USATU
Ufa State Aviation Technical University

ITU AI/ML in 5G Challenge

ITU-ML5G-PS-038: Traffic recognition and long-term traffic forecasting based on AI algorithms and metadata

Team USATU

Speaker Alexey Kovtunenko

Problem description

Global problem statement (Saint-Petersburg University of Telecommunication)

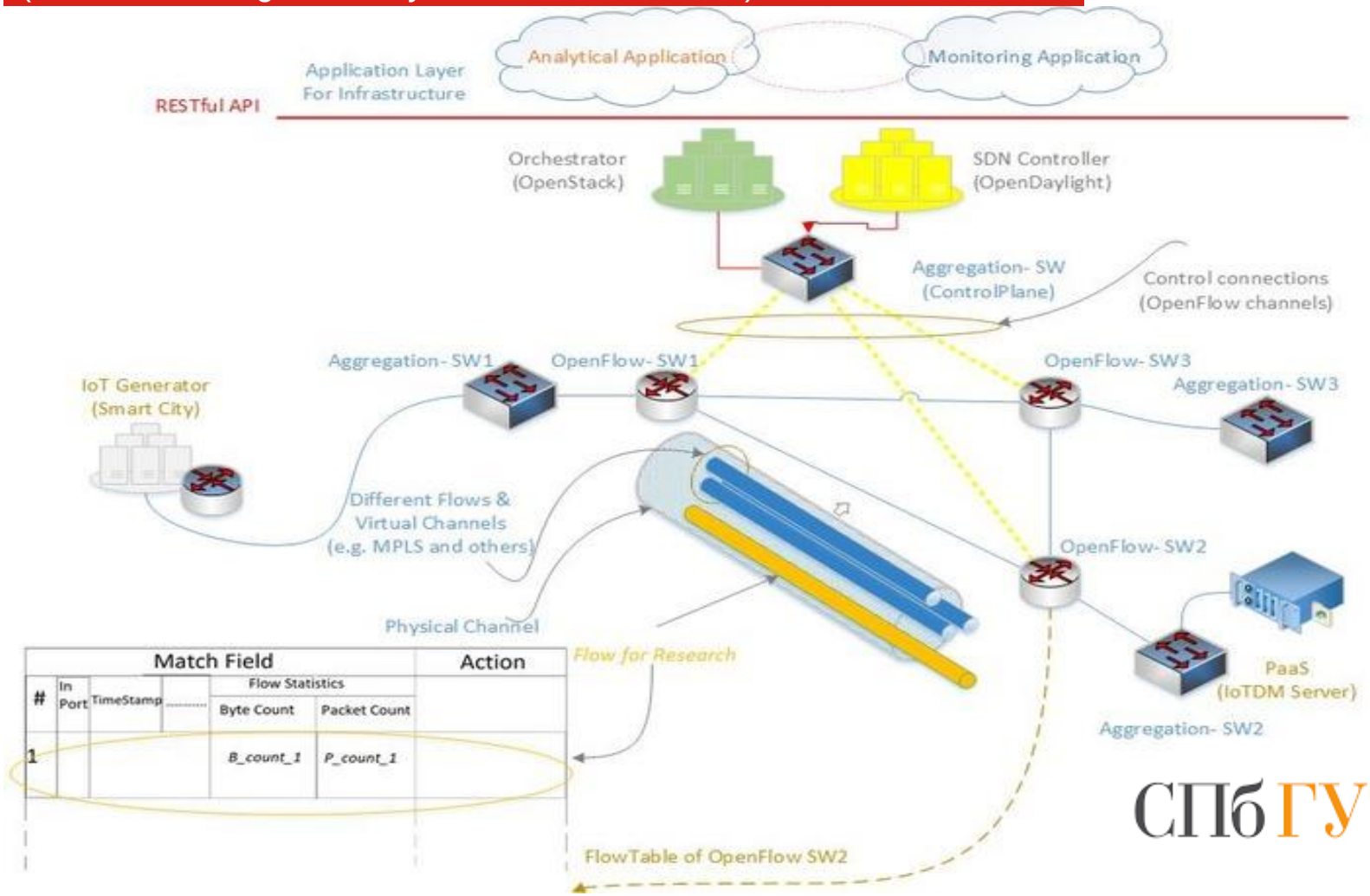
The 5G/IMT-2020 network will require robust smart algorithms to adapt network protocols and resource management for different services in different scenarios. According to the International Telecommunication Union recommendation ITU-R M.2083-0 IMT vision - “Framework and overall objectives of the future development of IMT-2020 and beyond”, infrastructure should be based on Software-Defined Networking (SDN) and Network Function Virtualization (NFV) for providing new quality level and service control possibility.

