Novel AI-Based Scheme for Traffic Detection and Recognition in 5G Based Networks

Volkov Artem[1], Abdelhamied A. Ateya[1,2(✉)], Ammar Muthanna[1,3], and Andrey Koucheryavy[1]

[1] St. Petersburg State University of Telecommunication,
22 Prospekt Bolshevikov, St. Petersburg, Russia
Artem.n@5glab.ru, ammarexpress@gmail.com,
akouch@mail.ru
[2] Electronics and Communications Engineering, Zagazig University,
Zagazig, Egypt
a_ashraf@zu.edu.eg@gmail.com
[3] Peoples' Friendship University of Russia (RUDN University),
6 Miklukho-Maklaya St, Moscow 117198, Russia

Abstract. With the dramatic increase in the number of connected devices, the traffic generated by these devices puts high constraints on the design of fifth generation cellular systems (5G) and future networks. Furthermore, other requirements such as the mobility, reliability, scalability and quality of service (QoS) should be considered as well, while designing such networks. To achieve the announced requirements of the 5G systems and overcome the high traffic density problems, new technologies, such as the mobile edge computing (MEC) and software defined networking (SDN), and novel schemes, such as artificial intelligence (AI) algorithms and offloading algorithms, should be introduced. One main issue with the 5G networks is the heterogeneous traffic, since there are enormous number of applications and sub-networks. The main design challenge with the 5G network traffic is the recognition and classification of heterogeneous massive traffic, which cannot be performed by the current traditional methods. Instead, new reliable methods based on AI should be introduced. To this end, this work considers the problem of traffic recognition, controlling and management; mainly for ultra-dense 5G networks. In this paper, a novel AI algorithm is developed to detect and recognize the heterogeneous traffic at the core network. The algorithm is implemented at the control plane of the SDN network, located at the core network. The algorithm is based on the neural network. The system is simulated over a reliable environment for various considered cases and results are indicated.

Keywords: 5G/IMT-2020 · IoT · SDN · AI · RNN · Traffic recognition

1 Introduction

With the near release of 5G, it is expected that new enormous and cost efficient services will be supported [1]. Such services are expected to change the human life by introducing new paradigms such as the human to machine communications (H2M) [2].

Thus, creating an ecosystem for technical and business innovation for such systems become a great demand. There are many challenges associated with the design and realization of the 5G cellular system and announced use cases, these challenges are introduced in many literatures, such as [3, 4]. These challenges include the latency, reliability and availability. Some use cases of 5G systems requires an ultra-low end-to-end latency and ultra-high reliability and availability [5]. Tactile Internet is such example of these used cases, which require an end-to-end latency of 1 ms and a reliability of 99.99% [6]. Another main use case is the virtual reality applications that require a latency of 5 ms [7].

To enable such services, with the announced requirements, new technologies and infrastructure approaches should be involved. Another main challenge for 5G networks is the network traffic. By 2020, it is expected that the number of connected devices will be of order of billions, which puts high constraint on the design of networks that provides the coverage for such enormous number. This massive number of devices will generate a huge massive traffic that will pass over the network and load the core network [8]. For the current Internet, a significant part that always considered and monitored is the traffic generated by smart devices [9]. Thus, 5G networks should be provided with an infrastructure and novel technologies able to support and manage this enormous scalable number of traffic, with the required quality of services (QoS). One of the main novel technologies that will be deployed for 5G networks is the software defined networking (SDN). SDN enables the traffic management and control over the network separating data plane and control plane [10].

A significant number of the available Internet services and applications require the exact value of some network parameters such as latency, jitter, round trip time (RTT), and bandwidth. For such services and applications, SDN provides a vital solution that is able to extract and control such required network parameters. Furthermore, deploying SDN, especially for controlling network parameters, provides the way for implementing new services [11].

For the current networks, the QoS is measured and monitored using current existing differentiated services (DiffServ) technologies, as well as, some traffic engineering solutions [12]. A main example of these tools is the MPLS-TE, which uses the RSVP-TE protocol [13]. However, these solutions have several implementation disadvantages and cannot be used for a wide range of 5G/IMT-2020 services and applications. The main issues with these tools are the lack of dynamic control depending on the variability of the traffic type in the network, as well as the limited set of traffic classifiers.

