Knowledge Engineering and Data Science (KEDS)
Vol 4, No 2, December 2021, pp. 128–137

pISSN 2597-4602
eISSN 2597-4637

A Comparative Study of Machine Learning-based Approach for Network Traffic Classification

Kien Trang [a, b, 1], An Hoang Nguyen [a, b, 2, *]

[a] *School of Electrical Engineering, International University*
Quarter 6, Linh Trung Ward, Thu Duc City, Ho Chi Minh City 700000, Vietnam

[b] *Vietnam National University, Ho Chi Minh City*
Linh Trung Ward, Thu Duc City, Ho Chi Minh City 700000, Vietnam

[1] *tkien@hcmiu.edu.vn;* [2] *nhan@hcmiu.edu.vn**
** corresponding author*

ARTICLE INFO

Article history:
Submitted 7 December 2021
Revised 24 December 2021
Accepted 29 December 2021
Published online 31 December 2021

Keywords:
Artificial neural network
K-nearest neighbors
Machine learning
Network traffic classification
Random forest

ABSTRACT

Internet usage has increased rapidly and become an essential part of human life, corresponding to the rapid development of network infrastructure in recent years. Thus, protecting users' confidential information when joining the global network becomes one of the most significant considerations. Even though multiple encryption algorithms and techniques have been applied in different parties, including internet providers, and web hosting, this situation also allows the hacker to attack the network system anonymously. Therefore, the significance of classifying network data streams to improve network system quality and security is attracting increasing study interests. This work introduces a machine learning-based approach to find the most suitable training model for network traffic classification tasks. Data pre-processing is first applied to normalize each feature type in the dataset. Different machine learning techniques, including k-Nearest Neighbors (KNN), Artificial Neural Network (ANN), and Random Forest (RF), are applied based on the normalized features in the classification phase. An open-access dataset ISCXVPN2016 is applied for this research, which includes two types of encryption (VPN and Non-VPN) and seven classes of traffic categories classes. Experimental results on the open dataset have shown that the proposed models have reached a high classification rate – over 85% in some cases, in which the RF model obtains the most refined results among the three techniques.

I. Introduction

The accelerated development of the Internet has led to a new era of humans in the last decades. Nowadays, Internet applications are applied widely in different fields, including education and the working environment. Over a million learners are affected and need to switch to distance learning mode due to the outbreak of COVID-19 [1]. As the survey in [2], approximately 37% of US residents work remotely full-time in the first quarter of 2020, which leads to the fact that the data usage of the Internet reaches a new record height. The emergence of the Internet of Things (IoT) has brought about a major shift in the growing number and variety of connected devices and different applications supported by the network service provider. Thus, network traffic classification can solve complex network management problems for Internet Service Providers (ISPs).

The goal of network traffic classification is to identify various types of network protocols and applications existing in a network to facilitate network management. The packets are classified to calculate the appropriate service policy for the routers. QoS, network planning, monitoring, traffic trend analysis, and firewall configuration benefit from traffic classification. Moreover, Internet traffic classification may play an important component of automated intrusion detection systems for automatically identifying denial of service attacks to allocate network resources to priority

https://doi.org/10.17977/um018v4i22021p128-137

customers [3]. The ISPs can also increase the quality of services by accelerating the incident management process based on the Internet traffic classification.

In network traffic classification, traditional methods have certain limitations. Firstly, packet marking is suggested to distinguish traffic based on its QoS class. Some common fields are used, such as Type of Service (ToS), Differentiated Services Code Point (DSCP), and Explicit Congestion Notification (ECN). Then, several protocols have been proposed for traffic classification, including Differentiated Services (DiffServ), Integrated Services (IntServ), and Multi-Protocol Label Switching (MPLS). Due to the system compatibility problem, these protocols are not widely deployed and applied in reality. Besides, Port-based and Payload-based are known as the commonly applied techniques in terms of tradition. Each packet is assigned a port number assigned by Internet Assigned Number Authority (IANA) for the port-based method. The classification can be obtained based on the registered port number. For instance, port 25 (SMTP) and port 110 (POP3) are used to send and receive mail, respectively. However, due to the increase of Internet applications, dynamic port numbers and tunneling are used to hide the port number leading to the limitations in this method [4]. For the payload-based method, the data packet's content is examined against the characteristics of network applications in Internet traffic. This technique is especially recommended for Peer-to-Peer (P2P) applications. However, this technique also has certain limitations due to the high demand for hardware to detect features in data packets and the incapacity of handling encrypted data traffic packets [5][6]. In general, these traditional approaches have drawbacks in terms of classified accuracy and resources.

