A Deep Neural Network Model for Short-Term Load Forecast Based on Long Short-Term Memory Network and Convolutional Neural Network

Chujie Tian [1,*]**, Jian Ma** [1]**, Chunhong Zhang** [2] **and Panpan Zhan** [3]

[1] Institute of Network Technology, Beijing University of Posts and Telecommunications, Xitucheng Road No.10 Hadian District, Beijing 100876, China; jian.j.ma@gmail.com

[2] School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Xitucheng Road No.10 Hadian District, Beijing 100876, China; zhangch@bupt.edu.cn

[3] Beijing Institute of Spacecraft System Engineering, 104 YouYi Road Hadian District, Beijing 100094, China; panpan3210@163.com

* Correspondence: tianchujie@bupt.edu.cn; Tel.: +86-151-0158-4889

Abstract: Accurate electrical load forecasting is of great significance to help power companies in better scheduling and efficient management. Since high levels of uncertainties exist in the load time series, it is a challenging task to make accurate short-term load forecast (STLF). In recent years, deep learning approaches provide better performance to predict electrical load in real world cases. The convolutional neural network (CNN) can extract the local trend and capture the same pattern, and the long short-term memory (LSTM) is proposed to learn the relationship in time steps. In this paper, a new deep neural network framework that integrates the hidden feature of the CNN model and the LSTM model is proposed to improve the forecasting accuracy. The proposed model was tested in a real-world case, and detailed experiments were conducted to validate its practicality and stability. The forecasting performance of the proposed model was compared with the LSTM model and the CNN model. The Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE) and Root Mean Square Error (RMSE) were used as the evaluation indexes. The experimental results demonstrate that the proposed model can achieve better and stable performance in STLF.

Keywords: short-term load forecast; long short-term memory networks; convolutional neural networks; deep neural networks; artificial intelligence

---

1. Introduction

Demand Response Management (DRM) is one of the main features in smart grid that helps to reduce power peak load and variation [1]. The DRM controls the electricity consumption at the customer side and targets at improving energy-efficiency and reducing cost [2]. Accurate load forecasting has been more essential after deregulation of electricity industry [3]. It can minimize the gap between electricity supply and demand, while any error in the forecasting brings additional costs. In 1985, it was estimated that a 1% increase in forecasting error increases the associated operating costs of up to 10 million pounds every year in the thermal British power system [4]. Power companies are beginning to work with experts to explore models obtaining more accurate results in load forecasts. For instance, the National Grid in the United Kingdom (UK) is currently working with DeepMind [5,6], a Google-owned AI team, to predict the power supply and demand peaks in the UK based on the information from smart meters and incorporating weather-related variables. Therefore, precise load forecast is expected to reduce operation costs, optimize utilities and generate profits.

Load forecasting in energy management systems (EMS) can be categorized into four types according to different length of forecast interval [7]: (1) very short-term load forecasting (VSTLF) forecasts load for few minutes; (2) short-term load forecasting (STLF) forecasts load from 24 h to one week; (3) medium-term load forecasting (MTLF) forecasts load more than one week to few months; and (4) long-term load forecasting (LTLF) forecasts load longer than one year. In this paper, we focus on STLF. STLF is essential for controlling and scheduling of the power system in making everyday power system operation, interchange evaluation, security assessment, reliability analysis and spot price calculation [8,9], which leads to the higher accuracy requirement rather than long-term prediction.

The STLF problem has been tackled with various methods. These methods can be loosely categorized into two groups, namely traditional and computational intelligence methods. Statistical methods are most frequently used in early literature, including multiple linear regression [10,11], exponential smoothing [12], and the autoregressive integrated moving average (ARIMA) [13]. However, due to the inherent non-linear and the high requirement of the original time sequences of the electrical load data, these methods perform poorly in the STLF.

Computational intelligence methods have achieved great success and are widely used in load forecasting based on the non-linear learning and modeling capability, including clustering methods [14], fuzzy logic system [15], support vector machine (SVM) [16,17] and artificial neural networks [18]. In [19], a methodology based on artificial neural networks methods reinforced by an appropriate wavelet denoising algorithm is implemented to obtain short-term load forecasting, and the results show that the proposed method greatly improves the accuracy. Recently, deep learning frameworks have gained a particular attention [20]. Compared to shallow learning, deep learning usually involves a larger number of hidden layers, which makes the model able to learn more complex non-linear patterns [21]. As a deep learning framework with powerful learning ability to capture the non-stationary and long-term dependencies forecasting horizon [22], recurrent neural networks (RNNs) are effective methods for load forecasting in power grids. In [23], A novel pooling-based deep RNN is applied for household load forecast and achieves preliminary success. Compared with the state-of-the-art techniques in household load forecasting, the proposed method outperforms ARIMA by 19.5%, SVR by 13.1% and RNN by 6.5% in terms of RMSE. In [24], a new load forecasting model that incorporates one-step-ahead concept into RNN model is proposed. The performance in high or low demand regions is outstanding, which proves that the proposed electricity loads forecasting model can extract tinier fluctuations in different region than the other models. However, the vanishing gradient point is a problem for RNNs to improve the performance. To solve this problem, the long short-term memory (LSTM) and gated recurrent units (GRU), which variants of RNNs, have been proposed and perform well in long-term horizon forecasting based on the past data [25–27]. In [28],the proposed LSTM-based method is capable of forecasting accurately the complex electric load time series with a long forecasting horizon by exploiting the long-term dependencies. The experiments show that the proposed method performs better in complex electrical load forecasting scenario.. In [29], a method for short-term load forecasting with multi-source data using gated recurrent unit neural networks, which are used for extracting temporal features with simpler architecture and less convergence time in the hidden layers, is proposed. The average MAPE can be low as 10.98% for the proposed method, which outperforms other current methods, such as BPNNs, SAEs, RNNs and LSTM.