The current Internet protocol (IP) systems deploys a field, in the packet IP header, dedicated with defining certain QoS requirements, which is referred to as the type of service (ToS) field [14]. The field of ToS can be used to indicate the requirement of high through put or request of a low latency route for the data packet associated with a certain application. Current IPv6 uses an alternate definition for the ToS fields, which is the differentiated services field (DS), which can be used for traffic classification. The DS field is defined by eight bits and contains two main sub-fields; differentiated services code point (DSCP) field and explicit congestion notification (ECN). The first six bits of the DS field define the DSCP sub-field and the last two bits define the ECN sub-field. Current existing mechanisms are efficient for ensuring QoS in applications and services such as telephony, video, television, etc. However, these mechanisms

represents an efficient solutions when using for ensuring QoS of modern and upcoming applications such as Internet of things (IoT) applications, VR applications and most of expected 5G services. The main issue with these mechanisms, that makes them inefficient for 5G networks, is the lake of prober traffic engineering solution.

Besides the massive increase of traffic, another problem associated with the traffic, in 5G networks, is the traffic heterogeneity [15]. Such example is the IoT traffic that can go along with the traffic of the Video on-demand service, Web traffic and others. Furthermore, the unpredictability of 5G networks' traffic represents a challenge; mainly for the network operators.

Due to traffic heterogeneity, the computations and calculations of changes in traffic profile become more complex and thus, current existing static computational methods become improper and lose their relevance [16]. One way to ensure a rapid response of control systems to the heterogeneous traffic growth, including periodicity, and changes in the traffic profile, is to consider novel methods for computations and control processes associated with network traffic. In order to ensure rapid response, the communication network and, accordingly, the traffic switching and routing device should have the necessary level of abstraction. This necessary level of abstraction from physical processes can be realized by introducing network softwarization; including SDN and network function virtualization (NFV) paradigms [17].

For SDN/NFV based networks, it is required to ensure a traffic detection and classification with very high precision, to ensure ultra-reliable and ultra-low latency system. This process should be hold in an intelligent way, so that the pre-registered traffic flows, no recognition tasks should be deployed; such example is the VoIP and IPTV traffic. To this end, this paper provides an intelligent scheme for managing, controlling and classifying previously unregistered traffic over SDN/NFV based networks. IoT is considered as the main application in this work, since it is one of the main use cases of the 5G.

An intelligent comprehensive analysis of mathematical classification methods is carried out, considering the characteristics of the analyzed input data. These characteristics include the traffic flows and the initial requirements. The proposed methods are developed based on the neural network. The neural network architecture is chosen in accordance to the objectives and the characteristics of the input data set. The experimental evaluation of the proposed methods has been carried out in our SDN lab. A server monitoring and performing predictive analysis of traffic flow over SDN based networks is developed to be connected to the SDN controller. Experimental results validate the proposed system and confirm the feasibility of the proposed methods in Traffic detection and classifications for IoT networks; as a main use case of 5G.

2   Proposed SDN/NFV System Structure

Recently, there are many approaches that have been developed for traffic recognition in current existing communication networks [18]. These approaches are mainly developed based on the periodic capture of traffic and the analysis of its headers. Such methods have several disadvantages that include the delaying stream, the implementation of the analytical module requires additional hardware-software solutions and the high complexity of such methods and even the additional required hardware. To overcome these

limitations, the proposed work aims to implement the analytical system at the service level of the proposed system. Figure 1 illustrates the proposed SDN based network structure.

The proposed SDN/NFV based structure ensures the system portability, the independence of the data transmission medium, and mostly the integration of data Plane. For an analytical system, all devices and flows are digital objects with a number of parameters and functionality represented by a set of methods. The level of abstraction of the proposed SDN/NFV based structure enables the implementation of an analytical system that works with traffic flows metadata. Thus, the proposed system doesn't introduce additional traffic delays, or make changes to its activity, e.g. changes in the distribution laws and intensity. Furthermore, the proposed system monitors the flow activity and provides a graphical indication of all routes among core network.
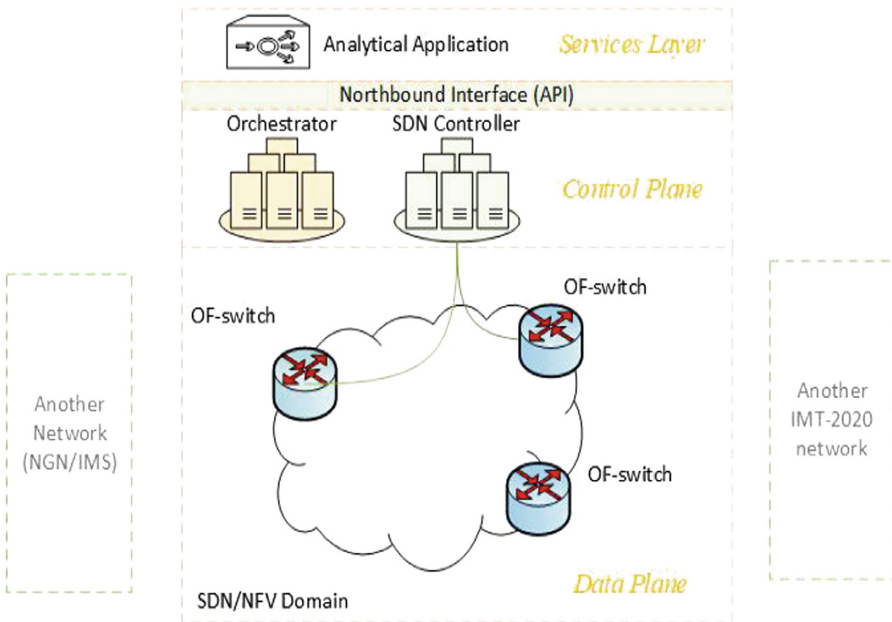

