Sensors and Materials, Vol. 34, No. 4 (2022) 1275–1285
MYU Tokyo

S & M 2890

Prediction of Short-term Load of Microgrid Based on Multivariable and Multistep Long Short-term Memory

Dashuang Li*

School of Artificial Intelligence and Smart Manufacturing, Hechi University,
Yizhou, Hechi 546300, China

(Received June 17, 2021; accepted January 12, 2022)

Keywords: microgrid, load prediction, LSTM, multivariable and multistep

In a microgrid system, a phasor measurement device (PMU) is used to measure the electrical quantities of nodes, which can provide accurate data for system stability control. How to use the data measured using a PMU to improve the stability of a microgrid is an important practical problem. The mismatch between generation power and load power in a microgrid system will cause oscillation in the system. To ensure accurate and rapid load forecasting in a microgrid system and the reliable and safe operation of the microgrid, deep learning is introduced into microgrid load prediction, and a method of predicting the short-term load for a microgrid based on multivariable and multistep long short-term memory (MM-LSTM) is proposed in this paper. The method considers the effects of meteorological factors on load data and forecasts the current load situation from the load data and the temperature and humidity data of the previous period. A Keras-based model of the short-term load for microgrid prediction based on MM-LSTM is built and its parameters are optimized. Then, the load of a microgrid is predicted using the power consumption and meteorological data. The average absolute percentage error between the experimental results and the actual power consumption is 8.827%, demonstrating the effectiveness of the method.

1. Introduction

The power generation and consumption of a microgrid in normal operation should be consistent; otherwise, the system frequency and voltage amplitude will fluctuate, affecting the stable operation of the system.[1,2] The power source of a microgrid system is mainly composed of distributed generation, such as photovoltaic power or wind power generation. Distributed generation exhibits randomness and volatility.[3] To maintain the balance of the system power, in addition to adopting advanced control methods in the output control of distributed generation to adjust the power output of distributed generation and respond to the load fluctuation in real time, the load of a microgrid system is predicted in order to make a system generation plan, to program the generation capacity of distributed generation in advance, and to maintain the balance of the system power. There are many different load components in a microgrid system, and the short-term load fluctuation is relatively large. The load, particularly residential electricity

*Corresponding author: e-mail: lidashuang@hcnu.edu.cn
https://doi.org/10.18494/SAM3468

ISSN 0914-4935
https://myukk.org/

consumption, is greatly affected by meteorological factors, such as temperature and humidity. Therefore, it is very important to consider the impact of meteorological factors on the load and to quickly, accurately, and efficiently forecast the short-term load on the basis of historical meteorological load data.

At present, the mainstream short-term load forecasting methods can be roughly divided into traditional analysis and machine learning methods.[4] Traditional analysis methods, such as time series prediction, the autoregressive moving average method, and the multiple linear regression method, mainly use time series analysis.[5–7]

The model used in the above methods is simple and these methods are fast, but they have high requirements for the time series of data and cannot perform nonlinear fitting. The accuracy of these methods cannot meet the needs of power systems with the rise of regional interconnected and large-scale power grids, and the load data of power systems is growing explosively.[8,9] To deal with large-capacity data and nonlinear coupling data, and improve the accuracy of power system load forecasting, machine learning methods, such as the neural network, random forest (RF), and support vector machine (SVM) methods, have attracted increasing attention.[10,11] Machine learning methods can solve the nonlinear relationship between data; however, it is necessary to add time characteristics artificially to ensure the prediction accuracy in load forecasting.

