User Classification and MCS Prediction with LSTM Neural Networks from LTE Traffic Data

Gianluca Fighera*, Yuri Nalesso* and Giovanni Perin*

Abstract— **Because of the beginning of the 5G era, predictive analysis on mobile traffic data is becoming of extreme relevance. Actually, knowing in advance the demand for resources of UEs implies that BSs are able to optimize the allocation and the network will benefit in terms of spectral efficiency as well as the users in terms of QoS and QoE. In this work, part of the SCAVENGE Data Challenge, we clusterize users having similar behaviors and propose a prediction framework for the best Modulation and Coding Scheme (MCS) that should be used in the following time step with a certain class of users. Moreover, from the predicted patterns and the belonging cluster, we try to infer users' mobility, doing anomaly detection to find instants where the users' pattern changes, possibly due to a modification in their mobility. Data is gathered from the Physical Downlink Control CHannel (PDCCH) of the LTE. Results show that our 3 class unsupervised clustering is effective in grouping similar users and that the LSTM neural networks employed for the prediction are very accurate, reaching low NRMSE, of the order of 10^{-2}.**

I. Introduction

With the advent of 5G communications, the need for low latency and reliable communications will increase and reach higher levels than ever. Actually, even though this problem started to be of the utmost relevance already within the now ending 4G age, it is estimated that, thanks to the IoT revolution, the number of connected devices will exceed 30 billions units [11] by 2020. The problem of allocation of physical resources such as the spectrum, will be thus of extreme importance, because the demand will increase correspondingly to the number of devices. Jointly, it would be desirable to improve at the same time the Quality of Service and the Quality of Experience. A method to improve both of them, as well as the spectral efficiency, is to be able to select a proper Modulation and Coding Scheme (MCS) at the BS. Analyzing past data, the network could be aware of the UEs demands and be able to properly respond to allocation of resources. Therefore, being able to predict the upcoming traffic demand and its quality will be fundamental in the near future.

In Long Term Evolution (LTE), the Channel Quality Indicator (CQI) is an index in $[0, 15]$ sent by the User Equipment (UE) as a feedback to indicate the data rate that can be supported by the transmission and it can be mapped to the highest MCS that allows the UE to decode the transport block with less than 10% error rate probability [6], [3]. In this work, part of the SCAVENGE Data Challenge contest [2], we exploit the state-of-the-art of machine learning frameworks to perform a deep analysis of the MCS sequences of many different users within three different eNodeBs. As on field data about the 4G LTE technology are now widely available, we use them to reach these goals:

- Classification of the users given the observed MCS pattern at the eNodeB;
- Implementation of a temporal predictor for the following employed MCS for each of the users classes;
- Inference of the variability of mobility of the users basing on the observed channel quality experienced.

More in the detail, we use a completely unsupervised method, based on spectral clustering, to divide users in three classes. Each cluster should contain users that behave in a similar way and this may be useful for improving performances in the following steps. The second objective is reached training a separate Long-Short Term Memory (LSTM) neural network for each of the users' clusters. We chose such networks because they have shown to outperform performances of simpler Recurrent Neural Networks (RNN) and Hidden Markov Models (HMM) for the task of time series prediction. Finally, we used the resulting temporal predictive LSTM models to infer variability in the MCS pattern, employing them as anomaly detection tools.

The rest of the paper is organized as follows: in Section II we show the work that has been done so far in the literature on mobile data analysis; Section III is dedicated to inspect the available dataset and its statistics, as well as the preprocessing steps that were performed; in Section IV we give some theoretical background on the employed methods, i.e. details about the unsupervised clustering of users and the temporal predictor built with LSTMs; Section V is devoted to the presentation of the reached results; finally, in Section VI we will drive some conclusions about the most relevant findings of our work and some possible extensions.

II. Related Work

Academic research and scientific literature have already focused on the task of user behavior prediction basing on mobile data. Nonetheless, as it is done in [12], often the employed datasets refer to Call Detail Records (CDRs). As they come indistinguishably from text, voice and data, the limit of this is that no information about how users employ physical resources can be assessed. More recently, however, LTE data started to be available and the scientific community employed them using different frameworks. As

*These authors contributed equally.