In addition to the above representative methods, the convolutional neural networks (CNNS) have been widely applied in the field of prediction. CNN can capture local trend features and scale-invariant features when the nearby data points typically have strong relationship with each other [30]. The pattern of the local trend of the load data in nearby hours can be extracted by CNN. In [31], a new load forecasting model that uses the CNN structure is presented and compared with other neural networks. The results show that MAPE and CV-RMSE of proposed algorithm are 9.77% and 11.66%, which are the smallest among all models. The experiments prove that the CNN structure is effective in the load forecasting and the hidden feature can be extracted by the designed 1D convolution layers. Based on the above literature, LSTM and CNN are both demonstrated to provide high accuracy

prediction in STLF due to their advantages to capture hidden features. Therefore, it is desired to develop a hybrid neural network framework that can capture and integrate such various hidden features to provide better performance.

This paper proposes a new deep learning framework based on LSTM and CNN. More specifically, it consists of three parts: the LSTM module, the CNN module and the feature-fusion module. The LSTM module can learn the useful information for a long time by the forget gate and memory cell, and CNN module is utilized to extract patterns of local trend and the same pattern which appears in different region. The feature-fusion module is used to integrate these hidden features and make the final prediction. The proposed CNN-LSTM model was developed and applied to predict a real-word electrical load time series. Additionally, several methods were implemented to be compared to our proposed model. To prove the validity of the proposed model, the CNN module and the LSTM module were also tested independently. Furthermore, the test dataset was divided into several partitions to test the stability of the proposed model. In summary, this paper proposes a deep learning framework that can effectively capture and integrate the hidden feature of the CNN model and the LSTM model to achieve higher accuracy and stability. From the experiments, the proposed CNN-LSTM model takes advantage of each components and achieves higher accuracy and stability in STLF.

The major contributions of this paper are: (1) a high precision STLF deep learning framework, which can integrate the hidden feature of the CNN model and LSTM model; (2) demonstrating the superiority of the proposed deep learning framework in real-word electrical load time series by comparisons with several models; (3) validating the practicality and stability of the proposed CNN-LSTM model in several partitions of test dataset; snf (4) a research direction in time sequence forecasting based on the integration of the hidden features of the LSTM and CNN model.

The rest of this paper is structured as follows: In Section 2, the RNN, LSTM, and CNN are introduced. In Section 3, the proposed CNN-LSTM neural network framework is proposed. In Section 4, the proposed model is applied to forecast the electrical load in a real-world case. Additionally, comparisons and analysis are provided. In Section 5, the discussion of the result is shown. Finally, we draw the conclusion in Section 6.

2. Methodologies of Artificial Neural Networks

This section provides brief backgrounds on several artificial neural networks, including RNN, LSTM, and CNN.

2.1. RNN

RNN is a kind of artificial neural network shown to have a strong ability to capture the hidden correlations occurring in data in applications for speech recognition, natural language processing and time series prediction. It is particularly suitable for modeling sequence problems by operating on input information as well as a trace of previously acquired information due to recurrent connections [32]. As shown in Figure 1, the mapping of one node S_t and the output O_t can be represented as:

$$S_t = f(U \times X_t + W \times S_{t-1}) \tag{1}$$

$$O_t = g(V \times S_t) \tag{2}$$

where S_t is the memory of the network at time t; U, W and V are the share weight matrix in each layer; X_t and O_t represents the input and the output at time t; and $f(.)$ and $g(.)$ represent the nonlinear function.
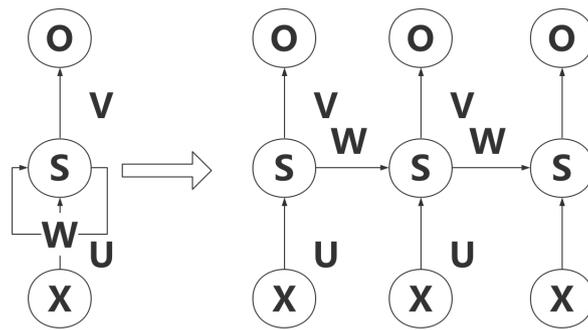
Figure 1. A simple recurrent neural network structure.

Unlike the weight connection established between the layers in the basic neural network, RNN can use the internal state (memory) to process sequence of inputs [33]. The hidden state captures the information at the previous point time, and the output is derived from the current time and previous memories. RNN performs well when the output is close to its associated inputs because the information of the previous node is passed to the next node. In theory, RNN is also able to deal with long dependencies. However, in practical applications, RNN cannot memorize the previous information well when the time interval is long due to the gradient vanishing problem. To solve these weaknesses and enhance the performance of the RNN, a special type of RNN architecture called LSTM was proposed.

2.2. LSTM

To overcome the aforementioned disadvantages of traditional RNNs, LSTM combines short-term memory with long-term memory through the gate control. As shown in the Figure 2, a common unit consists of a memory cell, an input gate, an output gate, and a forget gate. The input X_t at time t is selectively saved into cell C_t determined by the input gate, and the state of the last moment cell C_{t-1} is selectively forgotten by the forget gate. Finally, the output gate controls which part of the cell C_t is added to the output h_t.
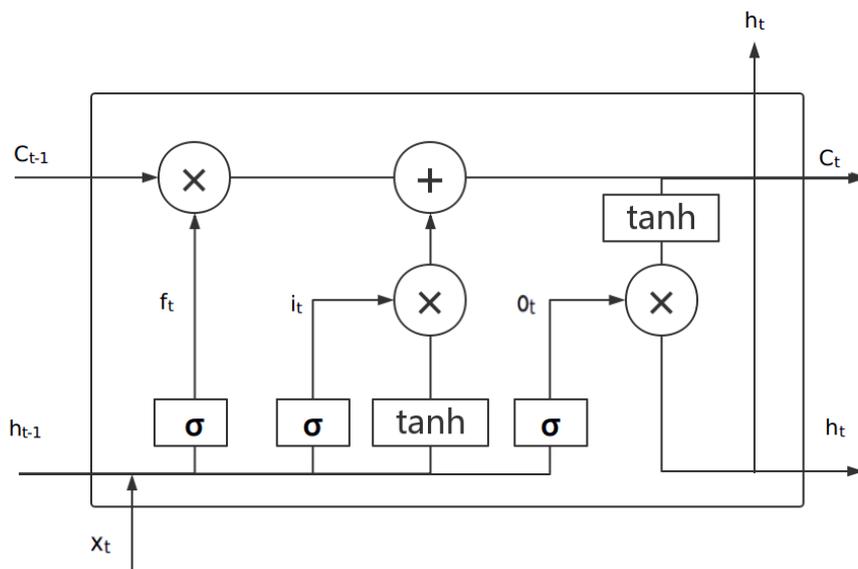
Figure 2. Inner structure of LSTM.