Over the last few years, in artificial intelligence (AI) research, machine learning (ML) has achieved remarkable success that allows automatic identification and classification without human intervention in some cases. Some recent research is gradually switching towards machine learning applications in network traffic classification. Yuan *et al*. [7] introduced using an advanced version of the decision tree called Hadoop C4.5 to classify the network traffic. The applied dataset contains eight classes which have 248 properties for each class. The results give an improvement in terms of classified speed and accuracy compared to the original method – reaching over 80%. The study in [8] mentioned the Netmate tool to select 23 core features before training for classification. Different algorithms are applied for comparting, including C4.5, Support Vector Machine (SVM), BayesNet, and Naive Bayes. Among these experiments, C4.5 gives the highest accuracy – 78.9%, while the lowest is 68.1% belongs to BayesNet. Similarly, Y. Ma *et al*. also applied the C4.5 decision tree to classify Internet traffic – reaching 88% in average accuracy. SVM and K-means are employed based on the realistic traces of the Internet in the research of Z. Fan *et al*. [9]. They apply the feature selection before the training stage. Different training and test set ratios are conducted, and the overall results are about 98% for both classifiers. According to a study of these classification outcomes, the classification model based on supervised learning algorithms has greater precision than the classification model based on unsupervised learning methods. Four distinguish feature selection methods also are discussed as a pre-processing step in [10] to improve the efficiency of the computation process and limit the error in classification. Besides, they also conduct experiments on different classifiers, including k-Nearest Neighbors (KNN), Random Forest (RF), and Gradient Boosting. The accuracy of feature selection methods and classifiers is approximately 85% in general. NaiveBayes classifier also is applied in work [11], [12], and [13], reaching over 90%, 93% and around 55%, respectively.

As a result of the support from hardware, deep learning becomes one of the most helpful assistants in the task of classification. Convolutional Neural Network (CNN) is introduced as one of the powerful methods to deal with the complicated image-based classification for the huge dataset. The end-to-end architecture of CNN can feed the input data directly without feature extraction or pre-processing and output predicted probabilistic or predicted class. Although many proposed models are established for graphical classification usage, inspired by previous studies, many researchers try to adjust these models to fit with the network traffic classification. F. Zhang *et al*. [14] proposed an improved version for Capsule Neural Network (CapsNet) to identify network traffic. A conversion step and normalization are conducted to turn the features into the two-dimensional array before feeding into the networks. Three versions of CNN are compared in the experiments with an average accuracy of over 95%. Besides, the study [15] introduced using the pre-trained model ResNet and self-developed CNN. The result from ResNet outperforms self-developed CNN, which reaches nearly 95.5% and 97%, respectively. The author explains that ResNet has the pre-trained weight and more complex architecture than the rest. However, deep learning is not a universal method to apply to every case; indeed, the dependence on the dataset is one of the big challenges.

There are three distinguished methods conducted in the study [16], including Random Forest (RF), Linear Discriminant Analysis (LDA), and Deep Neural Network (DNN). Regarding accuracy, two traditional machine learning methods, RF and LDA, have higher results than DNN for scenario A, while there is an improvement of DNN over RF in scenario B. L. Zhipeng *et al*. [17] discussed using two famous pre-trained CNN models: ResNet50 and GoogleNet. Since these two models are used for images, one-hot encoding transforms the symbolic features to the binary features stored as vectors. Afterward, the binary vectors are converted to grayscale images. The results give about 81% for the two given models. To deal with the limited samples dataset, the work in [18] proposed using Deep Convolutional Generative Adversarial Network (DCGAN) to generate more samples before training progress.

This network can perform semi-supervised learning with the existent samples and create more new data to enrich the dataset. By this method, the learning for classification would have more data for training and testing, which can improve the generalization and prevent the overfitting problem. As the baseline CNN results, this study achieves 89% and 78% for self-collected and ISCX datasets, respectively. The research in [19] proposed employing feature extraction based on a convolutional recurrent autoencoder neural network. The proposed approach is established based on the autoencoder architecture, consisting of the encoder, latent space, and decoder. Different DNNs are applied to verify the performance, including CNN, Sparse Autoencoder (SAE), and Long Short-Term Memory (LSTM). Ultimately, the Stacked-CNN–LSTM architecture reaches the highest performance in almost all metrics. Table 1 shows the summary of the related studies in Network Traffic Classification.

