A novel LSTM based deep learning approach for multi-time scale electric vehicles charging load prediction

Juncheng Zhu, Zhile Yang, Yan Chang, Yuanjun Guo, Kevin Zhu, Jianhua Zhang

Abstract—**Short-term load forecasting is an important issue in energy management system and a key measure to maintain the stable and effective operation of power systems, providing reasonable future load curve feeding to the unit commitment and economic load dispatch. Specifically, load forecasting at different time scales has various and corresponding roles in power systems. In recent years, plug-in electric vehicles have gradually become popular in major cities, and the number of electric vehicles is growing rapidly. However, the mass roll out of EVs may cause severe problems to the power system due to the huge charging power and stochastic charging behaviours of the EVs drivers. The accurate model of EVs charging load forecasting is therefore an emerging topic. In this paper, artificial neural networks and a long short term memory model based deep learning approaches are employed and compared in forecasting the EVs charging load from the charging station perspective. Numerical results show that the long short term memory model has demonstrated better performance and provided a model of higher accuracy in short-term EVs load forecasting comparing with the traditional artificial neural networks.**

I. Introduction

Power load forecasting has long been an important component in power and energy sectors and well studied since the 1980s [1]. Featured load forecasting is based on the operating characteristics of the system, capacity-enhancing decisions, natural conditions as well as the social impacts, determining load demand at particular time slots in the future under conditions and providing fundamental reference for power system scheduling. Accurate load forecasting could significantly contribute to various aspects including economical and reasonable arrangement of the start and stop of internal generator sets of a power grid, maintaining the safety and stability of the grid operation, reducing the unnecessary rotating reserve capacity, and eventually improving economic and social benefits. Load forecasting could be divided into ultra-short-term load forecasting, short-term load forecasting, medium-term load forecasting and long-term load forecasting depending on the expected time length. The ultra-short-term load forecasting refers to the load forecast within 1 hour ahead. In the safety monitoring state, the predicted values of 5 to 10 second or 1 to 5 minute are required, and the preventive control and emergency state processing require a predicted value of 10 minute to 1 hour. Various research approaches have been developed in predicting the power demand in multiple time scales.

The state-of-the-art power load forecasting models can be categorized into traditional statistical models and artificial intelligence (AI) models. Traditional forecasting methods majorly include the time series method [2], autoregressive integrated moving average [3], regression analysis [4], Kalman filtering [5], etc. On the other hand, AI methods consist of artificial neural networks (ANN) [6], support vector machines (SVM) [7], and deep learning methods [8]. Before the 21st century, due to the strong adaptive, self-learning and generelization ability, ANN had become an important technique in load forecasting. Hippert et al. [9] reviewed the application of ANNs for load forecasting and claimed that ANNs have effectiveness for load forecasting in terms of the accuracy and efficiency.

In recent years, the deep learning methods have been obtained wide attractions and used in image semantic segmentation [10], classification [11], target detection [12], natural language processing [13] and many other science and engineering fields. The network structure constructed by the deep learning methods are more complex with a large number of hidden layers and/or recurrent structure, which endowing stronger learning and self-adaptive ability than ANN methods. Therefore, it has also been paid attention in the field of load forecasting. In 1998, Vermaak and Botha [14] used recurrent neural network (RNN) for the first time to establish a short-term load forecasting model. However, the conventional RNN would suffer from the gradient vanishing problem and the long short term memory (LSTM) network is an effective approach to relief the issue. More recently, Marino et al. [15] proposed a LSTM architecture to forecast the load of individual residences [16]. Kong et al. [17] combined the energy consumption of a residence with the behaviour of a resident, converted the behaviour patterns of energy consumers into a sequence of

This research is financially supported by China NSFC under grants 61773252, Natural Science Foundation of Guangdong Province under grants 2018A030310671, China Post-doctoral Science Foundation (2018M631005), and State Key Laboratory of Alternate Electrical Power System with Renewable Energy Sources (Grant No. LAPS18020)

