Journal of Network Intelligence ©2022 ISSN 2414-8105 (Online)
Taiwan Ubiquitous Information Volume **7**, Number **4**, November **2022**

Short-term Load Prediction Based on Multi-Channel CNN-LSTM

Rongqun Hu*

College of Computer Information and Engineering
Nanchang Institute of Technology
Nanchang 330044, China
hrq2001@nut.edu.cn

Yanyan Gong

School of Electronics and Information
Nanchang Institute of Technology
Nanchang 330044, China
2411238635@qq.com

Jinliang Huang

School of Electronics and Information
Nanchang Institute of Technology
Nanchang 330044, China
1429963063@qq.com

Xiaojian Qiu

Nanchang Hangtianguangxin Technology Co., Ltd.
Nanchang 330044, China
Qxj9998@163.com

Keyong Shen

College of Computer Information and Engineering
Nanchang Institute of Technology
Nanchang 330044, China
549601097@qq.com

*Corresponding author: Rongqun Hu

Received July 26, 2022, revised September 10, 2022, accepted October 12, 2022.

Abstract. *Short-term load prediction plays a critical role in the stable operation of power systems. With the development of artificial intelligence, many advanced algorithms and models have been applied in short-term load prediction. To address the problem of inconsistent sampling intervals of meteorological data and load data, a short-term load prediction model based on multi-channel convolutional neural networks - long short-term memory (CNN-LSTM) was proposed. In this model, load data was input into multi-channel CNN-LSTM to extract the multi-scale and dependency features of the data, meteorological data was input into LSTM channels to extract the dependency features of the data, and the feature information of load data and meteorological data were concatenated and then input into the fully connected layer to output the final prediction results. The experimental results indicated that multi-channel CNN-LSTM addressed the issue of inconsistent sampling intervals and effectively extracted the multi-scale and dependency features of load data and dependency features of meteorological data, thus improving the short-term load prediction accuracy.*
Keywords: Load prediction; Sampling interval; Multi-scale features; Dependency features; Convolutional neural network; Long short-term memory

1. **Introduction.** Power load prediction is the prediction of power load, demand, situation, and other indicators at a certain time in the future based on historical data and other factors. According to the prediction time span, power load prediction is divided into three types, that is, long-term, mid-term, and short-term load prediction. Herein, short-term load prediction is of great significance for the stable operation of power systems, plays a key role in the power distribution planning of power systems, and can help improve the economic efficiency of power systems [1]. Nevertheless, since power load is featured by instability, limited by conditions and time [2], and affected by various factors such as atmosphere temperature, precipitation, humidity, and holidays, which increases the difficulty of load prediction, the development of load prediction methods with high accuracy has become a necessity [3].

Conventional load prediction methods include multiple regression analysis (MRA) [4], auto regressive moving average (ARMA) [5], and Kalman filtering (KF) [6]. These methods only consider the time-series features of load data and show poor nonlinear load data fitting performance [7]. In virtue of great advances in artificial intelligence [8, 9], advanced machine learning algorithms have been widely applied in swarm intelligence optimization algorithm [10], speech synthesis [11], medical image [12, 13], and outher fields [14, 15, 16]. Also, machine learning algorithms have been applied in short-term load prediction. Load prediction methods based on machine learning include support vector machine (SVM) [17], random forest (RF) [18], and artificial neural network (ANN) [19]. ANN is a complex network structure consisting of a large number of interconnected neurons and is an abstraction of the organizational structure and operation mechanism of the human brain. ANNs consist of an input layer, a hidden layer, and an output layer and perform well in load prediction. ANNs mainly include convolutional neural networks (CNNs) and recurrent neural networks (RNNs). CNNs have the advantages of sharing convolutional kernels and being suitable for extracting potential features of high-dimensional data [20], but CNNs have no memory and cannot extract the dependency features of data. RNNs have the characteristics of parameter sharing and memory and can overcome the problem of poor non-linear data fitting performance of conventional prediction methods. Vermaak and Botha [21] proposed a short-term load prediction model based on RNNs. The model effectively captured the features of the input data, but the gradient disappearance and explosion problems of RNNs were not alleviated in short-term load prediction. Hence, Hochreiter et al. proposed the long short-term memory (LSTM). The LSTM introduced a gate mechanism to control information transfer and effectively compensated for the shortcomings of RNNs. Kong et al. [22] applied LSTM in load prediction and demonstrated the superior performance of LSTM in load prediction as it solved the gradient disappearance and explosion problems of RNNs.

