFERENCES

[1] R. L. Z. Fan, Investigation of machine learning based network traffic classification,, Bologna: IEEE, 2017.

[2] M. Shaygan, C. Meese, W. Li, X. Zhao and M. Nejad, Traffic prediction using artificial intelligence: Review of recent advances and emerging opportunities, BlueHalo , Rockville , MD 20855: Elsevier, 2022.

[3] P. sanjeev, A. Gupta, N. S. Kumari, M. singh and V. Sharma, Network Traffic Classification Analysis Using Machine Learning Algorithms, Greater Noida: IEEE, 2018.

[4] S. m. Rachmawati, D.-s. Kim and J.-M. Lee, Machine Learning Algorithm in Network Traffic Classification, Jeju Island: IEEE, 2021.

[5] T. T. Nguyen, A survey of techniques for internet traffic classification using machine learning, IEEE, 2008.

[6] J. W. Azzedine Boukerche, Machine Learning-based traffic prediction models for Intelligent Transportation Systems, Computer Networks,, sciencedirect, 2020.

[7] Z. M. Fadlullah, State-of-the-Art Deep Learning: Evolving Machine Intelligence Toward Tomorrow's Intelligent Network Traffic Control Systems, IEEE, 2017.

[8] A. N. Hanan Almukhalfi, Traffic management approaches using machine learning and deep learning techniques: A survey, sciencedirect, 2024.

[9] M. H. S. M. M. Woldeyohannis, Customer Churn Prediction Using Machine Learning: Commercial Bank of Ethiopia, Bahir Dar: IEEE, 2022.