Juncheng Zhu is the Zhengzhou University, Zhengzhou, 450001, China (e-mail: jczhu_siat@163.com)

Zhile Yang and Yuanjun Guo are with Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, Guangdong, 518055, China (e-mail:zl.yang,yj.guo@siat.ac.cn)

Yan Chang is with University of Science and Technology of China, Hefei 230026, China (e-mail: sa517011@mail.ustc.edu.cn)

Kevin Zhu is with Lynbrook High School, San Jose, CA, USA, (email:Kevinrzhu123@gmail.com)

Jianhua Zhang is with the State Key Laboratory of Alternate Electrical Power System with Renewable Energy Sources, North China Electric Power University, Beijing 102206, China (e-mail: zjh@ncepu.edu.cn)

978-1-7281-3520-5/19/$31.00 ©2019 IEEE

input features to the network, thanks to which the accuracy of the load forecasting was improved. Some other studies [18, 19, 20, 21] also used LSTM in forecasting load demand, whereas Lu et al. [22] used gated recurrent units and achieved a more accurate results. Experimental results of these studies show that deep learning models have higher accuracy in load forecasting than the traditional methods.

In the recent years, plug-in electric vehicles (PEVs) have been widely emerged in the global market [23]. The large penetration of PEVs would bring significant uncertainty to the power demand and challenge the current power system scheduling to a large extent. A relatively accurate and efficient PEVs charging load forecasting is critical for the maintenance and operation of charging aggregator and is potential to relief the spike brought by stochastic PEVs charging. Mu et al. [24] presented a Spatial-Temporal model to evaluate the impact of large scale deployment of PEVs on urban distribution networks. Qian et al. [25] proposed a methodology for modeling and analyzing the load demand in a distribution system due to EV battery charging by Monte Carlo Simulation. Alizadeh et al. [26] proposed a stochastic model based on queuing theory for EV and PHEV charging load demand analysis. Luo et al. [27] proposed a Monte Carlo simulation based model in order to forecast the charging load of PEVs in China in 2015, 2020, 2030. However, these traditional methods see large difficulties in quantifying the external factors which affect the charging load of PEVs.

In this paper, a novel LSTM based model is established particular for forecasting EVs charging station load, where actual EVs charging stations dataset are adopted for model training and validation. In terms of the time slot, ultra-short-term load forecasting at two different time scales are adopted. The rest of the paper is organised as follow: Section II briefly describes the long short term memory networks and the load forecasting framework based on the LSTMs; Section III introduces the dataset and proposes the data pre-processing method; the experimental results are shown in Section IV, followed by Section V which concludes the paper and outlooks the future research.

II. Preliminaries

Deep learning models have strong learning and generalization ability and would be competitive in complex forecasting tasks. This chapter introduces the structure of the LSTM and the novel load forecasting framework for PEVs load forecasting.

A. Long-short term memory

The recurrent neural network is a model based on sequence input information. Traditional RNN structure has a gradient explosion and the gradient disappears problem when dealing with time-series and long-delay tasks. The LSTM neural network is a variant of the RNN. It is first proposed by Hochreiter and Schmidhuber [16] in 1997 and has been proved to have good performance in dealing with the gradient vanish problem. The basic LSTM structure is shown in Fig. 1, where the formulation of updating the cell states and parameters are as follows:

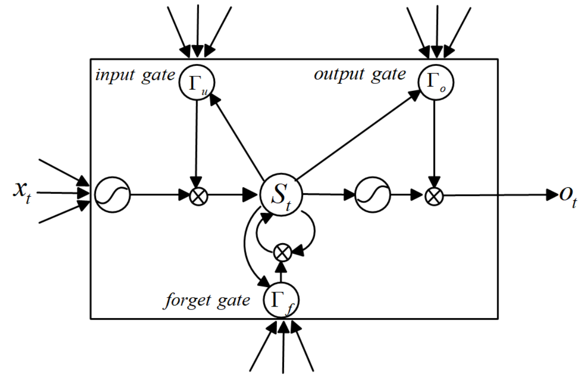

Fig. 1. Basic LSTM structure

$$\Gamma_f = \sigma(W_f[o_{t-1}, x_t] + b_f) \tag{1}$$

$$\Gamma_i = \sigma(W_i[o_{t-1}, x_t] + b_i) \tag{2}$$

$$\Gamma_o = \sigma(W_o[o_{t-1}, x_t] + b_o) \tag{3}$$

$$\widetilde{S}_t = tanh(W_s[o_{t-1}, x_t + b_s]) \tag{4}$$

$$S_t = \Gamma_i * \widetilde{S}_t + \Gamma_f * S_{t-1} \tag{5}$$

$$o_t = \Gamma_o * tanh(S_t) \tag{6}$$

The problem with traditional RNN is that due to the imperfect neuron function, the network will remember the information that should be forgotten, and forget the information that should be remembered. To tackle this, LSTM creates a path that allows the gradient to flow for long periods of time by introducing an intelligently controlled self-circulating cycle. These controlled self-circulations are composed of the input gate (Γ_u), forget gate (Γ_f) and output gate (Γ_o). The input gate determines the update of the hidden layer information, whereas the forget gate determines whether the updated information contains the information of the last moment. The output gate determines which part of information will be selected. Moreover, o_{t-1} is the output at $t-1$ time slot, and x_t is the input at current moment, $\widetilde{S}_t$ is the new candidate values, and S_t is the memory from current block, S_{t-1} is the memory from previous block, W is the weight, b is the bias, and symbol $*$ is the element-wise multiplier. In addition, σ is another activation function as follows:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{7}$$

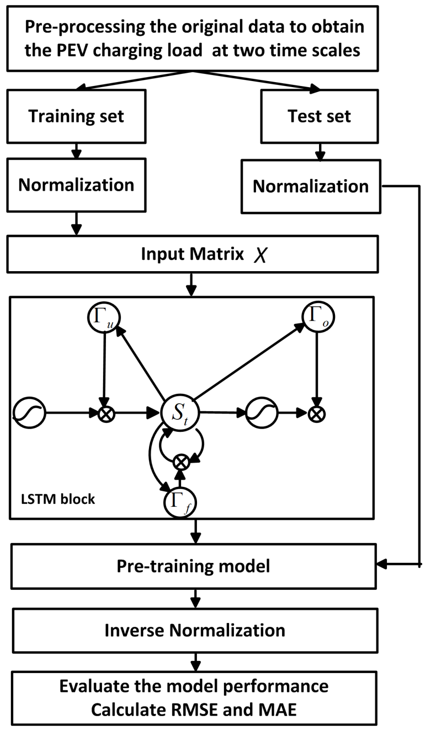

Fig. 2. Deep learing model based forecasting framework

B. PEVs Load forecasting framework based on LSTM

In this section, the ultra short-term forecasting framework for PEVs load is given in Fig. 2. The framework starts from data pre-processing for the inputs. Then, the data set is divided into a training set and a test set and all normalized. Further, the training set data is fed into the LSTM for pre-training, where 10% of the training set is used as a validation set. According to the performance of the model on the validation set, the hyper-parameters are well tuned to achieve the best performance. Finally, the pre-training model is obtained with the performance of the model evaluated by using the test set data, and the root mean squared error (RMSE) and mean absolute error (MAE) are calculated to verify the validity of the model. The full training process of LSTM is shown as follows:

1) Calculate the output value of each neuron by forward propagation, i.e Γ_u, Γ_f, Γ_o, S_t and o_t.
2) Determine the optimization objective function, where mean squared error (MSE) shown below is adopted as loss function:

$$MSE = \frac{1}{N}\sum_{i=1}^{n}(\hat{y} - x)^2 \tag{8}$$

where N is the number of samples, $\hat{y}$ is the forecasting value, and x is actual value.
3) The network weighting parameters are updated according to the gradient guidance of the loss function. The back-propagation of the LSTM error term consists of two levels: one is spatially level where the error term is propagated to the upper layer of the network. The other is the time level, which propagates back in time, that is, from the current t time, the error of each moment is calculated.
4) Jump to step (1) and repeat steps (1)-(3) until the network error converges.