Power load is affected by many factors, and load data have multi-scale features, mainly in the time period such as year, month, and day. Xiao et al. [23] proposed a short-term load prediction model based on multi-scale jump-connected LSTM with the concatenated load data and meteorological data such as the data of temperature, humidity, and precipitation as the input of LSTM, greatly improving short-term load prediction accuracy. Additionally, the multi-scale features of load data could be extracted by an integrated model. Zhang et al. [24] used orthogonal wavelet decomposition to decompose load data into multiple scales and then used LSTM to model the decomposed data. To better exploit the effective information contained in the massive data so as to improve short-term load prediction accuracy, Lu et al. [25] proposed a short-term load prediction method based on CNN-LSTM integrated neural network model. In this model, CNN was used to extract feature vectors, and the feature vectors were constructed into a form of time series and used as the input data of LSTM, and then LSTM was employed for short-term load prediction. Wang et al. [26] pointed out that the need for multi-step forecasting is more urgent in actual power load forecasting. According to the characteristics of CNN and bidirectional long short-term memory network (BiLSTM), a CNN-BiLSTM load forecasting model was established, and the offline mode was improved to achieve online multi-step prediction.

The above-mentioned studies considered the multi-scale features of load data. However, since inconsistent sampling intervals of meteorological data and load data, including daily average temperature and humidity, may lead to varying scale features, inputting the two types of data into the model at the same time seems inappropriate. In this study, a short-term load prediction model based on multi-channel convolutional neural networks - long short-term memory (CNN-LSTM) was proposed. In the proposed model, a multi-channel network model was designed, and the multi-scale features of load data were extracted by setting CNNs with different void coefficients, then the data was input into LSTM to extract dependency features; meanwhile, LSTM was used to process meteorological data and then extract the dependency features of meteorological data; after that, the feature information of load data and meteorological data was concatenated and input into the fully connected layer for short-term load prediction. This model not only solves the problem of inconsistent sampling intervals of load data and meteorological data but also effectively extracts the multi-scale and dependency features of load data and dependency features of meteorological data, greatly improving short-term load prediction accuracy.

2. **Basic Network Architecture.**

2.1. **CNN.**

Lecun et al. [27] proposed the CNN model and applied it in image processing fields such as image recognition and computer vision [28]. CNN uses local connection and weight sharing to abstract the original data and automatically extract the internal features of the data, and it mainly consists of a convolution layer, a pooling layer, and a fully connected layer. 1D convolution network is often used to process time series data. Suppose the time series data is $[x_1, x_2, x_3, x_4] \in R^{t \times n}$, where t is the input time step and n is the number of features of the input data. Then, the convolutional operation formula is:

$$y = f(x \otimes W_o + b_o) \tag{1}$$

where X is the input data, W_o is the 1D convolutional weight, b_o is the bias, $\otimes$ denotes the convolutional operation, and $f()$ is the activation function, where $relu()$ function as follows is generally used:

$$relu(x) = \begin{cases} x & x \geq 0 \\ 0 & x < 0 \end{cases} \tag{2}$$

2.2. **LSTM.** To address the problem that RNNs cannot extract long-term dependencies effectively, Hochreiter [29] proposed the LSTM and solved the gradient disappearance and explosion problems. Different from RNNs, LSTM introduced a gate mechanism to control information transfer [30]. Figure 1 shows its unit structure. Herein, forget gate f_t indicates how much of the state information from the previous cell is used to calculate the current cell information:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{3}$$

where W_f is the weight of the forget gate, h_{t-1} is the state information of the previous time step, x_t is the input information of the current time step, b_f is the bias of the forget gate, and $\sigma()$ is $sigmoid()$ activation function.

$$sigmoid(x) = \frac{1}{1 + e^{-x}} \tag{4}$$

The input gate i_t determines how much information of the current time step is retained, and the formula is as follows:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{5}$$

$$\tilde{c}_t = tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{6}$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{c}_t \tag{7}$$

where W_i is the weight of the forget gate, W_c is the weight of the updated state information, b_i is the bias of the forget gate, b_c is the bias of the update state information, $\tilde{c}_t$ is the updated value of the cell state at moment t , and C_t is the internal state used to record all historical information.

The output gate o_t is the output of the current time step, and the formula is as follows:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{8}$$

$$h_t = o_t * tanh(C_t) \tag{9}$$

where W_o is the weight of the output gate; b_o is the bias of the output gate; h_t is the hidden information at moment t.

