Deep Learning with Long Short-Term Memory for Time Series Prediction

Yuxiu Hua, Zhifeng Zhao, Rongpeng Li, Xianfu Chen, Zhiming Liu, and Honggang Zhang

Abstract—**Time series prediction can be generalized as a process that extracts useful information from historical records and then determines future values. Learning long-range dependencies that are embedded in time series is often an obstacle for most algorithms, whereas Long Short-Term Memory (LSTM) solutions, as a specific kind of scheme in deep learning, promise to effectively overcome the problem. In this article, we first give a brief introduction to the structure and forward propagation mechanism of LSTM. Then, aiming at reducing the considerable computing cost of LSTM, we put forward a Random Connectivity LSTM (RCLSTM) model by introducing stochastic connectivity to conventional LSTM neurons. Therefore, RCLSTM exhibits a certain level of sparsity and leads to a decrease in computational complexity. In the field of telecommunication networks, the prediction of traffic and user mobility could directly benefit from this improvement as we leverage realistic dataset to show that for RCLSTM, the prediction performance comparable to LSTM is available, whereas considerably less computing time is required. We strongly argue that RCLSTM is more competent than LSTM in latency-stringent or power-constrained application scenarios.**

I. Introduction

The analysis and prediction of time series has always been the key technique in an array of practical problems, including weather forecasting, transportation planning, traffic management, and so on. In the domain of telecommunications, intelligent mechanisms have already been designed to track and analyze a large number of time-dependent events, such as data traffic, user location, channel load, and service requests, etc [1]. On the other hand, with the explosive proliferation of mobile terminals as well as the expansion of mobile Internet, the Internet of Things (IoT) and cloud computing, the mobile communication network has become an indispensable social infrastructure that is bound up with people's lives and various areas of society [1]. However, it remains a challenging issue that how to guarantee the quality of service (QoS) and the quality of experience regardless of the dynamics of network traffic and user movements. One promising solution is to predict the varying pattern of data traffic and the location at which a mobile user will likely demand network service [1]. Accordingly, network operators can reserve some network resources, and react effectively to network changes in near real time [2]. However, since misplaced reservation of network resources and outdated predicted information will not only fail to support the desired QoS but also likely degrade the performance of the overall network, the prediction accuracy and complexity is of vital importance [3].

Time series analysis and prediction have been intensively studied for 40 years [4]. In statistical signal processing, the Autoregressive Integrated Moving Average (ARIMA) model has been used to study time-varying processes. However, one limitation of ARIMA is its natural tendency to concentrate on the mean values of the past series data. Therefore, it remains challenging to capture a rapidly changing process [5]. Support Vector Regression (SVR) has been successfully applied for time series prediction, but it also has disadvantages like the lack of structured means to determine some key parameters of the model [5]. In recent years, owing to the flexible structure, deep learning models are increasingly used in time series prediction [6]. Specifically, Recurrent Neural Networks (RNNs), one of deep learning models,

Y. Hua, Z. Zhao, R. Li and H. Zhang are with Zhejiang University, Hangzhou 310027, China, (email: {21631087, zhaozf, lirongpeng, honggangzhang}@zju.edu.cn).

X. Chen is with VTT Technical Research Centre of Finland, Oulu FI-90571, Finland (email: xianfu.chen@vtt.fi).

Z. Liu is with China Mobile Research Institute, Beijing 100053, China (email: liuzhiming@chinamobile.com).

This work was supported in part by National Key R&D Program of China (No. 2018YFB0803702), National Natural Science Foundation of China (No. 61701439, 61731002), Zhejiang Key Research and Development Plan (No. 2018C03056).

establish the reputation to cope with time series by recurrent neural connections. However, for any standard RNN architecture, the influence of a given input on the hidden layers and eventually on the neural network output would either decay or blow up exponentially when cycling around recurrent connections. To tackle this problem, Long Short-Term Memory (LSTM) has been revolutionarily designed by changing the structure of the hidden neurons in traditional RNN [7]. Today, research and applications of LSTM for time series prediction are proliferating. For example, Wang *et al.* [2] used LSTM-based model to predict the next-moment traffic load in a specific geometric area and Alahi *et al.* [3] predicted the motion dynamics in crowded scenes based on LSTM.