An effective optimizer is crucial in deep learning model. In this paper, several optimizers including Adam [28], classic stochastic gradient descent (SGD), Adagrad [29], Adadelta [30] and RMSProp [31] have been adopted and compared, and Adam achieved better results than other optimizers. Therefore, Adam is selected as the optimizer for the training process.

III. Data analyis and forecasting process

Deep learning approaches are a kind of data-driven methods. Data analysis and data processing are the premise of building a load forecasting model. This section will give a brief introduction to the data set used, and carefully explain the adopted data preprocessing methods.

A. Introduction of the Dataset

The data set used in this paper was provided by a charging station operating company of PEVs in Shenzhen. The charging load data from July 2017 to July 2018, for a whole year about 60,000, was selected. The charging station has 24 charging piles. Due to the random charging start time of each vehicle and the selection of charging piles, we need to integrate and split the data in time dimension, and the original data will have outliers and missing values, therefore, data pre-processing is crucial for the accuracy of load forecasting.

B. Data pre-processing

The data process in this paper is divided into three stages including outlier processing step, time interval processing step and normalized processing step, all of which are shown as below subsections.

1) Box plot analysis: The box plot provides a standard for identifying outliers, i.e. the values above or below the upper and lower bounds of the box plot setting are outliers. We define the upper quartile and the lower quartile. The upper quartile is set to U, which means that only 1/4 of all samples are larger than U. The lower quartile is set to L, which means that only 1/4 of all samples are less than L. The interpolation of the upper quartile and the lower quartile is IQR, i.e. $IQR = UL$, then the upper bound is $U + 1.5IQR$ and the lower bound is: $L - 1.5IQR$. Next the samples that exceed the upper and lower bounds of the box plot are found, and the average processing is conducted to replace the outliers with the mean of the sequence data of the day.

2) Time interval processing: The data for one year was selected and the data was converted into training for each of the 30-minute multi-time scales data every 30 minutes. In the division of training data and test data, the ratio of 7:3 is roughly divided. The sliding window is used to set the time window to 1, the step size is 1 to intercept the matrix, and the last column of the matrix is used as the label.

3) Normalized processing: Since this experiment is based on various neural network models, the data measure bar has a great influence on the training effect and the results of the model. Therefore, the data should be standardized first to eliminate the influence of the data dimension on the result. The standardized method uses normalization, which is to scale all data to 0-1, called maximum and minimum normalization. The detailed process is shown as follows:

$$y = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{9}$$

IV. Experimental Results and Discussion

In this section, the preprocessed data was fed into four deep learning models for performance evaluation and comparison. All training process for PEVs charging load forecasting are implemented in a workstation with 3.0 Ghz Intel i7 and 64GB RAM, of which the GPU is Geforce Nvidia GTX-1080Ti and all codes are run in Keras library [32] with Tensorflow [33] backend.

A. Model Evaluation

To access the effectiveness of the models, two widely used metrics are employed including RMSE and MAE [34]. Other metrics are not proper considering the features of EVs load characteristics.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \tag{10}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{n}|\hat{y}_i - y_i| \tag{11}$$

B. Experimental Results

As shown in Fig. 3 and Fig. 4, the data set at 30-minute intervals and the data set at 15-minute intervals can converge during training process and exhibit the best performance on the validation set.