The calculation of the input gate i_t and forget gate f_t can be, respectively, expressed as:

$$i_t = \sigma(W_i \times [h_{t-1,x_t}] + b_i) \tag{3}$$

$$f_t = \sigma(W_f \times [h_{t-1,x_t}] + b_f) \tag{4}$$

where W_i and W_f are the weight matrices, h_{t-1} is the output of the previous cell, x_t is the input, and b_i and b_f are the bias vectors.

The next step is to update the cell state C_t, which can be computed as:

$$C_t = f_t \times C_{t-1} + i_t \times (tanh(W_c \times [h_{t-1}, x_t] + b_c)) \tag{5}$$

where W_c is the weight matrix, b_c is the bias vector, and C_{t-1} is the state of the previous cell.

The output gate o_t and the final output h_t can be expressed as:

$$o_t = \sigma(W_o \times [h_{t-1,x_t}] + b_o) \tag{6}$$

$$h_t = \sigma_t \times tanh(C_t) \tag{7}$$

where W_o is the weight matrix and b_o is the bias vector.

2.3. CNN

CNN is a kind of deep artificial neural networks. CNN is most commonly applied to deal with tasks in which data have high local correlation, such as visual imagery, video prediction, and text categorization. It can capture when the same pattern appears in different regions. CNN requires minimal preprocessing by using a variation of multilayer perceptrons, and is effective at dealing with high-dimensional data based on their shared-weights architecture and translation invariance characteristics.

CNN usually consists of convolutional layers, pooling layers and fully-connected layers. Convolutional layers apply a convolution operation to the input. The purpose of the convolution operation is to extract different features of the input, and more layers can iteratively extract complex features from the last feature. As shown in Figure 3, each convolutional layer is composed of several convolutional units, and the parameters of each convolution units are optimized by a back propagation algorithm. Generally, features with a large dimension are obtained after the convolutional layer, which need to be dimension-reduced. Pooling layers combine the outputs of neuron clusters at one layer into a single neuron in the next layer. Fully-connected layer, which combines all local features into global features, is used to calculate the final result.
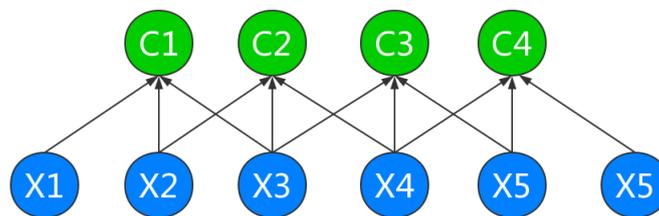
Figure 3. The one-dimensional convolutional layer.

3. The Proposed Method

In this section, we describe our CNN-LSTM based hybrid deep learning forecasting framework for STLF. It is motivated by the combination of CNN and LSTM, which considers the local trend and the long-term dependency of load data.

3.1. The Overview of the Proposed Framework

The structure of the proposed hybrid deep neural network is shown in Figure 4. The inputs are the information of the load value in the past few hours, and the outputs represent the prediction of the future load values. The proposed framework mainly consists of a CNN module, a LSTM module and a feature-fusion module.
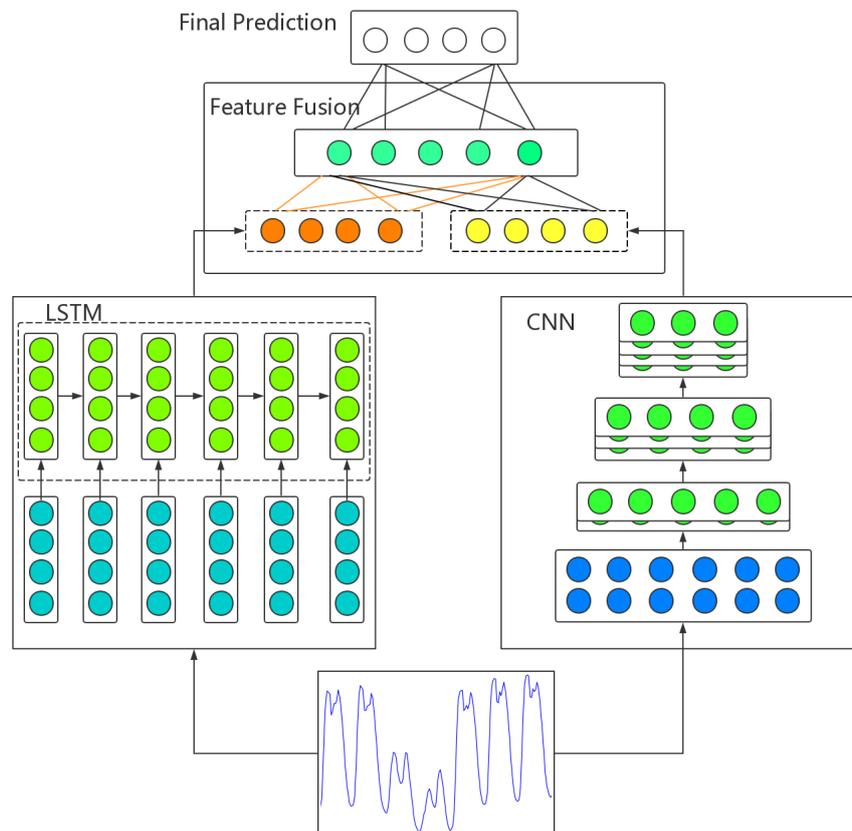
Figure 4. The proposed framework.

