UDC 621.391

ADJUSTING THE PARAMETERS OF MACHINE LEARNING ALGORITHMS TO IMPROVE THE SPEED AND ACCURACY OF TRAFFIC CLASSIFICATION

Andrii A. Astrakhantsev, Larysa S. Globa, Andrii M. Davydiuk, Oleksandra V. Sushko

Educational and Research Institute of Telecommunication Systems
Igor Sikorsky Kyiv Polytechnic Institute, Kyiv, Ukraine

Background. Telecommunications developments lead to new mobile network technologies and especially 5G, which has only recently been launched, sixth generation of which is already under active development. The development of new technologies influence on both types of mobile traffic (V2V, IoT) and leads to the significant increase in the volume of existing traffic types. Currently, existing methods of traffic processing are not adapted to such changes, which may lead to a deterioration in the quality of service.

Objective. The purpose of the paper is to analyze the effectiveness of machine learning algorithms to solve the task of traffic classification in mobile networks in real time.

Methods. The method of solving the problem of increasing the efficiency of information processing is the introduction of new algorithms for traffic classification and prioritization. In this regard, the paper presents the urgent task of analyzing the effectiveness of machine learning algorithms to solve the task of traffic classification in mobile networks in real time.

Results. Comparison indicated the best accuracy of the ANN algorithm that was achieved with the number of hidden layers of the network equal to 200. Also, the research results showed that different applications have different recognition accuracy, which does not depend on the total number of packets in the dataset.

Conclusions. This proceeding solves the urgent problem of increasing the efficiency of the mobile communication system through the use of machine learning algorithms for traffic classification. In this regard, it can be concluded that the most promising is the application of algorithms based on ANN. In future the aspect of anomaly detection based on traffic classification and traffic pattern preparation should be investigated, as this process allows detecting attacks to network infrastructure and increase mobile network security.

Keywords: *traffic classification; anomaly detection; machine learning technics; artificial neural network; feature analysis; traffic patterns; 5G network; accuracy; classification speed.*

Introduction

The intensive development and implementation of the latest 5G networks in recent years and the prospects for the implementation of 6G have revealed a number of new problems of modern networks [1]. Compared to traditional 3G/4G communication and traffic in cellular networks, 5G adds a large number of semi-autonomous and autonomous cars, various smart products, and many different sensors. All of this can cause serious problems in both the core network and the radio access network (RAN) as it leads to congestion and reduced quality of service. In order to prevent congestion, it is necessary to improve the existing methods of preliminary classification of traffic and its further distribution for processing. A key feature for effective traffic processing in 5G/6G networks is network slicing [2], which allows system resources to be distributed depending on the type of application and to process each slice separately. For the effective operation of this feature, preliminary classification and marking (labeling) of traffic is also very important.

The classification of network traffic allows you to organize its differentiated service in accordance with the requirements for the quality of service (QoS) level [3], which allows you to allocate network resources to ensure optimal QoS indicators for different classes of traffic. For example, high-priority network traffic or specific criteria matching traffic should be singled out for special processing, thereby helping to achieve peak performance for both applications and the network as a whole.

In general, the running time of an algorithm increases with the number of inputs, so it is common practice to represent the running time of a program as a function of the number of inputs. Thus, a sharp increase in traffic will reduce the effectiveness of algorithms.

Quality-of-service traffic handling may include faster forwarding using intermediate routers and switches, or reducing the likelihood that traffic packets were dropped due to lack of resources at intermediate nodes.

The purpose of the work: analysis of the effectiveness of machine learning algorithms for solving the task of traffic classification in 5G/6G mobile networks according to the criteria of classification quality and speed. Based on the results of the analysis, recommendations for the application of machine learning algorithms should be formed and their optimal parameters should be determined.

1. Existing approaches

Classification of network traffic can be done using information from various OSI layers. In our work we rely on L2/L3 information only (IP headers, ports).