Fig. 1. Principle structure of the proposed scheme.

3 Input Data Set

The proposed SDN/NFV based system structure enables the implementation of the analytical system in the form of an application at the service level of the SDN/NFV network. The data received at the core network, via the northern interface of the SDN controller, represented the input data to the proposed analytical application. Since, one of the main objectives of this work is to examine the analytical activity of flows and to

detect the flows at the data plane level, the analytical system has the ability to request flow tables data from all switches connected to the SDN control. The flow tables contain two main fields; the match field and the action filed. Analyzing data of both match and action fields, enables the development of a meta-model of flows. The general structure of the flow table is introduced in Fig. 2.

The circled data, in Fig. 2, is the data that should be used to form the meta-model for the network flow. These data includes two main counters; Byte Count and Packet Count. In addition to these counters, flow table contains another main parameter, which is the Time Stamp. Time Stamp enables the instantaneous calculation of ByteCount-delta and PacketCount-delta. One of the main features of these data is that, based on the "Byte Count" and "Packet Count" counters, it is impossible to accurately determine the exact packet length in the packet stream. Accordingly, based on these data it is impossible to accurately determine the length of each packet registered in the stream during a time interval ΔT.
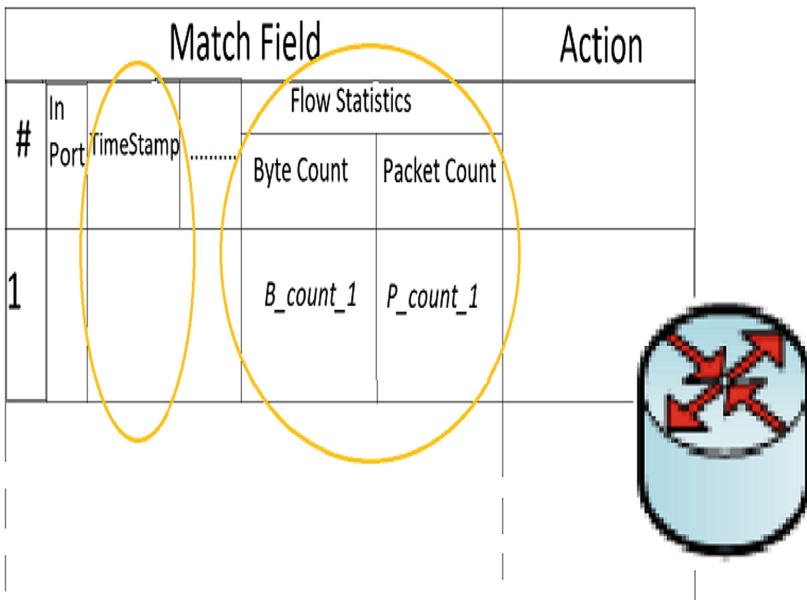

Fig. 2. Main fields of flow table.

However, for an arbitrary period of time ΔT, having samples of [Byte Count], [Packet Count], [Time Stamp] values, it is possible to create a data set with an established data structure, where each sample displays the instantaneous value [ByteCount-delta] and [PacketCount-delta]. The values of [Byte Count], [Packet Count] and [Time Stamp] are generated by instantaneous requests received by the SDN controller, via the RESTful application programming interface (API).

The data structure is formed on the basis of the $DataSet_{RQ}$ queries with "raw" data as in (1). Formula (2) is used to convert the data structure to the required $DataSet_{ML}$ format, while the instantaneous values are evaluated based on (3).