The application of machine learning in the load forecasting of power systems has been studied. Munkhammar *et al*. utilized the Markov chain mixture distribution model (MCM) for the very short term load forecasting of residential electricity consumption to forecast one-step-ahead (30 min resolution) residential electricity consumption data from Australia.[12] Wang *et al*. proposed a multi-energy load prediction model based on deep multitask learning and an ensemble approach for regional integrated energy systems to ensure their operational efficiency and reliability, for which the accurate prediction of energy demand has become a crucial task.[13] Power system load series are nonlinear and nonstationary, which affect the prediction accuracy. To solve this problem, Niu and Zhang proposed a method in which the sequence is decomposed by the variable mode decomposition leaky integrator echo state network method combined with multistep prediction to predict the short-term power load of the combined model component sequence.[14] At the same time, the load data of a power system has large capacity. To improve the ability of the load forecasting method by feature extraction, Yuan *et al*. proposed a method that introduced an extreme learning mechanism into a traditional deep belief network to combine the strong generalization ability of extreme learning and the advantages of a deep belief network.[15] Wang *et al*. proposed a forecasting method based on pole symmetrical empirical mode decomposition (ESMD)–permutation entropy (PE) and an adaptive deep belief network (ADBN) to improve the short-term power load forecasting performance.[16] In addition to historical load data, other factors, such as electricity price, affect load changes. Wang *et al*. clustered the factors that affect load forecasting, and the clustering data were used to train a model to comprehensively consider the factors that affect the load and improve the prediction accuracy.[17] The above methods are examples of offline prediction. Fekri *et al*. proposed an online adaptive recurrent neural network (RNN) for load forecasting that is capable of continuously learning from newly arriving data and adapting to new patterns.[18]

The above machine learning methods achieved good results in power system load forecasting. However, the amount of data needed to be collected and processed by the above methods was huge, the models were complex, and the effects of meteorological factors on load, especially those of temperature and humidity on residential electricity consumption, were not fully considered. Through appropriate sensors installed in a microgrid, power consumption, temperature, and humidity data can be obtained. In this study, we process and analyze these data to achieve accurate load forecasting. To ensure the rapidity, accuracy, and efficiency of load prediction in a microgrid system, deep learning is introduced into microgrid load prediction, and we propose a method for the short-term load prediction of a microgrid system based on multivariable and multistep long short-term memory (MM-LSTM) that considers the effects of meteorological factors, such as temperature and humidity, on load data. The prediction model is based on Keras, an open-source artificial neural network library written in Python for the design, debugging, evaluation, application, and visualization of deep learning models, and the power consumption and meteorological data of six districts of New York are used for load prediction to verify the effectiveness of the method.

2. Construction of MM-LSTM Model

2.1 Introduction of LSTM model

The RNN is effective for processing data with sequence characteristics. However, the learning ability of the RNN deteriorates with increasing distance between the past information and the current prediction information, which is called gradient disappearance. Long short-term memory (LSTM) was proposed to solve the problem of RNN gradient disappearance.[19]

A cell processor is added to the LSTM algorithm with three gates shown in Fig. 1: the forget gate f_t, input gate i_t, and output gate o_t.[20] The memory unit update process is used to determine whether information is adopted after it enters the LSTM. Only the information that conforms to

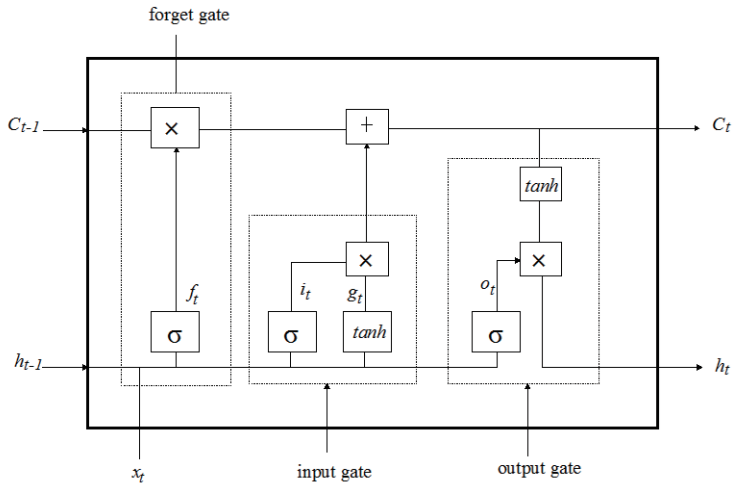

Fig. 1. Standard structure of LSTM.

the algorithm authentication is retained; otherwise, it is eliminated through the forget gate to solve the problem of long-term dependence of data. The gradient disappearance problem of the RNN can be solved by this process.

2.2 Update process of LSTM memory unit

2.2.1 Forget gate

The function of the forget gate is to determine how much information can be transferred from the previous cell state C_{t-1} to the current cell state C_t. If its output is 0, the information representing the cell state C_{t-1} at the previous moment is eliminated; if the output is 1, all the information representing the cell state C_{t-1} at the previous moment is retained. The process can be expressed as

$$f_t = \sigma(W_f \times [h_{t-1}, X_t] + b_f), \tag{1}$$

where σ is the activation function sigmoid, W_f is the matrix weight, which is equal to the product of the output h_{t-1} of the hidden layer of the forget gate f_t at the previous moment and the input x_t at the current time, and b_f is the bias.