A major goal is to design greener traffic-aware 5G/IMT-2020 network (SDN/NFV) networks with efficient resource allocation to ensure good quality of service.

СПбГУТ)))

Problem description

Experimental network for generating network traffic data (IoT and Video).
(Saint-Petersburg University of Telecommunication)



Problem description

Challenge tasks

Traffic analysis is required to efficiently use network resources.

Accurate recognition and prediction of network traffic at access points enables efficient resource allocation to ensure good quality of service.

So, the tasks are:

1. To develop a traffic recognition model based on the data of passive observation of a network;

2. To develop a long-term traffic prediction model.

Task 1. Network traffic recognition

Task 1. Network traffic recognition

A survey

Let’s consider three types of network traffic classification methods:

In **port-based** methods, network traffic can be identified based on port numbers that are assigned by the Internet Assigned Numbers Authority (IANA). The categorizer examines TCP SYN packets to identify the server side of a new TCP client / server connection. Using the target TCP port number of the SYN packet, the application will be directed to the corresponding [*T. Nguyen and G. Armitage, “A survey of techniques for Internet traffic classification using machine learning”, IEEE Communications Surveys & Tutorials, Vol. 10, No. 4, fourth quarter 2008, pp 56-76.*].

Payload-based methods are based on two assumptions: (1) a third party can see the payload; (2) the payload structure of each application's package is known to the classifier. These techniques, also known as Deep Packet Inspection (DPI), inspect the contents of packets by matching them with typical network application signatures stored in a database. These methods provide more accurate results than port-based methods and are successfully used in P2P networks [*S. Sen, O.Spatscheck and D.Wang, ”Accurate, scalable in network identification of P2P traffic using application signatures”, in WWW2004, New York, ny, USA, May 2004*].

Task 1. Network traffic recognition

A survey

Finally, the **machine learning** based approach is the most modern and effective in traffic recognition.

Muhammad Shafiq et al. [Muhammad Shafiq, Xiangzhan Yu, Asif Ali Laghari, Lu Yao, N abin Kumar Karn, Foudil Abdessamia, “Network Traffic Classification Techniques and Comparative Analysis Using Machine Learning Algorithms”, 2nd IEEE International Conference on Computer and Communications, 2016, pp2451-2455] offered a network traffic classification framework for identifying/classifying unknown classes of network traffic using supervised learning methods. This framework was applied to four C4.5 machine learning algorithms, SVM, BayesNet and NaïveBayes to build a classification model using tenfold cross validation.

T. Nguyen and G. Armitage [*Zhong Fan and Ran Liu, "Investigation of Machine Learning Based Network Traffic Classification", International Symposium on Wireless Communication Systems (ISWCS) 2017, pp1-6.*] considered two machine learning algorithms: support vector machine (SVM) and K-means. The influence of features selection and model settings on the quality of classification is studied. The results were obtained using a five-fold cross-validation.

Fatih Ertam et al. [*Fatih Ertam, Ilhan Firat Kilinçer, Orhan Yaman, "Intrusion Detection in Computer Networks via Machine Learning Algorithms", International Artificial Intelligence and Data Processing Symposium (IDAP), 2017, pp 1-4*] conducted analyzes to distinguish between normal and abnormal data , from the Internet. The dataset used was KDD Cup 99 and was classified using the classifiers Naïve Bayes (NB), Bayes Net (bN), Random Forest (RF), MLP, and SMO. The best precision values were obtained using the RF, SMO and MLP classifiers. The best mean accuracy was obtained by RF and MLP classifiers.

Task 1. Network traffic recognition

Our solution. Mathematical problem statement

Let:

$x = (\text{Bytes}, \text{Packets}), \quad x \in X \subseteq \mathrm{Z}_+^2$ – a pattern of traffic at some time interval,

$y \in Y, \quad Y = \{\text{IoT}, \text{Video}\}$ – predefined types of traffic

$S \subseteq X \times Y, \; S = \{(x^1, y^1), (x^2, y^2), \ldots, (x^N, y^N)\}$ – train sample, a set of supervised observations of network, where for each traffic pattern assigned an a priori known type of traffic.