Widely applied approaches [1-4] are RF, KNN, ANN, AdaBoost and SVM. These algorithms were used for comparative analysis in this paper. In addition to machine learning algorithms, Deep Packet Inspection (DPI) techniques can be used to classify traffic. Deep Packet Inspection gives the best accuracy, but it is not applicable for our case is the most advanced technology for traffic classification as it is the most accurate method. However, the actual performance of DPI is still unclear as the limited number of public datasets limits the comparability and reproducibility of the results [5].

2. Research pre-conditions and metrics

For traffic classification, a labelled datasets from [6] were used. Whereas most network traffic classification datasets only aim to identify the type of application using the IP flow (www, dns, ftp, p2p, telnet, etc.), this dataset goes a step further by allowing the detection of specific applications such as Facebook, YouTube, Instagram, etc., from IP flow statistics.

An important feature of this dataset, in addition to a large number of network applications, is a significant number of packet fields, which are used as features for model training and packet classification.

At the first stage of research, a list of applications that could not be successfully classified based on the data of the studied dataset was filtered. The number of available packets was used as a filtering criteria. The 25 applications with the fewest packets (fewer than 500 packets in the dataset) were discarded.

Various metrics [1] will be used to evaluate the effectiveness of machine learning algorithms: accuracy, precision, recall, and F1 metric.

Accuracy means the ratio of correctly classified samples (packets) of the traffic flow to the total number of samples:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}\ , \tag{1}$$

where TR (True Positive) is the number of packets that were correctly classified to a certain application/service; TN (True Negative) is the number of packets that were correctly classified as not matching the application/service; FP (False Positive) is the number of packets that were incorrectly assigned to an application/service; FN (False Negative) is the number of packets incorrectly classified as not belonging to the application/service. In some cases where the dataset has an application/service representing the majority of sample values, the accuracy score value may not accurately reflect the performance of the classifier model. To avoid this obstacle, other performance evaluation metrics such as precision and F1 metric was also used.

Precision is a measure of the ratio of positive, correctly predicted packets in traffic to the total number of positive classification predictions:

$$Precision = \frac{TP}{TP+FP}\ . \tag{2}$$

Recall measures the ratio of actual positive, correctly predicted packets in traffic:

$$Recall = \frac{TP}{TP+FN}\ . \tag{3}$$

The F1 metric represents the average of clarity and recall:

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision+Recall}\ . \tag{4}$$

3. Results of the classifier efficiency

For the balanced dataset, the best model parameters and classification accuracy was determined. Brief results of the comparison of classification accuracy when using different algorithms are given in Table 1. Detailed description of used algorithms was describe in [7]. Table 1 shows the best values for RF, ANN, KNN algorithms with training dataset and AdaBoost, SVM algorithms with test dataset.

Table 1. Comparison of classification efficiency for machine learning algorithms

Algorithm	Accuracy	Precision	F1-score
ANN	0.991	0.992	0.991
KNN	0.933	0.933	0.933
RF	0.993	0.993	0.993
AdaBoost	0.828	0.806	0.764

According to Tab. 1, overall classification results are quite high, but these values are averaged and different apps are classified with different degrees of accuracy. The work also evaluated the classification accuracy depending on the type of application. Tab. 2 shows the results for the apps with the best and worst recognition rates.

Table 2. Results of classification of apps by the ANN algorithm

App (protocol) with the highest accuracy	value	App (protocol) with the lowest accuracy	value
IP_ICMP	1	CITRIX_ONLINE	0.83
NTP	1	UPNP	0.82
TEAMVIEWER	1	GMAIL	0.79
DNS	1	WAZE	0.77
SSH	1	TWITTER	0.77
FTP_CONTROL	1	SKYPE	0.77

Results in Tab.1 received while whole dataset was use for training and the same set of data was use for test. On practice, it is an assumption and need to use different parts of the dataset for training and testing. This will bring the situation closer to reality. Tab. 3 showed accuracy results for different ways of dividing the dataset into training and test parts.

Table 3. Results of classification of apps by the ANN algorithm

test_size	accuracy
0.1	0.699
0.2	0.708
0.3	0.688
0.4	0.676
0.5	0.685
0.6	0.658
0.7	0.649
0.8	0.609
0.9	0.565

According to Tab. 3, split 90%-10% between the training and test portions, allows getting an accuracy of 0.699. When the split is 50%-50%, it is to get an accuracy of 0.685. If the size of the training part to be less than 50%, the accuracy deteriorates significantly to 0.56-0.65.