2.2.2 Input gate

The function of the input gate is to determine which information in the new input information needs to be transferred to the memory unit. It uses the activation function sigmoid to filter out the values used for updating. The function of the tanh layer is to generate candidate states of new memory units at the current time. It can be seen from Fig. 1 that these two parts of the process can be expressed as

$$i_t = \sigma(W_{it} \times [h_{t-1}, X_t] + b_i), \tag{2}$$

$$C_t = \tanh(W_c \times [h_{t-1}, X_t] + b_c), \tag{3}$$

where W_{it} is the matrix weight of input gate i_t at the current time, W_c is the matrix weight of the newly generated information at the current time, and b_c is the bias.

2.2.3 Updating the cell state

The updating of the cell state is a process of discarding unnecessary and increasing information. This process involves the addition of two parts to obtain candidate values: the product of the output of the forget gate f_t and the cell state C_{t-1} at the previous moment, and the product of the input i_t of the input gate and the current cell state C_t. The process is expressed as

$$C_t = f_t \times C_{t-1} + i_t \times C_t. \tag{4}$$

2.2.4 Hidden layer output and output gate

The initial output of LSTM is obtained by controlling the output gate o_t using the sigmoid function, and then the cell state C_t is compressed to (−1, 1) by the tanh activation function. The output of the model is the initial output multiplied by the value of the cell state C_t treated by the tanh activation function. The output is expressed as

$$o_t = \sigma(W_{ot} \times [h_{t-1}, X_t] + b_o), \tag{5}$$

$$h_i = o_i \times \tanh(C_i), \tag{6}$$

where W_{ot} is the matrix weight of the output gate o_t at the current time, and the tanh function can compress the information and stabilize the numerical value.

2.3 Operating principle of LSTM

The operating principle of LSTM is shown in Fig. 2. LSTM consists of t time steps and two network layers. The load data constitute the time series $x_1, x_2, x_3, x_4, \ldots, x_t$. The first work unit of the first layer of the network calculates the current response with the initial unit and hidden states after receiving the data x_1, and transmits the response to the second and first units of the upper layer. At the next moment, the second unit of the first layer transmits the calculation results to the third and second units of the second layer according to the input data x_2 and the state quantity of the previous unit h_1^1, and so on. The unit input of the second layer is the output of the first layer. By passing the result of the calculation in the same way as for the first layer, the output response is calculated. The output of each time step is the hidden state of each unit h_t^k. In

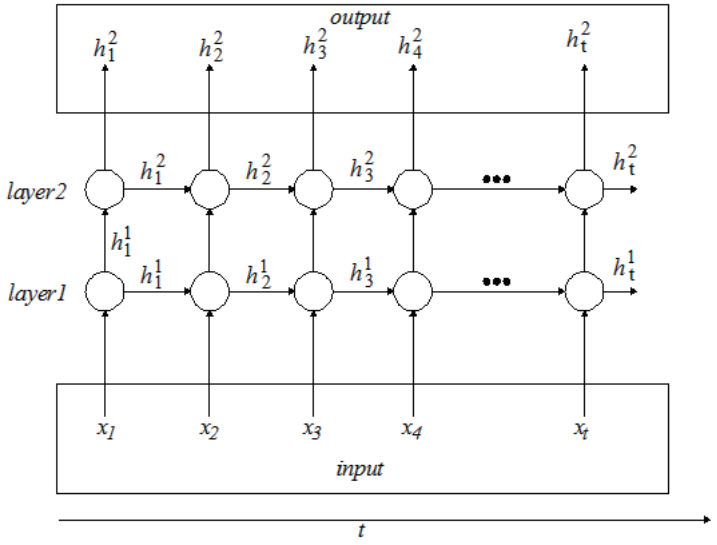

Fig. 2. Network layer of MM-LSTM.

the LSTM network, the time series is divided into several ordered data segments according to the time step, and the output results are calculated with different weights in the work units of each data segment. The prediction method with a time step of one is called the single-step prediction method, and the prediction method with a time step greater than one is called the multistep prediction method.[21]

It can be seen that when the time step is set to one, only one cell provides the weight and the response of the output and input, and all data pass through one unit to establish the information persistence of time series data. In a network with multiple time steps, multiple units provide different weights for time series. Moreover, the persistence of information is established not only in the unit itself, but also for the time series between the linked units. It also increases the correlation of time series data in time. Therefore, to a certain extent, the multi-time-step network model can be time-dependent. The sequence provides more abundant weight links and multiple information correlations in the response process. On this basis, an MM-LSTM method is proposed.