In the data preparation step, null values are checked and the load data are split into training and test sets. Then, the origin data are transferred into two different datasets. The CNN module is used to capture the local trend and the LSTM module is utilized to learn the long-term dependency. The two hidden feature are concatenated in the feature-fusion module. The final prediction is generated after a fully-connected layer. In the following, the detailed structure of each components is described.

In the CNN module, the main target is to capture the feature of the local trend. The inputs are the standardized load datasets, and the outputs are the prediction of the trend in next few hours. The main structure of the CNN module is performed by three convolution layers (Conv1, Conv2, and Conv3). Convolution layers are one-dimensional convolutions, and the activation function is the Rectified Linear Unit (RELU). The hidden feature of the CNN module is constructed to integrate with the feature of the LSTM module in the feature-fusion module.

The LSTM module is used to capture the long-term dependency. The inputs are reshaped for LSTM structure, and the prediction target is the maximum value of the next few hours. The hidden neurons of the output of the LSTM module are same as the CNN module.

After the process of the CNN module and the LSTM module, the outputs of the two modules are concatenated in the merge layer of the feature-fusion module. The final prediction is generated after a fully-connected layer.

3.2. Model Evaluation Indexes

To evaluate the performance of the proposed model, the Mean Absolute Error (MAE), the Mean Absolute Percentage Error (MAPE) and the Root Mean Square Error (RMSE) are employed. The error measures are defined as follows:

$$\text{MAE} = \frac{1}{N} \sum_{L=1}^{N} |(\widehat{y_L} - y_L)| \tag{8}$$

$$\text{MAPE} = \frac{1}{N} \sum_{L=1}^{N} |(\frac{\widehat{y_L} - y_L}{y_L})| \tag{9}$$

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{L=1}^{N} (\widehat{y_L} - y_L)^2} \tag{10}$$

where N is the size of training or test samples, and $\widehat{y_L}$ and y_l are the predicted value and actual value, respectively.

The MAE is the average of the absolute errors between the predicted values and actual values, which reflects the actual predicted value error. The MAPE further considers the ratio between error and the actual value. The RMSE represents the sample standard deviation of differences between the predicted values and the actual observed values. The smaller are the values of MAE, MAPE and RMSE, the better is the forecasting performance.

4. Experiments and Results

The proposed model was applied to forecast the electrical load in a real-world case. In this section, the experiments are described in detail, and comparisons with LSTM, CNN and the proposed model are also presented.

4.1. Datasets Description

In the experiment, the electric load dataset in the Italy-North Area provided by entsoe Transparency Platform was used. The period of the particular dataset used in this paper is from 1 January 2015 to 31 December 2017. The data sampling was one hour. The electrical dataset contains a total of 26,304 samples. In this study, the load data for first two years were chosen as the training set. The test data were collected in 2017. An example of the test dataset is shown in Figure 5.
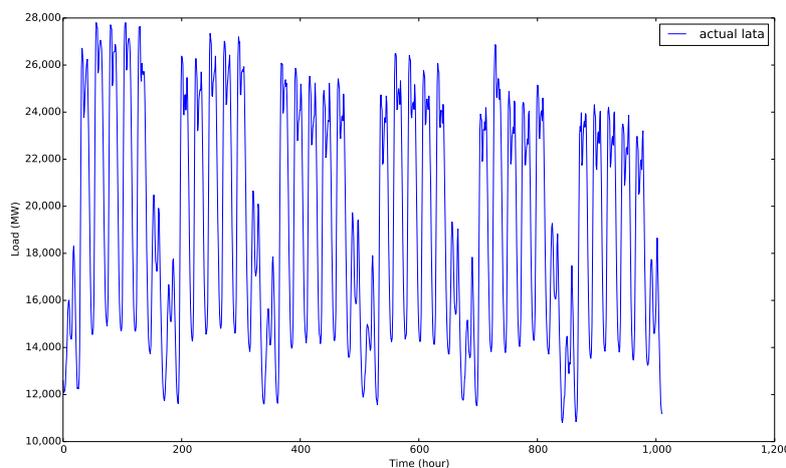
Figure 5. An example of the load data in test dataset.

4.2. The Detailed Experimental Setting

The past 21 × 24 h load data were selected as the input variable of the model, and the output was the load in the next 24 h. In the CNN module, the kernel sizes of the convolutional layer are 5, 3, and 3, and the filter sizes are 16, 32, and 64. The feature maps are all activated by the Rectified Linear Unit (ReLU) function. The hidden neuron of the LSTM module was set as 100. The sigmoid function was chosen to be the active function of the fully-connected layer. The training process continued until the MSE value had no improvement in 500 iterations or the maximal number of epochs was reached.

