# Optimizing application and algorithm complexity of machine learning methods in traffic classification

**Qiyuan Tan**

Camford Royal School, Beijing, 100144, China

tanqiyuan@STUDENT.WUST.EDU.PL

**Abstract.** As the Internet continues to evolve, it has become crucial for Internet Service Providers to analyze and classify their network flows. This enables them to identify suspicious activities and offer personalized services. Machine learning has been extensively deployed in network traffic classification, presenting a promising but challenging avenue. One of the primary challenges in applying machine learning to network traffic classification is reducing the computational resources used in training and implementing the model. By devising lightweight algorithms, traffic flow can be classified using fewer computational resources, effectively curtailing the escalating costs associated with the growing volume and transmission rate of traffic. In this study, we compare the performance of three classic machine learning algorithms - logistic regression, support vector machine, and shallow feedforward neural network - by employing them to classify mobile countries of origin, aiming to use as few features as possible. Remarkably, by utilizing only four features from the dataset, these three algorithms achieved an accuracy rate of 89%. This underscores the potential for computational and cost efficiency in network traffic classification with optimized machine learning methods.

**Keywords:** machine learning, network traffic classification, network traffic analysis, supervised learning.

## 1. Introduction

Classifying and analyzing network traffic flows have become essential tasks for Internet Service Providers (ISPs) in today's interconnected world. By accurately classifying and scrutinizing network traffic, ISPs are able to detect potential threats and subdue malicious activities. Furthermore, this allows for the provision of personalized services based on specific network requests.

Machine learning techniques play a pivotal role in network traffic classification. Utilizing flow-level measurements, machine learning algorithms can discern the statistical characteristics of specific flows, thereby classifying network traffic effectively. In the wake of several decades of continuous development, supervised machine learning algorithms are now highly advanced, even though more innovative solutions continue to emerge. The main challenges confronting machine learning techniques stem from the growing prevalence of traffic encryption and protocol encapsulation within the network, as well as the steadily increasing volume of traffic and transmission rates [1]. Consequently, contemporary researchers are focused on developing more lightweight algorithms to conserve computational resources and facilitate real-time analysis for higher transmission speeds.

This paper contrasts the performance of three classic machine learning algorithms - logistic regression, support vector machine, and shallow feedforward neural network - by employing them to classify the mobile country of origin with an emphasis on using as few features as possible to minimize computational resource consumption. This paper also outlines the process of building these models and provides a concise overview of relevant techniques.

## 2. Related theories

The process of training a network traffic classifier using supervised learning consists data collection, data process and model training. This article was managed according to this process [1].

### *2.1. Data set*

In this study, a subset of the Cross Market mobile application dataset was used. The dataset was originally gathered to examine the privacy of the 100 most popular iOS and Android applications in India, China, and the US. In this dataset, the same applications network captures are used but with the goal of determining the country of origin of the application. This dataset contains 10625 samples. The distribution of the samples was shown in figure 1.

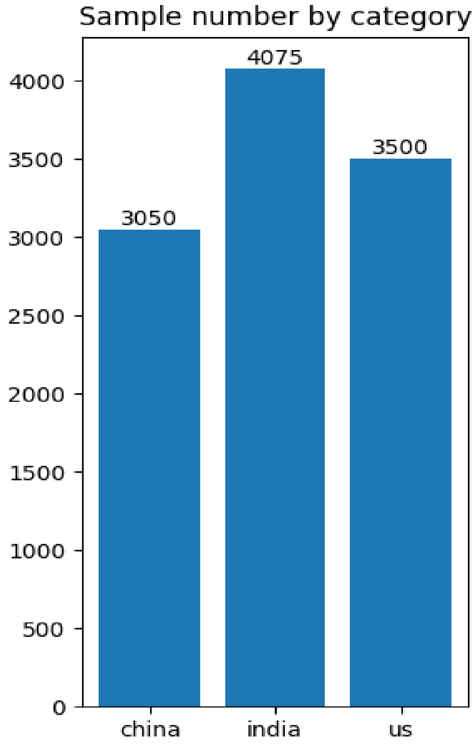


**Figure 1.** Number of samples by category.
Photo/Picture credit: Original

### *2.2. Data processing*

In this study, source IP, source host, destination IP, and destination host were selected as features since these are the characteristics that are closely related with countries. Natural language processing (NLP) techniques was implemented to vectorize the data and principal component analysis (PCA) was implemented to reduce the dimension of the data. The overall procedure of how the data was processed was shown in figure 2.