3. Experimental Analysis

3.1 Data selection

After 2014, New York has reduced the 100 h electricity demand with the highest annual value by using the distributed energy management method, which can delay the power system and reduce the electricity cost. It is estimated that this will save the state government $1.2–1.7 billion every year. With the continuous improvement of its power data monitoring and forecasting system, New York gives full play to the potential of distributed power resource management and control in each region, improving the load factor and thus helping improve the utilization rate of infrastructure, reduce excessive investment, and ensure the reliability of the power system. The Brooklyn & Queens Power Demand Management Project is the first such project in New York. At present, the whole state is testing the effect of such projects. As another example of a demonstration project, United Edison created a “virtual power plant” by integrating distributed solar photovoltaic and battery energy storage resources, similarly to a traditional power plant, providing important services such as capacity and frequency regulation to the power grid. In cooperation with two private enterprises, United Edison has installed 1.8 MW solar photovoltaic power generation facilities, as well as a battery energy storage system in 300 users’ homes, and added cloud technology to ensure that power companies can directly control the integrated system.

Through the tracking forecast and real-time management strategy of the power demand and electricity price, the district power company will buy electricity from power plants one day or several days in advance. Therefore, the real-time prediction of power demand and electricity price can accurately predict the future power demand and electricity price trends on the basis of previous data, optimize the power dispatch of the state grid, and ensure the stable and safe operation of the power system.

Here, we consider the power consumption, temperature, and humidity data from June 1, 2020 to June 25, 2020 in six districts of New York: Capital (CAPITL), Central (CENTRL), Dunwoodie (DUNWOD), Genesee (GENESE), Hudson Valley (HUD VL), and Long Island (LONGIL). The unit of power consumption is MWh, the update frequency of the power consumption data is 5 min, and no data are missing. The integrity of the data set is very high, making it very suitable for research on microgrid load forecasting and the optimal regional power distribution. A total of 7200 load data and corresponding temperature and humidity data are used as the original data for analysis and prediction.

3.2 Establishment of prediction and evaluation indexes

In this paper, root mean square error (*RMSE*), mean absolute error (*MAE*), and mean absolute percentage error (*MAPE*) are used as the prediction and evaluation indexes:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=0}^{n}(y_i - y_p)^2}, \tag{7}$$

$$MAE = \frac{1}{n}\sum_{i=0}^{n}\left|y_i - y_p\right|, \tag{8}$$

$$MAPE = \frac{1}{n}\sum_{i=0}^{n}\left|\frac{y_i - y_p}{y_i}\right| \times 100\%. \tag{9}$$

Here, y_i is the true value, y_p is the forecast value, and n is the number of load points to be predicted.

3.3 Algorithm verification

3.3.1 Network parameter training

In this study, an MM-LSTM prediction model based on Keras is built. The simulation is carried out on a computer with an Intel (R) Core (TM) i7-6700, 3.4 GHz CPU with 32 GB RAM.

To verify the effect of the step size on the accuracy of the prediction model and select the best step size, we consider data in CAPITL as the original data for analysis and prediction with Adam as the optimization algorithm. The step size is set to 1, 2, 5, 10, 15, 20, and 25.

The obtained prediction and evaluation indexes are shown in Table 1. With increasing step size, the prediction performance of the model increases, the error decreases, and the training time increases. However, when the step size is increased from 20 to 25, there is negligible improvement in accuracy, but the training time further increases. Considering the accuracy and training time, the step size of the prediction model in this study is set to 20.

Table 1
Comparison of prediction and evaluation indexes for different step sizes.

Step size	*RMSE*	*MAE*	*MAPE* (%)	Training time (s)
1	15.875	11.914	0.757	6.7510
2	15.395	11.546	0.737	8.3350
5	15.143	11.247	0.709	11.4360
10	14.013	10.196	0.642	14.2250
15	13.533	9.894	0.623	17.6410
20	13.520	9.752	0.617	20.7410
25	13.512	9.733	0.611	28.7720

Table 2
Comparison of prediction and evaluation indexes for different optimization algorithms.