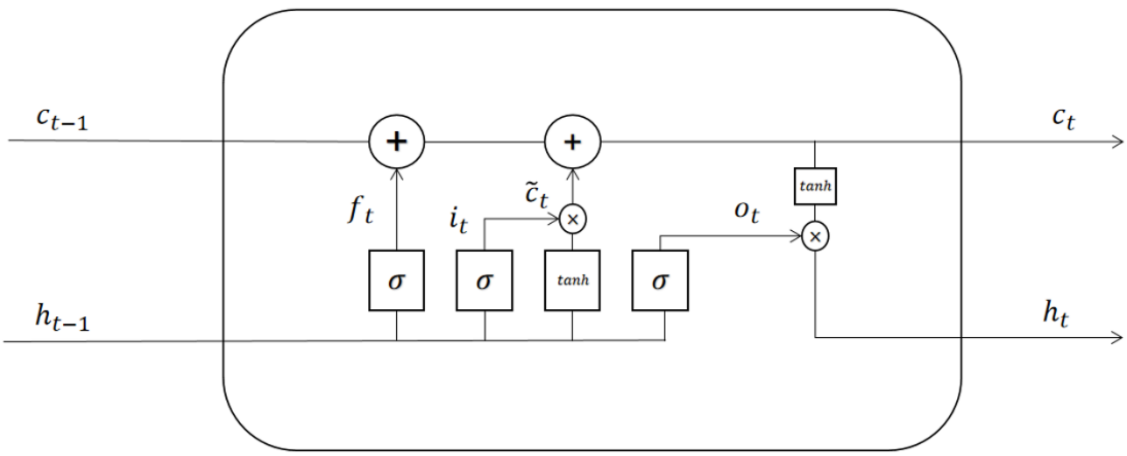


FIGURE 1. Structure of LSTM

3. **Short-Term Load Prediction Model Based on Multi-channel CNN-LSTM.** The short-term load prediction model based on multi-channel CNN-LSTM consists of multiple layers of networks operating in parallel and Figure 2 shows the model results. Load data and meteorological data are input into multi-channel CNN-LSTM and LSTM, respectively. Herein, CNN uses 1D CNN to extract the multi-scale features of the load data by varying void coefficients; meteorological data are input into LSTM channels to extract the long short-term dependency information of the meteorological data; then the extracted feature information of load data by CNN-LSTM and the extracted long short-term dependency feature information of meteorological data by LSTM are concatenated, and the final output of the fully connected layer is the prediction value.

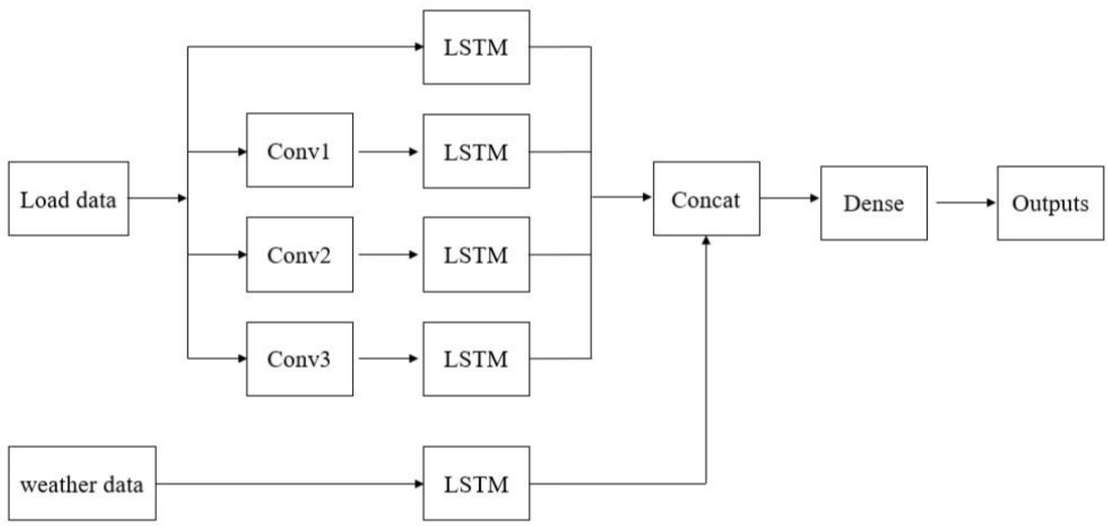


Figure 2. Structure of short-term load prediction model based on multi-channel CNN-LSTM

As shown in Figure 2, $Cov1$, $Cov2$ and $Cov3$ are convolution layers with different void coefficients, denotes matrix concatenation, and denotes the fully connected layer.

Assuming that the convolution layer input is $[x_1, x_2, \cdots, x_{3T}]$, as shown in Figure 3, the output obtained after the 1D convolutional operation is $[y_1, y_2, \cdots, y_T]$, i.e., the scale feature at time scale T extracted.

