Traffic Classification using Deep Learning Approach for End-to-End Slice Management in 5G/B5G

Noor Abdalkarem Mohammedali
Centre for Smart and Advanced Technologies (CAST)
University of Northampton
Northampton, UK
noor.mohammedali@northampton.ac.uk

Triantafyllos Kanakis
Centre for Smart and Advanced Technologies (CAST)
University of Northampton
Northampton, UK

Ali Al-Sherbaz
School of Computing and Engineering
University of Gloucestershire
Gloucestershire, UK
aalsherbaz@glos.ac.uk

Michael Opoku Agyeman
Centre for Smart and Advanced Technologies (CAST)
University of Northampton
Northampton, UK

***Abstract*—Network slicing is a key role in future networks. 5G networks are intended to meet the different service demands of an application offered to users. The 5G architecture is used to match the requirement of the Quality of Service (QoS) by addressing different scenarios in terms of latency, scalability and throughput with different service types. Using machine learning with network slicing allows network operators to create multiple virtual networks or slices on the same physical infrastructure. These slices are independent and customized. Precisely, it will manage dynamically according to the requirements defined between the network operators and the users.**

For this research, multi-machine learning algorithms are used to train our model, classify the traffic and predict accurate slice type for each user. After the traffic classification, we compared and analysed the performance of various machine learning algorithms in terms of learning percentage, accuracy, precision and F1 score.

***Index Terms*—5G, Machine Learning, Network Slicing, services, NFV, SDN, Deep Learning, End-to-End, classification**

I. Introduction

Future networks are intended to meet the different service requirements of an applications. For this, 5G systems plan to increase the resources offered to users by means of new radio access technologies and bands in the spectrum, such as massive multiple inputs, multiple outputs (MIMO). 5G networks will not use the monolithic architecture networks, since such architecture is unable to meet the different service of the most diverse use cases. Such use cases include, for example, vehicles autonomous, intelligent hospitals and wearable technologies. For that, End-to-end slices isolation introduced in [1] to isolate the physical infrastructure to virtual networks.

Network virtualization and Softwarization is a trend research area with many open research issues waiting to be addressed. Amongst the main areas that need further investigation are mobility management, resource allocation and slice management [2]. In the mobility management, the network manages based on the customer requirement called the Service Level Agreement (SLA) between the service provider and the customer (organization). In addition, there is another agreement between the organization and the end user called Business Level Agreement (BLA). In BLA, the user has limited service and specific criteria. On the other hand, in slice management, the slice (per flow, per service, etc.) is managed dynamically based on the agreement between the customer and the service provider [3].

There are three types of service that Network Slicing support eMBB (enhanced Mobile Broadband), URLLC (ultra-Reliable Low Latency Communications) and MIoT (Massive IoT) [12]. So, each type of service is served by a different slice. In eMBB, this service can handle consumer mobile broadband, including a High-Quality Video stream and Fast large file transfers. In URLLC, this service requires high reliability and low latency used in Vehicle-to-Vehicle (V2V), Vehicle-to-Infrastructure (V2I), and the device-to-device (D2D) communications concepts [4]. In MIoT, this service handles a large number of IoT devices efficiently and cost-effectively [5]. Within these services, each application has different priority level: high priority or low priority [6]. In addition, managing the priority of the slice that the user service belongs to and the priority of the service that the end user requires in each slice is remained an open issue. Finally, there are two types of slice priority: inter-slice priority and intra-slice priority [7].

To support the most diverse use cases, the 5G networks aim to integrate the concept of Network Slicing in its architecture. Although Network Slicing is a new technology, only recently has it been introduced in wireless networks. Virtualization of the network infrastructure can be performed, for example, from a hypervisor. The concept of Network Slicing in 5G networks is realized thanks to the of network software technologies, such as Network Virtualization Functions (NFV) and Software Defined Networks (SDN). These are intended to bring the benefits of software to networks. The flexibility, modularity are examples of such benefits [8].