*Department of Information Engineering, University of Padova, Padova, 35131, Italy {gianluca.fighera, yuri.nalesso, giovanni.perin.2}@studenti.unipd.it

for what concerns users classification, in [10], authors builds a robust binary classifier based on observed CQI's variance to distinguish between users with high and low mobility patterns. Classical approaches such as Wiener filters, cubic spline extrapolation and short-term average are studied in [14] to analyze elder CQI signals sent by the UEs and predict the ones that with higher probability will be used in the following time steps. With this method authors show to improve the average throughput with respect to the baseline. In another recent work Gaussian Process Regression is used for CQI-SNR mapping and to reduce the overhead due to CQI feedback while keeping unvaried the BLER [4], [5]. Hidden Markov Models are instead used for the location prediction problem in LTE networks [7]. However, it tackles the problem with a different approach which does not employ neither CQI nor MCS sequences but rather the topology of the network and routing information.

In our work, we employ LSTM neural networks for temporal prediction of mobile data, as it is done recently in [13] with promising results. We also propose novel methods, never employed before in this field to the best of our knowledge:

- User classification building an affinity matrix and clustering it with the spectral clustering algorithm. We borrowed this method from network science, where it is commonly used;
- Inference of variability in MCS pattern, possibly due to user mobility, using the LSTM networks as anomaly detection tools.

III. Dataset and Preprocessing

A. Dataset Exploration

The dataset provided contains traces from the 4G control channel of 3 different cells. The data gathering length one week observing the mobile users both uplink and downlink. The channel used is the LTE Physical Downlink Control Channel (PDCCH) that is responsible for the communication of the scheduling information. The information available spans 6 columns containing:

- The *Timestamp*, following the format Y/m/d/H/M/S;
- The temporal user identifier *RNTI* (Radio Network Temporary Identifier) which is renewed after a short time of inactivity (about 10 seconds);
- The transmission *direction* distinguishing the uplink (0) and downlink (1);
- The Modulation and Coding Scheme (*MCS*), which goes from 0 to 31. As the number grows also the spectral efficiency of the channel gets better;
- The *subframe number*, going from 0 to 1023 and representing the LTE clock;
- The *Subframe index*, going from 0 to 9 and further dividing the *subframe number*.

The date column has been replaced by *delta*, that tracks the time passed from the first data captured of the session.

The modulation used is slightly dependent on the direction of the transmission as we can see from its distribution in

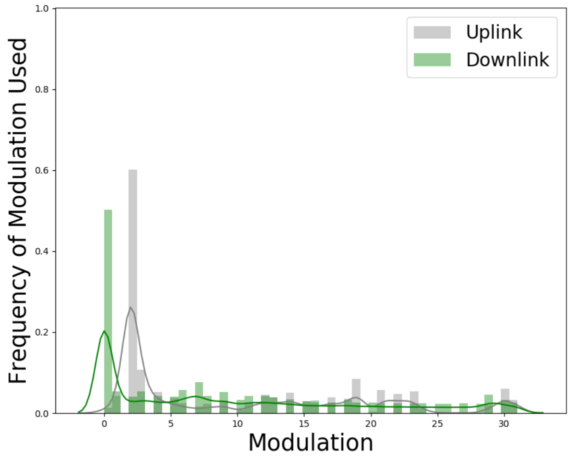

Fig. 1: MCS distribution plot

Fig. 1: actually, for the the downlink channel, the most used MCS is number 0 whereas for the uplink number 2. To assess this, we used the data available as features in a regression approach to fit the MCS pattern employed at the eNodeB. Random forests regressors confirm that the *direction* and the *delta* are the most important features related to the employed MCS whereas the *subframe number* only represents noise.