Need to find:

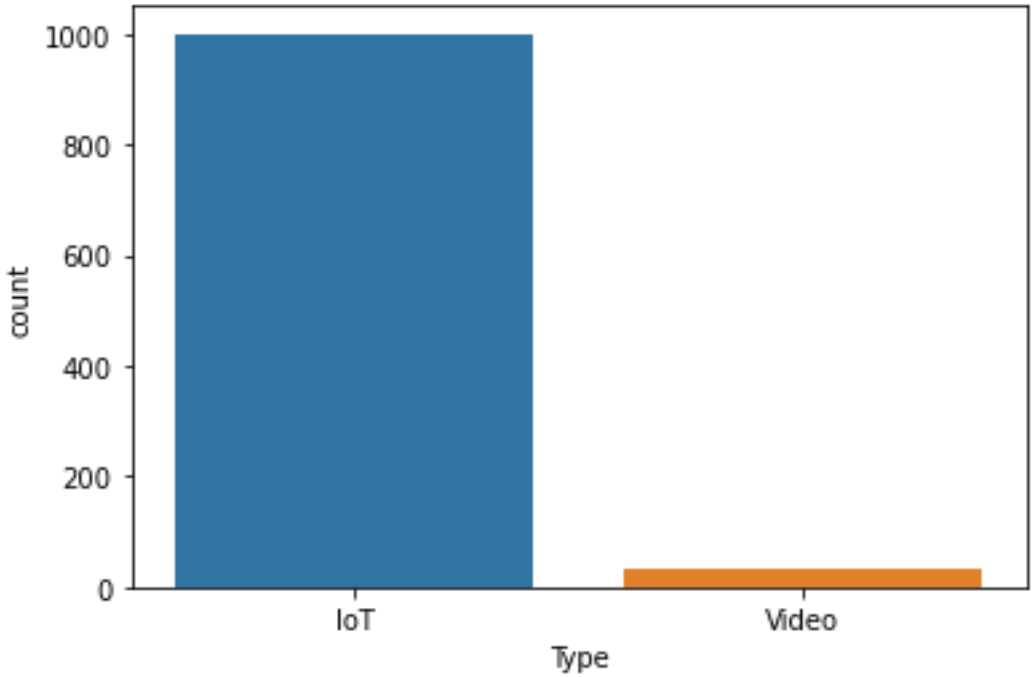
A function ***D*** that assigns a type of traffic ***y**** to each unknown pattern **x***