Generally, without customized hardware and software acceleration, the computing time of LSTM is proportional to the number of parameters. Given this disappointing characteristic, in this article, we present an approach to decrease the number of involved parameters, and thus put forward a new model that reduces the computational cost. Inspired by the interesting finding that Feed Forward Neural Networks (FFNNs) with sparse neural connections have a similar or even superior performance in many experiments compared to the conventional FFNNs [8], we introduce random connectivity to the conventional LSTM, thus forming a new architecture with sparse neural connections, called the Random Connectivity Long Short-Term Memory (RCLSTM).

Our simulation model is a three-layer stack RCLSTM neural network with a memory cell size of 300 per layer. Taking account of the significance of important typical scenarios and avoiding possible negative impact from datasets, we leverage both practical network traffic data from GÉANT networks—a pan-European research network [9], and realistic user-trajectory data [10]. Our simulation results show that when compared to LSTM, RCLSTM is highly capable of traffic prediction and user-location forecasting with less than half the neural connections. Particularly, in traffic prediction task, RCLSTM with even 1% neural connections performs better than ARIMA, SVR, and FFNN, while reducing the computing time by around 30% compared with the conventional LSTM. Moreover, the result of mobility prediction indicates that the prediction performance of RCLSTM is also comparable with the conventional LSTM.

II. An Overview of Artificial Neural Networks and LSTM

Artificial Neural Networks (ANNs) are constructed as a class of machine learning models that can eliminate the drawbacks of the traditional learning algorithms with rule-based programming [11]. Depending on the existence of the connection between the neurons within the same layer, ANNs can be classified into two main categories—FFNNs and RNNs. In FFNNs, there is no connection between the neurons within the same layer, and all neurons cannot be connected across layers, which means the information flows in one direction, from the input layer, through the hidden layers (if any), to the output layer. Instead, as depicted in Fig. 1, RNNs allow neurons within the same hidden layer to be connected, by calculating the output of the current moment from the input of the current moment and the hidden state of the previous moment. Therefore, RNNs embed historical input information in the network's internal state, and are thereby capable of mapping all of the historical input data to the final output. Theoretically, RNNs are more competent than FFNNs to handle such long-range dependencies. However, in practice, RNNs seem unable to accomplish the task. This phenomenon has been explored in depth by Hochreiter and Schmidhuber [7]. They explained some pretty fundamental reasons why such learning might be difficult.

Long Short-Term Memory networks, usually just called "LSTMs", are a special RNNs that are suitable for learning long-term dependencies [7]. The key part that enhances LSTMs' capability to model long-term dependencies is a component called memory block [7]. As illustrated in Fig. 1, the memory block is a recurrently connected subnet that contains functional modules called the memory cell and gates. The memory cell is in charge of remembering the temporal state of the neural network and the gates formed by multiplicative units are responsible for controlling the pattern of information flow. According to the corresponding practical functionalities, these gates are classified as the input gate, the output gate and the forget gate. The input gate control how much new information flows into the memory cell, while the

