This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LWC.2018.2795605, IEEE Wireless Communications Letters

Spatio-Temporal Wireless Traffic Prediction with Recurrent Neural Network

Chen Qiu, Yanyan Zhang, Zhiyong Feng, Ping Zhang, Shuguang Cui

***Abstract*—Accurate prediction of user traffic in cellular networks is crucial to improve the system performance in terms of energy efficiency and resource utilization. However, existing work mainly considers the temporal traffic correlations within each cell while neglecting the spatial correlation across neighboring cells. In this paper, machine learning models that jointly explore the spatio-temporal correlations are proposed. Specifically, several recurrent neural network structures are utilized. Furthermore, a multi-task learning approach is adopted to explore the commonalities and differences across cells in improving the prediction performance. Base on real data, we demonstrate the benefits of joint learning over spatial and temporal dimensions.**

***Index Terms*—Spatio-temporal Model, Recurrent Neural Network, Multi-task Learning.**

I. Introduction

With the rapid development of wireless communication networks, there is an increasing demand of accurate cellular traffic prediction to improve the network performance. For example, to reduce the energy consumption of cellular networks, the functional base station sleeping mechanism could be adopted based on the knowledge of future traffic [1].

However, most existing prediction methods only consider the temporal traffic correlation within each cell to learn its pattern [2], neglecting the potential benefits of jointly considering spatial correlations across the entire network. Some efforts have already been made to model the spatio-temporal characteristics of wireless traffic [3], [4]. Since users continuously move within a given cellular network, the traffic flows across neighboring base stations are correlated, such that learning over both the spatial and temporal dimensions would improve the traffic prediction performance.

Artificial neural networks could be easily adapted to learn and predict the base station traffic over the temporal dimension. The authors in [1], [5] applied general artificial neural networks to predict the base station traffic under different wireless network setups. However, a regular neural network is hard to be generalized into the joint spatio-temporal setup, since it could not distinguish the spatial and temporal correlations. On the other hand, Recurrent Neural Network (RNN) is an extended form from regular neural networks such that it is capable of keeping the internal memory while processing the sequential inputs [6], which provides an effective way to jointly explore the spatio-temporal relationships. In this paper, we thus adopt RNN to exploit the spatial and temporal correlations among neighboring base stations.

In addition, multi-task learning is a promising way to improve the learning and predicting performance by jointly considering multiple inputs, while the different features between tasks could be utilized effectively [7]. In applying multi-task learning under our problem setup, we develop a multi-task learning approach and analyze the corresponding experimental results.

The rest of this paper is organized as follows. Section II describes the basic system model. In Section III, the learning architectures over the spatio-temporal model are proposed. In Section IV, experimental results are provided along with detailed analysis. Finally, Section V concludes the paper.

II. System Model

A. Problem Formulation

We first start with a general formulation of the problem. Consider a system with N base stations and let the observation of traffic volumes be over the past K time slots. In time t, let $\boldsymbol{x}_t = \left[x_t^1, x_t^2, \cdots, x_t^N\right]$ be the input vector with length N, which denotes the traffic volumes of all the base stations at time t. We know that if we merely consider the temporal correlation within each base station, the input sequence $\{\boldsymbol{x}_t\}$ would be degraded to a scalar sequence containing the current local traffic volume. Here our objective is to find a prediction function $\hat{\boldsymbol{x}}_{t+1} = f(\boldsymbol{x}_t, \boldsymbol{x}_{t-1}, \cdots, \boldsymbol{x}_{t-K+1})$ that achieves:

$$\min_f \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} L(\hat{\boldsymbol{x}}_{t+1}, \boldsymbol{x}_{t+1})$$

where the loss function $L(\cdot)$ measures the difference between the predicted and real traffic values. Nevertheless, a general solution for minimizing the above objective function would be intractable, which encourages us to approximate the optimal predicting function with a pre-defined structure.

B. Recurrent Neural Network

The RNN model can be expressed as $\boldsymbol{h}_{t-k} = g(\boldsymbol{x}_{t-k}, \boldsymbol{h}_{t-k-1}), k \in \{0, 1, \cdots, K-1\}$, where g is the transfer function applied on the observation window recursively, and $\boldsymbol{h}_{t-k}$ is the hidden state at time slot $t-k$, which is a function of both the previous neural network state $\boldsymbol{h}_{t-k-1}$ and the current input vector $\boldsymbol{x}_{t-k}$. As the previous state is

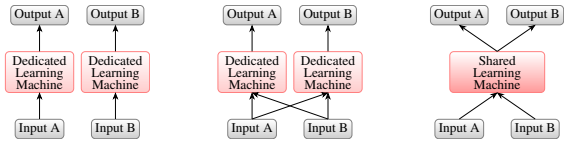
Chen Qiu, Zhiyong Feng and Ping Zhang are with Beijing University of Posts and Telecommunications, Beijing 100876 China (email: jp092983@bupt.edu.cn, fengzy@bupt.edu.cn, pzhang@bupt.edu.cn).