4.3. Experimental Results and Analysis

In this application, random forest (RF), decision tree (DT), DeepEnergy (DE) [28] and the proposed CNN-LSTM model were implemented and tested in the prediction of the next 24 h load forecast. Besides, the CNN module and the LSTM module were also extracted and tested to demonstrate the superiority of our proposed model. The result obtained by the proposed CNN-LSTM model is illustrated in Figure 6. To evaluate the performance and stability of the proposed model, the test dataset was divided into eight partitions. The detailed experimental results of each model are illustrated in Tables 1–3.
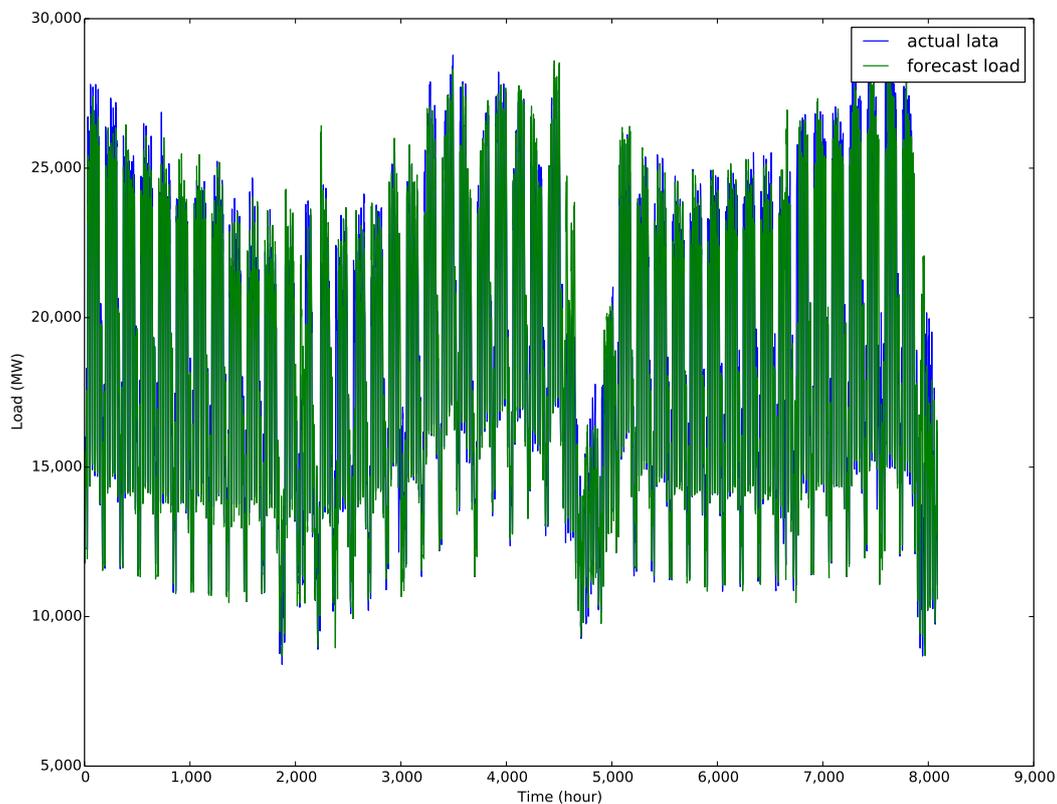
Figure 6. The forecast result using the proposed CNN-LSTM model for test data.

Table 1. The experimental results in terms of Mean Absolute Error (MAE).

Test	DT	RF	DE	CNN	LSTM	CNN-LSTM
Test-1	669.3277	509.4035	467.4859	476.7259	480.7812	470.3272
Test-2	841.6884	664.7461	610.9937	730.0540	729.0658	537.0471
Test-3	1085.2633	981.2681	1160.7618	1055.0703	1140.5746	1018.6435
Test-4	986.4937	851.9130	583.1347	753.4446	758.6537	495.8678
Test-5	1638.9530	1252.3642	880.7453	865.2113	1115.0330	858.6237
Test-6	741.9390	614.0473	439.6090	620.5186	539.8259	455.3989
Test-7	769.4685	718.3953	817.0670	892.1418	740.9322	741.5478
Test-8	1339.2667	1107.9678	1050.3163	1097.9277	1095.2742	959.7009
Test-avg	1009.0500	837.5132	751.2642	811.3868	825.0176	692.1446

Table 2. The experimental results in terms of Mean Absolute Percentage Error (MAPE).

Test	DT	RF	DE	CNN	LSTM	CNN-LSTM
Test-1	0.0332	0.0250	0.0236	0.0239	0.0249	0.0235
Test-2	0.0531	0.0414	0.0378	0.0465	0.0465	0.0327
Test-3	0.0686	0.0628	0.0726	0.0684	0.0734	0.0606
Test-4	0.0489	0.0425	0.0289	0.0371	0.0383	0.0244
Test-5	0.0977	0.0755	0.0531	0.0517	0.0669	0.0516
Test-6	0.0385	0.0314	0.0239	0.0336	0.0299	0.0241
Test-7	0.0428	0.0390	0.0447	0.0489	0.0411	0.0407
Test-8	0.0800	0.0658	0.0652	0.0672	0.0631	0.0594
Test-avg	0.0578	0.0479	0.0437	0.0472	0.0480	0.0396

Table 3. The experimental results in terms of Root Mean Square Error (RMSE).

Test	DT	RF	DE	CNN	LSTM	CNN-LSTM
Test-1	977.2206	755.5147	643.8908	627.4642	617.5835	612.4874
Test-2	1393.2847	1056.4105	907.0599	1146.4771	1119.7467	719.9939
Test-3	2070.3786	1880.0600	2102.2027	1906.8154	2054.4484	1859.0252
Test-4	1481.5294	1269.0204	753.5586	1027.8778	1109.1206	656.5774
Test-5	2364.5579	1876.6200	1323.3404	1295.1156	1569.6155	1340.6214
Test-6	1346.3700	1101.5862	585.0608	818.5694	798.6015	604.4891
Test-7	1444.5131	1373.6364	1669.9279	1624.9380	1487.9632	1467.0496
Test-8	2313.3100	1959.8469	1986.0953	1903.6605	1835.2911	1813.1891
Test-avg	1673.8955	1409.0869	1246.3920	1293.8648	1324.0463	1134.1791

As shown in Tables 1–3, the averaged MAE, MAPE and RMSE of the decision tree are the largest in the six models. The performance of the deep neural networks is much better than the decision tree and random forest. The results of the CNN module is a little better than the LSTM module, while they are both higher than the DeepEnergy. Although the performance of the independent CNN module and LSTM module is a little worse than the DeepEnergy, the proposed model, which integrates these two modules, provides better result. The average indexes of the proposed CNN-LSTM model are the minimum among all models: 692.1446, 0.0396 and 1134.1791. From the point of view of these three indexes, t the proposed model can improve performance by at least 9% compared to the DeepEnergy, 12% compared to the CNN module, and 14% compared to the LSTM module. Therefore, it is proven that our proposed CNN-LSTM model can make more accurate forecast by integrating the hidden feature of the CNN module and the LSTM module. According to the average indexes, it is demonstrated that our proposed CNN-LSTM model can achieve the best performance in STLF.