B. Preprocessing

The main preprocessing step is the user assignment due to the 10 seconds renewal time of the RNTI. This part needs to be a sequential reading of the data, depending thus linearly on the length of the dataset. The users are divided in a list and those that have less than N entries were discarded in order to improve the stability of the classification system. For eNodeB A and B, we discarded user with less than 500 and 300 data points respectively whereas, for eNodeB C, N was fixed to 1500 because of the huge amount of data available at this cell. Moreover, we avoided sequences of all zeros MCSs because they were not interesting for the task of prediction, being deterministic. We also discarded values of the MCSs in the range $[29, 31]$ because they are reserved for HARQ (Hybrid Automatic Repeat reQuest). This choice was done considering that they represented noise in the prediction of the MCS because the modulation employed is not linearly dependent on the index in the same way as other indeces.

IV. Methods

A. Users Clustering

The first goal of our project is the classification of UEs based on observed MCS patterns. Since we lack class labels and we only have data as described in Section III, we had to figure out a fully unsupervised clustering algorithm.

One of the major challenges was due to the fact that sequences lengths were very different, depending on for how long users filled the channel. Metrics often employed in this case are Dynamic Time Warping (DTW) based distances.

Algorithm 1: MCS clustering

Input : MCS sequences s_k for N users, processed removing the mean value $\bar{s_k}$;
Gaussian kernel width σ;
Number of clusters K.
Output: Cluster labels for each sequence.

```
begin
    initialize: D and A all zeros N × N matrices;
    for i = 0 to N − 1 do
        for j = i + 1 to N do
            compute the cross correlation of s_i and s_j;
            compute the shift corresponding to the
              argmax of the correlation;
            s_1 ← shorter sequence;
            map s_1 on the other sequence with the
              computed shift;
            s_2 ← window of the longer sequence where
              lies the mapping of s_1;
            d_{i,j} ← MAE(s_1, s_2);
        end
    end
    mirror D with respect to the diagonal;
end
begin
    foreach d_{i,j} ∈ D do
        a_{i,j} ← RBF(d_{i,j}, σ);
    end
    labels ← Spectral Clustering(A, K);
end
```

However, the complexity of this algorithm makes infeasible to compute pairwise distances for many and long time series. This is why we opted for the much simpler but effective following algorithm. After removing the mean value from signals, we computed the cross-correlation for each pair of them. Then, the mapping of the shorter sequence on the longer one was performed, basing on the index where correlation was maximized. We then computed the distance in the window defined by the matching using the Mean Absolute Error (MAE) criterion. In this way a distance matrix is obtained, further transformed to a similarity one applying the RBF transformation

$$s_{i,j} = \exp\left(-\frac{d_{i,j}^2}{2\sigma^2}\right)$$

with $\sigma = 7$ free parameter representing the width of the Gaussian kernel. We finally used the standard Spectral Clustering algorithm [9] to this affinity matrix looking for $K = 3$ clusters. The algorithm is resumed in Alg. 1.

B. LSTM Neural Networks

Long-short term memory networks are an extension of the recurrent neural networks, developed to face the problem of the vanishing gradient. Actually, when back-propagating the errors in standard RNN, long-time dependencies between data are lost due to the amount of multiplications. Instead of relying on set of neurons, each one with a unique operation and activation function, LSTM networks are composed by LSTM cells (Fig. 2). Each cell is basically a memory structure with some channels that allow for the information flow from one cell to another. Every channel is controlled by some learnable parameters, tuned during training, whose aim is to select the amount of information to retain and to discard in the cell. Also, they decide how much information to propagate to the subsequent LSTM cell.

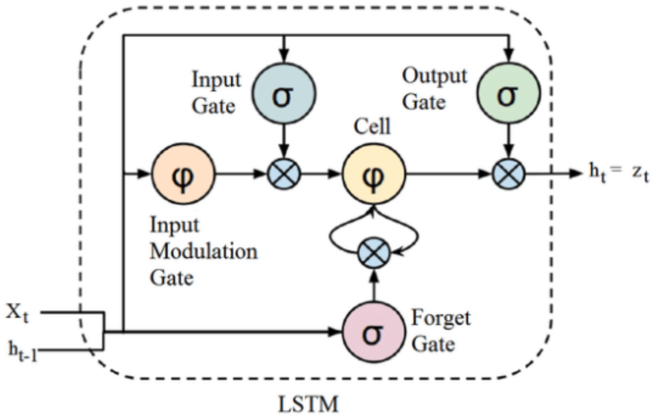

Fig. 2: LSTM cell that forms the basis of the LSTM network.

