Multivariate Spatio-temporal Cellular Traffic Prediction with Handover Based Clustering

Evren Tuna[1,2] and Alkan Soysal[2,3]

[1]5G Research & Development, Turkcell Technology, Istanbul 34880, Turkey
[2]Department of Electrical and Electronics Engineering, Bahcesehir University, Istanbul 34353, Turkey
[3]Calhoun Honors Discovery Program, Virginia Tech, Blacksburg, VA 24061, USA

2022 56th Annual Conference on Information Sciences and Systems (CISS) | 978-1-6654-1796-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/CISS53076.2022.9751165

***Abstract*—We consider an RNN-based traffic volume prediction, which is a critical problem for network slice management and resource allocation in slicing-enabled next generation cellular networks. We propose to use a novel cost function that takes SLA violations into account. Our approach is multivariate and spatio-temporal in three aspects. First, we consider the effects of several other RAN features in a cell besides the traffic volume. Second, we introduce feature vectors based on peak hours of the day and days of the week. Third, we introduce feature vectors based on incoming handover statistics from the neighboring cells. Our results show about 60% improvement over MAE-based univariate LSTM models and about 20% improvement over SLA-based univariate models.**

I. Introduction

Cellular service providers need to transform their existing network design and operation approaches in order to meet various service demands of different verticals in 5G and beyond networks. Service-specific design and integration of key enablers such as network slicing, private networks and edge computing have increased the complexity of networks. Machine Learning (ML) methods can deal with this complexity and make the next generation cellular networks more robust, predictive, autonomous, and reliable [1]. The works in [2], [3] provide detailed surveys for studies on various deep learning applications in different domains of mobile networks by considering both today's challenges and future perspectives.

Traditionally, time-series problems similar to traffic volume prediction were handled by versions of Autoregressive Integrated Moving Average (ARIMA) models [4]. For example, [5] divides the time series mobile traffic volume data into a regular component and a random component. They predict the regular component using ARIMA and report an error of 30% on the regular component. They argue that random component cannot be predicted. However, recent years have seen a large number of works that apply ML methods to traffic volume prediction and perform significantly better than ARIMA models.

ML methods for cellular traffic prediction are centered around Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) based models. While CNNs are argued to capture the spatial correlation, RNNs are used for time-series data due to their cyclical architecture in which current output is related to the previous outputs. Traditional RNNs are insufficient for problems involving long-term dependency. For this reason, Long Short-Term Memory (LSTM) based RNN models that have more complex structure has been proposed. Although the learning capacity of LSTM has become superior using this complex structure, the computational load has increased due to additional parameters. For this reason, Gated Recurrent Unit (GRU) based RNN models with a simpler structure and a smaller number of parameters were proposed.

In this study, we focus on sector and carrier-based downlink (DL) traffic volume prediction of base stations (eNodeBs) using LSTM and GRU models. Our dataset is collected from a real and live LTE network operating in a highly dense urban area. In addition to DL traffic volume, the dataset includes several other Radio Access Network (RAN) features as well. Moreover, we introduce new Boolean feature sets to emphasize the busy hours of the day and the days of the week. Finally, in order to manage the spatio-temporal effect and the interaction of cells among themselves, we propose a new handover-based feature.

Network traffic data has spatial and temporal dependencies due to interactions between eNodeBs and daily/weekly trends in mobile data usage, respectively. In the literature, a line of research (e.g., [6]–[10]) considers CNN-based models to exploit spatial dependency. Of these, [6]–[8] utilize 3D-CNN structure that is borrowed from video processing applications. They assume that inputs at a given time are in a matrix form. Each entry of the input matrix is the aggregated traffic of eNodeBs that are in a corresponding square grid area. We believe that correlation structures between color intensity levels of neighboring pixels in an image and between data traffic levels of neighboring eNodeB sectors are significantly different. Data volume at a neighboring cell might affect the data volume at the intended cell if there is handover or interference between the cells. Using similar arguments against the grid structure, [9], [10] use graph convolutional networks (GCNs). In [9], the authors combine LSTM with GCNs for multi-step ahead prediction. Although their multi-step ahead results are much better, for the special case of one-step ahead prediction, their proposed model performs worse than vanilla LSTM and about the same as ARIMA. In addition to GCNs, [10] proposes to use handover data to improve performance. Mean Square Error (MSE) and Mean Absolute Error (MAE) results in [10] are 10-15% better than vanilla LSTM results.