Accordingly, for further research will be used distribution as 80%-20%.

4. Optimization of the features number

During traffic processing, the speed of the classifier determines the speed and efficiency of the network as a whole (expressed in packet processing speed) and possible packet losses rate. Therefore, the task of ensuring maximum performance of this device is relevant. Classifier performance depends directly on the number of fields to be processed. That's why in the given work the problem of optimization of number of fields used for classification has been solved.

As a basic method of classification ANN was used with different hyperparameters values. A dataset from [6] containing 82 features gives an accuracy of 0.707 (with a distribution of 80%-20%) was used.

4.1 Impact group of features on classification accuracy.

To perform this part, we will combine the classification features into groups. To do this, let's divide the initial set of 82 features into 21 groups according to their characteristics (Tab. 4).

Table 4. Distribution of features by groups

Group	**Features in group**

1	Source.IP, Source.Port, Destination.IP, Destination.Port
2	Protocol, Flow.Duration
3	Total.Fwd.Packets,Total.Backward.Packets, Total.Length.of.Fwd.Packets, Total.Length.of.Bwd.Packets
4	Fwd.Packet.Length.Max, Fwd.Packet.Length.Min, Fwd.Packet.Length.Mean, Fwd.Packet.Length.Std
5	Bwd.Packet.Length.Max, Bwd.Packet.Length.Min, Bwd.Packet.Length.Mean, Bwd.Packet.Length.Std
6	Flow.Bytes.s, Flow.Packets.s
7	Flow.IAT.Mean, Flow.IAT.Std, Flow.IAT.Max, Flow.IAT.Min
8	Fwd.IAT.Total, Fwd.IAT.Mean, Fwd.IAT.Std, Fwd.IAT.Max, Fwd.IAT.Min
9	Bwd.IAT.Total, Bwd.IAT.Mean, Bwd.IAT.Std, Bwd.IAT.Max, Bwd.IAT.Min
10	Fwd.PSH.Flags, Bwd.PSH.Flags, Fwd.URG.Flags, Bwd.URG.Flags
11	Fwd.Header.Length, Bwd.Header.Length, Fwd.Packets.s, Bwd.Packets.s
12	Min.Packet.Length, Max.Packet.Length, Packet.Length.Mean, Packet.Length.Std, Packet.Length.Variance
13	FIN.Flag.Count, SYN.Flag.Count, RST.Flag.Count, PSH.Flag.Count, ACK.Flag.Count, URG.Flag.Count, CWE.Flag.Count, ECE.Flag.Count
14	Down.Up.Ratio,
15	Average.Packet.Size, Avg.Fwd.Segment.Size, Avg.Bwd.Segment.Size
16	Fwd.Header.Length, Fwd.Avg.Bytes.Bulk, Fwd.Avg.Packets.Bulk, Fwd.Avg.Bulk.Rate
17	Bwd.Avg.Bytes.Bulk, Bwd.Avg.Packets.Bulk, Bwd.Avg.Bulk.Rate
18	Subflow.Fwd.Packets, Subflow.Fwd.Bytes, Subflow.Bwd.Packets, Subflow.Bwd.Bytes
19	Init_Win_bytes_forward,Init_Win_bytes_backward, act_data_pkt_fwd, min_seg_size_forward,
20	Active.Mean, Active.Std, Active.Max, Active.Min
21	Idle.Mean, Idle.Std, Idle.Max, Idle.Min

For example, group 1 (Tab. 4) combines ports and sender/receiver addresses, group 3 includes the characteristics of the number of packets sent and received in general, and group 7 includes typical time intervals between adjacent packets of the same type (deviation time between two packets).

After this combination, the impact of each group will be assessed by not considering it during the classification process.

The calculation results are shown in Tab. 5.

As can be seen from the table, all groups except group 1 have a minor impact and the exclusion (loss) of one group does not reduce the overall classification accuracy, but since the groups influence each other, the loss of several groups at once can lead to a significant deterioration in accuracy.