The LSTM cell is composed by three gates: the forget gate, the input gate and the output gate. The forget gate is responsible for removing the part of the information from the previous time step that is not required by the network. The input gate is the one that takes the new information and saves it into the memory cell integrating it with the previous state. The output gate returns an output to feed the subsequent cell. Each of these channels have a different activation function that can be tuned for the specific application.

For this project, we used one LSTM network for each cluster of users. The model is implemented using the *Keras* library with *Tensorflow* backend. It is a double-layer network where the first layer is composed by $N = 100$ LSTM cells with *tanh* activation function and the second layer is a fully connected layer with a single scalar output. We used the Adam optimizer and trained the network to minimize the Mean Absolute Error. We explored different loss functions like Mean Squared Error, categorical cross entropy and others but we found that Mean Absolute Error gives the best results. The LSTM is trained with sequences of the MCS data of each user. After trying different processing steps, we discovered that giving original MCS data as input to the network without scaling them will result in a model that is not able to differentiate its output in a so wide range $[0, 28]$. Also, we found that scaling data in the range $[0, 1]$ helped the network but the best results were achieved scaling the MCS data in the range $[-1, 1]$. Short sequences of $T = 50$ time steps of the same user are concatenated before feeding the network. The LSTM model is trained to predict the MCS that the user is supposed to adopt in the subsequent time step. It is relevant to

	mean	std	iqr	variation
cluster 1	8.36 ± 2.51	6.20 ± 1.44	9.88 ± 4.37	0.78 ± 0.19
cluster 2	4.47 ± 3.51	2.84 ± 1.70	2.47 ± 2.93	0.69 ± 0.22
cluster 3	11.73 ± 2.18	7.84 ± 1.06	14.28 ± 4.41	0.69 ± 0.12

TABLE I: Statistical measures for the three users' clusters communicating with the eNodeB A.

notice here that MCS sequences can be considered unevenly sampled time series since the interval between a transmission and the following one of the same user may vary a lot. This is why we observe 50 samples before predicting the following one, because the network has to learn also the behavior of the UE about how it fills the channel.

V. RESULTS

A. Users classification

As better explained in Section IV-A, we implemented a spectral clustering based classification of the users' MCS sequences, building an affinity matrix derived as a Gaussian kernel transformation of the distance matrix obtained computing the mean absolute error between each pair of sequences. We decided to set the number of clusters to $K = 3$ and this was motivated by the following steps of our work. Actually, we did not want to have neither too many clusters, because this would have meant having as many different LSTM models, nor just two, because this choice could have led to an unacceptably high inter-class variability. We report here results from eNodeB A: plot examples of the three types of users that our algorithm classifies in different clusters can be seen in Fig. 3. In Tab. I common statistics for the three classes are shown. It is evident that the classes are significantly different in terms of either mean value, standard deviation and interquartile range. The variation coefficient is instead very similar among different clusters and this may tell that classes are correctly balanced. For eNodeB A 1266 useful sequences were recovered, and, after running our clustering algorithm, 60% of them were put in cluster 2 and the remaining 40% was equally split among the other two clusters. It is interesting to observe that clusters 1 and 3 have uplink/downlink distribution which is approximately balanced: uplink transmisisons represent the 50% and 57% respectively. Cluster number 2, instead, which is the one having the smallest variance, is composed for the 73% of uplink transmissions. Indeed, this is in perfect agreement with what was highlighted by Fig.1: the uplink channel is almost stable between MCSs 2 and 3. Results for eNodeB B and C were quite similar.

We also tried to run K-Means clustering using as features the ones reported in Tab. I and many other statistical measures, also applying Principal Component Analysis (PCA). Nevertheless, results were less effective and the method required also more human intervention in features selection.