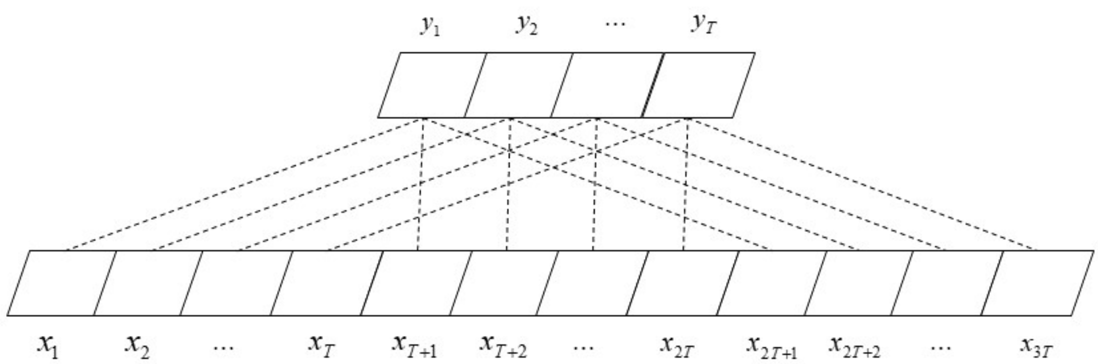


Figure 3. 1D void convolution

4. Case Study.

4.1. Data Analysis. Although load data are fluctuating, the fluctuation shows periodicity and continuity, and the periodicity is mainly reflected in the cycle period such as year, month, and day. Figure 4 shows the load data in one month with a sampling interval of 15 min. Time is an important factor influencing short-term power load prediction, which makes the load data show regular fluctuations. Given the periodic feature of short-term power load data, multi-channel CNN-LSTM used 1D convolution layers with varying void coefficients to extract the multi-scale features of load data.

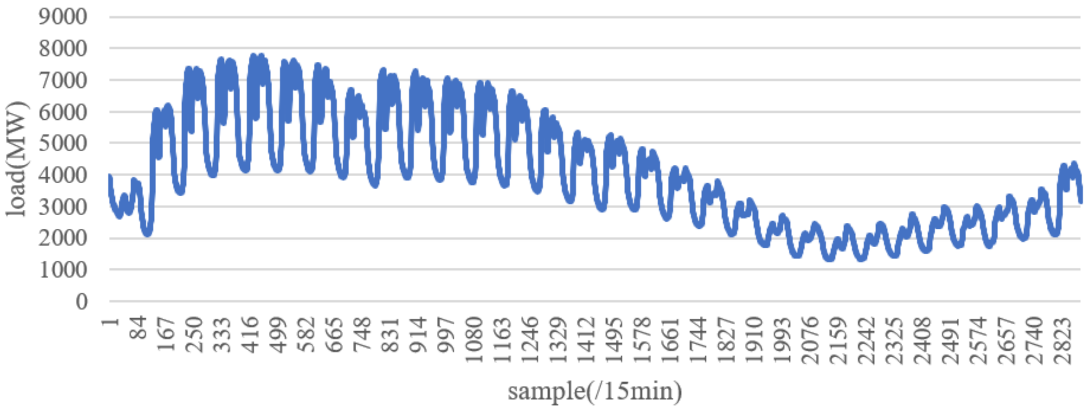


FIGURE 4. Curve of power load as a function of time

Meteorological data such as daily average temperature cannot be sampled once every 15 min. The daily average temperature data in January is shown in Figure 5. As observed, the data shows no obvious periodic pattern. Therefore, if meteorological data and load data are concatenated and then input into the load prediction model at the same time, inconsistent sampling intervals would lead to limited prediction accuracy.

4.2. Data Pre-processing. The experimental data are obtained from the power load data of some city from 2012 to 2015 with a sampling interval of 15 minutes and the unit of megawatts (MW). The data also include daily maximum temperature, daily minimum temperature, daily relative humidity, and other meteorological data. In this paper, 30 days of historical load data and meteorological data were used to predict the load for the next 24 hours (96 sampling points). Due to the large differences between load data and meteorological data, which is inconducive to model training, the data were normalized before the experiment. Also, the data were divided into training and testing sets in the ratio of 8:2, the training process uses the test set as the validation set. The data normalization formula is as follows:

$$X = \frac{X - min(X)}{max(X) - min(x)} \tag{10}$$

where X is the input data, $min()$ is the minimization function, and $max()$ is the maximization function.

4.3. Experimental Set-up. The experimental environment was set as follows: processor: Intel(R) Xeon(R) Gold5118 CPU @ 2.30GHz; memory: 64G; graphics card model: NVIDIA GeForce RTX 2080 Ti; operating system: Windows 10 Professional Workstation Edition; the deep learning framework Tensorflow was used to build all neural network models.

