Available online at www.ijpe-online.com

International Journal of Performability Engineering

vol. 15, no. 11, November 2019, pp. 2927-2935
DOI: 10.23940/ijpe.19.11.p11.29272935

A Novel Ensemble Forecasting Algorithm based on Distributed Deep Learning Network

Tao Ma*, Fen Wang, Yanshan Tian, Yan Ma, and Xu Ma

School of Mathematical and Computer Science, Ningxia Normal University, Guyuan, 756000, China

Abstract

This paper proposes an ensemble model based on distribution deep learning network. The ensemble model is composed of deep belief network (DBN) for reconstructing original data, and the bidirectional long short-term memory (BLSTM) method is used for prediction due to its good results in big data applications. The dynamic weighting strategies are proposed and applied to the sub models of the ensemble by a weighted least square method. The weight update with variable training sets and the predictions for each ensemble are obtained from the distributed computing engine Apache Spark. The performance of the proposed model is evaluated on wind data on the wind farm of the Hexi Corridor in China. The simulation results show that the dynamic ensemble algorithm performs well, which is a very valuable result for the forecasting of big data time series. Furthermore, the results are successfully compared with back propagation neural Network (BPNN), LSTM, BLSTM, and stacked LSTMs with memory between batches (SBLSTM), improving the accuracy of prediction.

Keywords: distributed deep learning network; wind time series; ensemble; forecasting; big data

(Submitted on August 7, 2019; Revised on October 11, 2019; Accepted on November 10, 2019)

1. Instruction

Information in big data can sometimes be difficult to understand or analyze. Data mining technology provides a solution to this problem and obtain useful knowledge from big data. In fact, data mining has shifted toward big data and has been used for actual forecasting and correlation analysis, such as predicting earthquakes, medical image analysis, urban traffic height [1], and even weather analysis [2]. Today, as data development progresses, collecting information has become easier. This results in the collection of thousands of petabytes of data per day, and new data mining techniques are needed to deal with such massive data. A new framework for the successful implementation of data mining methods has emerged. The most famous MapReduce framework was developed by Google [3], and it allows developer to compute data mining algorithms in parallel on a set of machines, making data analysis and processing faster and more efficient. The Apache Hadoop software [4] quickly became popular in the scientific community, because it successfully developed with the features of MapReduce paradigm and the project of open source software. However, Hadoop has had several problems in developing iterative calculation methods, so other distributed parallel computing methods have been introduced. The project of Apache Spark [5] has proven to be one of the most successful. Since whole calculations are performed in Spark's memory, the program will execute significantly faster than Hadoop. A daunting task is to configure Spark in the cluster, so its automation is a major research direction [6]. Nonetheless, there are several problems in the Spark framework. In order to effectively handle the problem of Spark subsequence sorting and developing deep learning based on it, it is a worthwhile research area to combine the Spark framework with deep learning and distributed computing.

Given the above, in order to better mine the knowledge in the big data time series and develop the deep learning prediction algorithm based on the Spark framework, we propose an ensemble method of dynamic weights to implement parallel processing of different sub-problems. First, the big data time series is divided into three data sets, which are a training set, a verification set, and a test set. The verification set is mainly used to calculate the weight of every sub-model

* Corresponding author.
E-mail address: matao@nxnu.edu.cn

of the ensemble, and the weight is updated by the periodic replacement verification set. Second, the high-dimensional feature vector contained in the complex data is characterized by the DBN method. The feature of the data input into the network is selected by the method of unsupervised greedy layer-by-layer mapping, with the visible layer and the hidden layer in the DBN network. The DBN network can reconstruct the original data in a smaller dimension. Thirdly, using the newly generated data to train the improved BLSTM model, the memory unit can fully exploit the information in the overall data by processing the different length sequences. Finally, the processed results are combined to obtain the predicted results.

In this work, we discuss the strategy of dynamic weight distribution and propose distributed prediction using DBNs for data reconstruction and parallel BLSTM as a sub-model of the ensemble to improve the utilization of wind energy resources and the decision-making department of the wind farm.

2. The Research

In general, the predicting method of time series can be divided into statistical methods and data mining methods, such as GARCH, support vector machine (SVM), and artificial neural network (ANN). Predicting time series within the energy field applies the classification of the above techniques [7].

