Applied Mathematics, Modeling and Computer Simulation
C.-H. Chen et al. (Eds.)

doi:10.3233/ATDE240776

Spectrum Hole Prediction in LTE-Based Cognitive Radio System Using Kolmogorov-Arnold Network

Rykhard BOHUSH [a,1], Yagor ADAMOVSKIY [a], Nikalay NAUMOVISH [b]
[a] *Polotsk State University, Polotsk, Belarus*
[b] *Belarusian State University of Informatics and Radioelectronics, Minsk, Belarus*

Abstract. This paper examines the problem of channel resource occupancy prediction using Radio Environment Maps (REM) based on Long Term Evolution (LTE) for a cognitive communication system. REM is a spatiotemporal database in the form of a resource grid with passing traffic in cells. For prediction, the Kolmogorov-Arnold network (KAN) architecture is used. A model structure has been developed that collects data, trains and tests KAN. The predictive model control algorithm is implemented in Python. MatLab was used to prepare input data and implement the LTE simulation model. Experiments are carried out and results of free channel resource prediction based on KAN and long-term memory are presented.

Keywords. Cellular communication system; KAN; artificial neural networks.

1. Introduction

Currently, communication systems face the problem of spectrum scarcity. One solution is dynamic spectrum access. Its implementation is the technology of cognitive radio (CR). Licensed primary users (PUs) do not always fully utilize the frequency resources allocated to them in the time and space domains. CR devices are capable of assessing the radio frequency (RF) environment around them, detecting currently idle channel resources, and using them for communication between secondary users (SUs). When PUs appear on the channel, such devices must immediately release the resources they occupy.

Traffic in communications systems is a partially stochastic process, but contains patterns that can be extracted and predicted. Among the artificial neural networks (ANN), the architectures used for prediction are feed-forward neural networks (FNN), convolutional neural networks (CNN), and recurrent neural networks (RNN).

FNN, or multi-layer perceptron (MLP), is a classic ANN architecture containing multiple unidirectional fully connected neural layers in the structure. FNN is a universal solution for forecasting problems, but requires a large amount of data and computing resources for training. CNN architecture is used for pattern recognition in images (feature extraction), but is also capable of making predictions and representing

[1] Corresponding Author, Rykhard BOHUSH, Polotsk State University, Polotsk, Belarus; E-mail: r.bogush@psu.by.

the results as indices from a list of output states (classification). CNN has the disadvantages of FNN and also requires tuning many internal parameters to achieve high accuracy. RNN implements feedback between layers, which allows you to remember information obtained in previous stages of training.

The recurrent LSTM architecture has been successfully applied to prediction and classification problems based on large data sets [1], including communication systems [2]. LSTM is superior in accuracy to the statistical model autoregressive integrated moving average (ARIMA) [3]. Next, the specifics of using LSTM in traffic prediction for communication systems are considered.

In [4], the authors propose a spatiotemporal cross-domain neural network (STC-Net) architecture based on hybrid conv-LSTM layers [5] for 5G wireless network for traffic analysis on a 100×100 grid using input data: week number, hour of the day and type of day (weekend or weekday).

Work [6] shows traffic prediction architecture similar to [4], but using fully connected, convolutional, and LSTM layers in a single model. In contrast to [4], forecasting was performed for time series, i.e. for a sequence of traffic data over a time interval in a limited spatial area. The model demonstrates higher accuracy compared to other deep learning models: DenseNet, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Gate Recurrent Unit Network (GRU).

Therefore, different ANN architectures can be combined to solve the forecasting problem. At the same time, the developers [4] note that their model is highly complex and requires a lot of training time. It follows that with a similar approach (architectural combination), the model [6] will have similar disadvantages.