$D : X \to Y, \qquad y^* = D(x^*)$

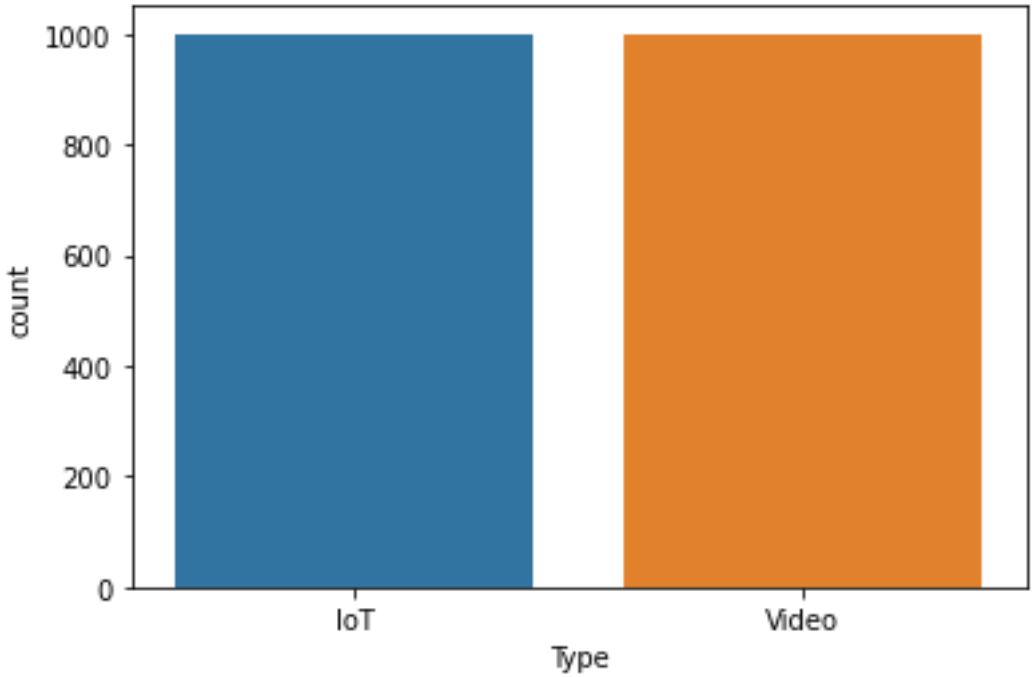
Task 1. Network traffic recognition

Our solution. Data preprocessing

There is a significant difference between amount of data of different types of traffic in given dataset. Therefore we use oversampling (python library **imblearn**) to equalize the amount of data.



Before oversampling



After oversampling

Task 1. Network traffic recognition

Our solution. Model of classification

To solve a problem of binary classification in our solution we used the method of gradient boosting

CatBoost is a high-performance open source library for gradient boosting on decision trees from the leader of the IT market in Russia – **Yandex N.V.**

It is open-source Python library, which distributes under the Apache 2.0 license. The site of the project is https://catboost.ai/

Yandex

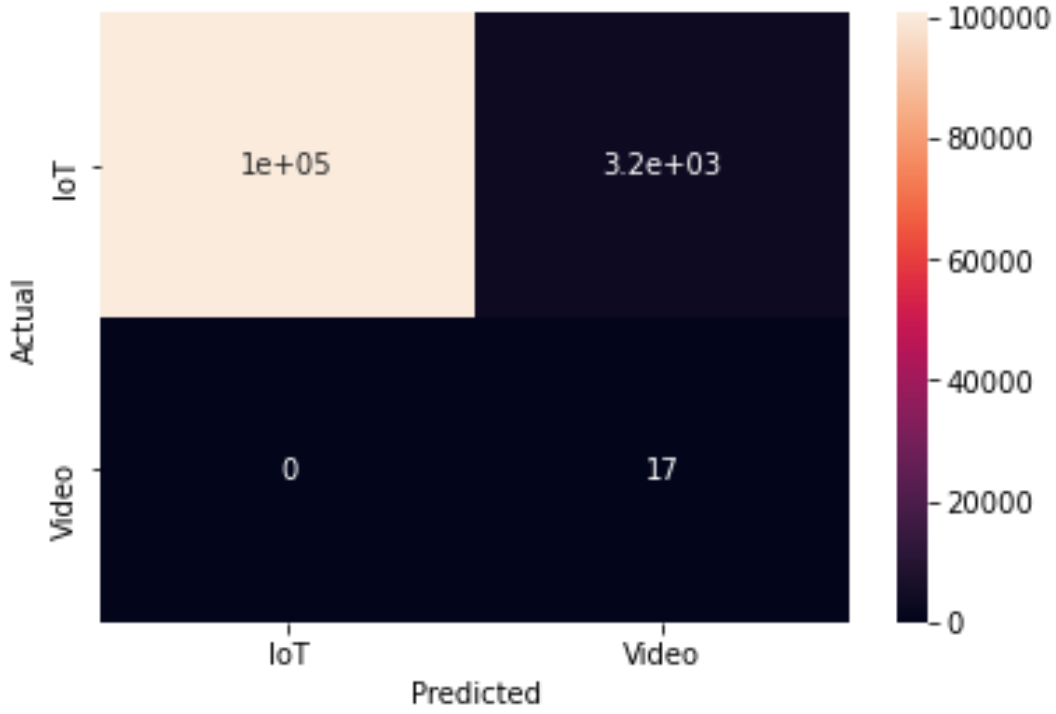
CatBoost

Task 1. Network traffic recognition

Our solution. Results

The accuracy obtained on the test dataset has reached 97 %

More clearly the results of classification can be presented in the confusion matrix.



Confusion matrix

Task 2. Network traffic prediction

Task 2. Network traffic prediction

A survey

Many works focus on network traffic prediction based on AI technologies.