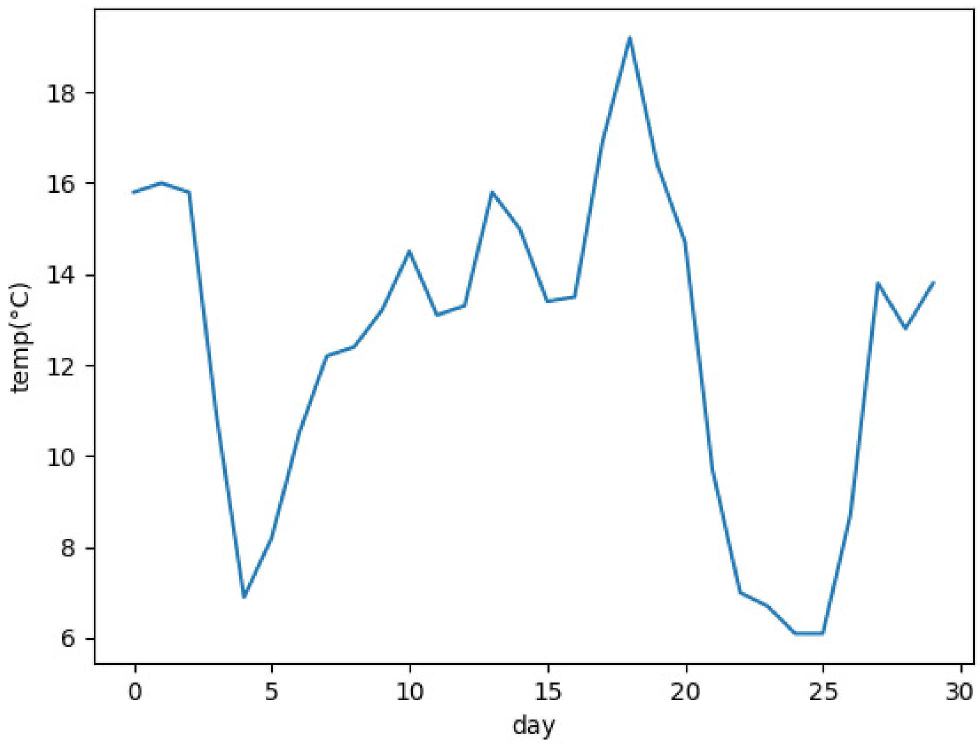


Figure 5. Daily average temperature curve

Four short-term load prediction models (RNN [21], LSTM [24], support vector regression (SVR) [31], and CNN-LSTM) were used to compare with the short-term load prediction model based on multi-channel CNN-LSTM. Herein, the CNN-LSTM was set to compare with multi-channel CNN-LSTM, so as to verify that multi-channel can effectively improve short-term load prediction accuracy.

The multi-channel CNN-LSTM was set as follows: four channels were set, where the first channel contains one layer of LSTM and the other three channels contain one layer of LSTM and one layer of 1D CNN; the number of neurons in LSTM was set to 16, and for the first channel, the number of convolutional kernels was set to 12, the size of the convolutional kernel was 10, the step size was 1, and the void coefficient was set to 96; for the second channel, the number of convolutional kernels was set to 12, the size of the convolutional kernel was 8, the step size was 1, and the void coefficient was set to 192; for the third channel, the number of convolutional kernels was set to 12, the size of the convolutional kernel was 4, the step size was 1, and the void coefficient was set to 288. The RNN model was set with one layer of neural network and the number of neuron units of 60. The LSTM model was set with one layer of neural network and the number of neuron units of 60. The CNN-LSTM model was set with a 1D CNN convolutional kernel size of 10, a step size of 1, and no void coefficients, and the number of neuron units in LSTM was set to 16. Gaussian kernel function was used as the kernel function of SVR.

1D CNN uses $relu()$ as the activation function, while other network layers use $tanh()$ as the activation function. The model was optimized using Adam's algorithm with 500 iterations and a learning rate of 0.001. To speed up the operation, the mini-batch technique was used, and the batch size was set to 320.

Mean square error (MSE) was used as the loss function to train the model, and mean absolute error (MAE) was used as the evaluation function to evaluate the prediction performance of the model. The formula is as follows:

$$MSE = \frac{1}{n}\sum_{1}^{n}(y^i_{pred} - y^i_{true})^2 \tag{11}$$

$$MAE = \frac{1}{n}|y^i_{pred} - y^i_{true}| \tag{12}$$

where n is the number of samples, y^i_{pred} is the prediction value, and y^i_{true} is the practical value.