In [7], the authors use a 4-layer (20% dropout) LSTM model to predict encrypted traffic from multiple LTE base stations (BSs) based on time series (32 input samples (1 hour interval) and 1 output sample). The advantage of LSTM over ARIMA is shown to be on average 12% in terms of the following indicators: mean square error (MSE), mean absolute error (MAE) and R2 score. However, the model accuracy [7] (MSE = 0.05) is not an ideal result, since the input data had a simple predictable structure, and training was carried out on a huge number (~1 million) of records. In [8], the authors propose a 4-layer LSTM model with exponential smoothing (SES) and a smaller dataset (~120,000 records) compared to [7] for mobile traffic prediction. The achieved MSE was 10-4–10-5 with an actual average prediction accuracy of 91%.

Based on the analysis, it follows that setting up an LSTM model with data in mind is a complex task, which is determined by many internal factors (type of input data, data size, layer configuration, etc.) and external factors (limited computing resources, training time and operating speed).

The LTE cellular communication system has features of traffic generation between network nodes. The work [9] compares different approaches to radio environment map (REM) state prediction based on real LTE data on a 100×100 grid: Received Signal Strength Indicator (RSSI), Reference Signal Received Power (RSRP), and Global Positioning System (GPS). According to RMSE (root MSE), the following approaches showed the greatest accuracy: extremely randomized tree regressor (ERTR), bagging, KNN and regression tree. However, [10] showed that SVM is comparable in accuracy and superior in speed, and [11] noted Naive Bayes classifier, but their accuracy is lower than deep learning approaches, including LSTM.

Since some aspects of REM can be represented as two-dimensional arrays, CNN models are also used for prediction. In [12], a DeepREM model based on a convolutional U-Net and a conditional generative adversarial network (CGAN) was used to analyze the BS coverage and RSRP parameters. However, such methods do not use a path loss model and are data-driven only.

Based on an analysis of existing solutions for the problem of traffic forecasting in communication systems, we can conclude that the best approach is to use LSTM (or its modifications). LSTM provides high prediction accuracy for large amounts of data containing hidden patterns.

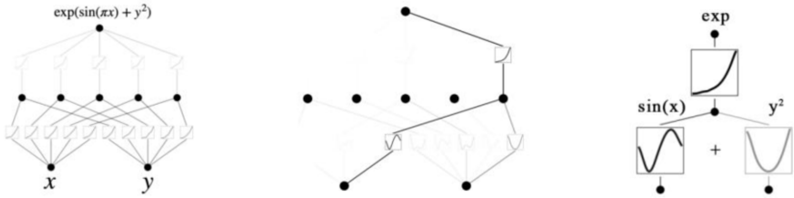
Recently, the KAN architecture [13] was proposed as a promising alternative to MLP: it demonstrates higher accuracy with a similar number of parameters and offers a visual interpretation of the network. A special feature of KAN is the implementation of activation functions on edges instead of nodes, i.e. training activation functions instead of using ready-made functions (Tanh, ReLU or Softmax). Currently, forecasting the channel resources state for communication systems using KAN is not performed, so this article evaluates the suitability of this architecture for this task.

In [14], a simulation model of an LTE communication system is presented. The model generates data reflecting the dynamics of the state of the radio environment in the form of a map (REM). The model simulates LTE BS and user equipment (UE) objects. Users act according to an individual schedule of communication sessions and movements across map cells, formed on the basis of a Markov process. In the LTE model, a KAN-based model will be used to predict occupied channel resources and compared with LSTM.

2. Kolmogorov-Arnold neural network

The KAN architecture implements a system of nodes and edges similar to MLP, but the edges contain parameterized curves (splines). Their parameters change during the training process, and the nodes simply sum them up as we can see in figure 1a. The authors of the original paper [13] compare the number of KAN and MLP parameters to be trained. The comparison shows higher requirements for KAN, but the authors argue that MLP requires more N neurons for each of the L layers to achieve the same accuracy.