However, when large data must be processed, existing data mining methods cannot be used for big data processing due to high computational cost. Thus, distributed computing develops methods for big data mining techniques [8] to solve classification, clustering, or regression tasks. The following is an abbreviated overview of research results in this area. Recently, several big data scene methods based on MapReduce have been applied for traditional tasks. The parallel-based SVM algorithms have been applied in the field of high performance computing [9]. Several parallel KNN algorithms based on Spark have been proposed in [10]. In addition, the computational cost of the kNN method has been reduced and the storage method has been simplified, as proposed in the MapReduce framework [11].

Deep learning methods have been successfully applied in the field of fraud detection and video processing. The distributed computing approach can support the computational requirements of deep learning networks and reduce training time by parallelizing the training phase of the network. Researching and building a distributed deep learning method network in Spark can realize the needs of large-scale machine learning. Existing research methods generally rely on Hadoop's distributed file system and the memory of Spark framework computing for parallel learning. The DBN is usually composed of a multi-layer constrained Boltzmann machine (RBM), which trains the model in a layer-by-layer unsupervised manner and adjusts the parameters in the DBN to optimize the network structure by using a labeled supervised method.

Most of the above research results are based on the application of Hadoop framework, but they lack analysis and applications for time series of big data. In this work, we focus on the combination of deep learning methods and distributed computing platforms. Aiming at the low density of massive historical data, we propose an ensemble method based on DBN and BLSTM to be implemented on the Spark framework. On this basis, we obtain the dynamic weight update algorithm to represent the relationship between sub-problems of the ensemble. The proposed prediction method can not only solve the high computational cost of the big data prediction algorithm, but also effectively analyze the current value and historical data of the time series, which will help wind farms utilize the wind energy resources more efficiently.

3. Methodology

In order to solve the shortcomings of the high computational cost of the data time series, we propose a distributed deep learning network method based on the dynamic weighting computing in Spark. The algorithm uses DBN and BLSTM as sub-models of the ensemble, and this method is called DBNLSTM. First, the raw data is divided into three parts: training data, verification data, and test data. Secondly, k sub-models are constructed respectively, in which DBN is used for data reconstruction. In this way, the reconstructed data is generated from DBN as the input data of BLSTM and predicted. The weight matrix is calculated from the error between the true value of the data and the predicted value of each sub model. Third, the test data is applied to the weight matrix and the trained ensemble model for prediction results. Finally, the training data is updated, the weight matrix is recalculated, and the updated matrix and every predicted value in the ensemble are integrated to obtain the final prediction result.

3.1. Dynamic Weight Method

To obtain the weighting coefficients for every sub model, the prediction error is minimized on the verification data set. The number of sub models is set as K in the Spark framework. The parameter h is the prediction range based on N instances for verification. Next, the error between the prediction results is minimized based on the weighted least squares method. The

required formula is as follows:

- Suppose M^j is a weight value matrix with prediction value of validation instances and k sub models in the j^{th} value of the forecasting range, W^j is a vector containing weights of each sub models, and p^j is the actual value of j^{th} values of the validation set.

$$M^j W^j = p^j \tag{1}$$

We use a weight matrix to predict the testing set when the weight matrix is obtained by minimizing the squared error. We assume that the test set consists of N_1 instances and the i^{th} prediction horizon represented by matrix W is calculated by the linear combination of prediction of the k sub model.

- Where M^i is a matrix with N_1 row and k columns and represents the i^{th} prediction with k sub models. W^i is the weight matrix obtained from Equation (1). W can update the weights after setting the time interval in the training set.

$$p^i = [M^1 W^1, \cdots, M^i W^i] \tag{2}$$

3.2. Proposed Ensemble Model

We propose a combined prediction ensemble method based on DBN for the feature selection of raw data and the prediction of reconstructed data by the improved LSTM model, which is called DBNLSTM. The method takes advantage of the deep learning framework. DBN characterizes the data, and the reconstructed data accurately represents the useful information in the original data. The method enables the LSTM unit to extract the overall information of the data, learn more useful rules, and achieve better predictive performance.