Let PacketCount-delta - PC_{delta},

ByteCount-delta - BC_{delta},

TimeStamp-deltas - TS = 1 [sec.] = const, Then:

$$DataSet_{RQ} = \begin{matrix} [TimeStamp] & [ByteCount] & [PacketCount] \\ TimeStamp_{11} & ByteCount_{12} & PacketCount_{13} \\ TimeStamp_{21} & ByteCount_{22} & PacketCount_{23} \\ \ldots & \ldots & \ldots \\ TimeStamp_{N1} & ByteCount_{N2} & PacketCount_{N3} \end{matrix} \tag{1}$$

$$\begin{cases} BC\_delta_{N2} = ByteCount_{N2} - ByteCount_{(N-1)2}, if \ N \geq 1 \\ PC\_delta_{N2} = PacketCount_{N2} - PacketCount_{(N-1)2}, if \ N \geq 1 \end{cases} \tag{2}$$

$$DataSet_{ML} = \begin{matrix} [TimeStamp] & [ByteCount] & [PacketCount] \\ TS & BC_{delta_{12}} & PC_{delta_{13}} \\ TS & BC_{delta_{22}} & PC_{delta_{23}} \\ \ldots & \ldots & \ldots \\ TS & BC_{delta_{N2}} & PC_{delta_{N3}} \end{matrix} \tag{3}$$

The total values of these parameters for a period of time are calculated as following:

$$ByteCount_{\Delta T} = \sum_{N=1}^{N=\Delta T/TS} BC_{delta_{N2}} \tag{4}$$

$$PacketCount_{\Delta T} = \sum_{N=1}^{N=\Delta T/TS} PC_{delta_{N2}} \tag{5}$$

4   Proposed RNN

The artificial neural network is one of the main AI approaches that is widely used to solve various real-life problems. Tasks such as speech recognition, text recognition, and prediction of complex models are, recently, solved with the help of neural networks; at high performance. One of the typical tasks introduced to the neural networks is the classification. There are many neural network based classification methods; one of the most efficient methods among these methods is the descriptions of objects using features [19]. Each object is characterized by a set of numeric or non-numeric features. However, for some types of data, open features do not provide classification accuracy, for example, the color of image points or the digital sound signal; this is because of these data contains hidden features. Another AI paradigm is the Deep Learning, which is a set of machine learning algorithms that attempt to simulate high-level abstractions in data and extract hidden features from input data.

Therefore, considering the object characteristics as traffic and its features as numerical - statistical series, a neural network with Deep Learning can be used for

detection and classifications. A recurrent neural network (RNN) with Long Short-Term Memory (LSTM) is chosen as the neural network architecture as introduced in Fig. 3. The LSTM network is a universal network, such that with a sufficient number of network elements, it can perform any calculation that a conventional computer can do. This requires an appropriate matrix of weights that can be considered as a program.
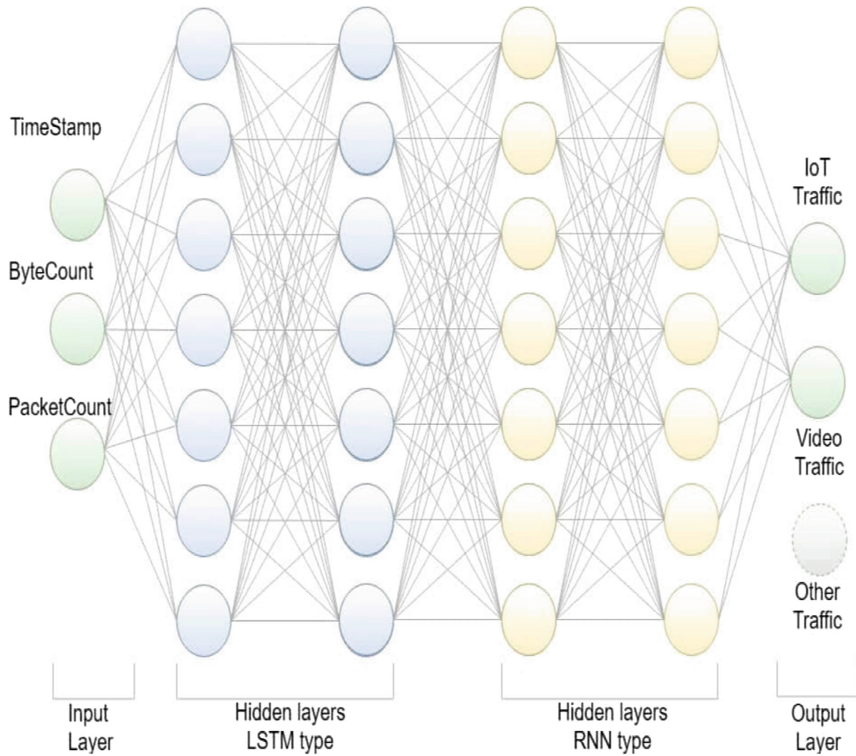

Fig. 3. The architecture of the proposed RNN.