Table 5. Impact of each group of features

Group	Accuracy	Precision
1	0.624	0.865
2	0.716	0.882
3	0.715	0.886
4	0.695	0.879
5	0.710	0.893
6	0.704	0.883
7	0.700	0.875
8	0.708	0.883
9	0.706	0.887
10	0.706	0.866
11	0.705	0.880
12	0.714	0.878
13	0.697	0.882
14	0.707	0.867
15	0.707	0.886
16	0.705	0.871
17	0.701	0.873
18	0.709	0.879
19	0.686	0.881
20	0.703	0.873
21	0.705	0.878

It is also important to emphasize the importance of group 1, which contains ports and IP addresses, for the classification, and not taking it into account immediately worsens the accuracy by 0.08.

4.2 Impact of individual features on classification accuracy.

Filtering of classification features that are present in only a small number of packets (rarely found in packets) will potentially speed up the classification process and simplify system setup at the expense of a slight deterioration in accuracy. As a result of such actions, the number of features was reduced to 54 (Fig.1) with similar accuracy 0.693 (Tab. 6). This feature set marked as "medium".

```
feats_importance_large = [
    'Destination.IP', 'Destination.Port',
'Source.IP', 'Init_Win_bytes_forward',
    'min_seg_size_forward',
'Fwd.Packet.Length.Max',
'Init_Win_bytes_backward',
'Flow.IAT.Max',
    'Source.Port', 'Flow.Duration',
'Fwd.Packet.Length.Std', 'Bwd.IAT.Total',
    'Avg.Fwd.Segment.Size',
'Fwd.Packets.s', 'Fwd.IAT.Total',
'Fwd.IAT.Max',
    'Fwd.Packet.Length.Mean',
'Subflow.Fwd.Bytes', 'Flow.Bytes.s',
'Min.Packet.Length',
    'Total.Length.of.Fwd.Packets',
'Bwd.IAT.Max', 'Packet.Length.Variance',
'Bwd.Packets.s',
    'Flow.IAT.Mean', 'Fwd.Header.Length',
'act_data_pkt_fwd', 'Max.Packet.Length',
    'Flow.Packets.s', 'Flow.IAT.Std',
'Packet.Length.Std', 'Idle.Max',
    'Fwd.Header.Length.1',
'Bwd.Packet.Length.Mean', 'Bwd.IAT.Std',
'Fwd.Packet.Length.Min',
    'Bwd.Packet.Length.Std',
'Avg.Bwd.Segment.Size',
'Average.Packet.Size',
'Total.Length.of.Bwd.Packets',
    'Packet.Length.Mean', 'Fwd.IAT.Mean',
'Fwd.IAT.Std', 'Flow.IAT.Min',
    'Bwd.IAT.Mean',
'Bwd.Packet.Length.Max',
'Subflow.Fwd.Packets',
'Total.Fwd.Packets',
    'Total.Backward.Packets',
'Bwd.Header.Length', 'Subflow.Bwd.Bytes',
'Subflow.Bwd.Packets',
    'Idle.Mean', 'Fwd.IAT.Min',
'Down.Up.Ratio', 'Idle.Min']
```

Fig. 1. Large set of classification features

The next step will be multi-criteria optimization of classification features by checking the level of influence of each feature on accuracy and further reducing the number of features with an acceptable loss of accuracy. As a result, a list of 18 most important features (Fig. 2) was obtained, which ensures an accuracy of 0.638 (Tab. 6). This feature set marked as “small”.

```
feats_importance_small =
['Destination.IP', 'Destination.Port',
'Source.IP','Fwd.Packet.Length.Max',
 'Source.Port', 'Flow.Duration',
'Fwd.Packet.Length.Std', 'Bwd.IAT.Total',
 'Fwd.Packet.Length.Mean',
'Subflow.Fwd.Bytes', 'Flow.Bytes.s',
 'Bwd.IAT.Max',
'Bwd.Packets.s','Flow.Packets.s','Bwd.IAT
.Std',
 'Fwd.Packet.Length.Min','Bwd.IAT.Mean',
'Subflow.Fwd.Packets']
```

Fig. 2. Small set of classification features

In addition to the size of the dataset, the accuracy and speed of classification are affected by the settings of two hyperparameters.