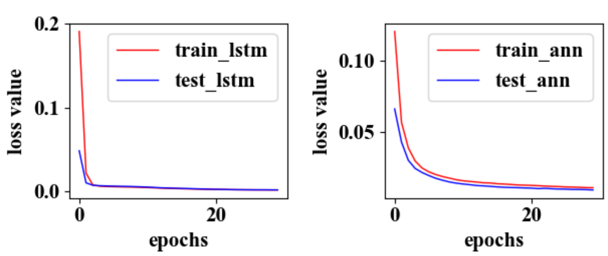

Fig. 3. Loss curve of the data set at 30-minute intervals

A load forecasting situation on a certain day in the training set and the day in the test set is randomly selected, and a picture is plotted to visually see the model prediction effect. Fig.5 is a data prediction result graph with 30-minute intervals and contains 48 load points. Fig.6 is a graph of data prediction results at intervals of 15 minutes, containing 96 load points.

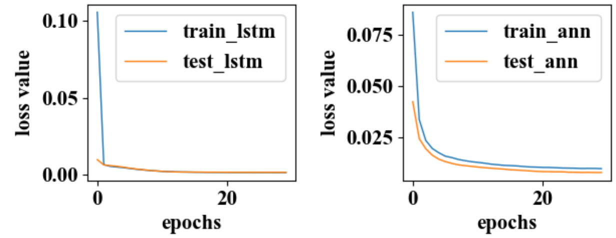

Fig. 4. Loss curve of the data set at 15-minute intervals

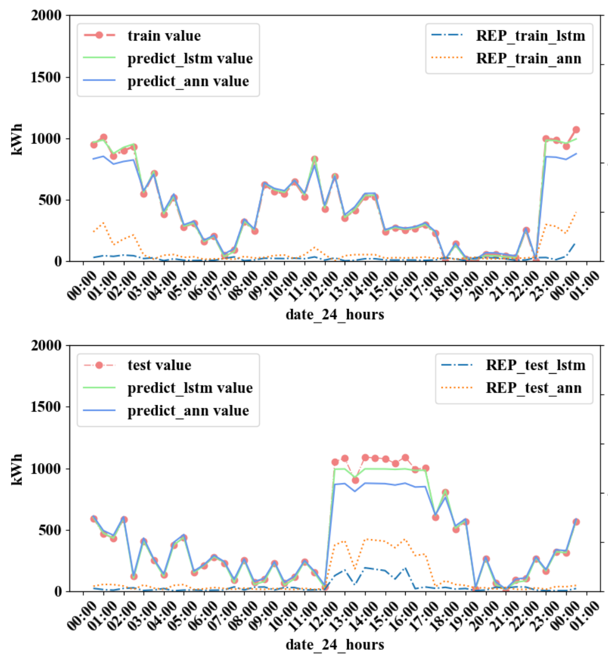

Fig. 5. Predictions and REP of the 30-minute intervals

Both figures show the degree of fit between the predictions of the ANN model and the LSTM model with the original data, where the real-time error curves for each point are also plotted. From the figure, it can seen that the two models are not ideal for daily peak load prediction, but the LSTM model has smaller errors.

During the experimental process, we verified the load prediction effects of the two models at two different time scales, and the performance comparisons are shown in Table I. It can be seen from the table that compared with the prediction error of the ANN model, the prediction error of the LSTM model on the two time scales is only one-third of that of the ANN model. Moreover, the RMSE and MAE of the LSTM model in the test set at 15-minute intervals is 12.302 and 5.509, while the ANN sees worse results as 36.630 and 21.755. Through the comparison the prediction errors at two different time scales, it could be found that with the time interval becomes smaller, the amount of data increases, and the prediction error will also decrease. The LSTM model has less than one-half the prediction error of the test set at intervals of 30 minutes

TABLE I
PERFORMANCE COMPARISON

Model	Time Scale	Train-RMSE	Test-RMSE	Train-MAE	Test-MAE
ANN	15min	28.378	36.630	17.445	21.755
	30min	59.187	76.440	34.699	43.107
LSTM	15min	9.284	12.302	3.763	5.509
	30min	22.008	28.250	13.562	16.854