A. Azzouniand et al. [A. Azzouni and G. Pujolle, "NeuTM: A neural network-based framework for traffic matrix prediction in SDN," NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium, Taipei, 2018, pp. 1-5, doi: 10.1109/NOMS.2018.8406199.] proposes a framework for network Traffic Matrix (TM) prediction based on Recurrent Neural Networks (RNN) equipped with the Long Short-Term Memory (LSTM) units.

Y. Liu and et al. [Y. Liu, H. Zheng, X. Feng and Z. Chen, "Short-term traffic flow prediction with Conv-LSTM," 2017 9th International Conference on Wireless Communications and Signal Processing (WCSP), Nanjing, 2017, pp. 1-6, doi: 10.1109/WCSP.2017.8171119.] proposes a convolutional and a recurrent module to extract both spatial and temporal information from the traffic flows

X. Cao et al. [X. Cao, Y. Zhong, Y. Zhou, J. Wang, C. Zhu and W. Zhang, "Interactive Temporal Recurrent Convolution Network for Traffic Prediction in Data Centers," in IEEE Access, vol. 6, pp. 5276-5289, 2018, doi: 10.1109/ACCESS.2017.2787696.] treats network matrices as images and use the Convolutional Neural Networks (CNN) to find the correlations among traffic exchanged between different pairs of nodes

Task 2. Network traffic prediction

Our solution. Mathematical problem statement

Let:

$Bytes[t], Packets[t], t \in T \subseteq Z_+$ – time series of amounts of bytes and packets respectively processed at some time interval

$T = T^1 \cup T^2 \cup ... \cup T^K$ – classes of equivalence that corresponds to conjunction of a set of initial time intervals (varying from 3 to 90 seconds) into larger and more regular (1 hour)

$B[\tau] = \sum_{t \in T^\tau} Bytes[t], P[\tau] = \sum_{t \in T^\tau} Packets[t], \tau \in \overline{1, K}$ – time series after discretization

Need to find:

Functions which are taking some previous values of a time series and returns a predicted value as follows (here *K* - seasonal component)

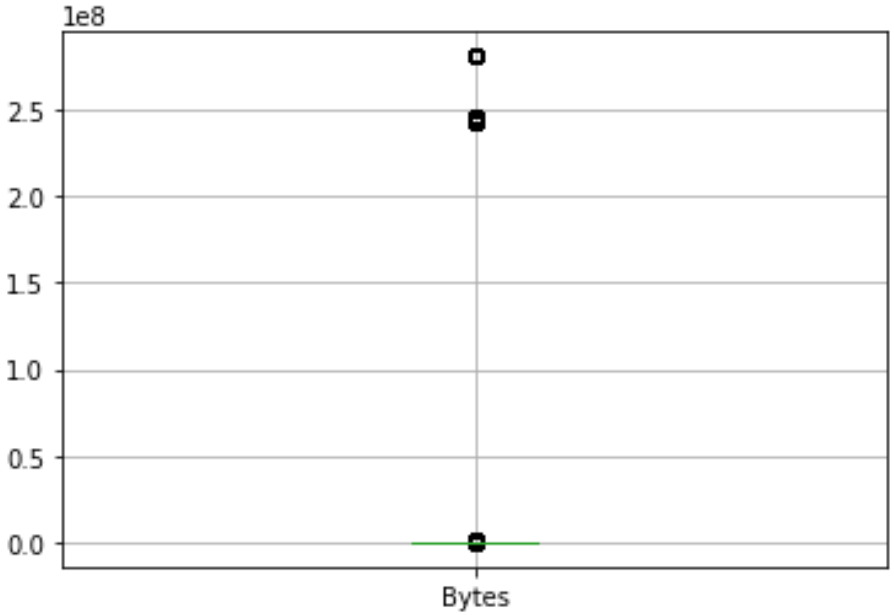
$$B[\tau] = FB(B[\tau - 1], ..., B[\tau - N], B[\tau - M])$$