Another line of research (e.g., [11]–[15]) considers RNN-

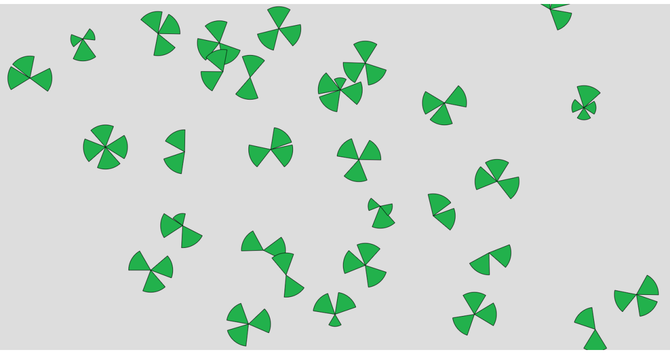

Fig. 1. Cells in a highly-dense urban area.

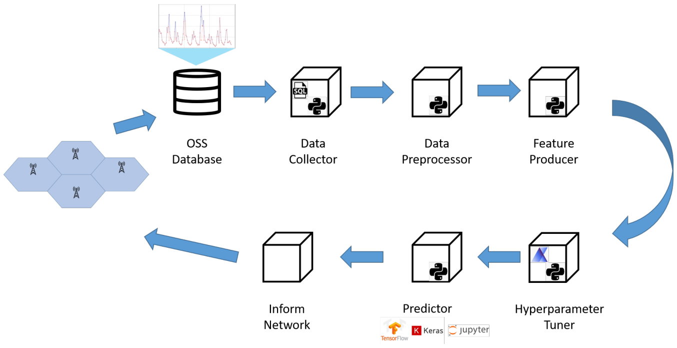

Fig. 2. Architecture of our method

based models to exploit temporal dependency. Since the end goal in [14], [15] is to optimize resource allocation using the predicted traffic, it is difficult to evaluate the performance of the network traffic prediction. On the other hand, [11], [12] also use a grid structure similar to [6]–[8]. Finally, [13] uses a private dataset and cell-clustering based on similarity of time-series trends.

To the best of our knowledge, this paper is the first work in the literature that exploits additional RAN data, peak hours in a day and days of the week to improve prediction performance. Moreover, unlike [10] where the transition probability matrix of the handover graph is used for graph convolutions, we use handover rates as a feature for our RNN models.

In addition, we propose a Service-Level Agreements (SLA) violation-based cost function. Traditional loss functions, like MSE and MAE, are not suitable for mobile operators. It is crucial for an operator to maximize resource utilization while avoiding any SLA violations. Although [8] uses a similar cost function, their analysis is CNN-based, univariate and limited to traffic volume only.

We compare our results to ARIMA, MAE-based univariate LSTM, SLA-based univariate LSTM methods. When all features proposed in this paper are considered, spatio-temporal multivariate LSTM method performs 67% better than ARIMA, 62% better than MAE-based univariate LSTM, and about 20% better than SLA-based univariate LSTM.

II. Network Architecture

This study considers RAN domain of a real-world LTE network operating in a highly dense urban area. As illustrated in Fig. 1, the network consists of eNodeBs having different number of sectors and carriers. Each eNodeB is represented by two letters throughout this study, e.g., GU, VO, SY, and ME. The sector and carrier number of the corresponding eNodeB are represented by two digits. For instance, GU14 denotes the first sector and fourth carrier of the eNodeB GU. In this study, we refer to a single carrier in a sector as a cell, e.g., GU14, GU12, VO14, and SY24.