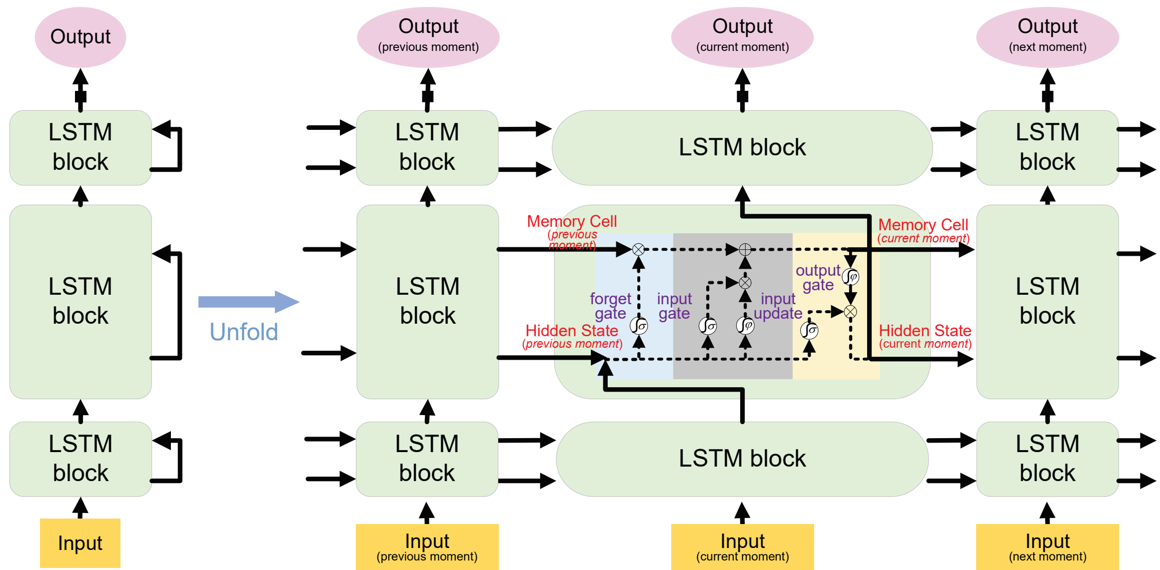

Fig. 1: An illustration of one three-layer LSTM network.

forget gate governs how much information of the memory cell still remains in the current memory cell through recurrent connection, and the output gate determines how much information is used to compute the output activation of the memory block and further flows into the rest of the neural network. Fig. 1 highlights the details of working mechanism of LSTM. Through the cooperation between the memory cell and the gates, LSTM is endowed with a powerful ability to predict time series with long-term dependences.

Since the invention of LSTM, a number of scholars have proposed several improvements with respect to its original architecture. Greff *et al.* [12] evaluated the aforementioned conventional LSTM and eight different variants thereof (e.g., Gated Recurrent Unit (GRU) [13]) on three benchmark problems—TIMIT, IAM Online and JSB Chorales. Each variant differs from the conventional LSTM by few single and simple changes. They found that the conventional LSTM architecture performs well on the three datasets, and none of the eight investigated modifications significantly improve the performance.

III. Random Connectivity for LSTM

The conventional LSTM (including its variants) follows the classical pattern that neurons in each block are fully connected and this connectivity cannot be changed arbitrarily. However, it has been found that for certain functional connectivity in neural microcircuits, random topology formation of synapses plays a key role and can provide a sufficient foundation for specific functional connectivity to emerge in local neural microcircuits [14]. This discovery is different from the conventional cases where neural connectivity is considered to be more heuristic so that neurons need to be connected in a more fully organized manner. It raises a fundamental question as to whether a strategy of forming more random neural connectivity, like in the human brain, might yield potential benefits to LSTM's performance and efficiency. With this conjecture, we built up the RCLSTM.

In RCLSTM, neurons are randomly connected rather than being fully connected as in LSTM. Actually, the trainable parameters in LSTM only exist between the input part—the combination of the input of the current moment and the output of the previous moment, and the functional part—the combination of the gate layers and the input update layer. Therefore, the LSTM architecture can be further depicted in Fig. 2. In our approach, whether the LSTM neurons are connected or not can be determined by certain randomness. Therefore, we use dashed lines to denote that the neural connections can be added or omitted, as depicted in the upper part of Fig. 2. If the neurons are fully connected, then it becomes a standard LSTM. On the other hand, if the neurons are randomly connected according to some rules (which are covered in detail below), then an RCLSTM is