Table 1. Comparative studies

Research	Method	Result	Number of Class
[7]	C4.5 Decision tree	Over 80%	8
[8]	C4.5 Decision tree SVM Bayes Net NaiveBayes	78.9% 74% 68.1% 71.8%	5
[9]	SVM K-means	~98% for both methods	6
[10]	KNN Random Forest Gradient Boosting	~85% for all cases	Not mentioned
[11]	NaiveBayes	Over 90%	7
[12]	NaiveBayes	93%	Not mentioned
[13]	NaiveBayes	54~55% for all cases	3
[14]	Improved Caspnet	Over 95%	12
[15]	ResNet self-developed CNN	~97% ~95.5%	8
[16]	Random Forest LDA DNN	95%, 42% for Scenario A, B 98%, 76% for Scenario A, B 69%, 74% for Scenario A, B	3
[17]	ResNet50 GoogleNet	81.5% 81.8%	5
[18]	DCGAN + base-line CNN	89% for self-collected dataset 78% for ISCX dataset	Not mentioned
[19]	CNN-SAE-CNN LSTM-SAE-NN CNN-LSTM-SAE-NN Stacked- CNN-LSTM-SAE-NN	> 95% for all cases	4

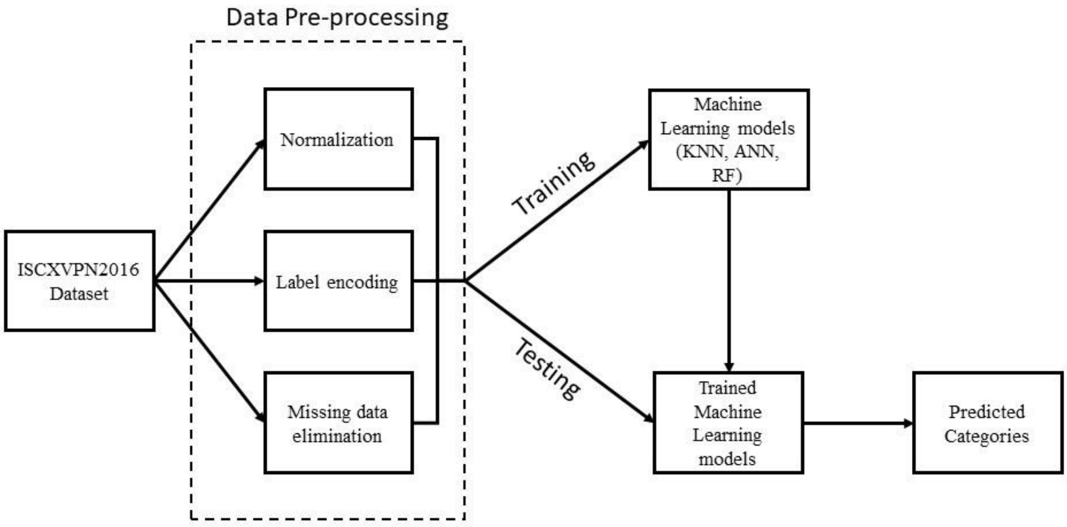

Fig. 1. The processing chart of the proposed algorithms

Although various parties have used different encryption methods and approaches, including internet service providers and web hosting companies, this circumstance allows a hacker to attack the network system anonymously. As a result, the importance of classifying network data streams in order to improve the quality and security of network systems is drawing an increasing amount of research interest. This work introduces a machine learning-based approach for determining the most appropriate training model for network traffic classification tasks, described in detail elsewhere.

II. Approach

Figure 1 depicts the processing chart of the proposed approach. The dataset applied for this work is taken from [20]. Before feeding into machine learning models, pre-processing is initially applied to meet some basic requirements, including normalization and data transformation. Then, the dataset is divided into two subsets: training and test set. Finally, different traditional machine learning models are applied to test different scenarios. From the comparative studies in the previous section, the traditional models almost give better performance than the advanced models, and it can be explained that different datasets may have a variety in size and latent properties, leading to the fact that deep learning techniques cannot perform well in some narrow size of the dataset. Thus, K-Nearest Neighbors (KNN), Artificial Neural Network (ANN), and Random Forest (RF) are chosen to apply in this study.

A. Data Pre-processing

Since the given dataset contained different types of features with various ranges, this leads to that pre-process step being applied to deal with the classification purpose of the proposed approach. Therefore, normalization is necessary to convert the numerical value to a similar scale without affecting the difference of value range. Therefore, scale normalization is applied by (1).

$$d_i^{'} = \frac{d_i - \min(d)}{\max(d) - \min(d)} \tag{1}$$

where d is the feature vector, d_i each element in the feature vector, and the corresponding normalized element. After this process, the feature would be in the range of 0 and 1. Besides, each class's label name, such as VPN-Mail and VPN-VOIP, needs to be converted into numeric values. Missing values in data can be caused by data corruption or a failure to record data which also influences classification performance. Since some machine learning algorithms are not able to work with missing values. The corresponding data element will be removed to prevent the impact of the training process to deal with this phenomenon.

B. Machine Learning Models

In general, machine learning is the process of seeking and describing structural patterns in a given data set. The output of the machine learning model will be a description of the learned knowledge which can be classification and regression.