Yanyan Zhang is with Texas A&M University, College Station, TX 77843 USA (email: yanyan1016@email.tamu.edu).

Shuguang Cui is with University of California at Davis, Davis, CA 95616 USA (email: sgcui@ucdavis.edu).

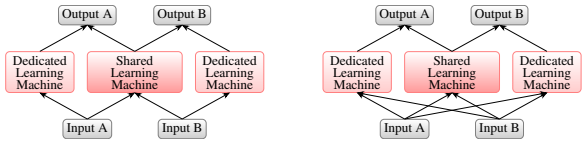
This work is supported in part by the National Natural Science Foundation of China with grants 61525101, 61631003 and the China Scholarship Council.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LWC.2018.2795605, IEEE Wireless Communications Letters



(a) 1-to-1 (b) n-to-1 (c) n-to-n

Fig. 1. Spatio-Temporal Learning Structures



(a) n-to-n with 1-to-1 (b) n-to-n with n-to-1

Fig. 2. Multi-task Learning Architectures

taken as one input, it carries the memory for learning from the internal correlations over time. When $k = 0$, the final $\boldsymbol{h}_t$ is considered to be a summary over all the past inputs, which can be used to produce the predicted cellular traffic volumes for the next time slot under our problem setup.

However, a simple RNN may have difficulties in handling long-term dependencies. Therefore, we adopt the widely used Long Short Term Memory (LSTM) structure [6] as a special RNN to realize the function g, which incorporates an additional vector $\boldsymbol{c}_{t-k}$ to carry the long-term memory. We refer readers to [6], [8] for more details about RNN and LSTM.

Based on LSTM, we further develop a multi-task learning approach that could deal with several prediction tasks at the same time to leverage the mutual benefits. By considering the traffic history over multiple base stations as samples drawn from different but related distributions, the joint spatio-temporal prediction is cast as simultaneous learning over several correlated tasks. As the wireless traffic volumes are generated in neighboring cells, the resemblance and dissimilarity across the multiple tasks are both important components to explore. Therefore, employing such a multi-task learning framework should lead to performance gains.

III. Learning Architecture

In this section, we first propose several spatio-temporal learning architectures for traffic prediction. Then we describe how to integrate those spatio-temporal learning architectures into a unified multi-task learning framework. Before we get into further details, let us consider a decomposition of our predictor f as $f = \psi \circ \xi$, where $\circ$ indicates applying function ψ on function ξ's output. The input data first go through the feature learning machine $\xi(\cdot)$, which is used to transform inputs into features. The second step involves the representation function $\psi(\cdot)$, which maps features into a prediction [9]. As we discussed in the previous section, we use RNN as the feature learning machine where we take the final hidden state $\boldsymbol{h}_t$, generated after applying the transfer function g for multiple steps over the observation window, as the output of $\xi(\cdot)$. The representation function $\psi(\cdot)$ that transforms the final state into a prediction is implemented as a fully connected feedforword neural network layer, given by $\psi(\boldsymbol{h}_t) = W\boldsymbol{h}_t$, where W is a trainable weight vector.

A. Basic Spatio-Temporal Learning Structures

As RNN naturally captures the temporal information, here we mainly focus on how to explore the spatial correlation across base stations. As shown in Fig. 1, three basic structures with different spatial information exploration schemes are first proposed, which could be later generalized into the multi-task learning framework. For simplicity, only a two-cell scenario is presented.

1) 1-to-1: The traffic volumes' history $X_t^i = \left[x_t^i, x_{t-1}^i, x_{t-2}^i, \cdots, x_{t-K+1}^i\right]^T$ of a particular base station i is used to predict its own traffic with a local learning machine ξ^i, which is actually a pure temporal model and mainly used as a benchmark. The prediction process for each base station is given by

$$\hat{x}_{t+1}^i = \psi^i \circ \xi^i(X_t^i). \tag{1}$$

2) n-to-1: In this architecture, the prediction for each base station would still be served by its own dedicated learning machine. However, the full set of traffic volumes $\boldsymbol{X}_t = \left[X_t^1, X_t^2, \cdots, X_t^N\right]$ from all base stations is provided to each learning machine for the joint exploration of the spatio-temporal information. The prediction process for each base station can be formulated as