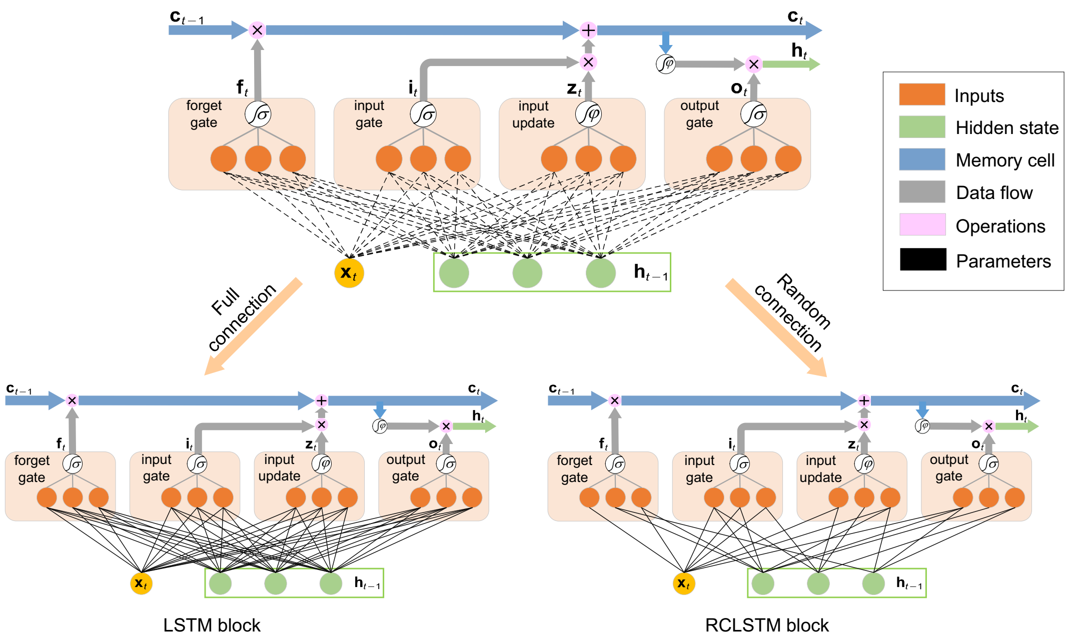

Fig. 2: The comparison between LSTM and RCLSTM in the view of generating process of neural connections.

created. The lower right part of Fig. 2 shows an example RCLSTM structure in which the neural connections are randomly sparse, unlike LSTM. The fundamental difference between RCLSTM and LSTM is illustrated in Fig. 2, so let us move to the implementation strategy of randomly connecting neurons.

First, we attach a probability value to each pair of neurons that are connected by a dashed line in the upper part of Fig. 2. The probability values can obey arbitrary statistical distributions, and we choose uniform distribution in our simulations given its computational efficiency. The probability value indicates the tendency that the corresponding pair of neurons will be connected. Then we assume all neurons are connected with the same probability and carefully set a threshold to determine the percentage of connected neurons. If the probability values are greater than the threshold, the corresponding pairs of neurons are connected. Otherwise, they are prohibited from being connected. This process can be visualized as turning dashed lines into solid lines, as shown in the right-hand transformation of Fig. 2. Therefore, the RCLSTM structure can create some sparsity, considerably decreasing the total number of involved parameters to be trained and reducing the computational loads of the whole RCLSTM network.

IV. Numerical Simulations for Traffic and Mobility Prediction

In this section, we focus on verifying the performance of the proposed RCLSTM on traffic prediction and user-location forecasting. In particular, we construct a three-layer RNN similar to the LSTM network in Fig. 1 but the recurrent memory blocks are replaced by the newly designed RCLSTM ones in Fig. 2 (for the sake of simplicity, this RNN is directly called RCLSTM in the following statement). First of all, we take advantage of RCLSTM to predict traffic and user mobility, particularly compare the prediction accuracy of RCLSTM with that of other algorithms or models. Then, we adjust the number of training data samples and the length of input sequences to investigate the influence of these factors on the prediction accuracy of RCLSTM.

A. Data Description and Evaluation Metrics

We evaluate the model's performance on traffic prediction depending upon real traffic data from a link in the GÉANT backbone networks [9]. GÉANT is a pan-European data source for the research and education communities. These traffic data are sampled every 15 minute during a 4-month period in 2005 and the unit for data points is Kbps. In this study, we select 10772 traffic data points therein. In order to make the training phase of ANN-based models converge faster and effectively

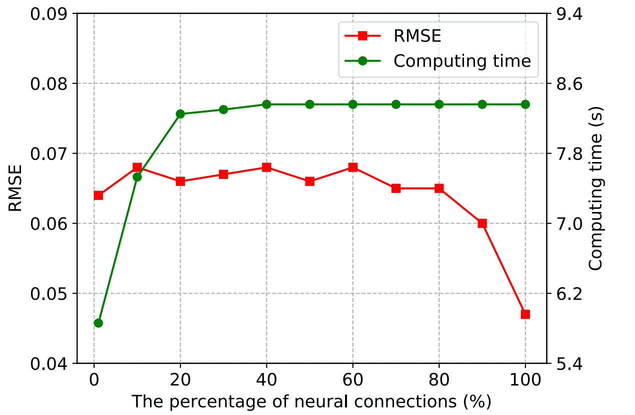

Fig. 3: The performance of RCLSTM in terms of RMSE and computing time.