DBNLSTM divides the raw data into a seven-window pattern using different time selection methods and passes the corresponding multi-factor data such as environmental data, meteorological data, and economic statistics to the DBN model for processing. The detailed steps are as follows:

Step 1 The raw data is processed according to different time window mode-consisting environmental characteristics in data. The processed data is DBN model input data, and the number of visible layers in the DBN is the same as the dimensions of input data. Because the DBNLSTM algorithm contains two restricted Boltzmann machine (RBM) networks, the visual layer can be reconstructed through a given hidden layer unit during each iteration as follows:

- Where v is a visible layer, h is a hidden layer, and p is an activation probability.

$$p(v, h^1, h^2) = p(v \mid h^1) p(h^1 \mid h^2) \tag{3}$$

Step 2 The generated data of DBN is used for prediction by the BLSTM network based on deep architecture.

- Assuming t is the overall size of the data and M is the size of the training data set, the predicted values can be expressed as follows:

$$\hat{y} = \{\hat{y}_{[M]}, \hat{y}_{[M+1]}, \cdots, \hat{y}_{[t]}\} \tag{4}$$

The vector of input of DBNLSTM can represented as follows:

- Where i represents the size of the windows, $tempe_{[t]}$ is the temperature variable at the t time point, and $hour_{[t]}$ is the time variable. The adaptive moment estimation (ADAM) method is used instead of the traditional stochastic gradient descent (SGD) algorithm, because ADAM has a lower error rate and faster convergence speed than the SGD algorithm.

$$i_{[t]} = [\hat{y}_{[t-1]}, tempe_{[t]}, hour_{[t-i]}, \cdots, hour_{[t-i+1]}] \tag{5}$$

$$f_{[t]} = [tempe_{[t]}, hour_{[t-i]}, \cdots, hour_{[t-i+1]}] \tag{6}$$

Step 3 The DBN network can represent the original data with less data through the process of multiple iterations and reverse adjustments. The feature-selected data is trained using the improved LSTM method. The sequence-to-sequence method and LSTM-used units of BLSTM are used to process time series of different lengths in the data.

Step 4 The coding network uses pre-training to generate a code in the training phase and minimize the prediction error. The decoding network updates the weights in the LSTN network by training the own network, and the trained network is used for wind speed prediction. The algorithm flow chart and network structure of DBNLSTM are shown in Figure 1.

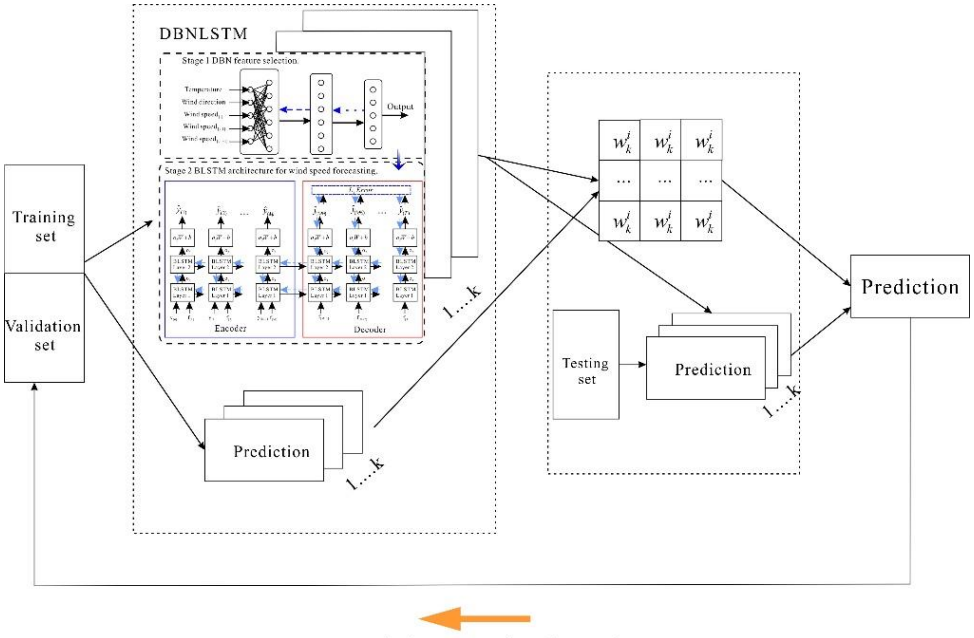

Figure 1. DBNLSTM algorithm workflow and network structure

Finally, the trained DBNLSTM model is used for short wind speed predication.