$$\hat{x}_{t+1}^i = \psi^i \circ \xi^i(\boldsymbol{X}_t). \tag{2}$$

3) n-to-n: Different from the previous setup, no dedicated RNN blocks are used. Instead, a shared RNN block is adopted. All the traffic volumes are provided to this shared block ξ^S to produce the shared features for the prediction of all the base station traffics at the same time. Then we have the n-to-n prediction process for each base station as

$$\hat{x}_{t+1}^i = \psi^i \circ \xi^S(\boldsymbol{X}_t). \tag{3}$$

B. Multi-task Learning Architecture

In the sense of simultaneous learning, the n-to-n architecture in Fig. 1(c) could be seen as one special case of multi-task learning. However, such a n-to-n model is still a simple sequential layout of neural networks. As can be seen in (3), the predictions for different base stations are based on the same set of features, which implies that the differences between tasks could not be expressed effectively. To further clarify this, let us assume without the loss of generality that the loss function takes the following form $L(\hat{\boldsymbol{x}}_{t+1}, \boldsymbol{x}_{t+1}) = \frac{1}{N}\sum_{i=1}^{N} ||\hat{x}_{t+1}^i - x_{t+1}^i||$ and take the derivative of the loss function with respect to features as:

$$\frac{\partial L}{\partial \xi_j^S} = \sum_{i=1}^{N} \frac{\partial L}{\partial \psi^i} \frac{\partial \psi^i}{\partial \xi_j^S}. \tag{4}$$

When using the gradient descent [9] to minimize the loss, each feature represented in the shared part is always influenced by the other tasks. Thus the ability to represent the difference between base stations is limited under such a fully shared architecture.

To overcome this problem, we propose the multi-task learning architecture, which combines the shared and dedicated learning machines. Hence, the task-specific features could be generated and exploited to improve the performance. More

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LWC.2018.2795605, IEEE Wireless Communications Letters

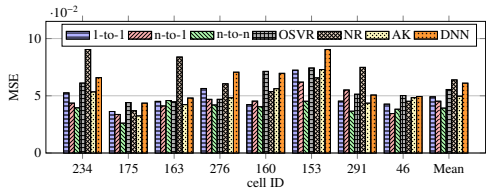
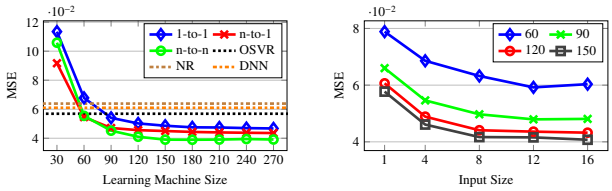


Fig. 3. Performance for Different Cells



(a) Effect of different learning machine size

(b) Effect of different input size

Fig. 4. Performance under different experiment parameters

specifically, the n-to-n architecture in Fig. 1 is combined with either the 1-to-1 or the n-to-1 architecture to form the multi-task learning architectures, as illustrated in Fig. 2. The predicting functions for the n-to-n with 1-to-1 and the n-to-n with n-to-1 models can be respectively cast as

$$\hat{x}^i_{t+1} = \psi^i \circ \left\{ \xi^S(\boldsymbol{X}_t), \xi^i(X^i_t) \right\}, \tag{5}$$

$$\hat{x}^i_{t+1} = \psi^i \circ \left\{ \xi^S(\boldsymbol{X}_t), \xi^i(\boldsymbol{X}_t) \right\}, \tag{6}$$

where the $\{\cdot,\cdot\}$ operator indicates the concatenation of two vectors. Under such a formulation there is one special set of features ξ^i generated for each base station i, which only serves a particular task, whose derivative in the loss function L is

$$\frac{\partial L}{\partial \xi^i_j} = \frac{\partial L}{\partial \psi^i} \frac{\partial \psi^i}{\partial \xi^i_j}. \tag{7}$$

These special feature sets are handled by the individual learning machines as shown in Fig. 2. The remaining n-to-n feature set ξ^S collects the common features shared among all the base stations, which is handled by a shared learning machine.

IV. Experiment Results

In this section, numerical experiments are conducted to demonstrate the effectiveness of the proposed spatio-temporal wireless traffic prediction framework. We first discuss the dataset and evaluation metrics. Then the results from different learning architectures are compared and analyzed.

A. Experiment Setup

Our methods are evaluated over a real cellular traffic data set collected from a big city in Asia. The data used in this work covers the traffic volumes of 16 different base stations within a 15-day period in year 2013, which is aggregated at one-hour intervals. Such a group of 16 base stations are located along some main streets; thus a high level of spatial correlations are presented.