Network Slicing can be done in both access networks: Radio Access Networks (RAN) and at the 5G core. Computer resources, spectrum, as well as network functions can be virtualized and distributed to the different slices belonging to the network. Mobile operators have been having difficulty accommodating in their network infrastructure traffic of its users, since there was an expressive increase of this traffic in the recent years due to the popularization of smartphones and tablets, as well as the video demand [9].

In Section II, we summarize the background research to develop deep learning approach for E2E slice management. In Section III, we propose Random Forest algorithm and classification model for the dataset. In Section IV, we explain and evaluate the performance of the model. Finally, in Section V, we conclude this paper.

II. Related work

Mobility management in 5G reviewed in [10] to match the demands of the Quality of Service (QoS) for different service types, such as: eMBB, mMTC and URLLC. Their solution tackled different scenarios in terms of throughput, latency and scalability.

Graph theory proposed in [11] to manage the inter-slice and saving the slice in queue based on the probability event. Moreover, predictive solutions is provided in their work to evaluate the network and improve the QoS in future networks.

The authors in [12] proposed a framework based on a model of proportional allocation of resources with the objective of realizing the concept of network slicing in multi-tenant networks. The proposed framework allows for dynamic sharing between the slices, increasing overall tenant performance. Besides that, such framework is generic and is not based on specific cellular technology.

Inter and intra slice management and network functions placement is mentioned in [13]. The authors highlighted that the network slicing approach needs a significant efforts when it is used with next generation mobile network.

Hybrid learning algorithm proposed in [14] to classify the slices in 5G system after optimizing the weight function using GS-DHOA, but their work needs further improvement to solve complex problems. Their dataset includes device types, reliability, duration, delay, jitter, bandwidth, speed and modulation type.

In [15], support vector machine (SVM) algorithm proposed to select slice features according to the IoT services. Unsupervised algorithm is used for grouping similar applications in one cluster called sub-slice clustering. On the other hand, K-means has some limitations in terms of latency.

In [16], K-means is used for clustering three slice types after identified 22 features from their dataset. Furthermore, with their classification results, they achieved high accuracy for all selected algorithms.

In [17], the author proposed a survey on how we can manage the slice using machine learning. Reinforcement learning and Neural networks algorithms summarized in their work.

In [18], they used regression tree for the classification and prediction. Their model accuracy in more than 90% for K-Nearest Neighbor and cosion KNN.

In [19], Machine learning model proposed for throughput prediction. They predicted the throughput for non-standard 5G networks, and their accuracy achieved 84% and 93%. On other hand, Chi-square method used for nonlinear dataset and their accuracy result was 99% for 25 features as explained in [20].

Transfer learning is proposed in [21] to accelerate the 5G resource allocation using a deep reinforcement learning algorithm on the radio access network. On the other hand, the deep reinforcement learning is implemented [22] to examine the effectiveness of the slice resources in 5G networks in terms of utilization and delay when considering the relationship between the node and the impact of surrounding nodes' resources.

For our model, we will use deep learning algorithm, and convolutional neural network (ConvNet) for the classification. In the model, we have eight features as input, four layer as a hidden layer and the 5G slice types as output; we will discuss more about it in the next section. This work is a continuous work for the research in [23]. We had been survived the statues-of-art for all the techniques that could be used to implement the slices for the 5G systems.

III. The Proposed Model

In this section, we will used filtered and cleared dataset which is used for training and testing the model. Their filtered file did not contain a header for each column, so based on the explanation in their research we identified how they prepared their model. After operating the model and separating the training model into many phases, the accuracy for the prediction of slices increased and the amount of loses decreased. Their model did not deal with real-time prediction or with other slice-management scenarios.

Number of the features in the dataset was 8 and we add new column for the location of the user to check the performance of the network when the user with in the home network and the visited network.

In this paper, we were trained 5G model using Deep Learning algorithm to train and predict a slice types for a device based on the information that calculated from previous connections. the dataset treated with a high-level API called (Keras) to build and train them in TensorFlow. Moreover, this machine learning model would be based on supervised learning because the dataset was big and structured. As a result, we were used a Random Forest algorithm and ConvNet for the traffic classification. furthermore, we were utilized various parameters to determine the network slicing: slice type, bandwidth, throughput, latency, equipment type, mobility, reliability, isolation and power.