3.3. Time Complexity Analysis

The DBNLSTM algorithm is divided into two parts. The first part is the process of selecting the original data by the DBN network. t_{rbm} represents the training time for each RBM, t_{tune} is the final adjustment time of the whole DBN network, and t_{bp} is the time of supervised fine tuning. The time spent by the training phase of DBN is as follows:

$$T_{dbn} = t_{rbm} + t_{tune} + t_{bp} \tag{7}$$

The second part is the process of training and predicting using the improved LSTM network, and the time of complexity of the second part can be expressed as follows:

- Where I is the input layer unit of LSTM, the number of units is H in the hidden layer of LSTM, and the number of output layer units is the K value.

$$T_{lstm} = 2 \times (4IH + 4H^2 + 3H + HS) \tag{8}$$

- Because DBNLSTM is deployed on the Spark platform, K DBNLSTM are trained and predicted in parallel. Therefore, the total time complexity of the algorithm should consider the computational cost of each sub model. The time complexity is shown as follows:

$$\begin{aligned} T_{dbnlstm} &= K \times (T_{dbn} + T_{blstm}) \\ &= t_{rbm} + t_{tune} + t_{bp} + 2 \times (4IH + 4H^2 + 3H + HK) \\ &= O(N^3) \end{aligned} \tag{9}$$

4. Experimental Results

To test the performance of the DBNLSTM model, the wind speed data of wind farms in the Hexi Corridor region of northwest China is used to test the performance of the proposed model. Because the wind speed data is influenced by multiple environmental conditions, there is high volatility and low correlation among the wind speed data, including environmental and meteorological characteristics. In order to comprehensively analyze the wind speed data from wind farms, this section analyzes and predicts the wind speed data according to different seasons. The predicted results of the DBNLSTM model are compared and analyzed with the predictions of the state of the art methods to verify the validity of the proposed model.

4.1. Data Collection

Wind turbines of wind farms mainly generate electricity by wind speed. In order to make wind turbines generate stable and available electric energy, it is necessary to improve the accuracy of wind speed prediction, which has higher requirements for prediction models. In this section, the wind speed data of wind farms in the Hexi Corridor area of China is used as the experimental data set to test the predictive performance of the DBNLSTM model.

The data set contains three years of data from January 2002 to December 2004, including wind speed data collected from a 10-meter-high observatory, current temperature data, and corresponding wind direction data. The data for the first two years is used as the training set and validation set, and the data for the last year is used as the test data set. The performance of the proposed method is compared with the current main prediction models, such as BPNN, SVM, LSTM, BLSTM, and SBLSTM. The three-year wind speed data and the average first week of the four seasons are shown in Figure 2.

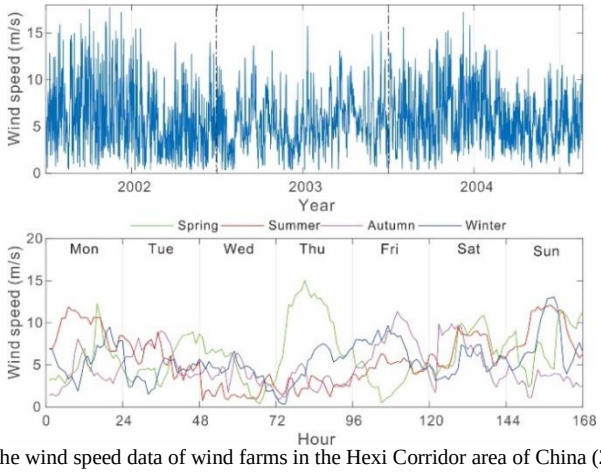

Figure 2. The wind speed data of wind farms in the Hexi Corridor area of China (2002-2004)

From Figure 2, it can be seen that the wind speed data in the region changes rapidly and fluctuates, and the wind speed data varies across the four seasons of the year. In addition, the wind speed and temperature are closely related. In order to analyze the correlation between the two factors, the Pearson correlation coefficient (PCC) is used for detection.

The wind speed data and corresponding correlation scatter plots and histograms for the four season are shown in Figure 3, and the correlation between the two sequences is calculated using the PCC. The correlation coefficients of wind speed and temperature in the four seasons are respectively -0.083, 0.296, 0.441, and 0.589. It is indicated that in the above three seasons, there is a significant correlation between wind speed and temperature, and the magnitude of wind speed is positively correlated with temperature.