Mobility is high in the area under consideration and there are unexpected increases in traffic demand especially in the afternoon. This is a region where shopping and business centers are located. Therefore, dynamically changing service demands pose challenges such as SLA violation and resource over-provisioning.

We can predict the traffic volume of an eNodeB either at the eNodeB level (aggregated over sectors and carriers), sector level (aggregated over carriers) or single carrier level. Aggregated eNodeB or sector-based traffic pattern analysis can be inefficient, especially in very crowded areas with dynamic user mobility. Each sector or carrier of an eNodeB may not be correlated with each other. Therefore, we consider deploying a model that will learn the cell-based traffic pattern.

III. Cellular Traffic Prediction

This study proposes a method that can be integrated into any live network. We train the ML model in our proposed method with time-series data that is extracted from the network. This method provides its prediction results directly to the relevant nodes or third party Self-Optimizing Network (SON) module. Fig. 2 illustrates the overall method that is integrated into live network data of eNodeBs. The method provides the traffic prediction in five main steps.

1. Data Collector: The dataset is collected from the Operations Support System (OSS) database of a live LTE network by using the operator-specific scripts. These scripts have unique formulas to calculate the values of specific RAN features by using counters. The dataset contains twenty RAN features that we selected after our technical discussions with network experts of the operator. Feature names are all beginning with a prefix that contains the measurement family name [16], [17]. For each cell (a sector in a carrier of an eNodeB), these features are recorded hourly for six months.

2. Data Preprocessor: The collected data includes all RAN feature values for sectors and carriers of all eNodeBs in the region in a single time axis. First, we organize the dataset based on cells. Then, we examine feature-based NULL statistics in the data of all cells and exclude the cells that contain more than a certain threshold of NULL elements from the scope of the problem. Next, the NULL elements in the remaining dataset are interpolated with a method based on the statistics of the relevant cell. Finally, we analyze all feature values statistically in order to check for an anomaly.

TABLE I
FEATURE LABELS AND ABBREVIATIONS

Feature label	Abbreviated feature name
F1	Nof_ERAB_Estab_Attempt
F2	RACH_Setup_Succ_PC
F3	Avg_RACH_TA
F4	NOF_RRC_ATT
F5	Nof_S1_Sig_EstabAtt
F6	DL_PDCP_Cell_Thput_Mbps
F7	UL_PDCP_Cell_Thput_Mbps
F8	DL_PDCP_User_Thput_Mbps
F9	UL_PDCP_User_Thput_Mbps
F10	DL_Traffic_Volume_Mbyte
F11	UL_Traffic_Volume_Mbyte
F12	Avg_UL_RSSI_Weigh_dBm_PUCCH
F13	Avg_UL_RSRP_PUSCH
F14	Avg_UL_RSRP_PUCCH
F15	Avg_CQI
F16	Avg_Active_Users_DL
F17	Avg_Active_Users_UL
F18	Nof_Avg_SimRRC_ConnUsr
F19	DL_PRB_Util
F20	UL_PRB_Util

3. Feature Producer: Table I shows the labels used for all features that we consider throughout this study. Fig. 3 illustrates the correlation heatmap for cell GU14. The darker the color in Fig. 3, the higher the correlation is between features. Since we focus on the prediction of feature F10, we examine the correlation of other features with F10. Instead of using all the features in the dataset, we apply a correlation threshold of 95% to decrease the dimensionality of the problem. As a result, in addition to feature F10, we include features F16, F17, F18, and F19 to our training set.

The feature producer also focuses on detailed analyses regarding the periodicity of features. We observe that DL traffic volume data in a cell has a 24-hour periodicity with a period of high volume during peak hours. In addition, there are statistical differences between weekday and weekend traffic patterns. Therefore, we include two Boolean feature vectors to our feature set. The first one is to differentiate peak hours from non-peak hours based on a daily peak hour statistical analysis. The second one is to differentiate a weekday from a weekend day. As a result, we highlight the peak hours of traffic volume in each day and weekdays/weekends. This allows the model to increase the prediction accuracy.