The goals for this model: A- Select a slice for a device. B- Select enough resources for the slice based on the traffic prediction. The dataset features description:

1) Device types: This column contains a group of devices: Mobile, VR, Healthcare, IoT, Gaming and Industry 4.0.
2) Device Category: 5G and LTE (4, 5, up to 20).
3) Technology Supported in terms of LTE, IoT, LTE-M, NB-IoT and 5G.
4) Duration: Connection duration in day and time.
5) Guaranteed Bit Rate (GBR) and Non-GBR.
6) Packet Loss Rate: Reliability for sending and receiving packets. For example: 0.000001 and 0.01.
7) Packet delay budget: Latency. For example: 10ms, 50ms and 75ms.
8) Roaming: Home and visited slice network.
9) Slice Type: eMBB, URLLC, mIoT and V2V.

IV. Result & Discussion

From the dataset features, we notice that it is a collection of Heterogeneous wireless networks (HetNets). In this case, some challenges appear with HetNets as explained in [24]. First, we need to know the HetNets patterns to obtain an accurate mathematical model to enhance the performance of the HetNets and network slicing. Second, how can we meet the QoS demands of the Slice based on SLA? Finally, how can we assign the spectrum dynamically to ensure the SLA for the slices?

The dataset held 66K rows and 9 columns. The rows stood for the 5G slice parameters, while the 8 columns held the Key Performance Indicators (KPI), and the last column corresponding to the slice types of the 5G networks. Each feature in the dataset had a label. The KPI parameters symbolized input and the last column symbolized the 5G slice types as output.

For the traffic classification, we would use popular machine learning algorithms to find the best algorithm that's fits the dataset, such as:

1) Naïve Bayes: The classification in this algorithm use Bayes' theorem to count the probability of a data that belongs to a particular type.
2) Support Vector Machine(SVM): It is work with liner and non-liner classifiers. SVM makes predication based on the support vectors.
3) Neural network (NN): It is work with non-linear algorithm.
4) Gradient Boosted Tree (GBT): with this algorithm trains multiple trees to reduce the cost function.
5) Random Forest (RF): It is choose a number of trees to do the classification. This algorithm will predict the finial class based on the majority votes.

In the training stage, all the data (X_train) sent to the machine learning algorithm to learn and come up with the correct answer for (y_train). The algorithm uses the following formula for that: f(X_train) = y_train. For the prediction, the algorithm took the output (y_train) and applied it to another formula y_train = f(X_train). In this case, the system will be able to predict the 5G slices for any new input that's contains all the 5G slice parameters.

In the validation stage, the cross-validation technique is used to evaluate the model and check if it is working dynamically by choosing a correct slice type when we add new data.

Afterwards, the model evaluation will be done by adding X_test to check if the model could predict the correct y_test.

Accuracy is used to view the relationship between the number of correct predictions and the total number of predictions. The accuracy formula is given below to evaluate the performance of the predictive model. Where, the accuracy formula contains True Positives (TP), True Negatives (TN), False Positives (FP) and False Negatives (FN) [25].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

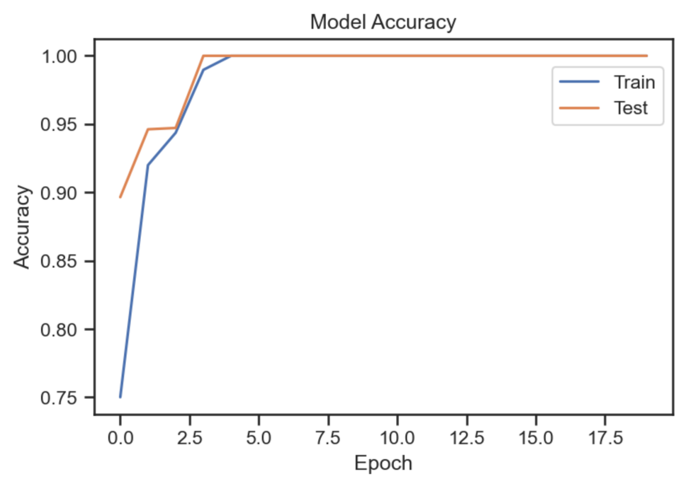

Fig. 1. Accuracy

Precision: The number of the TP, when it predicts yes, how often is it correct? [25]. The precision formula is given by:

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

Recall or sensitivity of the model when it is detected positive values. The Recall formula is given by [25]:

	precision	recall	f1-score	support
H_Net	1.00	1.00	1.00	141
V_Net	1.00	1.00	1.00	127
accuracy			1.00	268
macro avg	1.00	1.00	1.00	268
weighted avg	1.00	1.00	1.00	268

Fig. 2. Measures the precision, recall and F1

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

F1 score: This score represents the average of the precision and recall. The F1 formula is given by [25]:

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * TP}{2 * TP + FP + FN} \quad (4)$$

After we applied formulas on the dataset, the result shown in the Figure 2

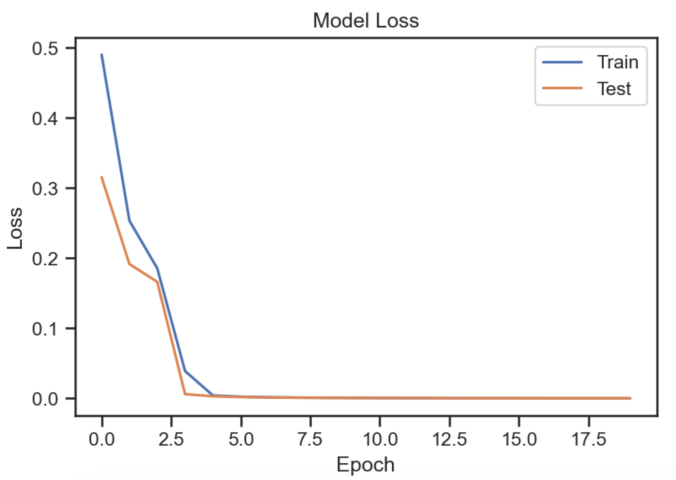

Fig. 3. Loss

Random Forest was implemented using Python with 10 trees. Small tree shown in Figure 4.

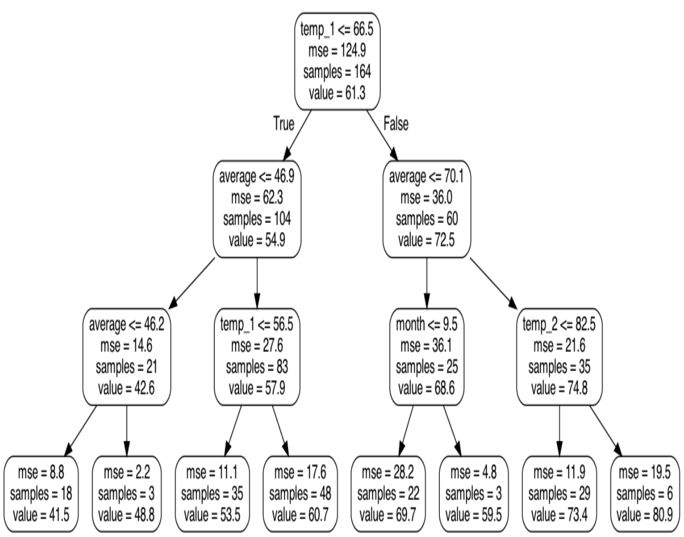

Fig. 4. Small Tree from the Decision Tree of the Random Forest

Confusion matrix is the relationship between the predicted values and the actual values. In this section, confusion matrix is used to check the performance of the classified model after the prediction.

The matrices from the RF algorithm as shown in Figure 5 and 6. First figure for the 5G slice types and the second figure for the user location in (home or visiting network).