1) K-Nearest Neighbors (KNN)

KNN is one of the most fundamental and simplest in the supervised machine learning algorithms, which operates by grouping all the samples having similar characteristics of the dataset [21]. Instead of learning from the training data, KNN mechanically memorizes all the data. Then, all the computational processes are conducted in the test phase, which means that every time a sample of the test dataset is input for classification, the algorithm computes the difference between the testing data point and the nearest ones. The predicted label is dependent on the label of the nearest data points having the minimum distance [22]. In addition, a voting process may be conducted in the case of many different labels in the data points.

Let $X = \{x_1, x_2, ..., x_n\}$ is a sample and $x_1, x_2, ..., x_n$ are the features of the sample. The majority rule specifies the classification procedure based on the number of k-nearest reference vectors from the projection of the sample X . An assumption of all samples in the data set corresponding to points that exist in an n-dimensional space denoted by $\mathfrak{R}^n$ is conducted. Distance metrics define the distance between points in the mentioned spatial dimension. The formula for calculating the distance between samples X_i and X_j is defined in (2).

$$d(X_i, X_j) = \left(\sum_{f=1}^{n} \left| x_f^i - x_f^j \right|^p \right)^{1/p} \tag{2}$$

where x_f^i and x_f^j are the corresponding value of the number of features f of the data sample X_i and X_j , respectively. Next, the algorithm selects k points corresponding to the number of samples in the training set with the closest distance to the sample at the input. The sample's label X will be classified based on the number of classes of the above k samples according to the rule of majority voting.

2) Artificial Neural Network (ANN)

ANN is a machine learning algorithm that simulates the biological neural activity of humans. This method consists of 3 main layers: input, hidden, and output. Each layer consists of many neurons which are connected to process information. Each neuron includes data inputs to receive and process to produce an output. In addition, the neuron output or the neuron processing result can be used as an input for other neurons. Independent values in the input are passed to the neural network node to produce dependent values in the output. The precondition is that those output values must correspond to the input data group as independent variables.

Each input value x_i is attached with corresponding weight w_i and bias b_i , representing the importance of that input value at the neuron node compared to other input values. The computation takes the summation of all input data values with the weights and biases for each neuron. These weights are set randomly by default at the initial. During the training, the updated weights are computed through the optimization process. Then, an activation function is applied to map the input values of a neuron node to the output. The mathematical representation is defined in (3).

$$m_i = \sum_{i=1}^{K} w_i x_i + b_i \tag{3}$$

where K is the number of input values passing through a neuron. Therefore, after mapping the activation, (3) is adjusted to (4).

$$y_i = f(m_i) = f\left(\sum_{i=1}^{K} w_i x_i + b_i \right) \tag{4}$$

3) Random Forest (RF)

Random Forest, developed in the study [23], is the combination of multiple decision trees referred to as the bagging method. The typical decision tree model classifies the data samples in the training dataset based on their features. The training process starts from the root with the complete dataset, splitting into smaller samples at the different terminal or intermediate nodes based on the values of specific calculation metrics, such as Entropy or Gini index, of one respective feature. The Entropy is the parameter indicating the randomness of the analyzing feature, which decides how the model splits the data into subsets based on that respective feature. Then, based on the Entropy values, the model calculates the Information gain, determining how well the data were split. The decision tree mostly tries to maximize the Information Gain while keeping the Entropy value minimum. The formula for calculating the Entropy and information gain is illustrated in (5) and (6).

$$E = \sum_{i=1}^{c} -p_i \log_2(p_i) \tag{5}$$

$$IG = E(t-1) - \sum_{j=1}^{k} E(j,t) \tag{6}$$

where c is the number of features, k is the maximum number of subsets divided. The random forest utilizes different inputs with different features corresponding to each decision tree for predictions. Multiple prediction outcomes are made to classify the data samples. The final classification step of the random forest model is made based on the majority rule of the outcomes of those decision trees. Therefore, the increase in the number of decision trees during the RF model creation helps to increase the accuracy level in classification decisions while avoiding the heavy computation load in the hyperparameter tuning process.

III. Results Analysis

This section presents the dataset and scenarios description, the evaluation metrics, and the discussions of the obtained results from the three machine learning models, being RF, KNN, and ANN, respectively.

A. Dataset Description

In the scope of this research, the dataset VPN – Non-VPN (ISCXVPN2016) [20] is implemented for the training and testing phases. It was created during an experiment from the New Brunswick University, Canada, in which the dataset generator created two user accounts to participate in different Internet services such as Facebook, uTorrent, and Skype.

Each class inside the dataset is also divided into two categories: Non-VPN and VPN encryption traffic classes. Therefore, the total number of labels for classification can be considered up to 14 classes. The dataset nature corresponds to the training, and the testing scheme is divided into two steps. The first step is to classify the two general classes, Non-VPN and VPN encrypted traffic flow. Afterward, for each class, seven distinguished traffic flows are classified. The detailed process in the classification task is described in Figure 2. Besides the types of Internet traffic-based classification, the data also includes time-based division. Therefore, for each step of the classification process, the data is divided into four categories 15, 30, 60, and 120s.