Another feature that we consider is based on handovers between cells. If we include the data from all cells in order to optimize the prediction in one cell, the complexity of the problem increases significantly. Therefore, we only consider traffic volume data of a cluster of cells around the cell under consideration. However, there is not any positive affect on the prediction performance if we choose the cells in the cluster depending only on proximity and/or intersection of coverage. On the other hand, we observe an increase in performance when we include cells with high handover percentage in the cluster. As a result, the proposed handover-based clustering feature uses the statistics of incoming handovers from neighboring cells. First, we calculate the ratio of incoming handover number to the total number of handovers. Then, we create a cluster with those cells above a pre-defined threshold value. As an example, Table II illustrates handover statistics for GU14. We observe that the highest rate of incoming handovers is from GU12. If we choose the handover threshold to be 5%, handover-based clustering for GU14 includes GU14, GU12, and VO14.

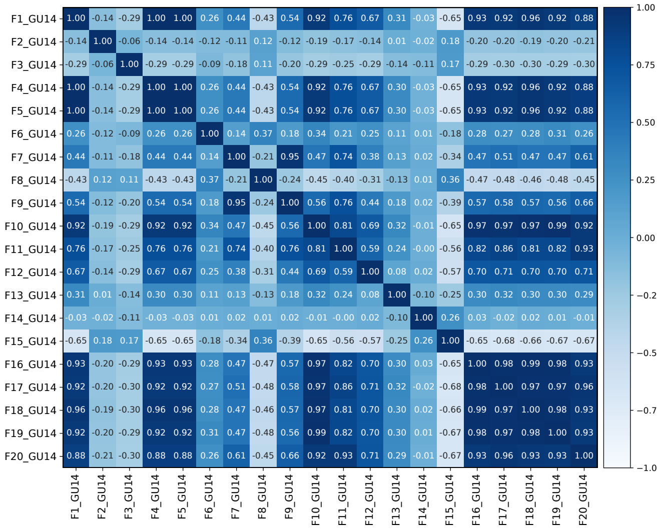

Fig. 3. Correlation heatmap of RAN features for GU14 cell.

TABLE II
THE RATES OF INCOMING HANDOVERS TO GU14

Cell	Handover percentage
GU12	71.97
VO14	7.49
SY24	4.80
VO12	4.17
ME34	3.57
GU24	2.48
ME32	1.85
SY22	1.40
SY34	0.79

4. Hyperparameter Tuner: We structure the data to form sequential samples with a sliding window method. The sliding window is designed to predict the value at the next hour using past values based on the window size. After evaluating the performance results, we observe that the optimum window size is 24. This means that each prediction is based on the traffic data of the past 24 hours.

We split the dataset into training, validation, and test sets

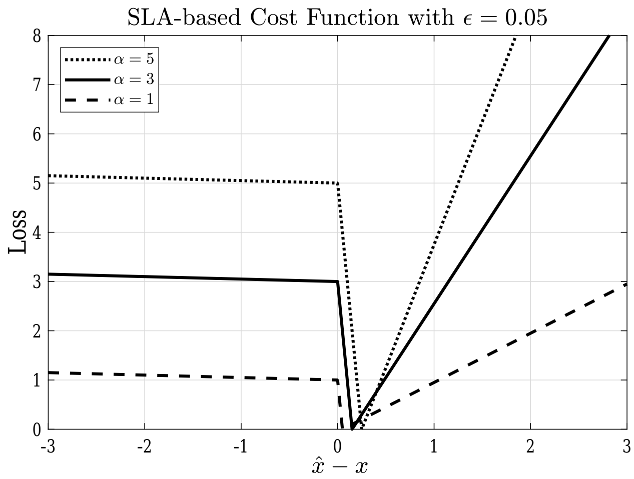

Fig. 4. SLA-based cost functions for different α values.

with the lengths of 18 weeks, 5 weeks, and 3 weeks, respectively. Then, we normalize feature sets by using the mean and standard deviation values calculated based on the training set. Next, using a grid search, we optimize the hyperparameters, such as learning rate, epoch number, L2 regularization penalty, LSTM or GRU layer number and hidden unit number. Finally, we choose the Initializer type to be Random normal or Glorot normal.