B. MCS prediction

In this project, we trained an LSTM model for each cluster of users. In particular, we took the MCS sequences

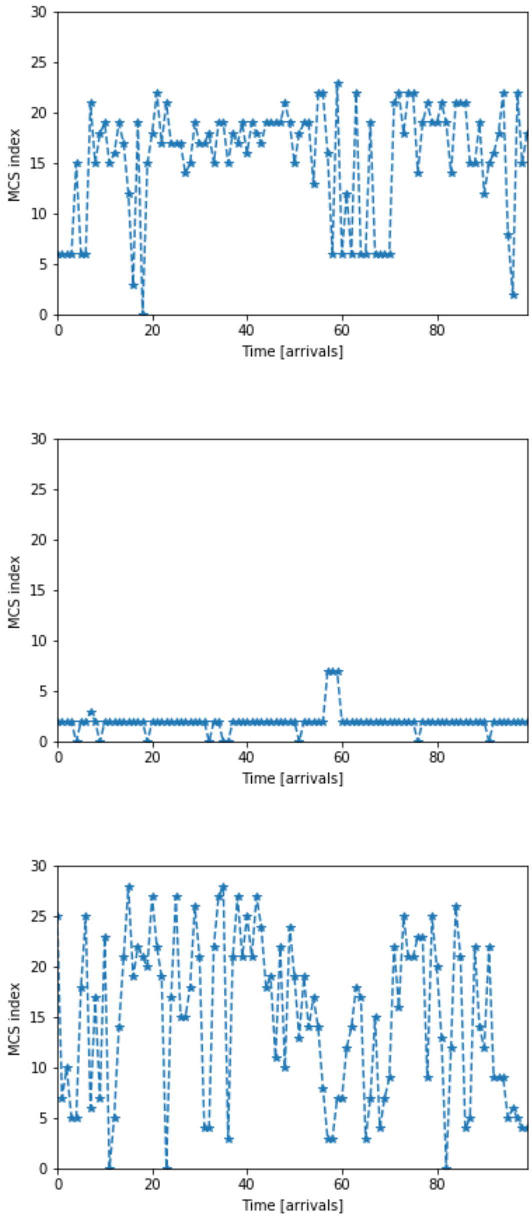

Fig. 3: Plot of three segments of MCS sequences (100 samples) belonging to different users' clusters. From top to bottom: cluster 1, cluster 2 and cluster 3. Data collected at eNodeB A.

of each user in the cluster and split them as sequences of 51 consecutive MCS. Then, we merged all the sub-sequences of all users in a dataset and shuffled it. Finally, we took 70% of these sequences as training set and the remaining 30% as test set. For each sequence in the training set, the LSTM network is trained to predict the 51^{th} MCS in the sequence, given the previous 50. We trained each LSTM for 10 epochs, since we observed that the loss converged quickly after $5-6$ epochs.

We tried also to include the timestamp and the direction of the transmission as features to help the LSTM prediction. However, no significant results were obtained.