LSTM part of the neural network enables the identification of patterns of influence of the considered samples, taking into account the correlations between samples' values. Such example is the periodicity in IoT traffic and the character self-similarity. Since, the selected neural network architecture implements the principle of data training, it is required to create training labeled data sets and save the state of the trained network. In order to train the neural network, the input $DataSet_{ML}$ is converted to a $DataSet_{MLtrain}$, by adding a new data column; each line of which contains the statistical sampling identifier. Accordingly, in order to recognize a wide range of traffic types, this training data set needs to be expanded by marking the appropriate statistical sample with a traffic label. Thus, the structure of the training $DataSet_{MLtrain}$, for IoT network, can be defined as following:

$$DataSet_{ML} = \begin{array}{ccccc} [TypeOfTraffic] & [TimeStamp] & [ByteCount] & [PacketCount] \\ IoT & TS & BC_{delta_{12}} & PC_{delta_{13}} \\ IoT & TS & BC_{delta_{22}} & PC_{delta_{23}} \\ \ldots & \ldots & \ldots & \ldots \\ Video & TS & BC_{delta_{N2}} & PC_{delta_{N3}} \\ others & TS & BC_{delta_{(N+1)2}} & PC_{delta_{(N+1)3}} \end{array}$$

The LSTM network receives a fixed length data entry, so the data is divided into 200-line segments or 10 s. Activity tags are converted into unitary code. Data is divided into training and practice sets in the ratio of 8:2.

The network model contains two fully connected RNN layers and two fully connected LSTM layers; each of which contains seven hidden nodes. The hyperparameters of learning are as following:

- Optimizer: Adam,
- Number of epochs: 40,
- Number of samples per iteration: 1024, and
- Learning speed: 0.0025

The resulting neural network architecture is presented in Fig. 3. As introduced in the figure, the input layer of the neural network consists of three neurons; the input of which is the corresponding data from the DataSet$_{MLtrain}$. The three input neurons are connected to the first hidden layer, consisting of LSTM neurons. The second hidden layer consists of neurons of the LSTM type and it links both first and third layers. The third and fourth hidden layers are built based on the fully connected neurons of the RNN. The output neurons are two neurons displaying the results of the neural network. Accordingly, for a DataSet$_{MLtrain}$ with a large number of types of tagged traffic statistics, the number of output neurons will be increased. For the proposed RNN, there is a certain defined threshold, which is used for defining number of neurons associated with each hidden layer; mainly introduced for optimizing neural network training.

5   Network Model

For performance evaluation of the proposed system, a network model is introduced. This model represents a real model for a smart city network, since it comprises large heterogeneous amount of traffic. In order to generate IoT traffic, IoT traffic generators are used. These IoT traffic generators are used with the Internet of Things Data Management system (IoTDM), which is built according to the specifications of the international community of OneM2 M. These traffic generators have been developed as part of a previous research work, the results of which are introduced in details in [9].

The analytical system is referred to as the network application. A part of this network application is the module for detecting the IoT flows. This module has been developed using Python programming language as a WEB server operating on the MVC model. This server runs on the Northbound API controller and the orchestrator. The used SDN controller is OpenDaylight Beryllium SR4 and the OpenStack is used as

the infrastructure orchestrator. The Data Plane layer is built on based of Mikrotik switches that support the OpenFlow protocol.

To train the data sets for the proposed neural network, the following assumptions are introduced.

1. Any traffic is turned off and the network is modified so that it could be accurately detect flows from smart city traffic in the switch flow table.
2. Next, application is given an exact flow number in the flow table and the application starts saving data.

A training data set for video on demand traffic has also been formed while software has been already used to generate video on demand traffic. After the trained data set is created, the developed neural network is activated and the state of the neural network is preserved after its successful training. At the next stage, the network application must get the ability to load the stored neural network to identify any flow in the network. To that end a testing data set is created, after specifying the flow number in the flow table. The end-to-end structure of the developed performance evaluation model is introduced in Fig. 4.

6   Performance Evaluation

For performance evaluation of the proposed RNN, the previous introduced network model is used. The proposed IoT traffic generator models the operation of 160 IoT devices. Each device generates an HTTP packet and transmits the values of three sensors in the packet body.