The architecture supports pruning, which removes nodes with weights below a threshold from the trained network as shown in figure 1b. This operation makes it possible to estimate the sufficient size of the model for the task, facilitates, and speeds up its work (see figure 1c).



a) b) c)

a- initial state of the network, input and output data are related by the formula $z(x,y)=e^{\sin(\pi x)+y^2}$

b- network after training c- network after pruning

Figure 1. KAN architecture diagram [13]

The authors of CAS note its low learning rate. The training was implemented on a central processing unit (CPU) with small data sets using software implemented in Python and made publicly available [15].

In [16], KAN is used to predict satellite traffic time series. The authors compare this approach with MLP and conclude that KAN has higher efficiency and performance. Their 4-layer MLP network with 329,000 parameters achieved an MSE of 6.12×10-3 after training, while the 4-layer KAN network with 109,000 parameters achieved an MSE of 5.08×10-3. As in the original work [13], a paper [16] indicates a positive effect of increasing the parameters G and k on the accuracy of the model.

The novelty of KAN determines the lack of assessments for its use in terms of speed and quality of training on large and complex data. There is no information about the specific KAN requirements for computing devices. This makes it difficult to use KAN effectively in the context of this work. However, KAN implementations already exist that enable training using graphics processing units (GPUs).

3. Structure of REM cell occupancy prediction model based on KAN

The LTE simulation model [14, 17] generates output files with a step of T LTE frames, their structure is presented in Table 1. The LTE network resource grid is defined as the passing traffic in a cell for 10 ms.

Table 1. Structure of the simulation model output file

Parameter	Data size and type	Example values	Description
rem	X×Y cells by L×20 bool	5×5 cells by 156×20 bool	REM grid
frame	unit	2900090000	LTE frame number

Each file contains an X×Y grid of REM cells represented by L rows of resource blocks (RBs) in the frequency domain and 20 slots (LTE frame) in the time domain [14, 18]. Each cell provides a matrix of L×20 samples that represents the busy state of a time-frequency resource of size [0.5 ms; 180 kHz].

We have developed an algorithm (see Figure 2) involving data collection, training and testing of KAN to predict the state of REM cells, which consists of the following steps:

1. Block reading the configuration file «save.mat» and output files of the LTE simulation model, unpacking them into a preliminary structure presented in Table 2. All structure values are normalized.

2. A block for reading the reference file of the user schedule «week.mat» with a table of correspondence between the day number and the following parameters: day of the week (Mon-Sun), type of day (working, weekend, holiday), week number (1-4).

Obtaining inputs and outputs for the predictive model. The input data is calculated based on the fields «line», «sample», «numday» and the content of the file «week.mat», while the output data corresponds to the field «status» (Table 2).