Performances of the model were evaluated using the

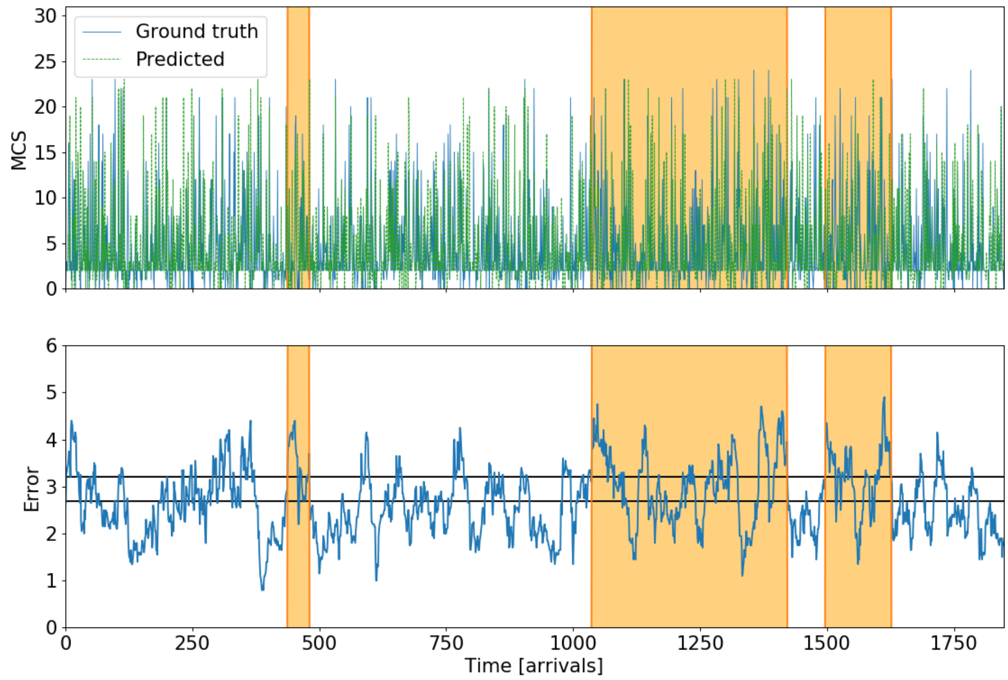

Fig. 4: Above: LSTM prediction and ground truth for an MCS sequence belonging to cluster 1 of eNodeB A. Below: mean absolute error of the prediction. In orange are highlighted the regions with high variability basing on the prediction error. The horizontal lines represent the mean value of the error and the mean added to one standard deviation.

normalized root mean squared error (NRMSE). The RMSE was calculated striding a window with size 20 and stride 1 over the predicted MCS sequence compared to the ground truth. Then, the RMSE was normalized by diving it by the mean of the ground truth. Results are illustrated in table II. For cluster 3, the NRMSE is lower because, while having the same prediction precision, the mean of the cluster is higher.

C. Mobility inference

The variability of the MCS sequence can be due to other factors besides users' mobility. As authors say in [10], one of the possible contributing cause is delay profile. Moreover, other possible momentary channel disturbances may modify the MCS employed. Nonetheless, user mobility is certainly a major cause factor in Signal to Noise Ratio variability and therefore also in the modulation used for transmissions. Therefore, a first consideration can be done just by looking at users' clusters and Tab. I: it is evident how variability changes a lot when comparing different clusters. For the example reported about eNodeB A, cluster 1 can be considered the one belonging to moderate mobility users whereas cluster 3 to high mobility ones. Cluster 2 should be the one relative to low mobility users, but, as already remarked, it is affected by the fact that transmissions are unbalanced towards the uplink channel, which prefers smaller and fewer MCSs.

	Cell A	**Cell B**	**Cell C**
cluster 1	0.96	0.87	0.96
cluster 2	0.95	0.85	0.98
cluster 3	0.96	0.35	0.71

TABLE II: Normalized RMSE for the clusters of the three cells.

Apart from these purely statistical considerations, we developed a method to infer variability about the channel experienced by a UE that can possibly be due to a change in the mobility. Our algorithm is straightforward and is based on the use of the LSTM as an anomaly detection tool. We considered an MCS sequence and computed the mean absolute error vector e of the prediction with respect to the ground truth. Then we took the values that lied outside the range $[0, \bar{e}+\sigma_e]$ where $\bar{e}$ is the mean value of the error vector and σ_e its standard deviation. We built an indicator vector having 1s in the positions were the error was greater than $\bar{e}+\sigma_e$ and opened an anomaly window if more than N consecutive 1s were found. The window was closed if more than M consecutive 0s were found in the indicator vector. We decided to set $N = 20$ and $M = 50$ and results are shown in Fig. 4.

VI. Conclusions

In this paper we presented a framework for predicting the best Modulation and Coding Scheme (MCS) for the following time step given the past experience about users or class of users. Actually, we also implemented an effective and completely unsupervised way to classify users in three clusters having different statistical metrics. Moreover, we proposed an additional mobility detector based on the developed LSTM models that highlights where the observed MCS patterns change behavior. Thanks to its high accuracy, our predictor can be used to improve the resource allocation in terms of spectral efficiency and Quality of Service when the last CQI received is outdated.