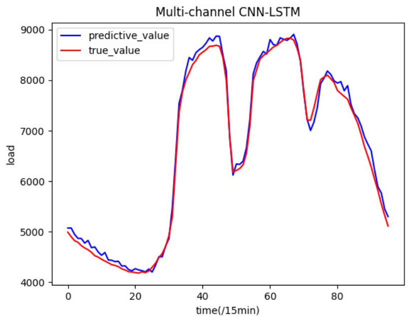
4.4. **Experimental Results and Analysis.** In this experiment, five testing sets were used to test different models, and the results were shown in Table 1. Among the five testing sets, the mean prediction error obtained using Multi-channel CNN-LSTM was 210.9436, which was much lower than the mean error obtained using the other four prediction models; The prediction error obtained using multi-channel CNN-LSTM was the lowest, which was 69.7485, indicating that multi-channel CNN-LSTM exhibited maximum prediction accuracy.

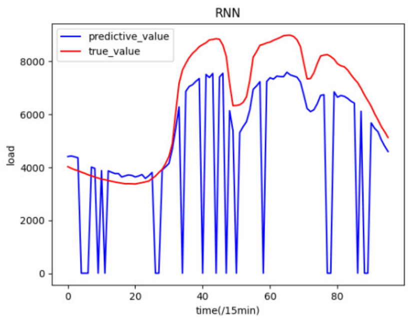
As shown in Table 1, since SVR is not applicable to load data and has poor generalization ability, compared with neural network models, SVR is higher in terms of the mean load prediction error, which reaches 2470.4871. RNN cannot obtain the long-term dependency of load data, while LSTM can extract the long-term dependency of load data by setting a gate structure, so the prediction error obtained by RNN is higher than that obtained by LSTM. Nevertheless, load data has multi-scale features, which can barely be extracted by SVR, RNN, and LSTM models. Hence, CNN-LSTM extracts the multi-scale features of load data by means of CNN, effectively improves the short-term load prediction accuracy, and decreases the mean prediction accuracy error to 324.0744. In addition to extracting the multi-scale features of load data, multi-channel CNN-LSTM considers the inconsistent sampling intervals of meteorological data (e.g., daily average atmosphere temperature, precipitation) and load data. By designing a multi-channel model and taking weather factors into account, multi-channel CNN-LSTM increases short-term load prediction accuracy and decreases the mean prediction error to 210.9436. In summary, multi-channel CNN-LSTM exhibited maximum prediction accuracy.

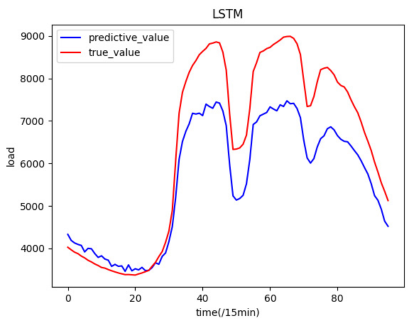
Table 1. Mean prediction errors of different models

Models	Test Sets					Average
	1	2	3	4	5	
Multi-channel CNN-LSTM	**168.9328**	**351.7530**	**282.1417**	**69.7485**	**182.1417**	**210.9436**
RNN	1623.7393	1864.7606	1748.8400	1835.9477	2074.4530	1829.5481
LSTM	895.2815	782.0772	1204.5942	689.4718	857.7419	885.8333
CNN-LSTM	427.3559	469.2293	294.1838	135.9040	293.6989	324.0744
SVR	2951.1747	3091.4231	3288.2584	1429.8903	1591.6891	2470.4871

Figure 6 illustrates load curves predicted by different models and the practical load curve. As observed, huge differences exist between the SVR prediction curves and the practical load curves. In short-term load prediction, the gradient disappearance and explosion problems of RNN are not alleviated, which leads to a large difference in the prediction results of RNN. Since LSTM compensates for the shortcomings of RNN, the prediction curve of LSTM is relatively close to the practical load curve. On the basis of LSTM, CNN-LSTM improves the feature extraction of load data by adding CNN and thus has its load prediction curve closer to the practical load curve than LSTM. All the above models use the concatenated load data and meteorological data as the input of the models to predict the future load, but multi-channel CNN-LSTM considers the inconsistent sampling intervals of load data and meteorological data, and designs multiple channels to extract the multi-scale features of load data, and input the meteorological data and load data into different channels for prediction, and obtained the prediction curve that is the closest to the practical load curve, indicating the highest prediction accuracy. The results shown in Figure 6 and Table 1 demonstrate that multi-channel CNN-LSTM performs better than other models.