Some other features are also available, like duration of the flow, packet lengths and time-series based features. The information within the packet may also be processed to be features. However, as the growing trend of traffic encryption, this method might have more limitations in the future.

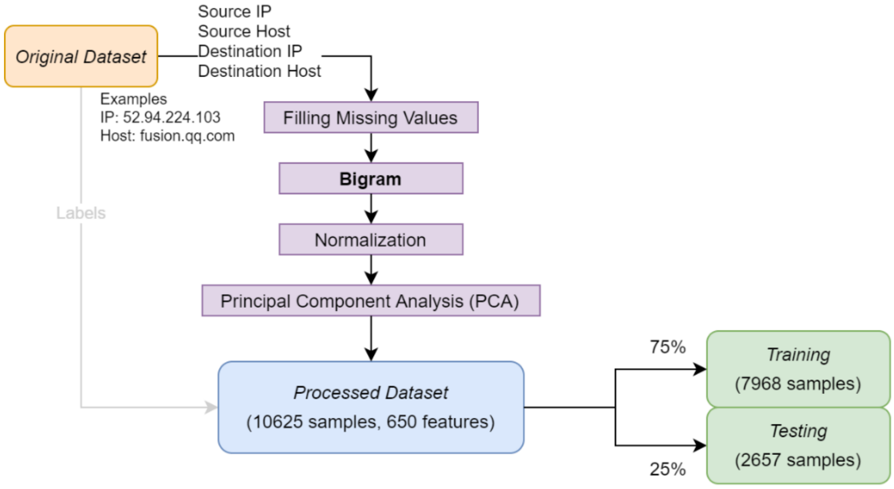


**Figure 2.** The overall procedure of data processing.
Photo/Picture credit: Original

### 2.3. *Feature extraction*

After the features were selected, these features need to be converted to a way that computers can understand. In this study, the bag-of-words model and bigram was used to vectorize Ips and host names. Strings separated by a dot was considered to be a word. Two adjacent words are grouped into a group, and the bag-of-words model generates a vector for the group of words.

There are other ways to represent network traffic data. In 2021, by aligning the binary data and remaining the semantic structure the protocols, Jordan Holland and his colleagues presented a new way to represent data that is amenable for machine learning [2].

NLP techniques has been implemented on network related classification. Apart from the bag-of-words model and bi-gram model that was used in this study, N-gram method was applied to extract contextual grammar features for domain name detection by Zhang, Sun and Wang [3]. Similarly, Doc2vec model was used to capture semantic features of domain names by Ma et al [4].

### 2.4. *Principal component analysis (PCA)*

After the features were converted to vectors, PCA was implemented to reduce the dimensions of the data. This technique reduced the dimension of the data from 4957 to 650 with 95% of variance explained figure 3.

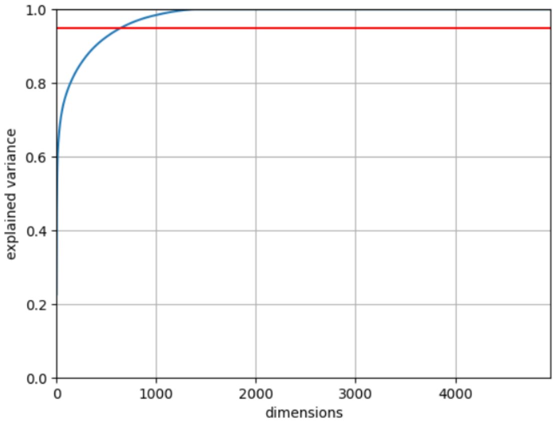


**Figure 3.** Relationship between number of dimensions and explained variance.
Photo/Picture credit: Original

By reducing the dimension of the dataset, the computation resource consumed during model training can be reduced and the generalization ability of the model can be enhanced.

### *2.5. Logistic regression*

Logistic regression is a simple classification algorithm with high computational efficiency. By applying a weight to each input and applying the sigmoid function, it can solve binomial classification problems with very high efficiency. By training multiple models, it can also be used to solve multi-classification problems.

In this study, L2 regularization was used, and the regularization parameter C was optimized by grid search and cross-validation. A small value of C would cause underfitting, a high value of C would consume more time. According to the grid search result, C was set to 1.