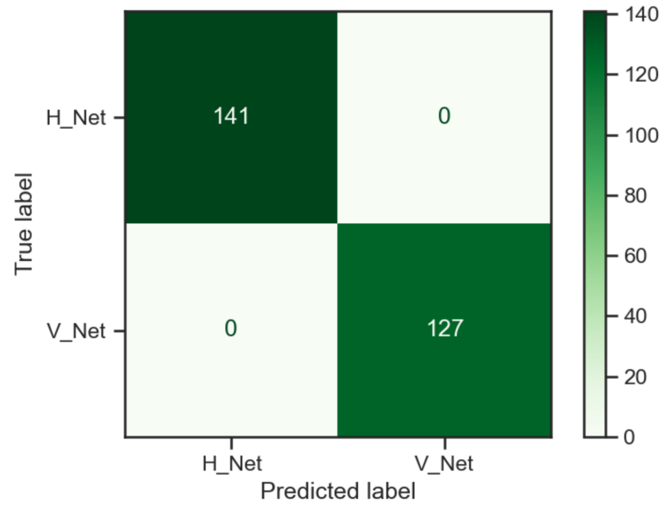

Fig. 5. Confusion matrix for home and visited network

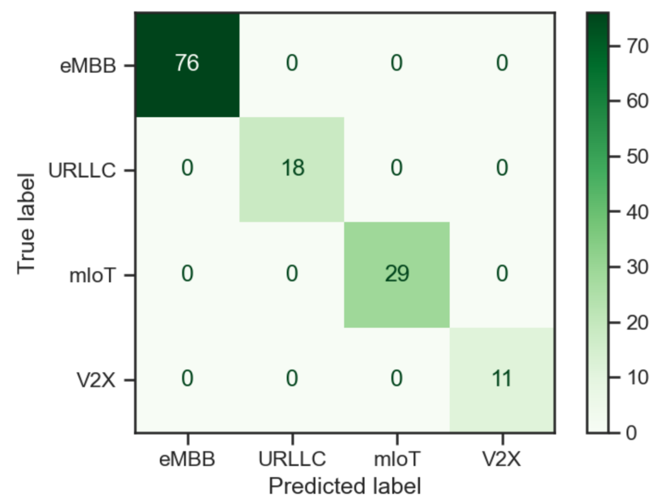

Fig. 6. Confusion matrix for the slice type

Confusion matrix for the classification model. The number of the correct predictions shown and for the models that not fitted with our model the number of wrong predictions appears in the matrices.

Confusion matrix terms:

1) True Positives (TP): we predicted yes; they have 5G slice type.
2) True Negatives (TN): we predicted no; they have 5G slice type.
3) False Positives (FP): we predicted yes; but they don't have a 5G slice type.
4) False Negatives (FN): We predicted no, but they do have a 5G slice type.

All the dataset features were trained with Trees, Support Vec-

TABLE I
CLASSIFY THE ML MODEL TO FIT THE PROTOTYPE

Class	Classification Model	Accuracy	Prediction Speed (obs/sec)	Train Time (sec)
Trees	Fine	100	400000	1202.9
	Medium	100	1300000	980.04
	Coarse	100	740000	2.5542
SVM	Linear	100	410000	27.097
	Quadratic	100	240000	52.742
	Cubic	100	230000	81.596
	Fine Gaussian	100	5600	680.26
	Medium Gaussian	100	150000	709.69
	Coarse Gaussian	100	160000	739.51
KNN	Fine	100	17000	757.04
	Medium	100	9600	785.85
	Coarse	100	3300	868.16
	Cosine	100	2800	972.43
	Cubic	100	10000	999.02
	Weighted	100	9800	1027.2
Ensemble	Boosted Trees	50.3	930000	1029
	Bagged Trees	100	160000	1042.1
	Subspace Discriminant	85.6	61000	1053.2
	Subspace KNN	98.8	920	1341.2
	RUSBoosted Trees	50.3	810000	1342
Neural Network	Narrow	100	650000	1347.5
	Medium	100	680000	1353.3
	Wide	100	480000	1367.3
	Bilayered	100	680000	1374.7
	Trilayered	100	600000	1385.2
Kernel	SVM	94.7	22000	1544
	Logistic Regression	92.7	22000	1616.5

tor Machine (SVM), K-Nearest Neighbor (KNN), Ensemble, Neural Network and Kernel as shown in Table I. Each model had accuracy, prediction's speed and time for the training. In terms of accuracy, the majority of classification model had 100% except for boosted trees, kernel, subspace discriminant, and RUSBoosted trees.