As a result, a $DataSet_{MLtrain}$ has been formed, which merges two labeled data sets, i.e. IoT and Video. Furthermore, the $DataSet_{MLtrain}$ is fed to the input of the proposed neural network. A scatter plot of the obtained $DataSet_{MLtrain}$ is introduced in Fig. 5. In Fig. 5, it is visually possible to identify several areas, i.e. clusters, of points' distribution, in the area where the corresponding values of the average value of the packet length in the stream prevails. For example, for the IoT flow, in most cases (dense area), the average packet length is approximately 400 bytes, and the parameter may vary. There is also a spike in the average packet length of 470 bytes.

During training, the parameter of accuracy is monitored, which is represented by the function inverse to the error function in recognition of activity. The other monitored parameter is the loss, which is the cross entropy determining a slightly near-predictable distribution to the true one. Figure 6 illustrates the training progress that indicates these parameters.
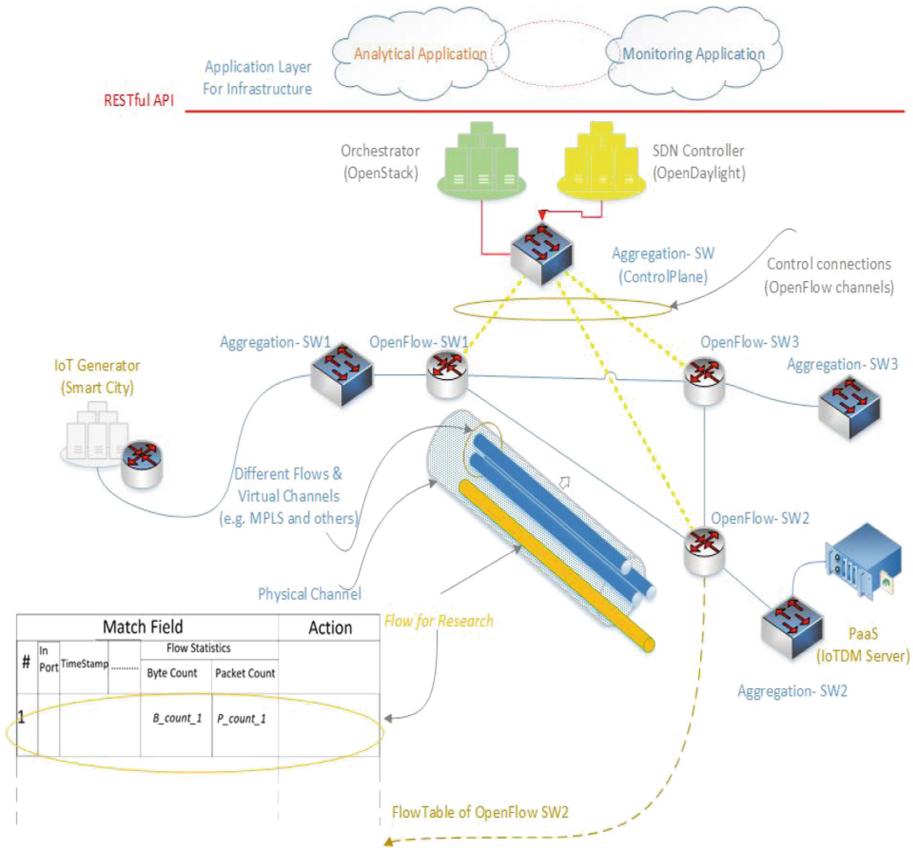
Fig. 4. End-to-end system structure of the considered network model.