B. Evaluation Metrics

In this study, the experiments are conducted on the Colab Pro-environment with 26 GB RAM and GPU NVIDIA Tesla P100. The applied dataset is separated into two subsets: training and test set

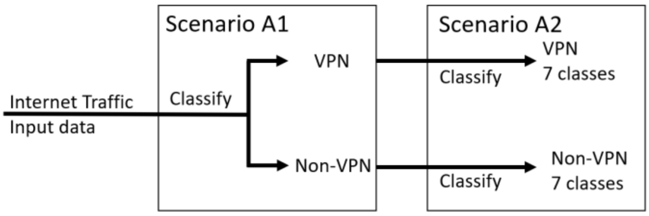

Fig. 2. Dataset classification scenarios

followed the ratio 80/20, respectively. Besides, cross-validation is not applied in this case due to the large dataset. In machine learning and artificial intelligence, one of the most common evaluation means is the usage of the confusion matrix. The confusion matrix is often implemented to evaluate the performance of a supervised learning model and the level of confusion while classifying the classes. The confusion matrix consists of the main four parameters, which are true positive (TP), false positive (FP), true negative (TN), and false-negative (FN).

The calculation from these four numbers could be implemented to examine the learning models through the frequently used evaluation metrics: accuracy, precision, Recall, and F1-score. Among the four metrics, the most famous indicator is the accuracy level, the number of samples classified correctly to their respective labels within the whole dataset. The formula to calculate the accuracy value is demonstrated in (7).

$$A = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

Even though the accuracy level is frequently used to obtain a basic understanding of the learning models, it is not recommended to neglect the number of falsely classified data samples into incorrect labels. Therefore, to perform a complete assessment over given learning models, the combination of other metrics is necessary. Precision is the number of samples data classified into a label belonging to that class. On the contrary, Recall is the number of samples accurately classified into a class over the total number of samples correctly and incorrectly classified into that respective class.

Finally, the F1-score is the combination metric from both Precision and Recall values, in which it is only archived high performance by having high results in those two metrics. Through the analysis of F1-score, the assessment process will acquire a thorough evaluation of the efficiency of the learning model. The formulas to calculate the above values are described in (8), (9), and (10), respectively.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{8}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{9}$$

$$\text{F1-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{10}$$

1) Scenario A1

In scenario A1, the primary purpose of classification is to distinguish the Internet traffic flow into two categories, Non-VPN and VPN encrypted traffic usage. The dataset is divided into four subsets

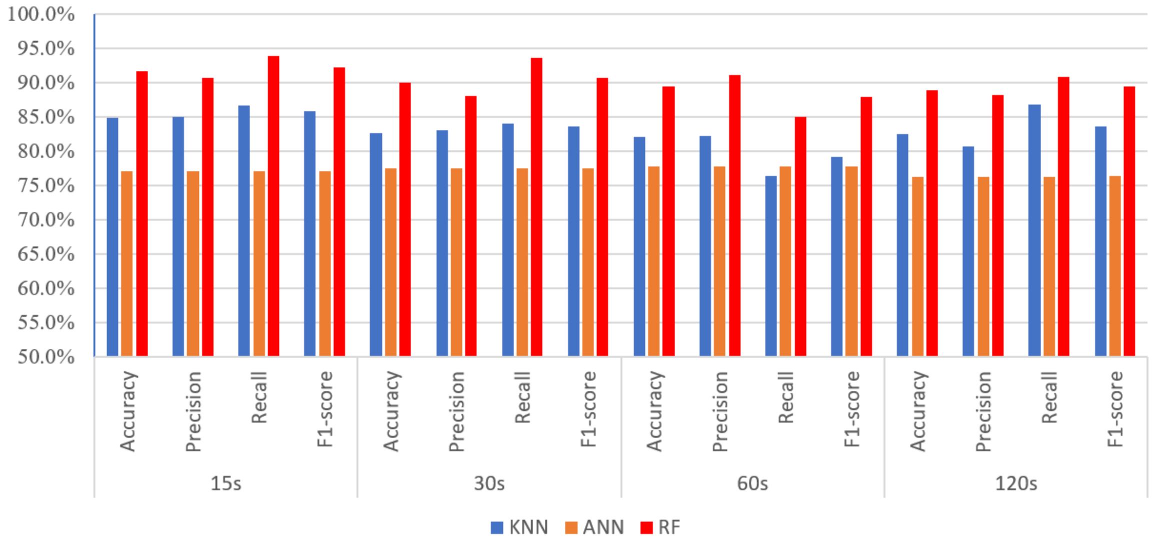

Fig. 3. Dataset classification scenarios

of data samples with different amounts of time recorded, which are 15, 30, 60, and 120s. The evaluation metrics in all four subsets of data samples of the three machine learning models are recorded in Figure 3.