5G services will be classified and predicted using supervised machine learning algorithms as shown in Table 1. Decision Tree, Support Vector Machine (SVM), K-Nearest Neighbors (KNN) and Kernel. MATLAB was used for the compared and contracted. K-Folds was applied using K = 5 for the cross-validation technique.

For the classification, MATLAB is used to check the best algorithms for the dataset. Table I shows different classification models. With compare and contrast, we can choose the best model for our data to do the classification.

Trees, SVM, KNN and neural network had best accuracy which is 100%, while Subspace Discriminant had 85% and the less accuracy was Boosted and RUSBoosted Trees as shown in the Table I.

We had four slice types need to be classified based on the selected features. The confusion matrices for the algorithms as is shown in Figure 7, 8 and 9.

Kernel was used as a numeric predictors for Naïve Bayes. In addition, the training time for this model was 1834.5 sec. After the training, the accuracy for the validation was 94.5%, the prediction speed was 150 obs/sec and the total cost was

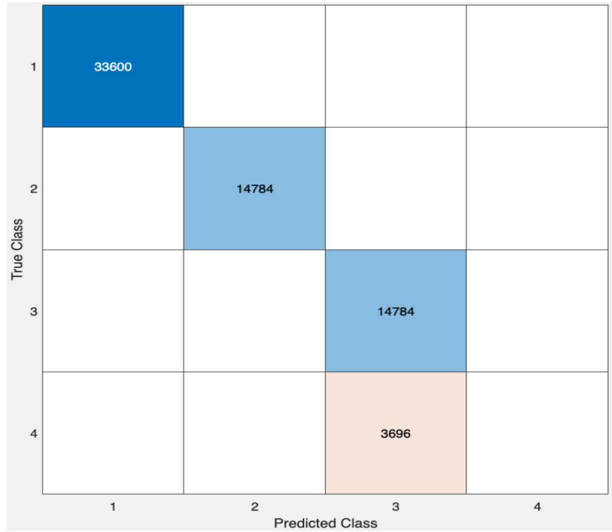

Fig. 7. Naïve Bayes

3696. The confusion matrix for the slice types prediction using Naïve Bayes as is shown in Figure 7.

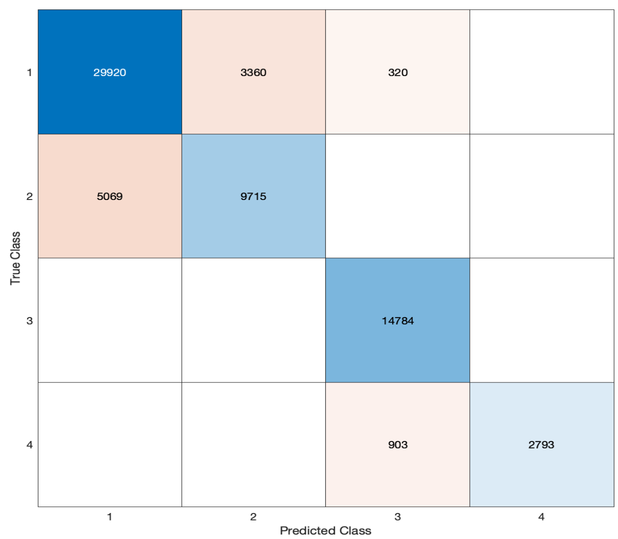

Fig. 8. Ensemble

The accuracy for the Ensemble model was 85.6 % and the training time for this model was 1053.2 sec with total cost was 9652. Furthermore, learner number was 30 and the confusion matrix for the slice types using Subspace Discriminant as is shown in Figure 8.