We analyze the performance of our ML models using a custom loss function. We use a modified version of the SLA-based cost function that is defined in [8]

$$L(\hat{x} - x) = \begin{cases} \alpha - \epsilon(\hat{x} - x), & \text{when} \quad (\hat{x} - x) \leq 0 \\ \alpha - \frac{1}{\epsilon}(\hat{x} - x), & \text{when} \quad 0 < (\hat{x} - x) < \epsilon\alpha \\ \alpha(\hat{x} - x) - \epsilon\alpha, & \text{when} \quad \epsilon\alpha \leq (\hat{x} - x) \end{cases}$$

where ϵ and α are the parameters of the function, and x and $\hat{x}$ represent the normalized actual traffic volume at a given time and its prediction using past values, respectively. As a result, $(\hat{x} - x) > 0$ means overprovisioning and $(\hat{x} - x) < 0$ means SLA violation. Fig. 4 plots the cost function for $\epsilon = 0.05$ and $\alpha = 1, 3$, and 5. We can see from Fig. 4 that our cost function penalizes underestimation more strictly than overestimation since underestimation results in SLA violations that have monetary consequences for the operator.

5. Predictor: We consider a multivariate dataset with dimensions $T \times F$, where T is the number of time stamps and F is the number of features. The value of F depends on the correlation and handover threshold. Let us consider cell GU14 with 95% correlation and 5% handover threshold. Then, we have five RAN features over 95% correlation threshold, and two traffic volume feature vectors from incoming cells with more than 5% handover rate. In addition, we have two Boolean features for peak hours and weekends. In total, we have nine input features, while the output is the predicted value of F10.

TABLE III
MODEL PERFORMANCE COMPARISON

Models	SLA-based cost
ARIMA	2.6300
MAE-based LSTM	2.3017
SLA-based LSTM	1.0764
mvLSTM-1	0.9582
mvLSTM-2	0.9882
mvLSTM-3	1.0169
mvLSTM-C	0.8737
mvGRU-1	0.9396
mvGRU-2	0.9711
mvGRU-3	0.9945
mvGRU-C	0.8781

IV. RESULTS

This section evaluates the performance of proposed multivariate LSTM and GRU models. Table III illustrates the SLA-based cost values of all models. We see that MAE-based univariate LSTM model performs very poorly and almost as bad as ARIMA when the performance metric is the SLA-based cost function. Using this justification, we use the SLA-based cost for the loss function in the remaining models in Table III. We consider SLA-based univariate LSTM model to be the baseline for our multivariate spatio-temporal models.

Our first multivariate model, mvLSTM-1, includes features F16 through F19 in addition to F10 to predict the future values of F10. This provides about 10% performance increase with respect to SLA-based univariate LSTM. The next multivariate model, mvLSTM-2, includes days of the week and peak hours feature vectors in addition to F10. The additional features in mvLSTM-2 are derived from the statistics of F10 and therefore mvLSTM-2 does not require any additional measurements besides F10. Since the performance of mvLSTM-2 is very similar to the one of mvLSTM-1, it can be a suitable alternative if other measurements are not available. The third multivariate model, mvLSTM-3, includes the traffic volume of the cells with large handover rates to the cell under consideration. The performance of this model is about 6% better than SLA-based univariate LSTM. Finally, we combine all features together in mvLSTM-C model whose performance is about 20% better than SLA-based univariate LSTM model and about 60% better than MAE-based univariate LSTM. We can also see from Table III that GRU based models perform very close to LSTM based models. However, they require a smaller number of parameters.