4.2. Experimental Simulation

The prediction process of the DBNLSTM model is divided into three steps. The first step is to process the temperature and

wind speed missing data and unreasonable data in the data set and select the wind speed data according to different time windows to generate data sets with different time intervals. According to different data sets and the corresponding temperature composition of the day, the data set is moved from one hour to seven hours in advance, and the corresponding DBN is used for data feature fusion.

The second step is to select data features for seven different DBN network structures for different input dimensions. The DBN algorithm is used to select the features of the data layer by layer to reduce the dimensions of the model. The data of the visible layer is reconstructed by the mutual mapping between the visible layer and the hidden layer and forwarded to the corresponding hidden layer activation unit. Then, supervised backpropagation is used to adjust the parameters in the DBN network and output the reconstructed data. In the third step, the DBN algorithm outputs data as input data for the BLSTM network, to be trained with the information transfer of the LSTM algorithm. The trained model is used to predict the wind speed for the four quarters of 2004 in northwest China's wind farms. In this paper, four seasons of the wind speed data are analyzed, and the average prediction error of the above six models in different time windows is shown in Figure 4.

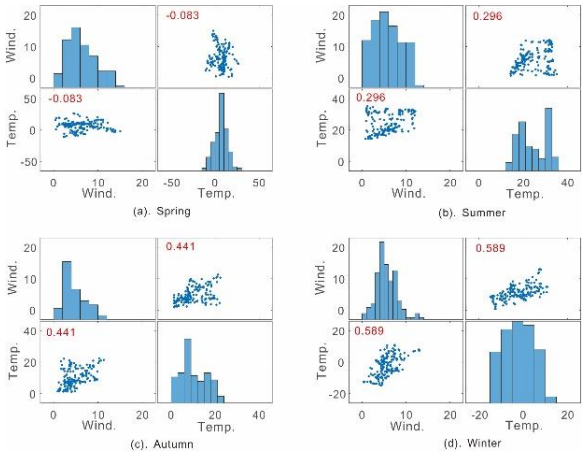

Figure 3. The correlation analysis between wind speed and temperature of wind farms

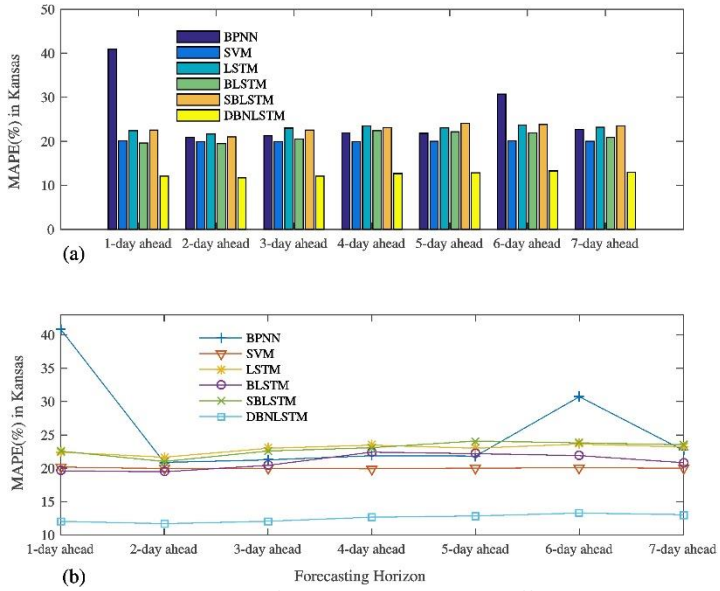

Figure 4. The average error of wind speed prediction in different time windows

In Figure 4, BPNN has the biggest performance change in the data selection method of the seven time windows. The predicted average errors of one hour ahead and six hours ahead are the largest of the six models, and they are 40.87% and 30.72%, respectively. The SVM model's performance is very stable for all four seasons, and the precision forecasting is always in the top two. This shows that the SVM model achieves good performance in the current wind speed prediction. The prediction errors of the LSTM model and the SBLSTM model are large, and the average MAPE exceeds 20%. The prediction error of the BLSTM model is smaller than that of LSTMT and SBLSTM. This shows that the BLSTM model can

better analyze historical information in the data to improve prediction performance. The DBNLSTM model has the least predictive average error of the above six models in seven time windows, which are 12.04%, 11.68%, 12.05%, 12.66%, 12.84%, 13.28%, and 13.04%, respectively. From the perspective of the time window, the prediction results based on data selection for one hour ahead, two hours ahead, and three hours ahead are better than the forecasting results based on data selection for four hours ahead, six hours ahead, and seven hours ahead. Therefore, we choose the way the data is based on two hours ahead to compare the predictive performance of six models.