3. Block for combining and splitting the received data into sets: training (60%), testing (20%) and validation (20%).

4. Block for initializing the KAN model, training, testing and saving for subsequent use in the LTE simulation model.

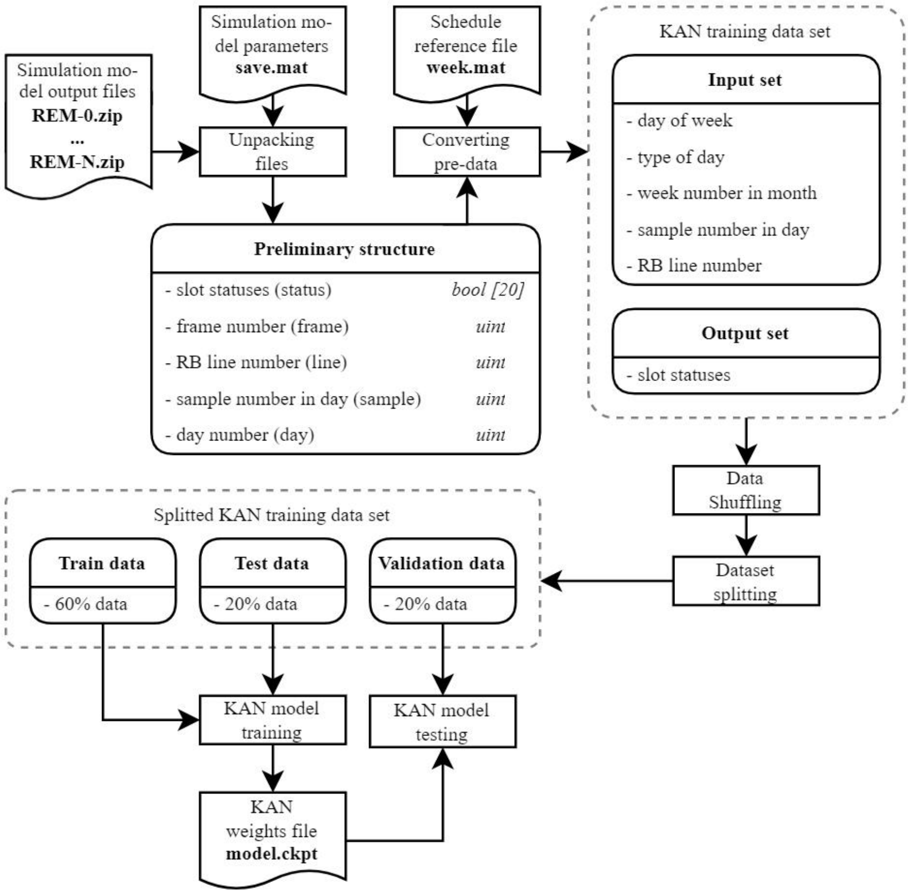


Figure 2. Data collection, training and testing of KAN to predict channel resources

Table 2. Preliminary data structure

Parameter	Data size and type	Range	Description
status	20 bool	0 or 1	Slot statuses on the RB line
frame	uint	0 ... N	Frame number
line	uint	1 ... 156	RB line number
sample	uint	1 ... 864	Sample number in day
numday	uint	1 ... 336	Day number

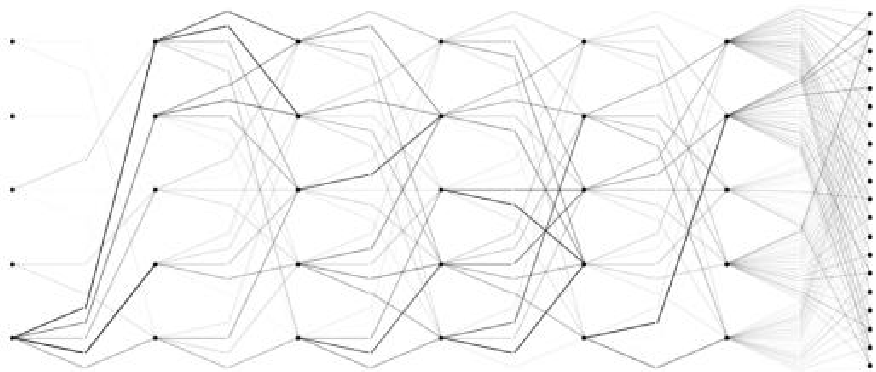
4. Experimental results

The developed forecasting model was implemented in Python 3.10. To implement KAN, the program code was adapted from the repository [19] using the PyTorch framework. The LTE model and data preparation are implemented in MatLab 2023.

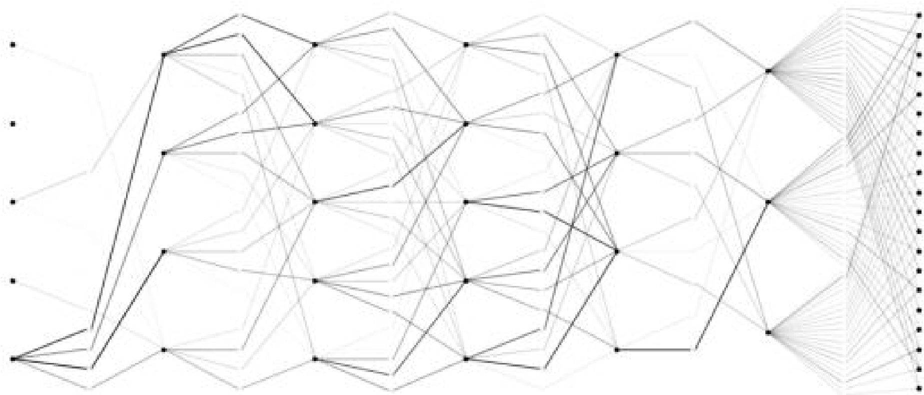
The LTE model generated data at 100 second intervals (T=10000) over simulated days over 12 months (864 samples per day). The generated data size was 894,660 records.