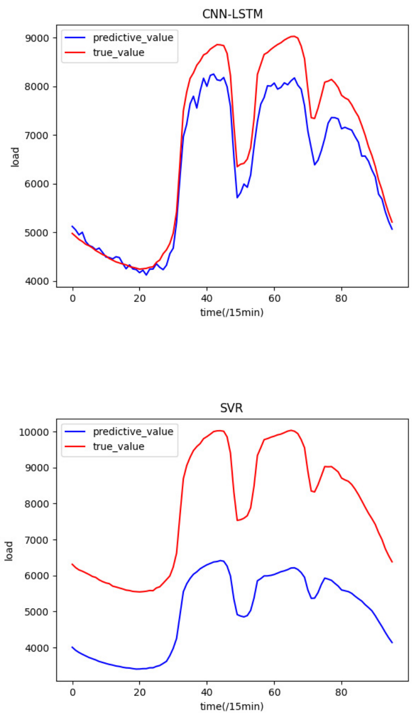


Figure 6. Load predictions by different models

5. **Conclusions.** Aimed at inconsistent sampling intervals of load data and meteorological data, we proposed the short-term load prediction model based on multi-channel CNN-LSTM. The proposed model extracted the multi-scale features of load data by setting CNNs with different void coefficients and then extracted the dependency features of load data by LSTM in each channel; after that, the feature information of meteorological data and load data processed by LSTM were fused and input into the fully connected layer to complete short-term load prediction. The experiment verified that multi-channel CNN-LSTM can effectively extract the multi-scale and dependency features of load data and the dependency features of meteorological data and solve the issue of inconsistent sampling intervals of load data and meteorological data, thus improving short-term load prediction accuracy.

Acknowledgment. This paper is supported by Science and Technology Research Project of Education Department of Jiangxi Province (No. GJJ212105)

REFERENCES

[1] Y. Zhao, H.-M. Wang, L. Kang, Z.-Y. Zhang, "Temporal convolution network-based short-term electrical load forecasting," *Transactions of China Electrotechnical Society*, vol. 37, no. 05, pp. 1242–1251, 2022.
[2] B.-Y. Yang, S.-J. Chen, "Overview of electrical load forecasting researches and forecasting analysis," *Sichuan Electric Power Technology*, vol. 41, no. 03, pp. 56–60+91, 2018.
[3] Z.-Y. Yang, J.-Y. Liu, Y.-B. Liu, L.-L. Wen, Z.-Q. Wang, S.-C. Ning, "Transformer load forecasting based on adaptive deep belief network," *Proceedings of the CSEE*, vol. 39, no. 14, pp. 4049–4061, 2019.
[4] G. Duket, "Pattern-based local linear regression models for short-term load forecasting," *Electric Power Systems Research*, vol. 130, pp. 139–147, 2016.
[5] H. Takeda, Y. Tamura, S. Sato, " Using the ensemble Kalman filter for electricity load forecasting and analysis," *Proceedings of the CSEE*, vol. 104, pp. 184–198, 2016.