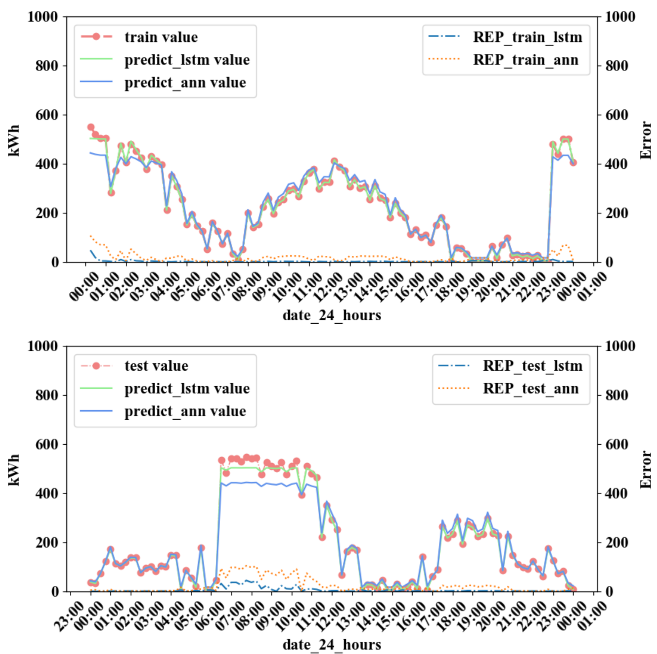

Fig. 6. Predictions and REP of the 15-minute intervals

comparing the error at 15 minute intervals.

V. CONCLUSION AND FUTURE WORKS

This paper proposes a novel LSTM model to predict the charging load of PEVs at two different time scales. Compared with traditional artificial neural networks, LSTM has lower prediction error at both time scales. In future work, we will continue to explore load forecasting at other time scales and increase input characteristics such as real-time electricity prices, temperature, holiday information, morning and evening peaks, peak load information, etc. In addition, the model proposed in this paper only performs single-step forecasting. To construct a model that can perform multi-step prediction will be addressed in the future research.

REFERENCES

[1] G. Janacek, "Time series analysis forecasting and control," *Journal of Time Series Analysis*, vol. 31, no. 4, pp. 303–303, 2010.

[2] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[3] L. Wei and Z. Zhen-gang, "Based on time sequence of arima model in the application of short-term electricity load forecasting," in *Research Challenges in Computer Science, 2009. ICRCCS'09. International Conference on*. IEEE, 2009, pp. 11–14.

[4] T. Haida and S. Muto, "Regression based peak load forecasting using a transformation technique," *IEEE Transactions on Power Systems*, vol. 9, no. 4, pp. 1788–1794, 1994.

[5] R. Shankar, K. Chatterjee, and T. Chatterjee, "A very short-term load forecasting using kalman filter for load frequency control with economic load dispatch." *Journal of Engineering Science & Technology Review*, vol. 5, no. 1, 2012.

[6] D. C. Park, M. El-Sharkawi, R. Marks, L. Atlas, and M. Damborg, "Electric load forecasting using an artificial neural network," *IEEE transactions on Power Systems*, vol. 6, no. 2, pp. 442–449, 1991.

[7] B.-J. Chen, M.-W. Chang *et al.*, "Load forecasting using support vector machines: A study on eunite competition 2001," *IEEE transactions on power systems*, vol. 19, no. 4, pp. 1821–1830, 2004.

[8] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[9] H. S. Hippert, C. E. Pedreira, and R. C. Souza, "Neural networks for short-term load forecasting: A review and evaluation," *IEEE Transactions on power systems*, vol. 16, no. 1, pp. 44–55, 2001.

[10] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[12] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[13] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[14] J. Vermaak and E. Botha, "Recurrent neural networks

for short-term load forecasting," *IEEE Transactions on Power Systems*, vol. 13, no. 1, pp. 126–132, 1998.