### *2.6. Support vector machine (SVM)*

SVM is a machine learning algorithm introduced by Vladimir Vapnik and his colleagues. This algorithm solves binary classification problems by mapping the input vector to a higher dimensional space using a kernel function and finds a hyperplane to distinguish the input vectors. Due to its good generalization ability and discriminative power, SVM algorithm has become one of the most popular machine learning algorithms. By training multiple models, SVM could also be used to solve multi-classification problems [5].

The function of kernel function is to map the original data to a higher dimensional space called "feature space". In the feature space, samples may be more easily separated by a hyperplane, thus achieving segmentation that was not possible in the original input space. Choosing a kernel is important for building a model using the SVM model, but here is no unanimous conclusion about which kernel is better or worse for specific applications [6].

In this study, grid search was used to determine the best kernel for this task. RBF was chosen according to the result of grid search shown in figure 4Figure.

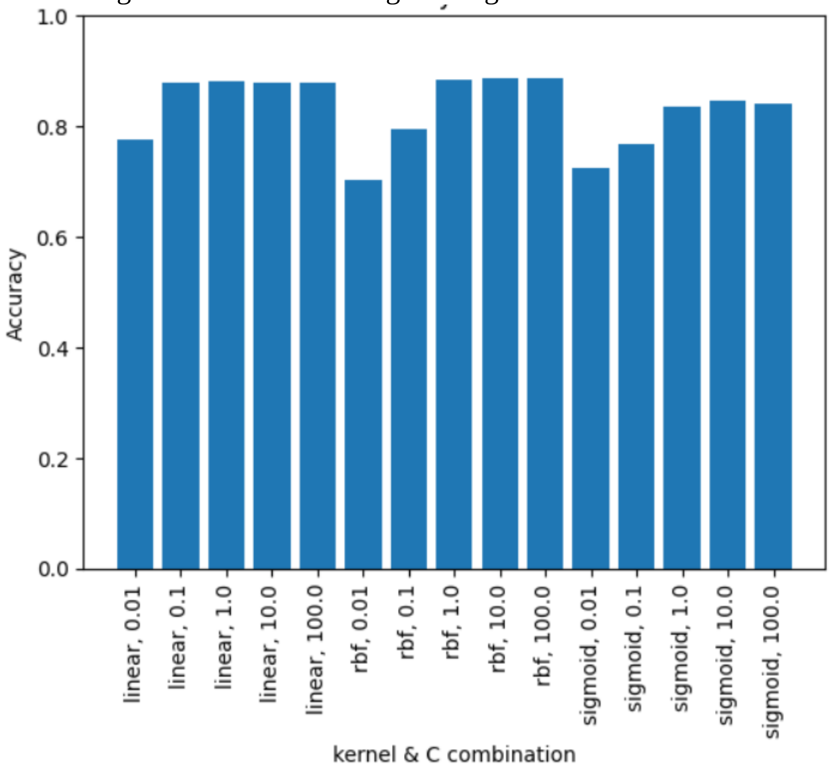


**Figure 4.** Grid search result for SVM.
Photo/Picture credit: Original

### *2.7. Feedforward neural network (FNN)*

**Table 1.** Parameters for neural network.

| Parameter | Explanation |
|---|---|
| Number of neurons | Number of neurons in a layer. |
| Activation function | The activation function used in neurons, affects the behavior of the neurons. |
| Optimizer | The algorithm used to update weights in backpropagation. |
| Learning rate | How much the weight can be changed. High learning rate may not reach the solution and causes vibration of accuracy near the end of the training. Low learning rate slow down the training process and my stuck in a local optimal solution. |
| Epochs | Number of rounds of backpropagation. Low epochs would cause underfitting and high epochs might cause overfitting. |
| Batch size | The number of samples that are input each time a neural network is trained. Larger batch sizes require a lot of memory and can easily cause overfitting, while smaller batch sizes require longer training times. |
| Loss function | The algorithm used to calculate the value of loss that was used in backpropagation. Affects the behavior of the neural network |

A neural network is a computational model inspired by the way neurons work in the human brain. A neural network consists of a large number of artificial neurons that communicate and transmit information through connections. Each neuron receives input from other neurons and produces an output. These inputs were processed according to a set of weight and then processed by an activation function to generate an output. These weight values can be learned and adjusted through training so that the network can adapt to different tasks and data. The neural network is known for its strong learning ability.