After the SVM kernal model trained, the total cost was 3534 and the confusion matrix for the slice types using SVM Kernel as is shown in Figure 9.

V. CONCLUSIONS & FUTURE WORK

In this paper, we compared different algorithms for our model with the selected features. The dataset held 66K rows

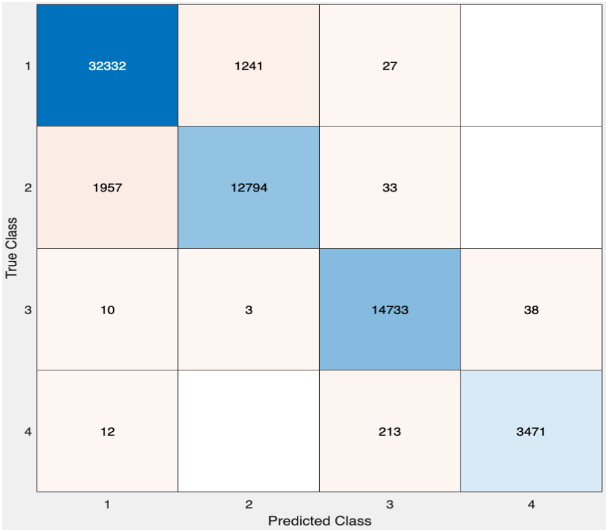

Fig. 9. SVM Kernel

and 9 columns as discussed before to choose a good algorithm to fit the model. Each feature in the dataset had a label. The KPI parameters symbolized input and the last column symbolized the 5G slice types as output. The accuracy result in this model was 100% which indicate that this model will give a good result for the real-time traffic classification. For future work, we will deal with real-time traffic and create a robust machine learning model to deal with our dataset. The real-time data will be collected from the End-to-End 5G network in our lab. A deep reinforcement learning algorithm will be used for slice prediction and classification to improve the accuracy and recall rates in future networks.

REFERENCES

[1] Z. Kotulski, T. W. Nowak, M. Sepczuk, and M. A. Tunia, "5g networks: Types of isolation and their parameters in ran and cn slices," *Computer Networks*, vol. 171, p. 107135, 2020.

[2] S. Cherrared, S. Imadali, E. Fabre, G. Gössler, and I. G. B. Yahia, "A survey of fault management in network virtualization environments: Challenges and solutions," *IEEE Transactions on Network and Service Management*, vol. 16, no. 4, pp. 1537–1551, 2019.

[3] A. Papageorgiou, A. Fernández-Fernández, L. Ochoa-Aday, M. S. Peláez, and M. S. Siddiqui, "Sla management procedures in 5g slicing-based systems," in *2020 European Conference on Networks and Communications (EuCNC)*. IEEE, 2020, pp. 7–11.

[4] Y.-B. Lin, C.-C. Tseng, and M.-H. Wang, "Effects of transport network slicing on 5g applications," *Future Internet*, vol. 13, no. 3, p. 69, 2021.

[5] S. Wijethilaka and M. Liyanage, "Survey on network slicing for internet of things realization in 5g networks," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 2, pp. 957–994, 2021.

[6] S. A. AlQahtani and A. S. Altamrah, "Supporting qos requirements provisions on 5g network slices using an efficient priority-based polling technique," *Wireless Networks*, vol. 25, no. 7, pp. 3825–3838, 2019.

[7] A. Othman and N. A. Nayan, "Efficient admission control and resource allocation mechanisms for public safety communications over 5g network slice," *Telecommunication Systems*, vol. 72, no. 4, pp. 595–607, 2019.

[8] H. Zhang, N. Liu, X. Chu, K. Long, A.-H. Aghvami, and V. C. Leung, "Network slicing based 5g and future mobile networks: mobility, resource management, and challenges," *IEEE communications magazine*, vol. 55, no. 8, pp. 138–145, 2017.