[15] D. L. Marino, K. Amarasinghe, and M. Manic, "Building energy load forecasting using deep neural networks," in *Industrial Electronics Society, IECON 2016-42nd Annual Conference of the IEEE*. IEEE, 2016, pp. 7046–7051.

[16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[17] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on lstm recurrent neural network," *IEEE Transactions on Smart Grid*, 2017.

[18] S. Bouktif, A. Fiaz, A. Ouni, and M. Serhani, "Optimal deep learning lstm model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches," *Energies*, vol. 11, no. 7, p. 1636, 2018.

[19] H. Zheng, J. Yuan, and L. Chen, "Short-term load forecasting using emd-lstm neural networks with a xgboost algorithm for feature importance evaluation," *Energies*, vol. 10, no. 8, p. 1168, 2017.

[20] A. Gensler, J. Henze, B. Sick, and N. Raabe, "Deep learning for solar power forecastingan approach using autoencoder and lstm neural networks," in *Systems, Man, and Cybernetics (SMC), 2016 IEEE International Conference on*. IEEE, 2016, pp. 002 858–002 865.

[21] J. Kumar, R. Goomer, and A. K. Singh, "Long short term memory recurrent neural network (lstm-rnn) based workload forecasting model for cloud datacenters," *Procedia Computer Science*, vol. 125, pp. 676–682, 2018.

[22] K. Lu, Y. Zhao, X. Wang, Y. Cheng, X. Pang, W. Sun, Z. Jiang, Y. Zhang, N. Xu, and X. Zhao, "Short-term electricity load forecasting method based on multilayered self-normalizing gru network [j]," *Energy Internet and Energy System Integration (EI2)*, 2017.

[23] Z. Yang, K. Li, and A. Foley, "Computational scheduling methods for integrating plug-in electric vehicles with power systems: A review," *Renewable and Sustainable Energy Reviews*, vol. 51, pp. 396–416, 2015.

[24] Y. Mu, J. Wu, N. Jenkins, H. Jia, and C. Wang, "A spatialtemporal model for grid impact analysis of plug-in electric vehicles ," *Applied Energy*, vol. 114, no. 2, pp. 456–465, 2014.

[25] K. Qian, C. Zhou, M. Allan, and Y. Yue, "Modeling of load demand due to ev battery charging in distribution systems," *IEEE Transactions on Power Systems*, vol. 26, no. 2, pp. 802–810, 2011.

[26] M. Alizadeh, A. Scaglione, J. Davies, and K. S. Kurani, "A scalable stochastic model for the electricity demand of electric and plug-in hybrid vehicles." *IEEE Transactions on Smart Grid*, vol. 5, no. 2, pp. 848–860, 2014.

[27] Z. Luo, Y. Song, Z. Hu, Z. Xu, Y. Xia, and K. Zhan, "Forecasting charging load of plug-in electric vehicles in china," in *Power and Energy Society General Meeting*, 2011.

[28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *Computer Science*, 2014.

[29] J. Duchi, E. Hazan, Singer, and Yoram, *Adaptive Subgradient Methods for Online Learning and Stochastic Optimization*. JMLR.org, 2011.

[30] M. D. Zeiler, "ADADELTA: an adaptive learning rate method," *CoRR*, vol. abs/1212.5701, 2012. [Online]. Available: http://arxiv.org/abs/1212.5701

[31] T. Tieleman and G. Hinton, "Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude," *COURSERA: Neural networks for machine learning*, vol. 4, no. 2, pp. 26–31, 2012.

[32] F. Chollet *et al.*, "Keras," 2015.

[33] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard *et al.*, "Tensorflow: a system for large-scale machine learning." in *OSDI*, vol. 16, 2016, pp. 265–283.

[34] C. J. Willmott, S. G. Ackleson, R. E. Davis, J. J. Feddema, K. M. Klink, D. R. Legates, J. O'donnell, and C. M. Rowe, "Statistics for the evaluation and comparison of models," *Journal of Geophysical Research: Oceans*, vol. 90, no. C5, pp. 8995–9005, 1985.