The KAN model was initialized and trained with parameters: G = 3, k = 3, LBFGS optimizer, MSE estimation, 500 epochs. The correct comparison of models for different architectures was ensured by choosing layer configurations in which the number of parameters for the models would be equal.

Figure 3a shows the KAN model after training. The input layer is shown on the left (5 nodes) and the output layer is shown on the right (20 nodes). The grey levels of the lines reflect the weight values of the model nodes. Figure 3b shows the KAN model after pruning. As a result of the operation, nodes with a weak effect on the prediction result were removed.



a)



b)

a- after training b- after pruning

Figure 3. KAN diagrams with different the number of nodes

The reduced model at the threshold th_{prune} =0.07 showed a decrease in accuracy by 5.72% when the number of parameters was reduced by 1.4 factor (model configuration after pruning [5, 4, 5, 5, 4, 3, 20]). Processing speed increased by 1.5 factor for the pruned model. Thus, in KAN applications where performance is important, the prune can be used with little loss of accuracy. Figure 4 shows the KAN and LSTM training graphs, and the resulting metrics are presented in Table 3.

Table 3. Test results of trained models based on various ANN architectures

Model	MSE in training	Actual accuracy, %
KAN	0.0499	92.564
Pruned KAN, th = 0.04	-	86.844
LSTM	0.0619	91.206

Actual accuracy was measured by binarizing the model results (y_{pred}) with a threshold of th_{bin} =0.5 and calculating the ratio of correctly predicted samples to all analyzed samples compared to the validation data y_{val} . The th_{bin} was selected experimentally using the formula:

$$th_{bin} = \min\left(MSE\left(y_{val}, y_{pred} > th\right)\right), \text{где } th = 0, 0.05, 0.1, ..., 1 \quad (1)$$

Figure 4 shows the KAN model learns faster and more accurately than the LSTM model with the same number of parameters. KAN provides a reduction in MSE by 1.5 factor compared to LSTM.

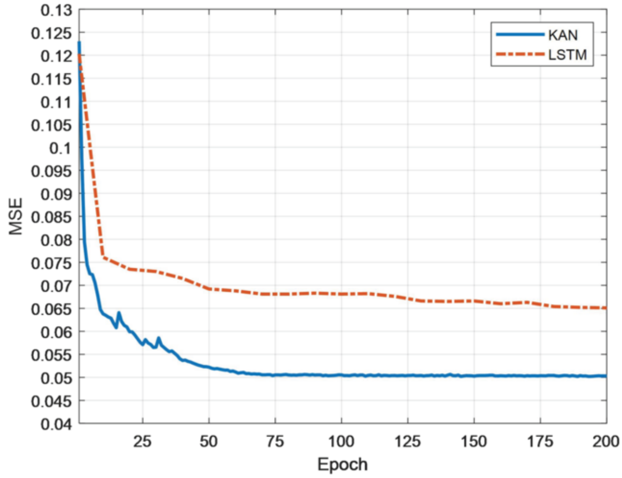


Figure 4. Prediction errors using KAN and LSTM models for first 200 training epochs

5. Conclusions

A model is proposed for predicting channel resources of an LTE cellular communication system, which can be used by secondary users with cognitive radio technology. An algorithm for creating and implementing a forecasting system is described, which includes the main steps: generating input data, training and testing a model based on the KAN architecture. The accuracy of the predictive model achieved was 92.564%. Comparison of forecasting results showed the advantage of using KAN over LSTM in terms of accuracy.