In the experiments, we use the first 70% samples to train the learning model, and the remaining 30% to validate the results. The Mean Squared Error ($\frac{1}{N}\frac{1}{T}\sum_{i=1}^{N}\sum_{t=1}^{T}(\hat{x}^i_{t+1} - x^i_{t+1})^2$) is employed to measure the accuracy of traffic prediction for N base stations over T time steps. To make the result more comparable, MSE is measured on normalized data with standard deviation equal to 1 for each base station. The LSTM implementation given by Keras [10] is used in our experiments, where the recurrent dropout [8] is adopted to improve the result.

B. Result of Spatio-Temporal Learning

In this section, the capabilities of our spatio-temporal models are investigated by comparing with other existing methods. The reference approaches selected include the Online Support Vector Regression (OSVR) [11], the Nonparametric Regression (NR) [12], the Adaptive Kalman (AK) filter [13], and the Deep Neural Networks (DNN) [14].

The performance comparisons among different models are illustrated in Fig. 3. Since the neural network based methods are influenced by random initialization, the results of our spatio-temporal models and DNN are evaluated by averaging over 100 different runs. In addition, we use the Bayesian optimization algorithm [15] to tune the hyper-parameters for the comparing methods and present the optimized results here. We see that the RNN based models outperform all comparing approaches in most cases. The AK model is a linear model that is not able to explore the non-linear correlations, even with a much longer observation window. Although the OSVR and DNN models are very powerful, the laking of certain recurrent structures would make it very difficult to capture temporal correlations. Meanwhile, the NR model is trying to mimic the historic data and fails to actually capture the characteristics.

Among those proposed RNN based models in Fig. 1, the pure temporal model (1-to-1) is very often the worst. Although at some base stations it outperforms the n-to-1 model, the n-to-1 model is still the better one in most cases.

Another important observation is that the n-to-n model outperforms the n-to-1 model, i.e. instead of training a model for each base station, predicting those base stations all together could provide us even better results. This observation may be somehow counter-intuitive that a multi-objective optimization solution can outperform the dedicated solutions. However, by predicting multiple base station traffic volumes at the same time, the mapping from multiple inputs to multiple outputs could provide extra information and encourage the model to explore the spatial correlation among base stations. In addition, from the feature learning point of view, by introducing additional optimization objectives, we enforce the network to extract more general features from the training data and prevent overfitting issues.

Some experiments are also designed to illustrate the impact of several experiment parameters, which can help us better understand the spatio-temporal information embedded in our data set.

1) Size of Recurrent Network: Fig. 4(a) shows the result under different numbers of RNN neurons. The best performance of OSVR, NR and DNN models are also drawn as a reference. When RNN does not have enough neurons, the information representation ability is limited, especially for the n-to-n

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LWC.2018.2795605, IEEE Wireless Communications Letters

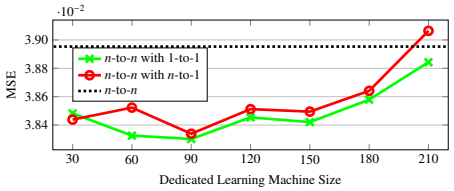


Fig. 5. Performance of multi-task learning

case, where overwhelmed information causes underfitting. By increasing the number of neurons, more features are extracted. However, the improvement stops after the learning machine size of 150 is reached. This experiment further shows that the n-to-n model explores extra information, which is extracted by the increased number of neurons.

2) Size of Spatial Input: In this experiment, the most correlated n "neighbors" with the highest correlation with the target base station are selected to provide the spatio-temporal information. As illustrated in Fig. 4(b), the different curves correspond to different sizes of the learning machine (number of neurons) used in the experiment. Although the n-to-n model is used, only the predicted result of the target base station is evaluated. In this way, the benefit of spatial information to a particular base station is presented. The overall performance is improved by the increased size of spatial inputs, but the improvement almost stops when the input size is greater than 8. The result is intuitive since the most correlated "neighbors" already contribute the majority of spatial information.

C. Result of Multi-task Learning

To validate the capability of multi-task learning, we chose the n-to-n model as the shared learning machine in the multi-task learning framework, where the size of the shared learning machines is set as 150. The best result achieved by the n-to-n model alone is also shown as a reference. As illustrated in Fig. 5, improved performance is achieved by both of two multi-task learning frameworks. However, the improvements in multi-task learning are not that obvious. The reason is that the n-to-n model and the multi-task learning frameworks share the same input and output structure. Since the same amount of information is presented to the learning machines, it is understandable that the improvements of multi-task learning are small, although different tasks in the n-to-n model may infer each other. In addition, our dataset is collected from a set of base stations covering similar geographic areas (along some main streets). As the data from different base stations share similar features with limited differences, the performance improvement of the multi-task learning models is limited.