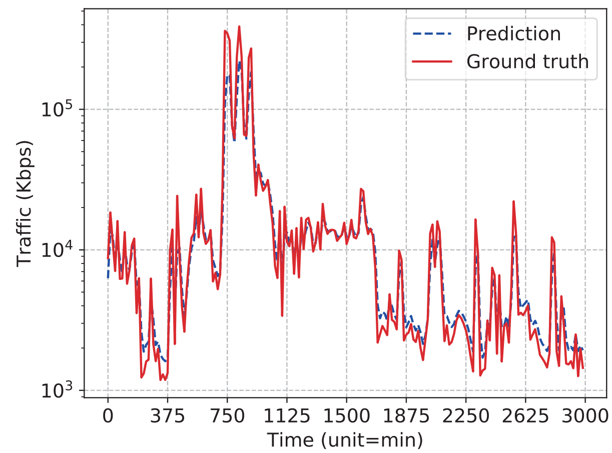

Fig. 4: Predicted traffic data and actual traffic data.

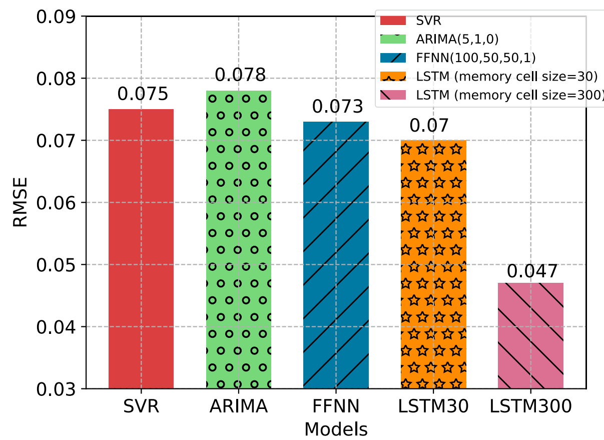

Fig. 5: Performance Comparison between RCLSTM and popular models.

avoid the bad local optimal solution [11], we first take the logarithm to base 10 of the raw data and then carry out a normalization process. Real-time prediction of data traffic requires continuous data input and learning. Hence, we introduce a notion of sliding window, which indicates a fixed number of previous timeslots to learn and then predict the current data traffic.

The other dataset to evaluate the capability of RCLSTM comes from [10], which contains the location history of several mobile users from 2015-08-06 to 2015-11-04 with manually marked important locations for every person in 1-hour intervals. The attributes of this mobility dataset include date-time, latitude, longitude, and assigned location ID. In this article, we select five users' trajectoires therein and assign their location IDs as one-hot vectors. Afterwards, consistent with the procedures to preprocess the GÉANT traffic dataset, we use the sliding window to slice the processed data and finally split them into training set and test set.

Finally, for the traffic prediction, Root Mean Square Error (RMSE) is applied to measure the difference between the predicted values and the actual ones. On the other hand, the accuracy level, which is defined as the ratio of the number of correct predictions to the total number of predictions, is chosen to evaluate the human mobility prediction results.

B. Testing Results and Analyses

1) Traffic Prediction

Fig. 3 reveals the RMSE and the computing time under different percentages of neural connectivity in RCLSTM (note that 100% connectivity means the conventional LSTM). Notably, the probability of neural connections obeys a uniform distribution between 0 and 1. In addition, the size of RCLSTM's memory cell is set at 300, while the ratio between the number of training samples and the number of test samples is set at 9:1, and the length of input traffic sequences is 100. Fig. 3 shows that the RMSE of RCLSTM is slightly larger than that of LSTM, but RCLSTM with very sparse neural connections (i.e. 1%) reduces the computing time by around 30% compared with the baseline LSTM. In addition, the computing time almost stops increasing when the percentage of neural connections is larger than 20%, which reflects that the method for accelerating calculation only works efficiently on extremely sparse matrices. Fig. 4 intuitively illustrates the actual and predicted traffic values by RCLSTM. It can be observed from the figure that the predicted values can match the variation trend and features of the actual values very well. Therefore, the simulation results indicate that RCLSTM can yield acceptable prediction capability, and effectively decrease the computational loads and complexity.