Finally, we list some possible future work and extensions. The user classification task can be improved finding a way to cope with the problem of hard classification. Some of the users, for example, have strong connections with users belonging to different clusters, because they presented patterns that varied during time, changing from one cluster to another one. A possible way to solve this problem is to use overlapping community detection to the same affinity matrix employed for spectral clustering. The MCS prediction task can be instead improved in many ways, for instance trying to predict more than one MCS or trying to predict the following best MCS at the eNodeB independently from the UE cluster. We also think, given the sparsity in the time domain of the sequences, that it would be worth it to try Time-LSTM [15], [1] for this task.

REFERENCES

[1] Inci M Baytas, Cao Xiao, Xi Zhang, Fei Wang, Anil K Jain, and Jiayu Zhou. Patient subtyping via time-aware lstm networks. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 65–74. ACM, 2017.

[2] Sustainable Cellular Networks Harvesting Ambient Energy. Mobile data challenge. http://www.scavenge.eu/dissemination/data-challenge/.

[3] TS ETSI. 136 213 v8. 3.0 (2008-11) technical specification. *URL: http://www.etsi.org/deliver/etsi_ts/136200_136299/136213/08.03.00_60/ts _136213v080300p.pdf.*

[4] Samira Homayouni, Stefan Schwarz, Martin Klaus Mueller, and Markus Rupp. Cqi mapping optimization in spatial wireless channel prediction. In *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*, pages 1–5. IEEE, 2018.

[5] Samira Homayouni, Stefan Schwarz, and Markus Rupp. On cqi estimation for mobility and correlation properties of gaussian process regression. In *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*, pages 1–5. IEEE, 2018.

[6] Mohammad T Kawser, Nafiz Imtiaz Bin Hamid, Md Nayeemul Hasan, M Shah Alam, and M Musfiqur Rahman. Downlink snr to cqi mapping for different multiple antenna techniques in lte. *International Journal of Information and Electronics Engineering*, 2(5):757, 2012.

[7] Qiujian Lv, Zongshan Mei, Yuanyuan Qiao, Yufei Zhong, and Zhenming Lei. Hidden markov model based user mobility analysis in lte network. In *2014 International Symposium on Wireless Personal Multimedia Communications (WPMC)*, pages 379–384. IEEE, 2014.

[8] Pankaj Malhotra, Lovekesh Vig, Gautam Shroff, and Puneet Agarwal. Long short term memory networks for anomaly detection in time series. In *Proceedings*, page 89. Presses universitaires de Louvain, 2015.

[9] Andrew Y Ng, Michael I Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in neural information processing systems*, pages 849–856, 2002.

[10] Jinping Niu, Daewon Lee, Tao Su, Geoffrey Y Li, and Xiaofeng Ren. User classification and scheduling in lte downlink systems with heterogeneous user mobilities. *IEEE Transactions on Wireless Communications*, 12(12):6205–6213, 2013.

[11] Amy Nordrum et al. Popular internet of things forecast of 50 billion devices by 2020 is outdated. *IEEE spectrum*, 18(3), 2016.

[12] Yuzuru Tanahashi, James R Rowland, Stephen North, and Kwan-Liu Ma. Inferring human mobility patterns from anonymized mobile communication usage. In *Proceedings of the 10th International Conference on Advances in Mobile Computing & Multimedia*, pages 151–160. ACM, 2012.

[13] Hoang Duy Trinh, Lorenza Giupponi, and Paolo Dini. Mobile traffic prediction from raw data using lstm networks. In *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pages 1827–1832. IEEE, 2018.

[14] Xiang Xu, Mingjian Ni, and Rudolf Mathar. Improving qos by predictive channel quality feedback for lte. In *2013 21st International Conference on Software, Telecommunications and Computer Networks-(SoftCOM 2013)*, pages 1–5. IEEE, 2013.

[15] Yu Zhu, Hao Li, Yikang Liao, Beidou Wang, Ziyu Guan, Haifeng Liu, and Deng Cai. What to do next: Modeling user behaviors by time-lstm. In *IJCAI*, pages 3602–3608, 2017.