Furthermore, Fig. 7 introduces the confusion matrix of the training neural network. Figure 6 mainly highlights the learning process of the network, in which it is clearly indicated that the network has successfully completed the learning process. As a result of the learning process of the neural network and testing its operation on test Data Sets, in the trained state, the developed neural network can identify the IoT flow with a probability of 99.97% and respectively the video stream. The confusion matrix indicates that the network fails identifying the Video flow; it is identified as an IoT flow, only one time. Furthermore, for IoT traffic it identifies all flows successfully. Thus, the proposed RNN architecture represents an optimal solution for traffic detection and classifications.
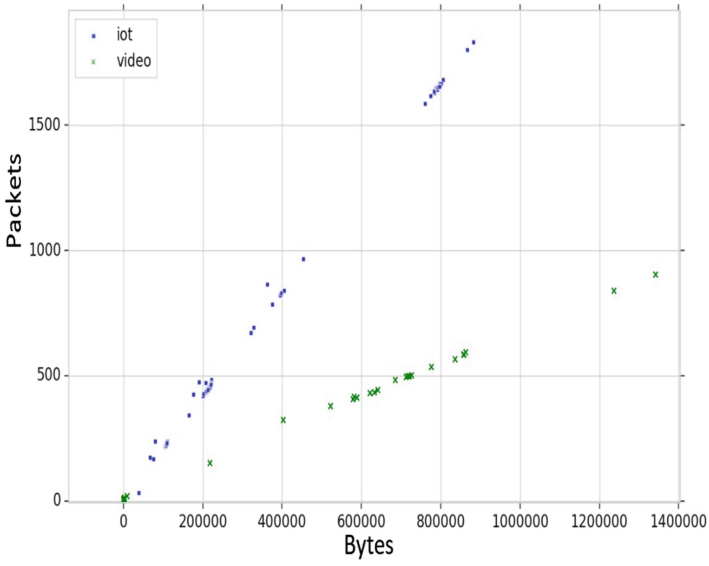
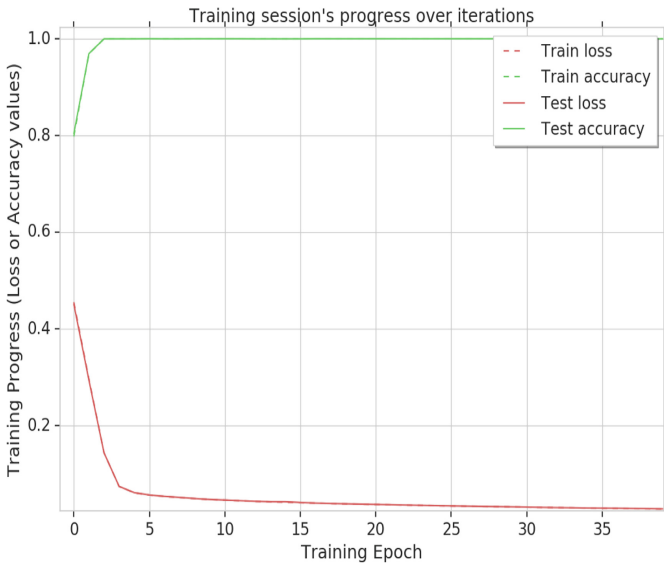**Fig. 5.** Scatter plot of the obtained DataSet$_{MLtrain}$.
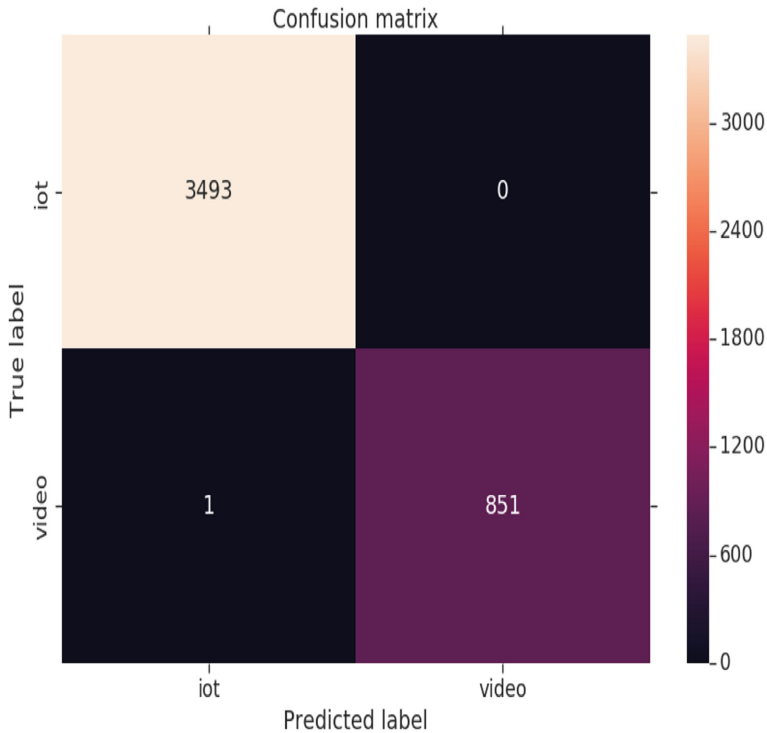

Fig. 6. Machine learning process.

Fig. 7. Confusion matrix.

7   Conclusion

This paper introduces a method for traffic detection and classification of 5G/IMT-2020 networks. The analytical module that implements the functionality of the proposed system is built at the service level of 5G networks, which does not interfere with the data plane level; this represents the novelty of the proposed work. Ultimately, this approach allows solving most of the traffic flow identification problems at the data plane level. The proposed method is based on artificial intelligence algorithms. The paper presents RNN architecture for detecting and classifying traffic in 5G networks. The proposed method can be scaled in two dimensions; of the first is the parallel study of multiple flows from different switches and the second is the uniquely identifications of flows, e.g., various IoT services, not just one Smart City service and video stream. A network model based on IoT traffic and smart city has been introduced to provide an input data set for the proposed RNN. The experimental evaluation of the proposed system has been introduced, which indicates that the proposed methods efficiently detect and classify the heterogeneous traffic.