$$P[\tau] = FP(P[\tau - 1], ..., P[\tau - N], P[\tau - M])$$

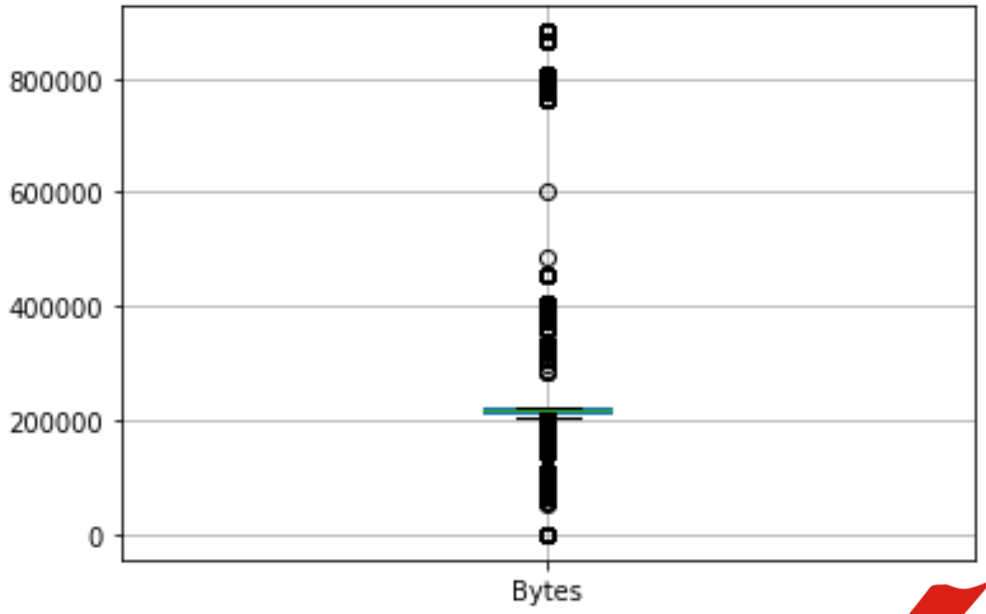
Task 2. Network traffic prediction

Our solution. Outliers filtering

Since the original data contained some outliers, the values outside the established boundaries were corrected by neighboring ones.