Optimization algorithm	*RMSE*	*MAE*	*MAPE* (%)	Training time (s)
AdaGrad	14.018	10.471	0.659	21.2530
RMSProp	25.694	20.928	1.285	23.4260
Adam	13.520	9.752	0.617	20.7410
AdaDelta	22.218	16.967	1.084	18.7720

To verify the effect of the optimization algorithm on the accuracy of the prediction model and select the best optimization algorithm, again we select the data in CAPITL as the original data for analysis and prediction with a step size of 20, and the results obtained with the AdaGrad, RMSProp, Adam, and AdaDelta algorithms are used for comparative analysis. The experimental prediction and evaluation indexes for the different optimization algorithms are shown in Table 2. The values of *RMSE*, *MAE*, and *MAPE* are the lowest for Adam. Considering the accuracy and training time, we select Adam as the optimization algorithm in this study.

3.3.2 Simulation analysis of prediction model

The step size of the prediction model is set to 20, that is, the load data of the first 100 min is used to predict the load data of the next sampling time. The optimization algorithm is Adam and the number of epochs is set to 500. In the MM-LSTM prediction model, 6700 data from the six districts of New York are selected as the training set and 500 data are selected as the test set to evaluate the performance of predicting the short-term load of the microgrid.

The actual and forecast load curves for the six districts are shown in Figs. 3–8. The forecast load is well fitted with the actual load for each district, showing the high prediction accuracy of the prediction model.

The prediction and evaluation indexes when the load data of different districts are used in the prediction model are shown in Table 3. The lowest values of *RMSE* and *MAE* are 8.827 and 6.775, respectively, for the load data sets from GENESE. The lowest value of *MAPE* is 0.617, which is for the load data sets from CAPITL. The highest values of *RMSE*, *MAE*, and *MAPE* are 19.745, 15.800, and 1.282, respectively, for the load data sets from HUD VL. The prediction of the model is relatively accurate and the error is relatively small for all six districts.

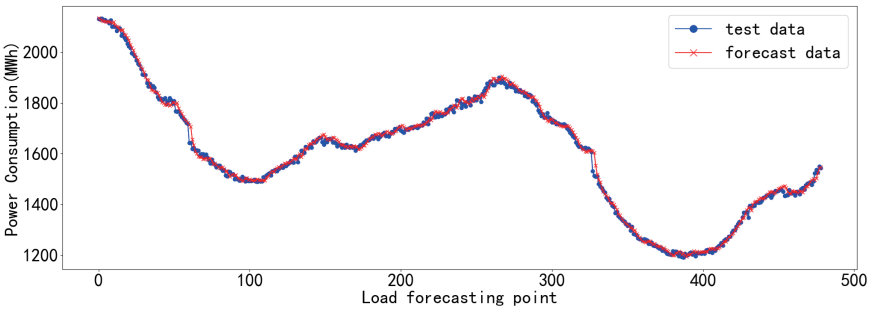

Fig. 3. (Color online) Actual and forecast load curves of CAPITL.

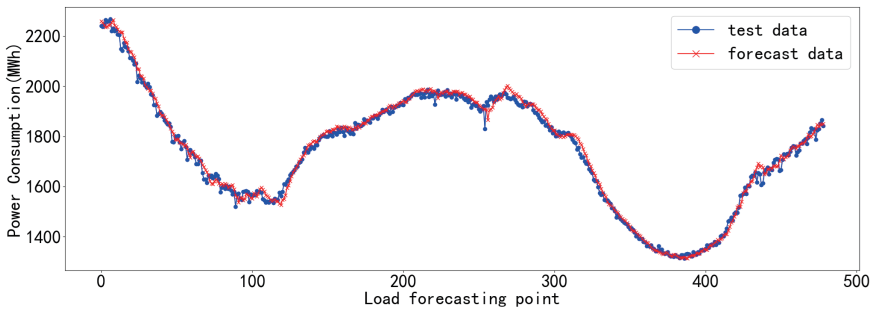

Fig. 4. (Color online) Actual and forecast load curves of CENTRL.