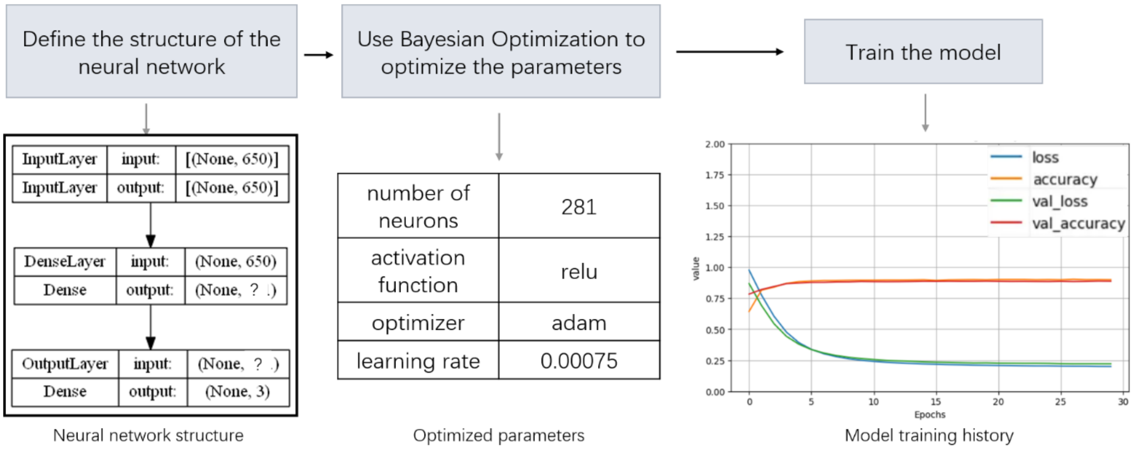
To train a neural network model, the structure of the neural network needs to be determined first. The structure of neural network can be classified to three types, which are feedforward neural network, or multi-layer perceptron (MLP), feedback neural network and Graph Neural Networks(GNN). In this study, a shallow feedforward neural network was used since it is the simplest structure of neural network and is enough for solving this problem. The process of tuning the parameters of a neural network model a laborious task. The neural network has the most tedious tuning process among the three algorithms used in this study. The parameters for a neural network were shown in table 1.

In this study, instead of using grid search, Bayesian optimization was used in order to reduce human effort. However, the process of optimizing the model still consumed lots of human effort and time.

The neural network built in this study uses only one dense layer because adding another dense layer does not show a better result on the test trail. After five rounds of Bayesian optimization, the results were shown in table 2. Maximum learning rate was set to void the vibration of accuracy near the end of the training and a validation set was used to prevent the model from overfitting [7]. The overall procedure of how this neural network was built was shown in figure 5**Figure 5.** Procedure of how this neural network was built.

**Table 2.** Parameters of the neural network trained in this study.

| Parameter | Value | Tuned by |
|---|---|---|
| Number of neurons | 281 | Bayesian optimization |
| Activation function | Relu | Bayesian optimization |
| Optimizer | adam | Bayesian optimization |
| Learning rate | 0.00075 | Bayesian optimization |
| Epochs | 30 | Hand |
| Maximum learning rate | 0.001 | Hand |
| Batch size | 200 | Hand |


Define the structure of the neural network
Use Bayesian Optimization to optimize the parameters
Train the model
InputLayer | input: | [(None, 650)]
InputLayer | output: | [(None, 650)]
DenseLayer | input: | (None, 650)
Dense | output: | (None, ? )
OutputLayer | input: | (None, ? )
Dense | output: | (None, 3)
Neural network structure
Optimized parameters
2.00
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
value
0
5
10
15
20
25
30
Epochs
loss
accuracy
val_loss
val_accuracy
Model training history

| | |
|---|---|
| number of neurons | 281 |
| activation function | relu |
| optimizer | adam |
| learning rate | 0.00075 |

**Figure 5.** Procedure of how this neural network was built.
Photo/Picture credit: Original

## 3. Experiments and analysis

### *3.1. Accuracy comparison*

The accuracy, precision, recall, and F1 score of the three models were shown in figure 6 and figure 7. The three model do not have too much different in accuracy. SVM and FNN had shown a higher accuracy, which is consistent with their ability of generalization. Considering the number features selected, this result is reasonable. This should be the highest accuracy that can be achieved without changing the features selected [8].

Due to the features used in this study, the advantage of SVM and neural network in dealing with complex on-linear problems cannot be seen.