First of them is ***batch size***. This is a hyperparameter, which defines the number of samples to work through before updating the internal model parameters. Second one – the ***number of epochs***. This is a hyperparameter, which defines the number times that the learning algorithm will work through the entire training dataset.

The two hyperparameters are interrelated and the results showed that reducing the batch parameter, leads to an increase in 1 epoch miscalculation time (Tab. 7):

Table 7. ANN hyperparameters

Batch size	Epoch calculation time
128	~ 6 sec
64	~ 10 sec
32	~ 17 sec

A summary result for classification speed and accuracy values are shown below (Tab. 8).

Table 8. Accuracy for different hyperparameters values and different features set size (82, 54, 18)

Batch size / epoch number	Classification time / Accuracy		
	Origin (82)	Medium (54)	Small (18)
128 / 10	70.1s / 0.700	**67.7s / 0.668**	**66.09s / 0.611**
64 / 10	108.28s / 0.711	105.1s / 0.699	98.21s / 0.646
32 / 10	189.1s / 0.723	184.7s / 0.714	174.3s / 0.655
128 / 50	**322.3s /0.773**	304.7s /0.765	281.7s /0.700
64 / 50	718.2s / 0.775	578.6s / 0.770	513.9s / 0.705
32 / 50	1166s / 0.784	897.3s / 0.772	805.6s / 0.71
128 / 100	**612.1s / 0.792**	**601.1s / 0.779**	547.6s / 0.719
64 / 100	1117.8s / 0.794	1043.7s/ 0.788	984.8s/ 0.725

As (Tab. 8) batch increases (64→128), performance improves from 105.1 sec to 67.7 sec, but accuracy deteriorates from 0.699 to 0.668 (medium feature set). Increase of epochs number (10→100) improves accuracy from 0.700 to 0.792, but performance significantly decrease from 70.1 sec to 612.1 sec (original features set).

Conclusion

The results of the comparative analysis showed that the best accuracy rates can be achieved when using ANN and RF algorithms.

During features set optimization, needed features number was decreased from 82 to 18 (reduced on 78%). Reducing the number of features allowed us to improve the classification speed by 8% for a large number of epochs (50) and by 12% for the number of epochs = 10, for batch parameter is 64.

The fastest performance can be obtained for hyperparameters batch/epoch = 128/10 by reducing the number of features to 54 (67.7 sec) and 18 (66.1 sec), however this leads to a significant drop in accuracy to 0.668 and 0.611 respectively.

In summary, if the highest performance is required, the recommended settings are batch/epoch = 128/10 and using a small (18) or medium (54) set of features.

The best accuracy is achieved by increasing the number of epochs and decreasing the batch parameter, which negatively affects the performance. Thus, the accuracy value on the test dataset 0.794 is achieved in ~1117 sec (with batch/epoch = 64/100), which may be unacceptable when it is necessary to ensure low latencies in the 5G network.

Thus, to ensure high classification accuracy at an acceptable speed, batch/epoch = 128 / 100 hyperparameter values are recommended for use, which provide an accuracy of 0.779 for the reduced set of 54 features and an accuracy of 0.792 for the original set of 82 features.

On the whole, the results (Tab. 8) show that reducing the number of features to 18 is not effective, since the gain in performance does not overlap the loss in accuracy, and almost all results of the minimal feature set (18) overlap the results of the medium (reduced) set (54) at other values of hyperparameters (batch/epoch). For example, classification time 98.21s and accuracy 0.646 for the minimum feature set (batch/epoch = 64/10) overlap with 67-70.1s / 0.67-0.7 for batch/epoch = 128 / 10 for the medium and original feature sets, respectively.

The scientific novelty of the work is to determine the parameters of machine learning algorithms that are optimal in terms of accuracy and speed to solve the problem of traffic classification in 5th and 6th generation mobile networks. In addition, scientific novelty should include an assessment of the importance of the parameters (fields) of the dataset for classification. The proposed algorithms and parameters are the first stage of multi-step processing of packets in the network, which, together with clustering, slicing and distributed processing, will improve the efficiency of the mobile communication system in general.