We further compare RCLSTM with three well-known prediction techniques—SVR, ARIMA, and FFNN. The hyper-parameters of these algorithms are as follows:

- SVR: The number of input features is 100, the kernel is a Radial Basis Function (RBF) and the tolerance for the stopping criterion is 0.001.
- ARIMA(p, d, q): The number of autoregressive terms (i.e. p) is 5, the number of nonseasonal differences needed for stationarity (i.e. d) is 1, and the number of lagged forecast errors in the prediction equation (i.e. q) is 0.
- FFNN: The number of input features is 100 and the numbers of neurons in both the first hidden layer and the second hidden layer are 50.

Since LSTM with a memory cell size of 30 has almost as many trainable parameters as RCLSTM with a memory cell size of 300 and 1% neural connections, we put it into the comparison list as well. The simulation results are shown in Fig. 5, which reveals that LSTM with a memory cell size of 300 performs much better than the others, followed by RCLSTM with the memory cell size of 300 and 1% neural connections. Interestingly, RCLSTM performs better than LSTM with the memory cell size of 30, which is probably due to a degree of overfitting that exists in the latter [11].

2) Human Mobility Prediction

The results of user-mobility prediction with RCLSTM are shown in Fig. 6, which demonstrates the prediction accuracy for the five users by RCLSTM, where the probability of neural connections obeys a uniform distribution between 0 and 1. In addition, the size of the memory cell is 150, the length of input sequences is 12, and the training samples account for 90% of the processed data. There is a slight difference in the prediction results for different users. For example, Users A and D are both university students with a part-time job, and thus they almost follow the same behavioral pattern in school or work, which results in highly expected predictability. On the other hand, User E is running his own business and is more likely to travel a lot with unfixed schedule, consequently having low expected predictability [10]. Although the prediction accuracy of RCLSTM is not as good as that of LSTM, RCLSTM with high sparsity of neural connections can compute faster than LSTM, similar to the traffic prediction scenario.

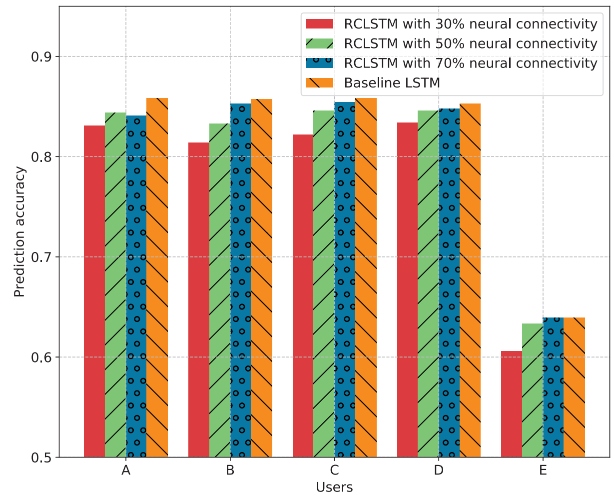

Fig. 6: Prediction accuracy of RCLSTMs for different users.

Remark. *RCLSTM shows promise for manifesting strong traffic and user-mobility prediction capabilities while reduces the number of parameters to be trained, which in effect decreases the computational load and complexity.*

V. Conclusion

In this article, we have addressed the importance of leveraging deep learning for time series prediction. In particular, we have reinvestigated the issues of traffic prediction and user-mobility forecasting with deep learning and proposed a new model named RCLSTM by revolutionarily redesigning the conventional LSTM. The basic idea behind RCLSTM is to construct neural networks by forming and realizing a random sparse graph. We have checked the effectiveness of RCLSTM by predicting the dynamics of traffic and user locations through various temporal scales. In traffic prediction, we have demonstrated that RCLSTM with 1% neural connections reduces the computing cost by 30% compared with the conventional LSTM. Although the characteristic of sparse neural connections may cause a performance degradation of approximate 25%, RCLSTM still outperforms SVR, ARIMA, FFNN, and LSTM with the same number of parameters. As for user-mobility prediction, we can safely draw the conclusion that RCLSTM is quite close to LSTM in terms of prediction accuracy. In summary, it can be expected that RCLSTM with lower computing costs and satisfactory performance will play an essential role in time series prediction in the future intelligent telecommunication networks.