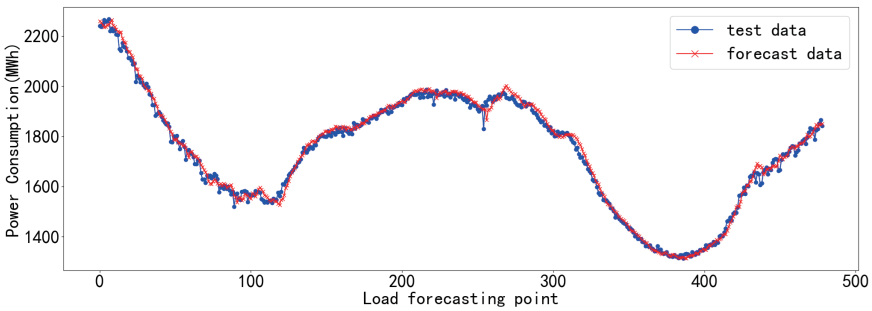

Fig. 5. (Color online) Actual and forecast load curves of DUNWOD.

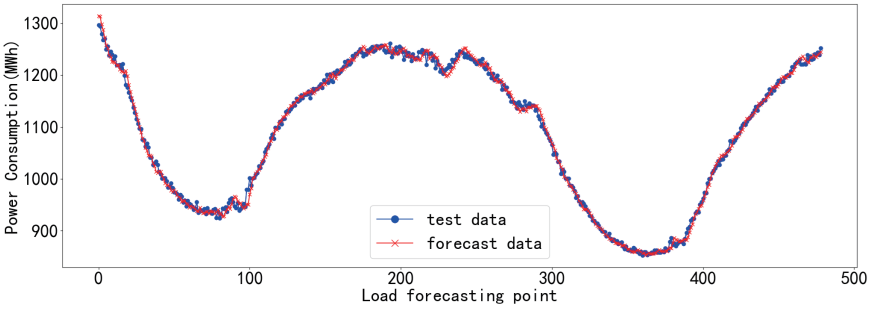

Fig. 6. (Color online) Actual and forecast load curves of GENESE.

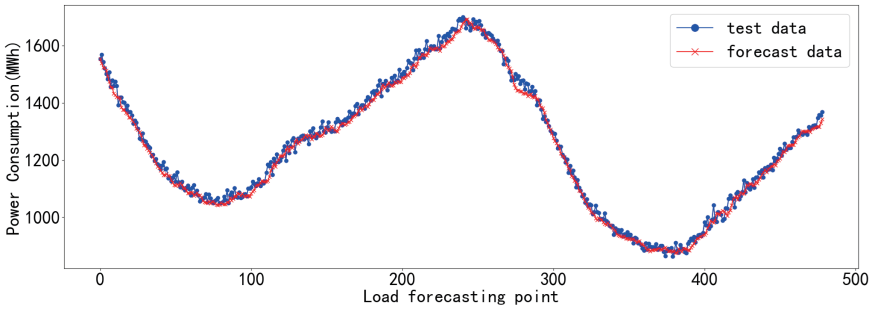

Fig. 7. (Color online) Actual and forecast load curves of HUD VL.

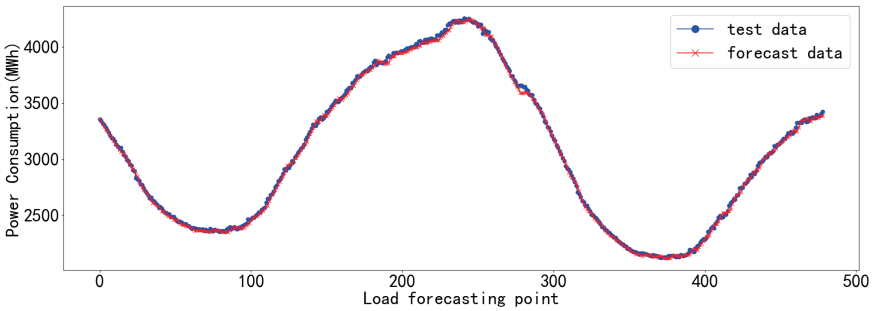

Fig. 8. (Color online) Actual and forecast load curves of LONGIL.

Table 3
Comparison of prediction and evaluation indexes in different regions.