The practical significance of the work lies in the possibility using the specified algorithms with the proposed parameters to improve the efficiency of packet classification in the 5th and 6th generation mobile communication network.

References

1. AlZoman, R.M.; Alenazi, M.J.F. (2020), “A Comparative Study of Traffic Classification Techniques for Smart City Networks”, Sensors, No. 21, 4677, pp. 1-17.

2. Salman, O., Elhajj, I.H., Kayssi, A., Chehab, A. (2020), "A review on machine learning– based approaches for Internet traffic classification", Annals of Telecommunications, No. 75(11), pp. 673–710.

3. Alqudah, N.; Yaseen, Q. (2020), “Machine Learning for Traffic Analysis: A Review”, Procedia Computer Science, No. 170, pp. 911-916.

4. Xie, J., Yu, F.R., Huang, T., Xie, R., Liu, J., Wang, C., Liu, Y. (2018), "A survey of machine learning techniques applied to software defined networking (SDN): Research issues and challenges", IEEE Communications Surveys & Tutorials, No. 21(1), pp. 393-430.

5. Bujlow, T., Carela-Español, V., Barlet-Ros, P. (2015), "Independent comparison of popular DPI tools for traffic classification", Computer Networks, No. 76, pp.75-89.

6. IP Network Traffic Flows Labeled with 75 Apps// Retrieved from https://www.kaggle.com/jsrojas/ip-network-traffic-flows-labeled-with-87-apps.

7. Study of the effectiveness of machine learning algorithms for traffic classification in mobile networks //A.A. Astrakhantsev, L.S. Globa, A.M. Davydiuk, O.V. Sushko / Problems of telecommunications. – 2022. – No. 1 (30). - pp. 3-17.

Астраханцев А.А., Глоба Л.С., Давидюк А.М., Сушко О.В.

Налаштування параметрів алгоритмів машинного навчання для підвищення швидкості та точності класифікації трафіку

Проблематика. Розвиток телекомунікацій призвів до нових технологій мобільних мереж, і особливо 5G було запущено лише нещодавно, шосте покоління якого вже активно розробляється. Розвиток нових технологій стосується обох типів мобільного трафіку (V2V, IoT), і призводить до значного збільшення обсягу існуючих типів трафіку. Наразі існуючі методи обробки трафіку не адаптовані до таких змін, що може призвести до погіршення якості обслуговування.

Мета досліджень. Аналіз ефективності алгоритмів машинного навчання для вирішення задачі класифікації трафіку в мобільних мережах у реальному часі.

Методика реалізації. Методом вирішення проблеми підвищення ефективності обробки інформації є впровадження нових алгоритмів класифікації та пріоритезації трафіку. У зв’язку з цим у роботі поставлено актуальну задачу аналізу ефективності алгоритмів машинного навчання для вирішення задачі класифікації трафіку в мобільних мережах у режимі реального часу.

Результати досліджень. Порівняння показало найкращу точність алгоритму ANN, яка була досягнута при кількості прихованих шарів мережі, що дорівнює 200. Також результати дослідження показали, що різні алгоритми мають різну точність розпізнавання, яка не залежить від загальної кількості мережих пакетів даних в датасеті.

Висновки. У цій роботі вирішується актуальна проблема підвищення ефективності системи мобільного зв’язку за рахунок використання алгоритмів машинного навчання класифікації трафіку. У зв'язку з цим можна зробити висновок, що найбільш перспективним є застосування алгоритмів на основі ШНМ. У майбутньому слід досліджувати аспект виявлення аномалій на основі класифікації трафіку та підготовки шаблонів трафіку, оскільки цей процес дозволяє виявляти атаки на мережеву інфраструктуру та підвищувати безпеку мобільної мережі.

***Ключові слова**: класифікація трафіку; виявлення аномалій; техніки машинного навчання; штучна нейронна мережа; аналіз ознак; паттерни трафіку; мережа 5G; точність; швидкість класифікації.*