4.3. Discussion of Results

The five predictive model mentioned above were compared the forecasting performance with DBNLSTM model. The data of one week was selected as the test data from the meteorological data corresponding to the four seasons in northwestern China in 2004. The six prediction models were forecasted by the two hours ahead data selection pattern, separately. The prediction results are shown in detail in Figures 5 to 8.

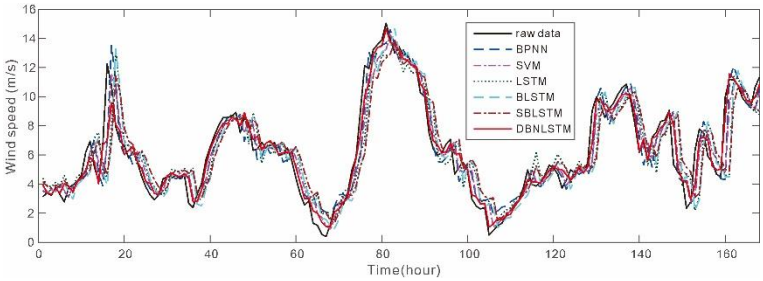

Figure 5. Prediction results of spring wind speed data of wind farms in northwest China based on two-hour data selection method

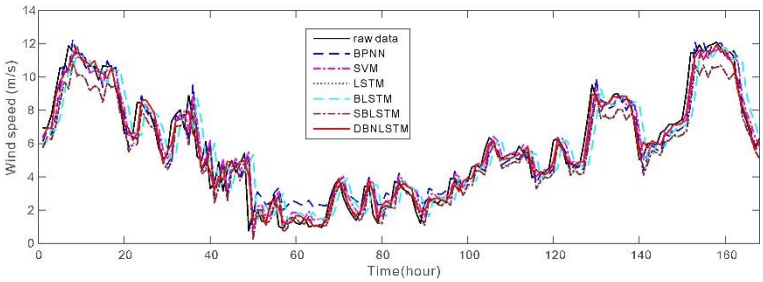

Figure 6. Prediction results of summer wind speed data of wind farms in northwest China based on two-hour data selection method

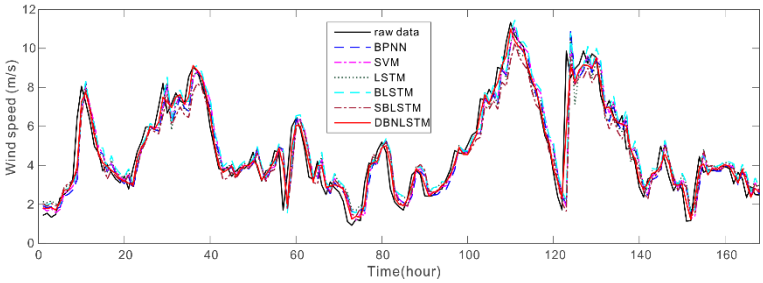

Figure 7. Prediction results of autumn wind speed data of wind farms in northwest China based on two-hour data selection method

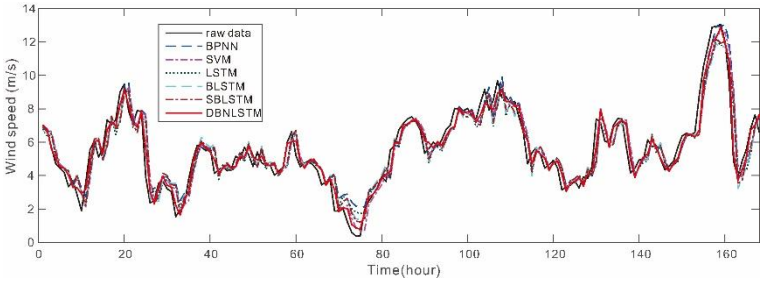

Figure 8. Prediction results of winter wind speed data of wind farms in northwest China based on two-hour data selection method