Furthermore, the n-to-n with 1-to-1 framework performs slightly better than the n-to-n with n-to-1 one. Since the shared learning machine has explored the spatial correlation, providing spatial information to the dedicated learning machine would not further improve the performance; it may make the training more difficult to converge. In addition, we see that the performance gets worse with the size of the dedicated learning machine increase. This degenerated result is caused by the large dedicated learning machine size that dominates the behavior of the overall multi-task learning. Since the shared learning machine has captured the shared features, the dedicated features for each base stations become limited. The dedicated learning machines may then experience overfitting issues.

V. Conclusion

In this work, we presented multiple RNN based learning models along with unified multi-task learning frameworks to explore spatio-temporal correlations among base stations, in the goal of improving the traffic prediction performance. Base on real data, we provided detailed evaluation on different learning models and demonstrate that the spatial correlation among base stations could provide valuable information to improve the prediction accuracy. In addition, we showed that the commonalities and differences across different base stations could be better exploited by the proposed multi-task learning frameworks.

References

[1] J. Hu, W. Heng, G. Zhang, and C. Meng, "Base station sleeping mechanism based on traffic prediction in heterogeneous networks," in *Proceedings of the 2015 International Telecommunication Networks and Applications Conference*, Sydney, Australia, Nov. 2015, pp. 83–87.

[2] T. Lagkas, *Wireless Network Traffic and Quality of Service Support: Trends and Standards*, P. Angelidis and L. Georgiadis, Eds. IGI Global, Mar. 2010.

[3] X. Chen, Y. Jin, S. Qiang, W. Hu, and K. Jiang, "Analyzing and modeling spatio-temporal dependence of cellular traffic at city scale," in *Proceedings of the 2015 IEEE International Conference on Communications*, London, United Kingdom., Jun. 2015, pp. 3585–3591.

[4] H. Wang, J. Ding, Y. Li, P. Hui, J. Yuan, and D. Jin, "Characterizing the spatio-temporal inhomogeneity of mobile traffic in large-scale cellular data networks," in *Proceedings of the 7th International Workshop on Hot Topics in Planet-scale mObile Computing and Online Social neTworking*, Hangzhou, China, Jun. 2015, pp. 19–24.

[5] I. Loumiotis, E. Adamopoulou, K. Demestichas, P. Kosmides, and M. Theologou, "Artificial neural networks for traffic prediction in 4G networks," in *Proceedings of the 8th International Wireless Internet Conference*, Lisbon, Portugal, Nov. 2014, pp. 141–146.

[6] R. Jozefowicz, W. Zaremba, and I. Sutskever, "An empirical exploration of recurrent network architectures," in *Proceedings of the 32nd International Conference on Machine Learning*, Lille, France, Jul. 2015, pp. 2342–2350.

[7] R. Caruana, "Multitask Learning," *Machine Learning*, vol. 28, no. 1, pp. 41–75, Jul. 1997.

[8] Y. Gal and Z. Ghahramani, "A theoretically grounded application of dropout in recurrent neural networks," in *Proceedings of the 29th Advances in Neural Information Processing Systems*, Barcelona, Spain, Dec. 2016, pp. 1019–1027.

[9] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, May 2015.

[10] F. Chollet *et al.*, "Keras," https://github.com/fchollet/keras, 2015.

[11] M. Castro-Neto, Y.-S. Jeong, M.-K. Jeong, and L. D. Han, "Online-SVR for short-term traffic flow prediction under typical and atypical traffic conditions," *Expert Systems with Applications*, vol. 36, no. 3, Part 2, pp. 6164–6173, Apr. 2009.

[12] S. Clark, "Traffic prediction using multivariate nonparametric regression," *Journal of transportation engineering*, vol. 129, no. 2, pp. 161–168, Feb. 2003.

[13] J. Guo, W. Huang, and B. M. Williams, "Adaptive kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification," *Transportation Research Part C: Emerging Technologies*, vol. 43, no. 1, pp. 50–64, Jun. 2014.

[14] Y. Lv, Y. Duan, W. Kang, Z. Li, and F. Y. Wang, "Traffic flow prediction with big data: A deep learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 865–873, Apr. 2015.

[15] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. d. Freitas, "Taking the human out of the loop: A review of bayesian optimization," *Proceedings of the IEEE*, vol. 104, no. 1, pp. 148–175, Jan. 2016.