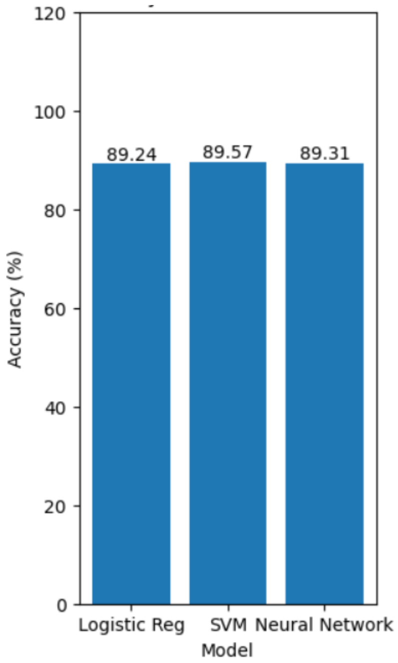


**Figure 6.** Comparison in accuracy for the three model.
Photo/Picture credit: Original

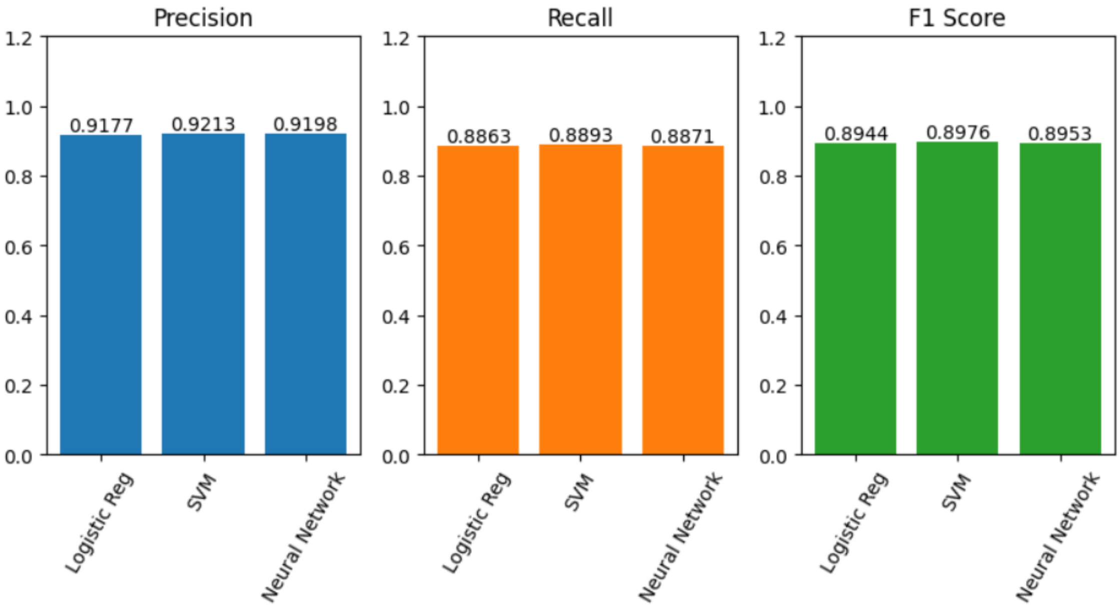


**Figure 7.** Comparison in precision, recall, and F1 score.
Photo/Picture credit: Original

### *3.2. Comparison of time*

The time consumption was measured on the same computer (i7-10875H, 16G RAM, RTX 2060) at the same condition, so the horizontal comparison should be fair. The comparison in time consumption was shown in figure 8. The logistic regression model had shown a very low time consumption on both training and testing trail while its accuracy is roughly the same compared to the SVM model and the neural network model. This result is probably caused by the feature selection. The small number of features used in this test and the feature processing technique may have caused the SVM and neural network cannot take an advantage on processing complex non-linear data, showing the advantage of processing speed of logistic regression [9].