Before filtering

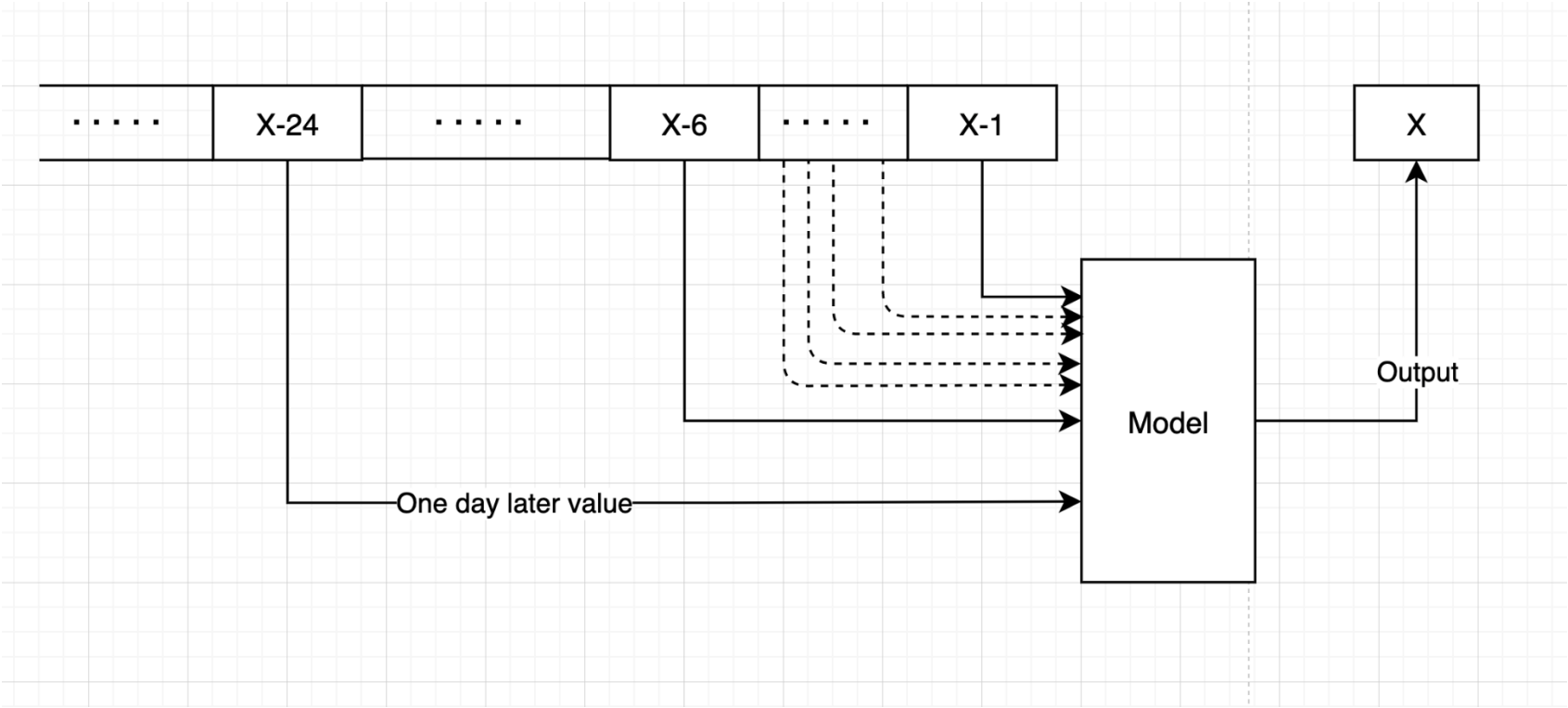


After filtering

Task 2. Network traffic prediction

Our solution. Model of predicting

It was experimentally found that the best forecasting results are achieved with the following model parameters: N=6, M=24

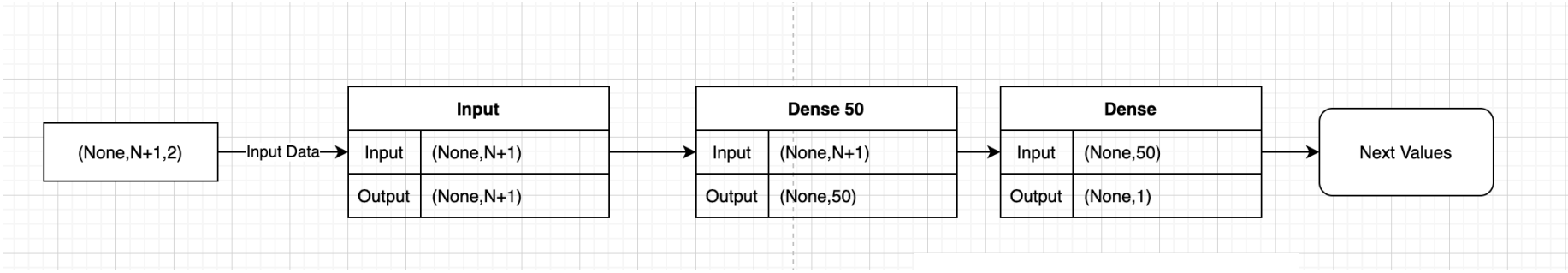


Task 2. Network traffic prediction

Our solution. MLP architecture

The model was implemented as a standalone console application on Python using a **tensorflow** library. The **pandas** library was used to load and process the raw data.