[9] M. Peng, Y. Li, Z. Zhao, and C. Wang, "System architecture and key technologies for 5g heterogeneous cloud radio access networks," *IEEE network*, vol. 29, no. 2, pp. 6–14, 2015.

[10] N. Akkari and N. Dimitriou, "Mobility management solutions for 5g networks: Architecture and services," *Computer Networks*, vol. 169, p. 107082, 2020.

[11] B. Bordel, R. Alcarria, D. Sánchez-de Rivera, and Á. Sánchez, "An inter-slice management solution for future virtualization-based 5g systems," in *International Conference on Advanced Information Networking and Applications*. Springer, 2019, pp. 1059–1070.

[12] P. Caballero, A. Banchs, G. De Veciana, and X. Costa-Pérez, "Network slicing games: Enabling customization in multi-tenant networks," in *IEEE INFOCOM 2017-IEEE Conference on Computer Communications*. IEEE, 2017, pp. 1–9.

[13] A. A. Barakabitze, A. Ahmad, R. Mijumbi, and A. Hines, "5g network slicing using sdn and nfv: A survey of taxonomy, architectures and future challenges," *Computer Networks*, vol. 167, p. 106984, 2020.

[14] M. H. Abidi, H. Alkhalefah, K. Moiduddin, M. Alazab, M. K. Mohammed, W. Ameen, and T. R. Gadekallu, "Optimal 5g network slicing using machine learning and deep learning concepts," *Computer Standards & Interfaces*, vol. 76, p. 103518, 2021.

[15] S. K. Singh, M. M. Salim, J. Cha, Y. Pan, and J. H. Park, "Machine learning-based network sub-slicing framework in a sustainable 5g environment," *Sustainability*, vol. 12, no. 15, p. 6250, 2020.

[16] L.-V. Le, B.-S. P. Lin, L.-P. Tung, and D. Sinh, "Sdn/nfv, machine learning, and big data driven network slicing for 5g," in *2018 IEEE 5G World Forum (5GWF)*. IEEE, 2018, pp. 20–25.

[17] B. Han and H. D. Schotten, "Machine learning for network slicing resource management: a comprehensive survey," *arXiv preprint arXiv:2001.07974*, 2020.

[18] N. Salhab, R. Rahim, R. Langar, and R. Boutaba, "Machine learning based resource orchestration for 5g network slices," in *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[19] D. Minovski, N. Ogren, C. Ahlund, and K. Mitra, "Throughput prediction using machine learning in lte and 5g networks," *IEEE Transactions on Mobile Computing*, 2021.

[20] F. Xie, D. Wei, and Z. Wang, "Traffic analysis for 5g network slice based on machine learning," *EURASIP Journal on Wireless Communications and Networking*, vol. 2021, no. 1, pp. 1–15, 2021.

[21] A. M. Nagib, H. Abou-Zeid, and H. S. Hassanein, "Transfer learning-based accelerated deep reinforcement learning for 5g ran slicing," in *2021 IEEE 46th Conference on Local Computer Networks (LCN)*. IEEE, 2021, pp. 249–256.

[22] N. He, S. Yang, F. Li, and X. Chen, "Intimacy-based resource allocation for network slicing in 5g via deep reinforcement learning," *IEEE Network*, vol. 35, no. 6, pp. 111–118, 2021.

[23] N. A. Mohammedali, T. Kanakis, M. O. Agyeman, and A. Al-Sherbaz, "A survey of mobility management as a service in real-time inter/intra slice control," *IEEE Access*, vol. 9, pp. 62 533–62 552, 2021.

[24] H. Zhang, S. Xu, S. Zhang, and Z. Jiang, "Slicing framework for service level agreement guarantee in heterogeneous networks—a deep reinforcement learning approach," *IEEE Wireless Communications Letters*, vol. 11, no. 1, pp. 193–197, 2021.

[25] C. A. U. Hassan, M. S. Khan, and M. A. Shah, "Comparison of machine learning algorithms in data classification," in *2018 24th International Conference on Automation and Computing (ICAC)*. IEEE, 2018, pp. 1–6.