District	*RMSE*	*MAE*	*MAPE* (%)
CAPITL	13.520	9.752	0.617
CENTRL	19.376	14.540	0.838
DUNWOD	8.857	6.898	0.838
GENESE	8.827	6.775	0.620
HUD VL	19.745	15.800	1.282
LONGIL	18.324	14.136	0.471

4. Conclusion

To ensure accurate and rapid load forecasting in a microgrid system and improve the stability of microgrid operation, we proposed an MM-LSTM prediction method to forecast the short-term load of the microgrid, fully considering the impact of temperature and humidity and other meteorological data obtained from sensors installed in a microgrid on load forecasting, where the model is built using Keras. By considering different step sizes and optimizing algorithms in the prediction model simulation, the best step size and optimizing algorithm were selected. Load forecasting using the prediction method was carried out using the electricity consumption,

temperature, and humidity data of six districts of New York, United States. The experimental results show that the method has high accuracy. The method can be used to forecast the short-term load of microgrids and for the practical application of microgrid short-term load prediction.

Acknowledgments

This work was supported in part by the Young and Middle-aged Teachers' Basic Scientific Research Ability Improvement Project of Guangxi University Fund "Research on the Key Technologies of Transient Stability Assessment of Microgrid based on Deep Learning" (2019KY0620) and the Hechi University research project "Research on Short Term Load Forecasting Method of Microgrid based on Deep Learning" (2019XJQN012).

References

1 H. T. Cheng, T. J. Cheng, and H. M. Huang: Sens. Mater. **33** (2021) 1245. https://doi.org/10.18494/SAM.2021.3162
2 L. Y. Chang and S. F. Lin: Sens. Mater. **33** (2021) 379. https://doi.org/10.18494/SAM.2021.3021
3 H. T. Cheng, T. J. Cheng, and H. M. Huang: Sens. Mater. **32** (2020) 1689. https://doi.org/10.18494/SAM.2020.2693
4 C. Herui and P. Xu: Power Syst. Protect. Control **43** (2015) 108. https://doi.org/10.7667/j.issn.16743415.2015.04.017
5 K. Wang and R. Y. Liu: Power Syst. Technol. **36** (2012) 77. https://doi.org/10.13335/j.1000-3673.pst.2012.11.044
6 A. Vaghefi, M. A. Jafari, and B. Emmanuel: Appl. Energy **136** (2015) 186. https://doi.org/10.1016/j.apenergy.2014.09.004
7 L. M. Huang, M. Y. Lee, and X. J. Chen, H.-W. Tseng, C.-F. Yang, and S.-F. Lee: Sens. Mater. **33** (2021) 805. https://doi.org/10.18494/SAM.2021.3048
8 X. Y. Kong, F. Zheng, and J. Z. Er: Autom. Electr. Power Syst. **42** (2018) 133. https://doi.org/10.7500/AEPS20170826002
9 T. Zhao, L. T. Wang, Y. Zhang, and S. M. Tian: J. Chin. Electr. Eng. Sci. **36** (2016) 604 (in Chinese). https://doi.org/10.13334/j.0258-8013.pcsee.2016.03.003
10 J. H. Xiong and K. Niu: Power Syst. Protect. Control **45** (2017) 71. https://doi.org/10.7667/PSPC160981
11 K. Supat, K. Paiwan, and B. Chanin: Sens. Mater. **33** (2021) 2427. https://doi.org/10.18494/SAM.2021.3312
12 J. V. Munkhammar, V. M. Dennis, and W. Joakim: Appl. Energy **282** (2021) 99. https://doi.org/10.1016/j.apenergy.2020.116180
13 X. Wang, S. X. Wang, and Q. Y. Zhao: Int. J. Electr. Power Energy Syst. **126** (2021) 121. https://doi.org/10.1016/J.IJEPES.2020.106583
14 A. M. Niu and H. L. Zhang: J. Xinjiang Univ. (Natural Science Edition in Chinese and English) **37** (2020) 562. https://doi.org/10.13568/j.cnki.651094.651316.2019.12.20.0002
15 F. Yuan, X. S. Wang, and Y. J. Liang: Sci. Technol. Innov. **33** (2020) 1. https://kns.cnki.net/kcms/detail/detail

About the Author

Dashuang Li was born in Guilin, China. He received his B.S. degree in electrical engineering and automation and his M.S. degree in electric power systems and automation from Northeastern University, Shenyang, China, in 2013 and 2015, respectively. He is currently a senior engineer with Hechi University. His current research interests include optimization theory and its applications in smart grids and the energy internet.
(lidashuang@hcnu.edu.cn)