At a glance, the results recorded from the RF model are the highest values, followed by the KNN and the lowest results from the ANN model. Throughout the evaluation metrics Accuracy, Precision, Recall, and F1-score, the RF model consistently produces numbers around the 88-94% range. The Recall value of the RF model in the 60s dataset is the only exception at 85%, which is in the higher range than other models. The ANN model is the least effective in classification, with the average range staying approximately at 77%. However, compared to the other two models, the ANN is the most balanced since all four metrics are almost the same throughout the different time-based datasets. In other words, the time feature does not affect the performance of the ANN model. On the other hand, the KNN model provides relatively high results, with most metrics being approximately 80-86%. The 60s subset of data is the worst time-based samples toward this model, with 82%, 82.1%, 76.3%, and 79.13% recorded for Accuracy, Precision, Recall, and F1-score, respectively.

2) Scenario A2 – Non-VPN

In contrast with the total domination of the RF model in scenario A1, the remarkable evaluation metrics of the Non-VPN subsets divide alternately by the ANN and RF model. To be more specific, the RF model scores the highest mostly in the Accuracy and Precision aspects, whereas the Recall and F1-score are greater in the ANN model than the other two, as indicated in Figure 4. In the accuracy metrics, all of the time-based subsets produce more than 90% results for the RF model, with only the 120s dataset containing an exception of the KNN and RF sharing the same 92.8% value. For the Precision aspect, the upper range recorded in the RF model is around 86-88%. Another point to be noted is that, even though the ANN is not always the greatest model, the produced results are greater than 80%.

On the other hand, the Recall and F1-score metrics mark a big drop in the performance of RF and KNN models. All KNN models fall below 70%, with the lowest value being the Recall in the 60s dataset of only 61%. The drop in evaluation metrics also appears in the RF model as the time feature increases in the time-based subset of samples. The highest values are in the 15s dataset, with Recall of 81.5% and F1-score is 84.8%. In the 120s dataset, these values fall to 66.1% and 73.5%, respectively. In contrast, the ANN model shows the most stable values, mostly greater than 80%. The only exception is in the 60s data set in which the values are lower, around 2% than the RF model.

3) Scenario A2 – VPN

In the case of the VPN encrypted subset, the performance of the learning models recorded a significant drop in the Precision, Recall, and F1-score except for the ANN model. The results are illustrated in Figure 5. The RF model still provides three over four highest values in the Accuracy

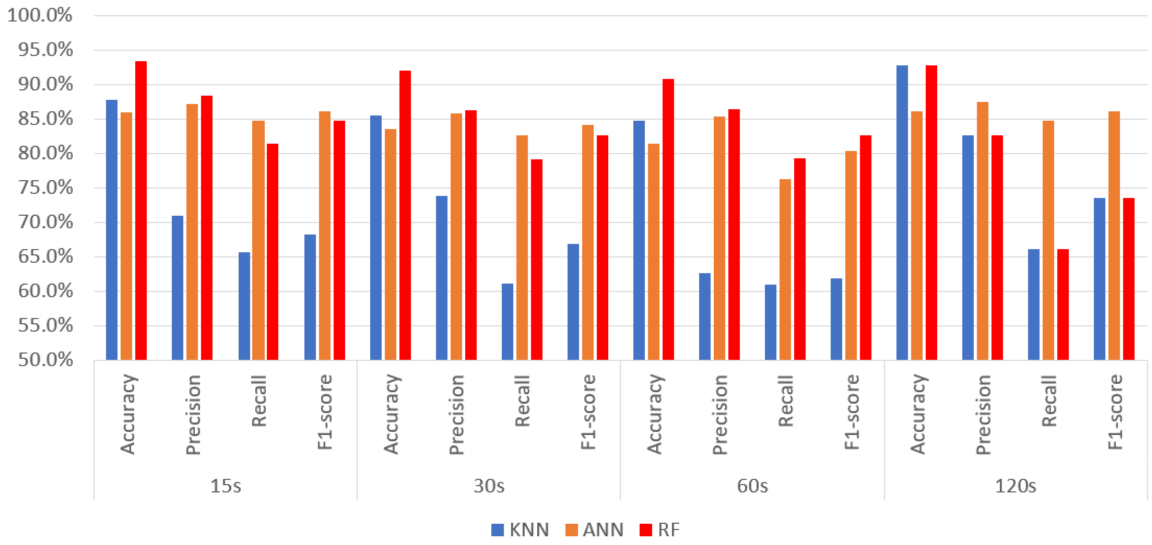

Fig. 4. Recorded evaluation metrics for KNN, ANN, and RF models – Scenario A2 – Non-VPN