Meanwhile, it is also evident the our proposed model is stable. In the eight partitions of the test dataset, the results of the proposed model prove the superiority compared to the other forecasting methods. For a better visualization, the results of six models in the eight partitions are also illustrated in Figures 7–9. As shown in Figures 7–9, the curves that denote the proposed CNN-LSTM model are approximately the minimum among all partitions. Specifically, the MAE, MAPE and RMSE of the

proposed model are the minimum in half of the eight test partitions, i.e., Test-2, Test-4, Test-7 and Test-8. In the other four partitions, the proposed model also provides accurate forecast result. The MAPE and RMSE of the proposed model are also the minimum in Test-1 and Test-3, and the MAE is only higher than DeepEnergy in Test-1 and random forest in Test-3. Although the performance of the proposed model is not the best in Test-5 and Test-6, it is still one of the best three results. On the other hand, the performances of the independent CNN module and LSTM module are not stable. The MAPE of the LSTM module is the largest in Test-3, while it is good in Test-7 and Test-8. The CNN module has good performance in Test-1 and Test-5, while it performs the worst in Test-7. It is obvious that the proposed model has good performance in all eight partitions, which proves that the proposed CNN-LSTM model can improve the stability of the load forecast.
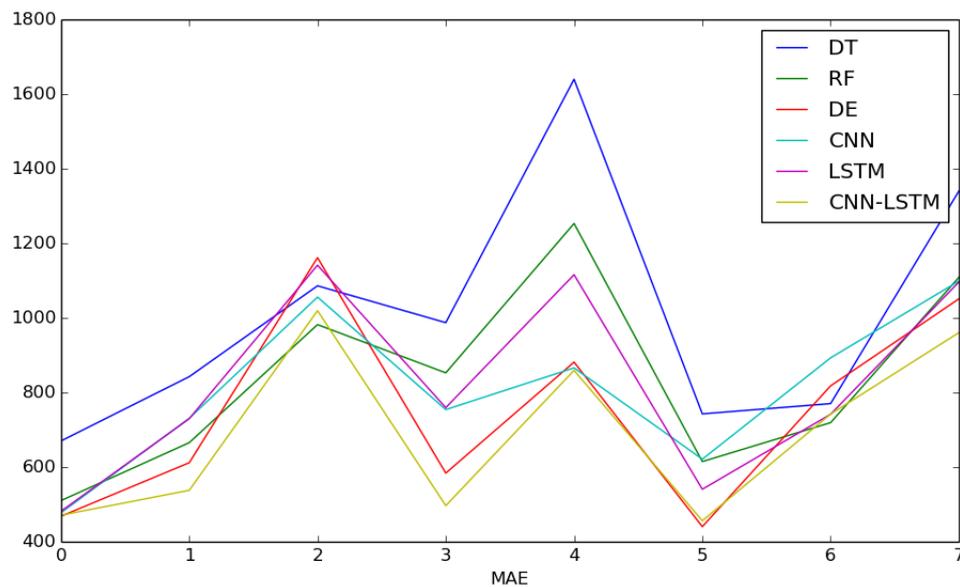
Figure 7. The comparison of the MAE in the six models.
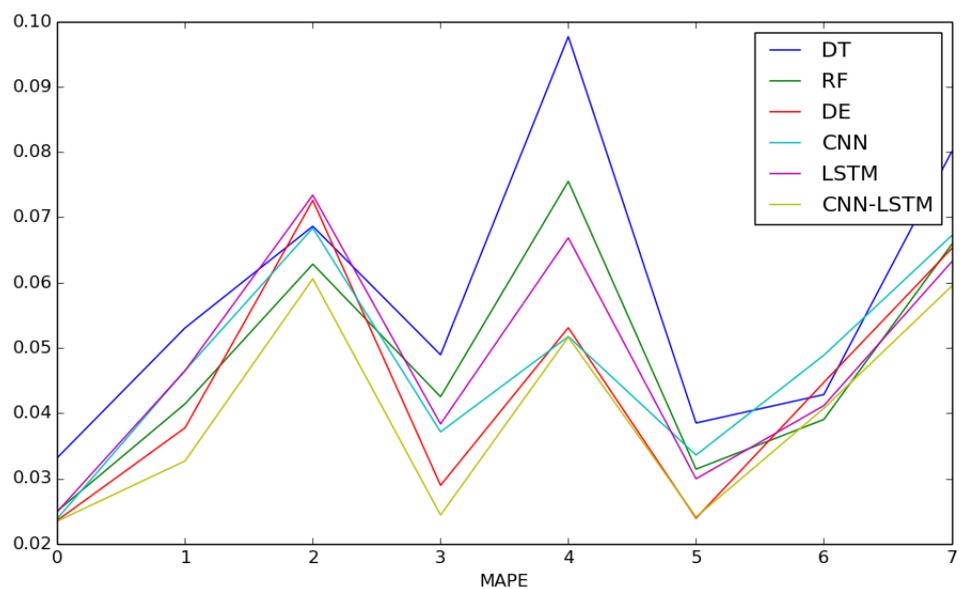
Figure 8. The comparison of the MAPE in the six models.
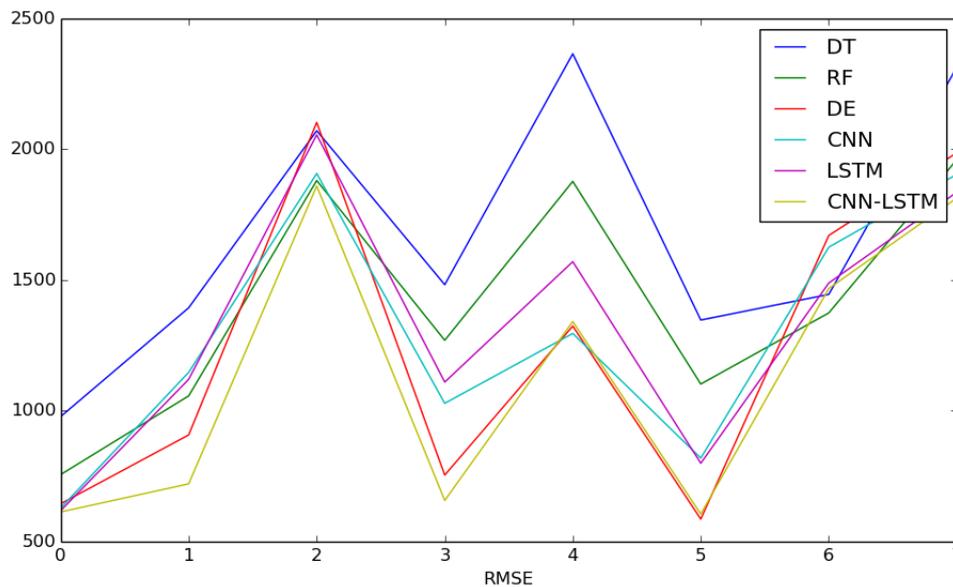
Figure 9. The comparison of the RMSE in the six models.