[6] C.-P. Chen, Y.-J. Fang, "Short-term load forecasting of power system for holiday point-by-point growth rate based on Kalman filtering," *Engineering Journal of Wuhan University*, vol. 53, no. 02, pp. 139–144, 2020.
[7] C.-Q. Kang, Q. Xia, B.-M. Zhang, "Review of power system load forecasting and its development," *Automation of Electric Power Systems*, vol. 28, no. 17, pp. 1–11, 2004.
[8] J. Zhao, D. Chen, R. Xiao, Z. Cui, H. Wang, I. Lee, "Multi-strategy ensemble firefly algorithm with equilibrium of convergence and diversity," *Applied Soft Computing*, vol. 123, pp. 108938, 2022.
[9] J. Zhao, W. Chen, R. Xiao, J. Ye, "Firefly algorithm with division of roles for complex optimal scheduling," *Frontiers of Information Technology & Electronic Engineering*, vol. 22, no. 10, pp. 1311–1333, 2021.
[10] J. Zhao, Z.-F. Xie, L. Lv, H. Wang, H. Sun, X. Yu, "Firefly Algorithm with Deep Learning," *Acta Electronica Sinica*, vol. 46, no. 11, pp. 2633–2641, 2018.
[11] J. Sarzynska-Wawer, A. Wawer, A. Pawlak, J. Szymanowska, I. Stefaniak, M. Jarkiewicz, L. Okruszek, "Detecting formal thought disorder by deep contextualized word representations," *Psychiatry Research*, vol. 302, pp. 114135, 2021.
[12] N. Feng, T.-Y. Wu, Y. Liang, "A deep dynamic neural network model and its application for ECG classification," *Journal of Intelligent & Fuzzy Systems*, vol. 43, no. 2, pp. 2147–2154, 2022.
[13] Y. Ma, Y. Peng, T.-Y. Wu, "Transfer learning model for false positive reduction in lymph node detection via sparse coding and deep learning," *Journal of Intelligent & Fuzzy Systems*, vol. 43, no. 2, pp. 2121–2133, 2022.
[14] M.-E. Wu, J.-H. Syu, C.-M. Chen, "Kelly-Based Options Trading Strategies on Settlement Date via Supervised Learning Algorithms," *Computational Economics*, vol. 59, no. 4, pp. 1627–1644, 2022.
[15] F. Zhang, T.-Y Wu, Y. Wang, R. Xiong, G. Ding, P. Mei, L. Liu, "Application of Quantum Genetic Optimization of LVQ Neural Network in Smart City Traffic Network Prediction," *IEEE Access*, vol. 8, no. 2, pp. 104555–104564, 2020.
[16] S.-M. Zhang, X. Su, X.-H. Jiang, M.-L. Chen, T.-Y. Wu, "A traffic prediction method of bicycle-sharing based on long and short term memory network," *Journal of Network Intelligence*, vol. 4, no. 2, pp. 17–29, 2019.
[17] G.-F. Fan, L.-L. Peng, W.-C. Hong, F. Sun, "Electric load forecasting by the SVR model with differential empirical mode decomposition and auto regression," *Neurocomputing*, vol. 173, pp. 958–970, 2016.
[18] Y. Li, Y.-J. Jia, L. Li, J.-S. Hao, X.-Y. Zhang, "Short term power load forecasting based on a stochastic forest algorithm," *Power System Protection and Control*, vol. 48, no. 21, pp. 117–124, 2020.
[19] M.-L. Wang, B.-M. Zhang, Q. Xia, "Short term load forecasting using a multilayer neural network with conjugate gradient learning algorithm," *Automation of Electric Power Systems*, no. 01, pp. 35+35–36, 1999.
[20] K.-K. Tseng, R. Zhang, C.-M. Chen, M.-M. Hassan, "DNetUnet: a semi-supervised CNN of medical image segmentation for super-computing AI service," *The Journal of Supercomputing*, no. 77, no. 4, pp. 3594–3615, 2021.
[21] J. Vermaak, E.-C. Botha, "Recurrent neural networks for short-term load forecasting," *EEE Transactions on Power Systems*, vol. 13, no. 1, pp. 126–132, 1998.
[22] W. Kong, Z.-Y. Dong, D.-J. Hill, Y. Xu, Y. Zhang, "Short-Term Residential Load Forecasting Based on LSTM Recurrent Neural Network," *IEEE Transactions on Smart Grid*, vol. 10, no. 1, pp. 841–851, 2019.
[23] Y. Xiao, K.-H. Zheng, Z.-J. Zheng, B. Qian, S. Li, Q.-L. Ma, "Multi-scale skip deep long short-term memory network for short-term multivariate load forecasting," *Journal of Computer Applications*, vol. 41, no. 01, pp. 231–236, 2021.
[24] L. Zhang, X.-P. Lai, S.-Y. Zhong, K.-Y. Li, "Electricity load forecasting method based on orthogonal wavelet and long short-term memory neural networks," *Modern Electric Power*, vol. 39, no. 01, pp. 72–79, 2022.
[25] J.-X. Lu, Q.-P. Zhang, Z.-H. Yang, M.-F. Tu, J.-J. Lu, H. Peng, "Short-term load forecasting method based on CNN-LSTM hybrid neural network model," *Modern Electric Power*, vol. 43, no. 08, pp. 131–137, 2019.
[26] H. Wang, P. Li, M. Cao, Y.-H. Sun, "Long-Term and Short-Term Power Load Online Prediction Method Based on CNN_BiLSTM," *Computer Simulation*, vol. 39, no. 03, pp. 96–103, 2022.
[27] Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[28] K.-K. Tseng, R. Zhang, C.-M. Chen, M.-M. Hassan, "DNetUnet: a semi-supervised CNN of medical image segmentation for super-computing AI service," *The Journal of Supercomputing*, vol. 77, no. 4, pp. 3594–3615, 2021.

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

[30] S. Kumar, A. Damaraju, A. Kumar, S. Kumari, C.-M. Chen, "LSTM Network for Transportation Mode Detection," *Journal of Internet Technology*, vol. 22, no. 4, pp. 891–902, 2021.

[31] R. Chen, C.-Y. Liang, F.-W. Xie, "Application of nonlinear time series forecasting methods based on support vector regression," *Journal of Hefei University of Technology(Natural Science)*, vol. 36, no. 03, pp. 369–374, 2013.