References

1. Feasibility Study on New Services and Markets Technology Enablers, document 3GPP TR 22.891, ver. 14.2.0, September 2016
2. Ateya, A.A., Muthanna, A., Makolkina, M., Koucheryavy, A.: Study of 5G services standardization: specifications and requirements. In: 2018 10th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), pp. 1–6. IEEE, November 2018
3. 5G PPP Architecture Working Group white paper, "View on 5G Architecture," July 2016
4. Jiang, D., Liu, G.: An overview of 5G requirements. In: Xiang, W., Zheng, K., Shen, X. (eds.) 5G Mobile Communications, pp. 3–26. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-34208-5_1
5. Tudzarov, A., Gelev, S.: Requirements for next generation business transformation and their implementation in 5G architecture. Int. J. Comput. Appl. **162**(2), 31–35 (2017)
6. Ateya, A. Vybornova, A., Kirichek, R., Koucheryavy, A.: Multilevel cloud based tactile internet system. In: IEEE-ICACT2017 International Conference, Korea, February 2017
7. Koucheryavy, A., Makolkina, M., Paramonov, A.: Applications of augmented reality traffic and quality requirements study and modeling. In: Vishnevskiy, V.M., Samouylov, K.E., Kozyrev, D.V. (eds.) DCCN 2016. CCIS, vol. 678, pp. 241–252. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-51917-3_22
8. Ateya, A.A., Muthanna, A., Gudkova, I., Abuarqoub, A., Vybornova, A., Koucheryavy, A.: Development of intelligent core network for tactile internet and future smart systems. J. Sens. Actuator Netw. **7**(1), 1 (2018)
9. Volkov, A., Khakimov, A., Muthanna, A., Kirichek, R., Vladyko, A., Koucheryavy, A.: Interaction of the IoT traffic generated by a smart city segment with SDN core network. In: Koucheryavy, Y., Mamatas, L., Matta, I., Ometov, A., Papadimitriou, P. (eds.) WWIC 2017. LNCS, vol. 10372, pp. 115–126. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-61382-6_10
10. Zaidi, Z., Friderikos, V., Yousaf, Z., Fletcher, S., Dohler, M., Aghvami, H.: Will SDN be part of 5G?. IEEE Commun. Surv. Tutorials **20**(4), 3220–3258 (2018)
11. Yousaf, F.Z., Bredel, M., Schaller, S., Schneider, F.: NFV and SDN-Key Technology Enablers for 5G Networks, arXiv preprint arXiv: 1806.07316 (2018)
12. BinSahaq, A., Sheltami, T., Salah, K.: A survey on autonomic provisioning and management of QoS in SDN networks. IEEE Access **7**, 2 (2019)
13. Attar, H., et al.: A mathematical model for managing the distribution of information flows for MPLS-TE networks under critical conditions. Commun. Netw. **10**(02), 31 (2018)
14. Haider, F., Chaudary, M.H., Naveed, M.S., Asif, M.: IPv6 QoS for multimedia applications: a performance analysis. In: Proceedings of the 2019 8th International Conference on Software and Computer Applications, pp. 501–504. ACM (2019)
15. An, J., Yang, K., Wu, J., Ye, N., Guo, S., Liao, Z.: Achieving sustainable ultra-dense heterogeneous networks for 5G. IEEE Commun. Mag. **55**(12), 84–90 (2017)
16. Qiu, T., Chen, N., Li, K., Qiao, D., Fu, Z.: Heterogeneous ad hoc networks: architectures, advances and challenges. Ad Hoc Netw. **55**, 143–152 (2017)
17. Haleplidis, E., Salim, J.H., Denazis, S., Koufopavlou, O.: Towards a network abstraction model for SDN. J. Netw. Syst. Manage. **23**(2), 309–327 (2015)
18. Velan, P., Čermák, M., Čeleda, P., Drašar, M.: A survey of methods for encrypted traffic classification and analysis. Int. J. Netw. Manage. **25**(5), 355–374 (2015)
19. Yang, Y., Luo, H., Xu, H., Wu, F.: Towards real-time traffic sign detection and classification. IEEE Trans. Intell. Transp. Syst. **17**(7), 2022–2031 (2015)