5. Discussion

Deep learning methods, such as CNN and LSTM, are widely used in many applications.In this study, CNN and LSTM provide more accurate results than random forest and decision tree. In aspect of the LSTM model, it can learn useful information in the historical data for a long period by the memory cell, while the useless information will be forgotten by the forget gate. According to the result, the LSTM module can make accurate load forecast by exploiting the long-term dependencies. On the other hand, the CNN model can extract patterns of local trend and capture the same pattern, which appears in different regions. The experiments also show that the CNN structure is effective in the load forecast. To further improve the accuracy and stability of the load forecast, a new deep neural network framework, which integrates the the CNN module and the LSTM module is proposed in this paper. In the experiments, our proposed CNN-LSTM model achieves the best performance among all models. Furthermore, the test dataset is divided into eight partitions to test the stability of the proposed model. The independent CNN module and LSTM module perform well in some partitions and poor in others, while the proposed model has good performance in all partitions. It demonstrates that the proposed model has better stability than independent module. The results prove that the integration of the hidden features of CNN model and LSTM model is effective in load forecast and can improve the prediction stability. This paper gives a new research direction in time sequence forecasting based on the integration of LSTM and CNN. Future studies can attempt to further improve the accuracy of the short-term electrical load forecast by more effective way to integrate the hidden features of LSTM and CNN.

6. Conclusions

This paper proposes a multi-step deep learning framework for STLF. The proposed model is based on the LSTM module, the CNN module and the feature-fusion module. The performance of the proposed model was validated by experiment with a real-world case of the Italy-North Area electrical load forecast. In addition, several partitions of test datasets were tested to verify the performance and stability of the proposed CNN-LSTM model. According to the results, the proposed model has the lowest values of MAE, MAPE and RMSE. The experiments demonstrate the superiority of the proposed model, which can effectively capture the hidden features extracted by the CNN module and LSTM module. The result shows a new research direction to further improve the accuracy and stability of the load forecast by integrating the hidden features of LSTM and CNN.

References

1. Chai, B.; Chen, J.; Yang, Z.; Zhang, Y. Demand Response Management With Multiple Utility Companies: A Two-Level Game Approach. *IEEE Trans. Smart Grid* **2014**, *5*, 722–731. [CrossRef]
2. Apostolopoulos, P.A.; Tsiropoulou, E.E.; Papavassiliou, S. Demand Response Management in Smart Grid Networks: A Two-Stage Game-Theoretic Learning-Based Approach. *Mob. Netw. Appl.* **2018**, 1–14. [CrossRef]
3. Chen, Y.; Luh, P.B.; Guan, C.; Zhao, Y.; Michel, L.D.; Coolbeth, M.A.; Friedland, P.B.; Rourke, S.J. Short-Term Load Forecasting: Similar Day-Based Wavelet Neural Networks. *IEEE Trans. Power Syst.* **2010**, *25*, 322–330. [CrossRef]
4. Bunn, D.; Farmer, E.D. *Comparative Models for Electrical Load Forecasting*; Wiley: New York, NY, USA, 1986; p. 232.
5. Oh, C.; Lee, T.; Kim, Y.; Park, S.; Kwon, S.B.; Suh, B. Us vs. Them: Understanding Artificial Intelligence Technophobia over the Google DeepMind Challenge Match. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, Denver, CO, USA, 6–11 May 2017; pp. 2523–2534.
6. Skilton, M.; Hovsepian, F. Example Case Studies of Impact of Artificial Intelligence on Jobs and Productivity. In *4th Industrial Revolution*; Skilton, M., Hovsepian, F., Eds.; National Academies Press: Washington, DC, USA, 2018; pp. 269–291.
7. Singh, P.; Dwivedi, P. Integration of new evolutionary approach with artificial neural network for solving short term load forecast problem. *Appl. Energy* **2018**, *217*, 537–549. [CrossRef]
8. Ertugrul, Ö.F. Forecasting electricity load by a novel recurrent extreme learning machines approach. *Int. J. Electr. Power Energy Syst.* **2016**, *78*, 429–435. [CrossRef]
9. Zhang, Z.; Liang, G.U.O.; Dai, Y.; Dong, X.U.; Wang, P.X. A Short-Term User Load Forecasting with Missing Data. *DEStech Trans. Eng. Technol. Res.* **2018**. [CrossRef]
10. Charytoniuk, W.; Chen, M.S.; Olinda, P.V. Nonparametric regression based short-term load forecasting. *IEEE Trans. Power Syst.* **1998**, *13*, 725–730. [CrossRef]
11. Song, K.B.; Baek, Y.S.; Hong, D.H.; Jang, G. Short-term load forecasting for the holidays using fuzzy linear regression method. *IEEE Trans. Power Syst.* **2005**, *20*, 96–101. [CrossRef]
12. Christiaanse, W.R. Short-term load forecasting using general exponential smoothing. *IEEE Trans. Power Appl. Syst.* **1971**, *2*, 900–911. [CrossRef]
13. Lee, C.M.; Ko, C.N. Short-term load forecasting using lifting scheme and ARIMA models. *Expert Syst. Appl.* **2011**, *38*, 5902–5911. [CrossRef]
14. Çevik, H.H.; Çunkaş, M. Short-term load forecasting using fuzzy logic and ANFIS. *Neural Comput. Appl.* **2015**, *26*, 1355–1367. [CrossRef]
15. Yu, F.; Hayashi,Y. Pattern sequence-based energy demand forecast using photovoltaic energy records. In Proceedings of the 2012 International Conference on Renewable Energy Research and Applications. Nagasaki, Japan, 11–14 November 2012; pp. 1–6.
16. Barman, M.; Choudhury, N.B.D.; Sutradhar, S. Short-term load forecasting using fuzzy logic and A regional hybrid GOA-SVM model based on similar day approach for short-term load forecasting in Assam, India. *Energy* **2018**, *145*, 710–720. [CrossRef]
17. Chen, Y.; Xu, P.; Chu, Y.; Li, W.; Wu, Y.; Ni, L.; Bao, Y.; Wang, K. Short-term electrical load forecasting using the Support Vector Regression (SVR) model to calculate the demand response baseline for office buildings. *Appl. Energy* **2017**, *195*, 659–670. [CrossRef]
18. Ding, N.; Benoit, C.; Foggia, G.; Besanger, Y.; Wurtz, F. Neural network-based model design for short-term load forecast in distribution systems. *IEEE Trans. Power Syst.* **2016**, *31*, 72–81. [CrossRef]