References

[1] Mahajan S, Harikrishnan R, Kotecha K. Prediction of Network Traffic in Wireless Mesh Networks Using Hybrid Deep Learning Model. IEEE Access. 2022, 10: 7003 - 7015
[2] Zhang C, Dang S, Shihada B, et al. Dual Attention-Based Federated Learning for Wireless Traffic Prediction. Proc. IEEE INFOCOM 2021 - IEEE Conference on Computer Communications. 2021:1-10
[3] Jaffry S, Syed F. Cellular traffic prediction using recurrent neural network. IEEE 5th International Symposium on Telecommunication Technologies. 2020: 94-98
[4] Zeng Q, Sun Q, Chen G, et al. Traffic Prediction of Wireless Cellular Networks Based on Deep Transfer Learning and Cross-Domain Data. IEEE Access. 2020, 8: 172387-172397
[5] Zhang C, Patras P. Long-term mobile traffic forecasting using deep spatio-temporal neural networks. Proc. 18th ACM Int. Symp. Mobile Ad Hoc Netw. Comput., 2018: 231–240
[6] Fu Y, Wang X. Traffic Prediction-Enabled Energy-Efficient Dynamic Computing Resource Allocation in CRAN Based on Deep Learning. IEEE Open Journal of the Communications Society. 2022, 3. 159-175
[7] Kurri V, Raja V, Periasamy P. Cellular traffic prediction on blockchain-based mobile networks using LSTM model in 4G LTE network. Peer-to-Peer Networking and Applications. 2021, 14. 1088-1105
[8] Alsaade F, Al-Adhaileh M. Cellular Traffic Prediction Based on an Intelligent Model. Mobile Information Systems. 2021:1-15
[9] García C, Koo I. Extremely Randomized Trees Regressor Scheme for Mobile Network Coverage Prediction and REM Construction. IEEE Access. 2023, 11: 65170 - 65180
[10] Stepanov N, Alekseeva D, Ometov A, et al. Applying Machine Learning to LTE Traffic Prediction: Comparison of Bagging, Random Forest, and SVM. 12th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops. 2020: 119-123
[11] Hervis Y, Plets D, Martinez A, et al. Radio Environment Map of an LTE Deployment Based on Machine Learning Estimation of Signal Levels. IEEE International Symposium on Broadband Multimedia Systems and Broadcasting. 2022: 1-6
[12] Chaves-Villota A, Viteri-Mera C. DeepREM: Deep-Learning-Based Radio Environment Map Estimation from Sparse Measurements. IEEE Access. 2023, 11: 48697 - 48714
[13] Liu Z, Wang Y, Vaidya S, et al. KAN: Kolmogorov-Arnold Networks. arXiv preprint arXiv:2404.19756. 2024:1-50
[14] Adamovskiy Y, Chertkov V, Bohush R. Model for building of the radio environment map for cognitive communication system based on LTE. Computer Research and Modeling. 2022, 14 (1):127-146
[15] Liu Z. KindXiaoming/pykan. https://github.com/KindXiaoming/pykan
[16] Vaca-Rubio, C. J., Blanco, L., Pereira, R., & Caus, M. Kolmogorov-Arnold Networks (KANs) for time series analysis. arXiv preprint arXiv:2405.08790. 2024:1-7
[17] Chertkov V, Bohush R, Adamovskiy Y. User data separation model in an LTE-based cognitive communication system. Information and Control Systems. 2023, 5: 43-54
[18] Adamovskiy Y, Bohush R, Chertkov V. Constructing Radio Environment Map Model for a Cognitive Communication System Based on LTE. Digital Technologies in Education, Science and Industry. 2022, 3382: 1-10
[19] Bethell D. team-daniel/KAN. GitHub. https://github.com/team-daniel/KAN. 2024