References

[1] R. Li, *et al*, "Intelligent 5G: When Cellular Networks Meet Artificial Intelligence," *IEEE Wireless Commun.*, vol. 24, no. 5, Oct. 2017, pp. 175–183.

[2] J. Wang, *et al*, "Spatiotemporal Modeling and Prediction in Cellular Networks: A Big Data Enabled Deep Learning Approach," *Proc. 36th IEEE INFOCOM*, Atlanta, GA, USA, May 2017, pp. 1–9.

[3] A. Alahi, *et al*, "Social LSTM: Human Trajectory Prediction in Crowded Spaces," *Proc. 29th CVPR*, Las Vegas, NV, USA, June 2016, pp. 961–971.

[4] N. I. Sapankevych and R. Sankar, "Time Series Prediction Using Support Vector Machines: A Survey," *IEEE Comput. Intell. Mag.*, vol. 4, no. 2, May 2009, pp. 24–38.

[5] W.-C. Hong, "Application of Seasonal SVR with Chaotic Immune Algorithm in Traffic Flow Forecasting," *Neural Computing and Applications*, vol. 21, no. 3, Apr. 2012, pp. 583–593.

[6] I. H. Witten, *et al*, *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, Burlington, MA, USA, 4th edition, 2016, https://www.amazon.com/exec/obidos/ASIN/0128042915/departmofcompute, accessed on 11/30/2018.

[7] S. Hochreiter and J. Schmidhuber, "Long Short-term Memory," *Neural Computation*, vol. 9, no. 8, 1997, pp. 1735–1780.

[8] M. J. Shafiee, *et al*, "Stochasticnet: Forming Deep Neural Networks via Stochastic Connectivity," *IEEE Access*, vol. 4, 2016, pp. 1915–1924.

[9] S. Uhlig, *et al*, "Providing Public Intradomain Traffic Matrices to the Research Community," *SIGCOMM Comp. Commun. Rev.*, vol. 36, no. 1, Jan. 2006, pp. 83–86.

[10] J. Tkačík and P. Kordík, "Neural Turing Machine for Sequential Learning of Human Mobility Patterns," *Proc. IJCNN 2016*, Vancouver, BC, Canada, July 2016, pp. 2790–2797.

[11] Y. LeCun, *et al*, "Deep Learning," *Nature*, vol. 521, no. 7553, 2015, p. 436.

[12] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A Search Space Odyssey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, Oct. 2017, pp. 2222–2232.

[13] J. Chung, *et al*, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling," *CoRR*, vol. abs/1412.3555, 2014.

[14] S. L. Hill, *et al*, "Statistical Connectivity Provides a Sufficient Foundation for Specific Functional Connectivity in Neocortical Neural Microcircuits," *Proceedings of the National Academy of Sciences*, vol. 109, no. 42, 2012, pp. E2885–E2894.

Author Biographies

Yuxiu Hua is currently pursuing the doctor's degree with Zhejiang University, Hangzhou, China. His research interests include deep learning, reinforcement learning, and green communication.

Zhifeng Zhao is an Associate Professor with Zhejiang University, China. His research area includes collective intelligence and software-defined networks.

Rongpeng Li is now a assistant professor in Zhejiang University, Hangzhou, China. His research interests currently focus on multi-agent reinforcement learning and network slicing.

Xianfu Chen is currently a senior scientist with the VTT Technical Research Centre of Finland Ltd, Oulu, Finland. His research interests include the application of machine learning to cognitive radio networks, etc.

Zhiming Liu is with the China Mobile Research Institute. His research include wireless AI driven intelligent RAN optimization, etc.

Honggang Zhang is a Professor of Zhejiang University, China. He is currently involved in the research on Cognitive Green Communications.