In the figures above, it is shown that the BLSTM model and the DBNLSTM model are close to the actual wind speed curve, while the predicted curves of the SVM, BPNN, LSTM, and SBLSTM models are different from the original data. At the 18-20 points in Figure 5, the six models have large prediction errors. The data of this time period represents the peak of the day. Due to the large fluctuations before and after the peak data, the prediction results show a large deviation. However, at time points 75-85, both DBNLSTM and BLSTM accurately predict the peak data. The predictive performance analysis of the six models is shown in the Table 1.

Table 1. Analysis of wind speed prediction results based on two hours ahead pattern

Season	Evaluation Index	BPNN	SVM	LSTM	BLSTM	SBLSTM	DBNLSTM
Spring	RMSE	1.170	1.809	1.180	1.130	1.160	0.758
	MAE	0.850	0.847	0.900	0.840	0.850	0.490
	MAPE	23.930	22.410	19.650	17.360	19.420	12.432
Summer	RMSE	0.920	0.660	1.190	0.890	1.100	0.590
	MAE	0.700	0.618	0.950	0.640	0.860	0.450
	MAPE	21.510	17.586	19.370	16.820	19.240	12.630
Autumn	RMSE	0.990	0.838	1.050	1.090	1.070	0.600
	MAE	0.660	0.722	0.740	0.720	0.740	0.410
	MAPE	17.060	23.738	28.120	25.180	26.960	10.920
Winter	RMSE	1.240	0.684	1.170	1.170	1.180	0.550
	MAE	0.910	0.681	0.870	0.840	0.870	0.420
	MAPE	20.990	16.030	19.450	18.510	18.400	10.720
Average	RMSE	1.080	0.998	1.148	1.070	1.128	0.625
	MAE	0.780	0.717	0.865	0.760	0.830	0.443
	MAPE	20.873	19.941	21.648	19.468	21.005	11.676

According to the prediction error during the spring from Table 1, DBNLSTM is the best predictive model, with RMSE, MAE, and MAPE values of 0.758, 0.49, and 12.43%, respectively. The BPNN method has the worst prediction performance, and its MAPE reaches 23.93%. During the summer, the BPNN method also has the largest prediction error and a large deviation from the raw wind speed curve; the RMSE, MAE, and MAPE values are 0.92, 0.70, and 21.51%, respectively. The LSTM and SBLSTM methods also have large prediction errors, with MAEP values of 19.37% and 19.24%, respectively. This indicates that the LSTM method without feature selection processing does not fit the current raw wind speed data.

From Table 1, DBNLSTM has the best performance of the six methods in wind speed prediction for the four seasons, and it is also the best in term of average performance comparison. The average values of RMSE, MAE, and MAPE are 0.625, 0.443, and 11.676%, respectively. This is sufficient data to show that the LSTM method based on DBN feature selection pattern in the distribution framework can learn more useful information and rules from the raw data. The DBNLSTM method thus improves the prediction accuracy of wind speed data.

5. Conclusions

In this work, a dynamic ensemble algorithm was proposed to obtain big data time series predictions based on the Spark framework. The distribution framework was applied to predict the short wind speed data of wind farms, and it could also be applied to the forecasting of long time series wind data. The error in wind speed prediction results of wind farms in western China was decreased by an average of about 1.3%. Similarly, experiments were conducted based on the size of the distributed computer scale, and the number of master nodes and slave nodes for the method for predicting large time series was obtained.

One suggestion for future research is to optimize the weight update algorithm. Further studies should also analyze the relationship between deep learning and parallel computing architecture and the efficiency of the algorithm. In addition, it would be interesting to study artificial neural networks based on parallel framework. Finally, the proposed method must be validated by a wider range of applications with other data sets of larger sizes.

Acknowledgements

This work has been supported by the Natural Science Fund of the Ningxia Province of China (No. NXSFZDA1801, NXSFZDA1802, and 2018AAC03242), the High School Project of Ningxia Province of China (No. NGY2018-122), the First-Class Discipline Construction in Ningxia High School (No. NXYLXK2017B11), the Project of Ningxia Province of China Key R&D Program (No. 2018BEE03025 and 2018BEE03026).