19. Ekonomou, L.; Christodoulou, C.A.; Mladenov, V. A short-term load forecasting method using artificial neural networks and wavelet analysis. *Int. J. Power Syst.* **2016**, *1*, 64–68.

20. Merkel, G.; Povinelli, R.; Brown, R. Short-Term load forecasting of natural gas with deep neural network regression. *Energies* **2018**, *11*, 2008. [CrossRef]

21. Bouktif, S.; Fiaz, A.; Ouni, A.; Serhani, M. Optimal Deep Learning LSTM Model for Electric Load Forecasting using Feature Selection and Genetic Algorithm: Comparison with Machine Learning Approaches. *Energies* **2018**, *11*, 1636. [CrossRef]

22. Zheng, H.; Yuan, J.; Chen, L. Short-term load forecasting using EMD-LSTM neural networks with a Xgboost algorithm for feature importance evaluation. *Energies* **2017**, *10*, 1168. [CrossRef]

23. Shi, H.; Xu, M.; Li, R. Deep Learning for Household Load Forecasting—A Novel Pooling Deep RNN. *IEEE Trans. Smart Grid* **2018**, *9*, 5271–5280. [CrossRef]

24. Wei, L.Y.; Tsai, C.H.; Chung, Y.C.; Liao, K.H.; Chueh, H.E.; Lin, J.S. A Study of the Hybrid Recurrent Neural Network Model for Electricity Loads Forecasting. *Int. J. Acad. Res. Account. Financ. Manag. Sci.* **2017**, *7*, 21–29. [CrossRef]

25. Gensler, A.; Henze, J.; Sick, B.; Raabe, N. Deep Learning for solar power forecasting—An approach using AutoEncoder and LSTM Neural Networks. In Proceedings of the 2016 IEEE International Conference on Systems, Man, and Cybernetics, Budapest, Hungary, 9–12 October 2016; pp. 858–865.

26. Chen, Z.; Sun, L.X.; University, Z. Short-Term Electrical Load Forecasting Based on Deep Learning LSTM Networks. *Electron. Technol.* **2018**, *47*, 39–41.

27. Kuan, L.; Yan, Z.; Xin, W.; Yan, C.; Xiangkun, P.; Wenxue, S.; Zhe, J.; Yong, Z.; Nan, X.; Xin, Z. Short-term electricity load forecasting method based on multilayered self-normalizing GRU network. In Proceedings of the 2017 IEEE Conference on Energy Internet and Energy System Integration, Beijing, China, 26–28 November 2018; pp. 1–5.

28. Zheng, J.; Xu, C.; Zhang, Z.; Li, X. Electric load forecasting in smart grids using Long-Short-Term-Memory based Recurrent Neural Network. In Proceedings of the 51st Annual Conference on Information Sciences and Systems (CISS), Baltimore, MD, USA, 22–24 March 2017; pp. 1–6.

29. Wang, Y.; Liu, M.; Bao, Z.; Zhang, S. Short-Term Load Forecasting with Multi-Source Data Using Gated Recurrent Unit Neural Networks. *Energies* **2018**, *11*, 1138. [CrossRef]

30. Du, S.; Li, T.; Gong, X.; Yang, Y.; Horng, S.J. Traffic flow forecasting based on hybrid deep learning framework. In Proceedings of the 12th International Conference on Intelligent Systems and Knowledge Engineering, Nanjing, China, 24–26 November 2017; pp. 1–6.

31. Kuo, P.H.; Huang, C.J. A High Precision Artificial Neural Networks Model for Short-Term Energy Load Forecasting. *Energies* **2018**, *11*, 213. [CrossRef]

32. Hsu, D. Time Series Forecasting Based on Augmented Long Short-Term Memory. *arXiv* **2017**, arXiv:1707.00666.

33. Che, Z.; Purushotham, S.; Cho, K.; Sontag, D.; Liu, Y. Recurrent Neural Networks for Multivariate Time Series with Missing Values. *Sci. Rep.* **2018**, *8*, 6085. [CrossRef]