Actual and predicted DL traffic volumes are plotted in Fig. 5 for MAE-based univariate LSTM model and in Fig. 6 for SLA-based mvLSTM-C model. The prediction curve in Fig. 5 seems to follow the actual data more closely than the prediction curve in Fig. 6. On the other hand, error lines in Fig. 5 show significant SLA violations while there are very few SLA violations in Fig. 6. In general, MAE-based LSTM

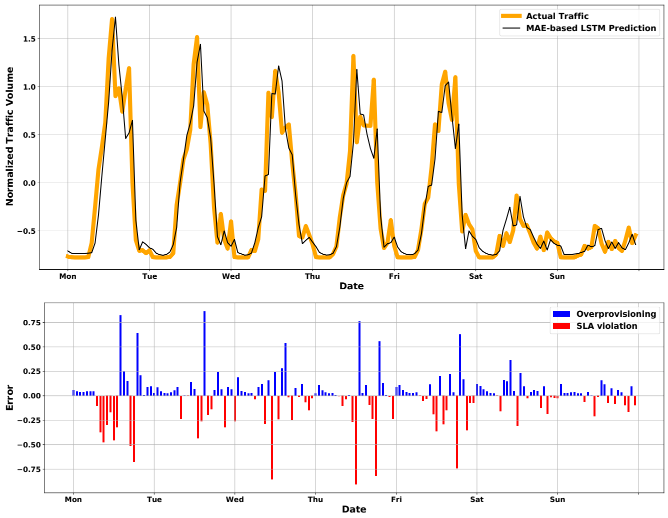

Fig. 5. Actual traffic, predicted traffic, and error values for MAE-based univariate LSTM model. Red lines are SLA violations.

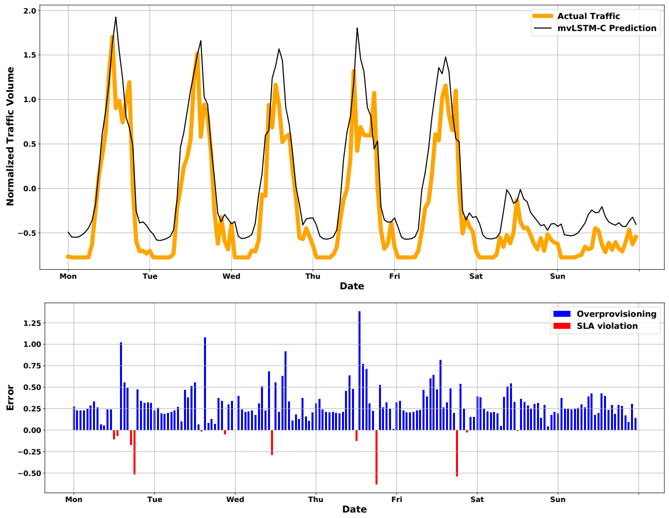

Fig. 6. Actual traffic, predicted traffic, and error values for SLA-based multivariate LSTM model. Red lines are SLA violations.

models result in SLA violations about half of the time and lag the actual traffic at peak hours. For SLA-based LSTM models, it is possible to adjust the level of SLA violations by tuning the parameter α.

V. Conclusion

In this paper, we propose SLA-based multivariate LSTM models that focus on DL traffic volume prediction of a specific sector and carrier of an eNodeB. The dataset is collected from a real and live LTE network operating in a highly dense urban area. When compared to MAE-based models, SLA-based models provide the operator the freedom to set the amount of SLA violations and to avoid associated SLA violation fees. This is particularly important for network slice management, since different slices might have different SLAs. Our future work includes predicting multiple steps ahead in addition to the next hour prediction, optimizing multiple cells at the same time, and investigating the effects of correlation and handover percentage thresholds.

References

[1] R. Shafin, L. Liu, V. Chandrasekhar, H. Chen, J. Reed, and J. C. Zhang, “Artificial intelligence-enabled cellular networks: A critical path to beyond-5G and 6G,” *IEEE Wireless Communications*, vol. 27, no. 2, pp. 212–217, April 2020.

[2] C. Zhang, P. Patras, and H. Haddadi, “Deep learning in mobile and wireless networking: A survey,” *IEEE Communications Surveys Tutorials*, vol. 21, no. 4, pp. 2224–2287, March 2019.