References

1. S. Singh and Y. Liu, "A Cloud Service Architecture for Analyzing Big Monitoring Data," *Tsinghua Science and Technology*, Vol. 21, No. 1, pp. 50-70, July 2016
2. M. M. Oliveira, A. S. Camanho, J. B. Walden, V. L. Migu ś, N. B. Ferreira, and M. B. Gaspar, "Forecasting Bivalve Landings with Multiple Regression and Data Mining Techniques: The Case of the Portuguese Artisanal Dredge Fleet," *Marine Policy*, Vol. 84, pp. 110-118, August 2017
3. G. Asencio-Cort és, E. Florido, A. Troncoso, and F. Mart ínez-Álvarez, "A Novel Methodology to Predict Urban Traffic Congestion with Ensemble Learning," *Soft Computing*, Vol. 20, No. 11, pp. 4205-4216, January 2016
4. J. Dean and S. Ghemawat, "Mapreduce: Simplified Data Processing on Large Clusters," *ACM Communications*, Vol. 51, No. 1, pp. 107-113, February 2008
5. V. Chang, "Towards Data Analysis for Weather Cloud Computing," *Knowledge-based Systems*, Vol. 127, No. 4, pp. 29-45, February 2017
6. S. Jeon, B. Hong, and V. Chang, "Pattern Graph Tracking-based Stock Price Prediction using Big Data," *Future Generation Computer Systems*, Vol. 80, No. 5, pp. 171-187, July 2018
7. A. M. Fern ández, J. F. Torres, A. Troncoso, and F. Mart ínez-Álvarez, "Automated Spark Clusters Deployment for Big Data with Standalone Applications Integration," *Lecture Notes in Artificial Intelligence*, Vol. 9868, No. 3, pp. 150-159, March 2016
8. A. Gounaris and J. Torres, "A Methodology for Spark Parameter Tuning," *Big Data Research*, Vol. 11, No. 5, pp. 22-32, February 2018
9. L. Mac ías-Garc á, J. M. Luna-Romera, J. Garc á-Guti érrez, M. Mart ínez-Ballesteros, J. C. Riquelme-Santos, and R. Gonz ález-Cámpora, "A Study of the Suitability of Autoencoders for Preprocessing Data in Breast Cancer Experimentation," *Journal of Biomedical Informatics*, Vol. 72, No. 4, pp. 33-44, October 2017
10. C. Ren, N. An, J. Z. Wang, L. Li, B. Hu, and D. Shang, "Optimal Parameters Selection for BP Neural Network based on Particle Swarm Optimization: A Case Study of Wind Speed Forecasting," *Knowledge-based Systems*, Vol. 56, No. 3, pp. 226-239, October 2014
11. S. S. Sancho, G. O. G. Emilio, M. P. Ángel, P. F. Antonio, and P. Luis, "Short Term Wind Speed Prediction based on Evolutionary Support Vector Regression Algorithms," *Expert Systems with Applications*, Vol. 38, No. 4, pp. 4052-4057, December 2011

Tao Ma received his Ph.D. in computer science from the School of Information Science and Engineering at Lanzhou University in 2017. He is currently an associate professor in the School of Mathematical and Computer Science at Ningxia Normal University. His research interests include data mining and algorithm optimization.

Fen Wang received her MSc. degree in computer science and technology from Shaanxi Normal University in 2004 and her M.S. degree in computer graphics theory from Ningxia University in 2010. She is currently an associate professor and senior engineer at Ningxia Normal University. Her research interests include data mining and face recognition algorithms.

Yanshan Tian received his Ph.D. in computer science from the School of Information Science and Engineering at Lanzhou University in 2018. He is currently an associate professor in the School of Mathematics and Computer Science at Ningxia Normal University. His research interests include embedded systems and parallel computing.

Yan Ma received her Ph.D. in basic physics theory from Shaanxi Normal University in 2018. She is currently an associate professor and senior engineer at Ningxia Normal University. Her research interests include ultrasonic engineering and ultrasonic cavitation.

Xu Ma received his master's degree in computer science from Xi'an University of Electronic Technology. He is currently a professor in the School of Mathematics and Computer Science at Ningxia Normal University. His research interests include cloud computing and smart computing.