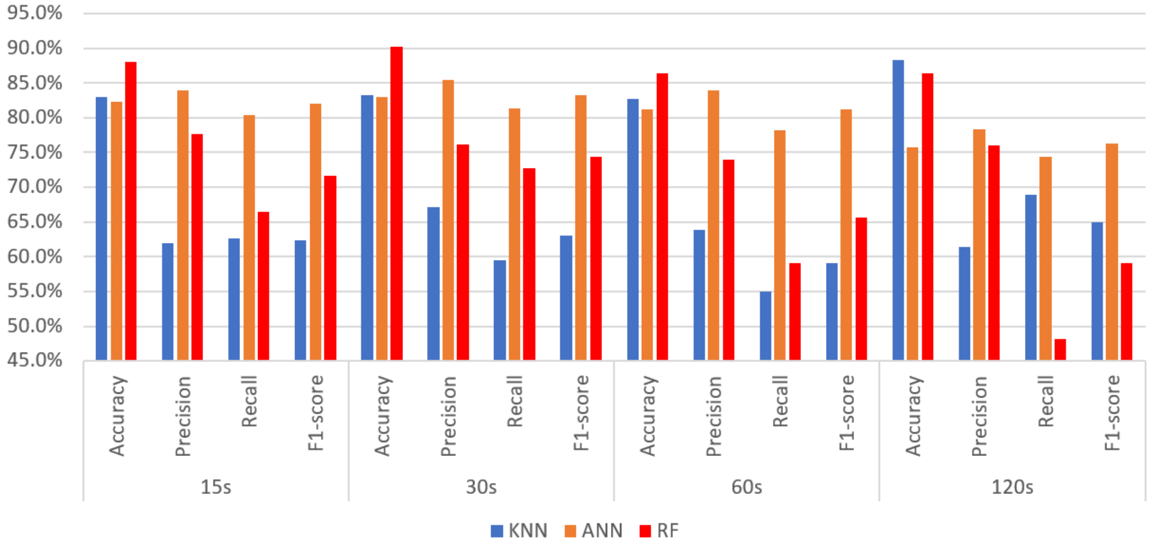

Fig. 5. Recorded evaluation metrics for KNN, ANN, and RF models – Scenario A2 –VPN

metric, all of which are larger than 86%. However, only in the 120s, the peak value belongs to the KNN model with 87.8% in classified accuracy. On the contrary, the ANN model displays total domination in the three remaining metrics. Most of the values in the 15, 30, and 60s dataset were recorded in the range of approximately 80.5% - 84%. The trend only reduces in the 120s dataset, with the Precision, Recall, and F1-score being 78.4%, 74.4%, and 76.3%, respectively.

IV. Conclusions

In this research, different machine learning models are recommended to classify multiple Internet traffic flows included in the open-access VPN – Non-VPN (ISCXVPN2016) dataset. The learning models include the Random Forest, the K-Nearest Neighbors, and the Artificial Neural Networks. The models are trained, then perform the classification task in two steps: the Non-VPN and VPN classification in scenario A1. Subsequently, the models classify each subset into seven different Internet traffic classes. Based on the obtained results, the Random Forest is the most suitable training model for this dataset, even though the classification results indicate that it is not accurate in the long-time data samples, such as the results in the 120s subset.

In future research, different datasets of more complex Internet traffic classification schemes and more effective yet suitable training models such as reinforcement learning models could be considered for further analysis. The ISCXVPN2016 dataset is well established with different categories and sub-scenarios. However, new Internet and communication protocols and applications are emerging daily, corresponding to the rapidly increasing Internet usage rate all over the world. The encryption protocols are also developed to protect the user's personal information and secure the Internet connection. Therefore, appropriate training models fitting in the purpose of the Internet flows classification, which is suitable for practical application and development, can be discovered and will be the main target for research in the field.

Declarations

Author contribution

All authors contributed equally as the main contributor of this paper. All authors read and approved the final paper.

Funding statement

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflict of interest

The authors declare no known conflict of financial interest or personal relationships that could have appeared to influence the work reported in this paper.

Additional information

Reprints and permission information are available at http://journal2.um.ac.id/index.php/keds.

Publisher's Note: Department of Electrical Engineering - Universitas Negeri Malang remains neutral with regard to jurisdictional claims and institutional affiliations.

References

[1] G.R. El Said, "How Did the COVID-19 Pandemic Affect Higher Education Learning Experience? An Empirical Investigation of Learners' Academic Performance at a University in a Developing Country", *Advances in Human-Computer Interaction*, vol. 2021, pp. 1–10, Feb. 2021.

[2] L. Yang, D. Holtz, S. Jaffe, S. Suri, S. Sinha, J. Weston, C. Joyce, N. Shah, K. Sherman, B. Hecht, and J. Teevan, "The effects of remote work on collaboration among information workers," *Nature Human Behaviour*, Sep. 2021.