[3] M. Wang, Y. Cui, X. Wang, S. Xiao, and J. Jiang, “Machine learning for networking: Workflow, advances and opportunities,” *IEEE Network*, vol. 32, no. 5, pp. 92–99, March 2018.

[4] N. Bui, M. Cesana, S. A. Hosseini, Q. Liao, I. Malanchini, and J. Widmer, “A survey of anticipatory mobile networking: Context-based classification, prediction methodologies, and optimization techniques,” *IEEE Communications Surveys Tutorials*, vol. 19, pp. 1790–1821, thirdquarter 2017.

[5] F. Xu, Y. Lin, J. Huang, D. Wu, H. Shi, J. Song, and Y. Li, “Big data driven mobile traffic understanding and forecasting: A time series approach,” *IEEE Transactions on Services Computing*, vol. 9, pp. 796–805, September 2016.

[6] C. Zhang, H. Zhang, D. Yuan, and M. Zhang, “Citywide cellular traffic prediction based on densely connected convolutional neural networks,” *IEEE Communications Letters*, vol. 22, pp. 1656–1659, August 2018.

[7] C. Zhang and P. Patras, “Long-term mobile traffic forecasting using deep spatio-temporal neural networks,” in *ACM Mobicom*, June 2018, p. 231–240.

[8] D. Bega, M. Gramaglia, M. Fiore, A. Banchs, and X. Costa-Pérez, “Deepcog: Optimizing resource provisioning in network slicing with AI-based capacity forecasting,” *IEEE Journal on Selected Areas in Communications*, vol. 38, pp. 361–376, February 2020.

[9] L. Fang, X. Cheng, H. Wang, and L. Yang, “Mobile demand forecasting via deep graph-sequence spatiotemporal modeling in cellular networks,” *IEEE Internet of Things Journal*, vol. 5, pp. 3091–3101, August 2018.

[10] S. Zhao, X. Jiang, G. Jacobson, R. Jana, W.-L. Hsu, R. Rustamov, M. Talasila, S. A. Aftab, Y. Chen, and C. Borcea, “Cellular network traffic prediction incorporating handover: A graph convolutional approach,” in *IEEE SECON*, Como, Italy, June 2020.

[11] J. Wang, J. Tang, Z. Xu, Y. Wang, G. Xue, X. Zhang, and D. Yang, “Spatiotemporal modeling and prediction in cellular networks: A big data enabled deep learning approach,” in *IEEE INFOCOM*, Atlanta, GA, USA, May 2017.

[12] L. Chen, D. Yang, D. Zhang, C. Wang, J. Li, and T. M. T. Nguyen, “Deep mobile traffic forecast and complementary base station clustering for C-RAN optimization,” *Journal of Network and Computer Applications*, vol. 121, pp. 59–69, 2018.

[13] J. Feng, X. Chen, R. Gao, M. Zeng, and Y. Li, “DeepTP: An end-to-end neural network for mobile cellular traffic prediction,” *IEEE Network*, vol. 32, pp. 108–115, November 2018.

[14] I. Alawe, A. Ksentini, Y. Hadjadj-Aoul, and P. Bertin, “Improving traffic forecasting for 5G core network scalability: A machine learning approach,” *IEEE Network*, vol. 32, pp. 42–49, November 2018.

[15] H. Chergui and C. Verikoukis, “Offline SLA-constrained deep learning for 5G networks reliable and dynamic end-to-end slicing,” *IEEE Journal on Selected Areas in Communications*, vol. 38, pp. 350–360, February 2020.

[16] ETSI, “LTE; Telecommunication management; Performance Management (PM); Performance measurements Evolved Universal Terrestrial Radio Access Network (E-UTRAN),” European Telecommunications Standards Institute (ETSI), Technical Specification (TS) 132.425, 11 2020, version 16.5.0.

[17] ETSI, “Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); LTE; Telecommunication management; Performance Management (PM); Concept and requirements,” European Telecommunications Standards Institute (ETSI), Technical Specification (TS) 132.401, 08 2020, version 16.0.0.