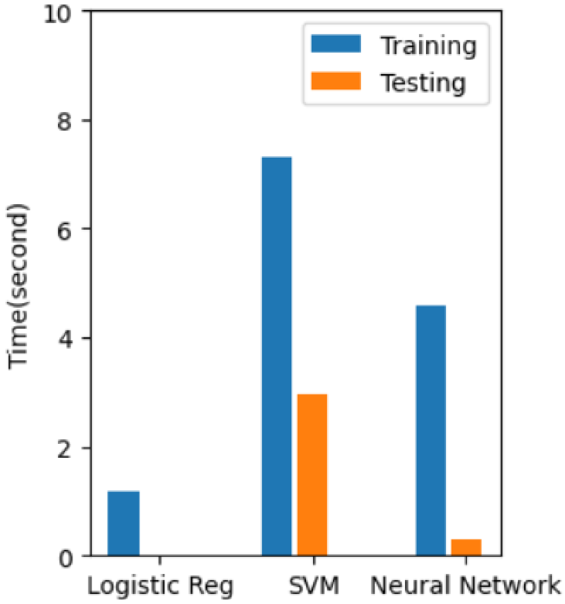


**Figure 8.** Comparison in time consumption.
Photo/Picture credit: Original

(Training stands for the time consumed in training and testing stands for the time consumed to generate the result form the test set). It is worth noting is that GPU acceleration is enabled in this comparison. Of the three algorithms, only neural networks can take advantage of GPU acceleration. The time spent of tuning the parameter of the three models cannot be measured. But the relationship of time spent on parameter tuning is neural network > SVM > logistic regression [10].

## 4. Conclusion

In summary, this study evaluated the performance of three machine learning algorithms - logistic regression, support vector machine, and neural networks - in the classification of traffic trajectories across different countries, and provided an overview of the related techniques used in constructing network classifiers. The three algorithms demonstrated comparable accuracy levels, which are considered to be the highest achievable without incorporating additional features. Notably, the logistic regression model was the least time-consuming during both training and testing phases, while the other two algorithms did not exhibit superior performance in handling complex non-linear data, likely attributable to the feature selection. The process of parameter tuning proved to be a considerable time and resource investment, particularly for neural networks. This study has a few limitations: firstly, the size of the training set was constant. Although one of the theoretical advantages of SVM is its superior performance with smaller training sets, the experimental design of this study might have precluded this advantage from manifesting. Additionally, numerical data relating to processing times could vary across different computer systems. For instance, a more powerful GPU could significantly reduce the processing time for neural networks. However, such variations should not be substantial enough to alter the comparative rankings of the algorithms in the results. Moving forward, the practice of classifying network traffic using supervised machine learning algorithms will likely improve with the development of more sophisticated data processing techniques and more efficient algorithms.

## References

[1] R. E. O. Noora AI Khater, "Network Traffic Classification Techniques and Challenges," The Tenth International Conference on Digital Information Management, 2015.

[2] P. S. N. F. a. P. M. J. Holland, "New Directions in Automated Traffic Analysis," Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security, p. 3366–3383, Nov. 2021.

[3] H. S. a. J. W. J. Zhang, "Malicious domain name detection model based on CNN-LSTM," Third International Conference on Computer Communication and Network Security, p. 57–62, Oct. 2022.

[4] S. Z. F. K. a. Z. F. D. Ma, "Malicious Domain Name Detection Based on Doc2vec and Hybrid Network," IOP Conf. Ser.: Earth Environ. Sci, vol. 693, p. 012089, Mar. 2021.

[5] Cortes, C., Vapnik, V. Support-vector networks. Mach Learn 20, 273–297 (1995). https://doi.org/10.1007/BF00994018.

[6] F. G.-L. L. R.-M. A. L. Jair Cervantes, "A comprehensive survey on support vector machine classification: Applications, challenges and trends," Neurocomputing, vol. 408, pp. 189-215, 2020.

[7] A. A. V. S. Varun Kumar Ojha, "Metaheuristic design of feedforward neural networks: A review of two decades of research," Engineering Applications of Artificial Intelligence, vol. 60, pp. 97-116, 2017.

[8] J. Liu, P. N. Pathiranage, J. Reyes, and A. Jukan, "Traffic Classification with Limited Internet Traffic Data Using Machine Learning", IEEE Access, vol. 7, pp. 107287-107301, Aug. 2019.

[9] S. Dulluri, R. S. Paithankar, and S. Nambiar, "Traffic Classification using Machine Learning Techniques: A Survey", Proceedings of Computing Conference 2019, pp. 856-860, Jul. 2019.

[10] Y. Cui, C. Zhang, Y. Jiang, M. Li, S. Li, and L. Li, "Traffic Classification Based on Machine Learning: A Survey", IEEE Access, vol. 6, pp. 20680-20697, Apr. 2018.