[3] L. Stewart, G. Armitage, P. Branch, and S. Zander, "An Architecture for Automated Network Control of QoS over Consumer Broadband Links," TENCON 2005 - 2005 IEEE Region 10 Conference, pp. 1-6, November 2005.

[4] T. Karagiannis, A. Broido, M. Faloutsos, and K. claffy, "Transport layer identification of P2P traffic," *Proceeding of the 4th ACM SIGCOMM conference on Internet measurement (IMC '04)*, New York, pp. 121–134, September 2004.

[5] P. B. Park, Y. Won, J. Chung, M. Kim, and J. W.-K. Hong, "Fine-grained traffic classification based on functional separation," *International Journal of Network Management*, vol. 23, no. 5, pp. 350–381, Aug. 2013.

[6] G. Aceto, A. Dainotti, W. de Donato and A. Pescape, "PortLoad: Taking the Best of Two Worlds in Traffic Classification," *2010 INFOCOM IEEE Conference on Computer Communications Workshops*, pp. 1-5, March 2010.

[7] Z. Yuan and C. Wang, "An improved network traffic classification algorithm based on Hadoop decision tree," *2016 IEEE International Conference of Online Analysis and Computing Science (ICOACS)*, pp. 53-56, May 2016.

[8] M. Shafiq, X. Yu, A. A. Laghari, L. Yao, N. K. Karn and F. Abdessamia, "Network Traffic Classification techniques and comparative analysis using Machine Learning algorithms," *2016 2nd IEEE International Conference on Computer and Communications (ICCC)*, pp. 2451-2455, October 2016.

[9] Z. Fan and R. Liu, "Investigation of machine learning based network traffic classification," 2017 *International Symposium on Wireless Communication Systems (ISWCS)*, pp. 1-6, August 2017.

[10] A. Pasyuk, E. Semenov and D. Tyuhtyaev, "Feature Selection in the Classification of Network Traffic Flows," *2019 International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon)*, pp. 1-5, October 2019.

[11] Y. Wang, Y. Xiang and S. Yu, "Internet Traffic Classification Using Machine Learning: A Token-based Approach," *2011 14th IEEE International Conference on Computational Science and Engineering*, pp. 285-289, August 2011.

[12] S. Dong and R. Jain, "Flow online identification method for the encrypted Skype," in *Journal of Network and Computer Applications*, vol 132, pp. 75-85.

[13] M. Dixit, R. Sharma, S. Shaikh and K. Muley, "Internet Traffic Detection using Naïve Bayes and K-Nearest Neighbors (KNN) algorithm," *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, pp. 1153-1157, May 2019.

[14] F. Zhang, Y. Wang and M. Ye, "Network Traffic Classification Method Based on Improved Capsule Neural Network," *2018 14th International Conference on Computational Intelligence and Security (CIS)*, pp. 174-178, November 2018.

[15] H. Lim, J. Kim, J. Heo, K. Kim, Y. Hong and Y. Han, "Packet-based Network Traffic Classification Using Deep Learning," *2019 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*, pp. 046-05, February 2019.

[16] J. Kwon, D. Jung and H. Park, "Traffic Data Classification using Machine Learning Algorithms in SDN Networks," *2020 International Conference on Information and Communication Technology Convergence (ICTC)*, pp. 1031-1033, October 2020.

[17] Z. Li, Z. Qin, K. Huang, X. Yang, and S. Ye, "Intrusion Detection Using Convolutional Neural Networks for Representation Learning," Lecture Notes in Computer Science, pp. 858–866, 2017.

[18] A. S. Iliyasu and H. Deng, "Semi-Supervised Encrypted Traffic Classification with Deep Convolutional Generative Adversarial Networks," in *IEEE Access*, vol. 8, pp. 118-126, 2020.

[19] G. D'Angelo and F. Palmieri, "Network traffic classification using deep convolutional recurrent autoencoder neural networks for spatial–temporal features extraction," *Journal of Network and Computer Applications*, vol. 173, pp. 102890, 2021.

[20] G. Draper-Gil, A. H. Lashkari, M. S. I. Mamun, and A. A. Ghorbani, "Characterization of Encrypted and VPN Traffic using Time-related Features," *Proceedings of the 2nd International Conference on Information Systems Security and Privacy (ICISSP2016)*, pp. 407-414, February 2016.

[21] H. A. H. Ibrahim, O. R. Aqeel Al Zuobi, M. A. Al-Namari, G. Mohamed Ali, and A. A. A. Abdalla, "Internet traffic classification using machine learning approach: Datasets validation issues," *2016 Conference of Basic Sciences and Engineering Studies (SGCAC)*, pp. 158-166, February 2016.

[22] A. Moldagulova and R. B. Sulaiman, "Using KNN algorithm for classification of textual documents," *2017 8th International Conference on Information Technology (ICIT)*, pp. 665-671, May 2017.

[23] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, Mar. 1986.