The architecture of multilayer perceptron (MLP) is follows



TensorFlow

pandas

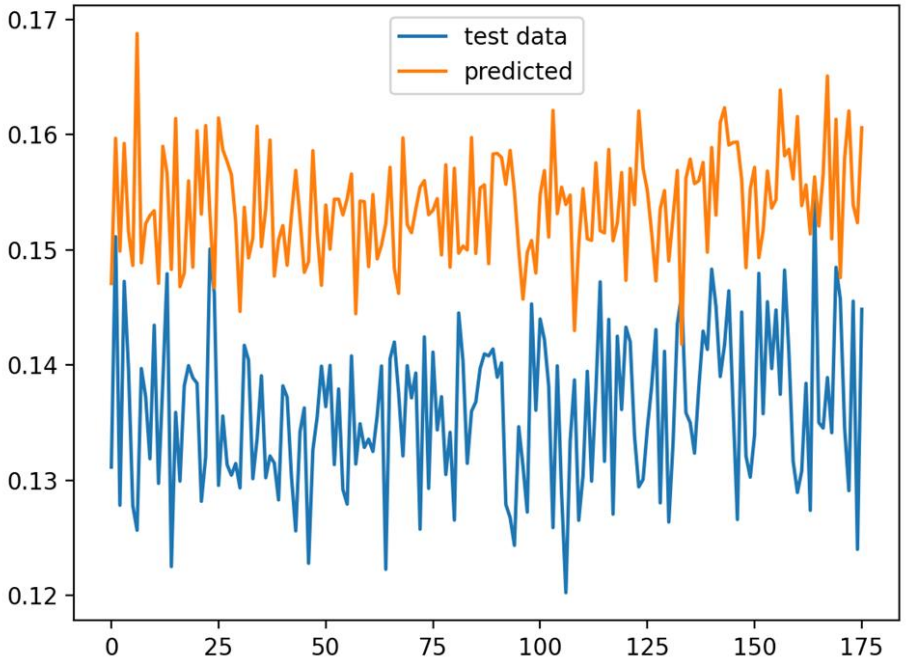
Task 2. Network traffic prediction

Our solution. Results

A separate predictive model was trained for each time series. The following training conditions were selected experimentally for each model.

Bytes

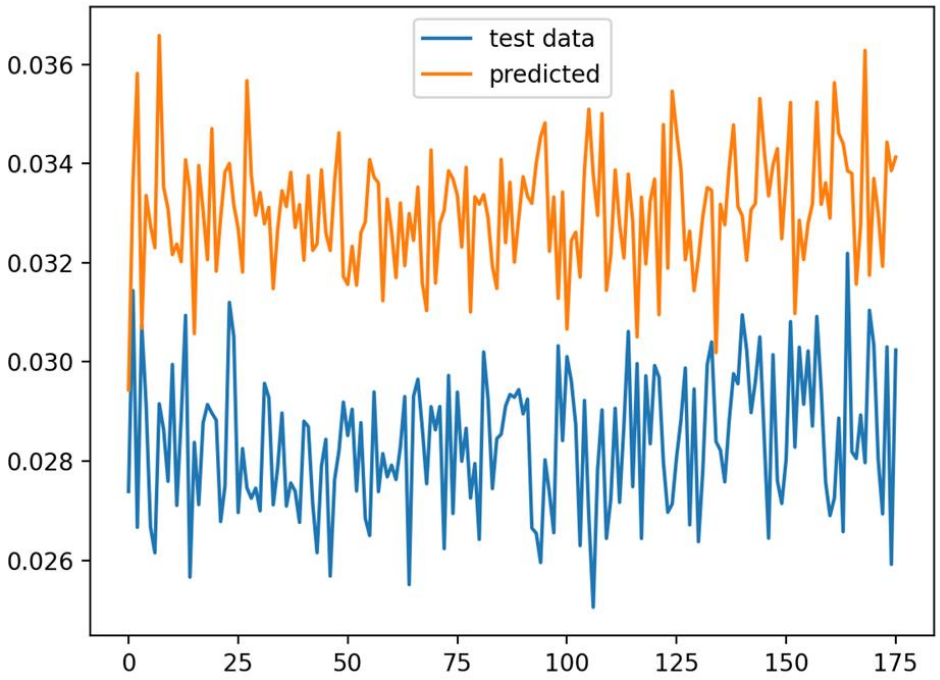
Optimizer – SGD, learning_rate = 0.005, momentun = 0.05



RMSE = 0.0195, MAPE = 13.43 %

Packets

Optimizer – AdaGrad, learning_rate = 0.05



RMSE = 0.005, MAPE = 16.781 %

CONCLUSIONS

Task1. Network traffic recognition

- **Method**: gradient boosting on decision trees
- **Used libraries**: imblearn, CatBoost
- **Achived accuracy**: 97 %

Task 2. Network traffic prediction

- **Method**: Multilayer Perceptron
- **Used libraries**: pandas, tensorflow
- **Bytes forecasting**:
- **RMSE** = 0.0195
  MAPE = 13.43 %
- **Packages forecasting**:
- **RMSE** = 0.005
- **MAPE** = 16.781 %

Our Team

Alexey Kovtunenko – Associated Professor in Department of Computer Science (askovtunenko@mail.ru) - Head ;

Denis Garaev (garaev.denchik@gmail.com); – 4-th year student;
Ainaz Hamidullin (khamidullinaynaz@mail.ru) – 3-th year student;
Artem Andrievsky (krekerr85@gmail.com)– 3-th year student;
Viktor Adadurov (viktor_1198@mail.ru). – 5-th year student