Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

An Efficient Method for Wind Power Generation Forecasting by LSTM in Consideration of Overfitting Prevention

Soichiro Ookura*, Hiroyuki Mori*

**Department of Network Design, Meiji University, Nakano-ku, Tokyo 164-8525, Japan (Tel: +81-3-5343-8292; e-mail: hmori@meiji.ac.jp).*

Abstract: This paper proposes an efficient method for wind power generation forecasting by Long Short Term Memory (LSTM) of Deep Neural Network (DNN). It is one of recurrent neural networks that make use of past output of the network, but replaces hidden layers of the conventional networks with the LSTM Block with memory and three gates of input, output and forget. Artificial and Deep Neural Networks are inclined to overfit leaning data in learning process. This paper proposes a modified LSTM that considers to prevent LSTM from overfitting with two strategies. One is Dropout to exclude some nodes randomly and change network topology while the other is Weight Decay that evaluates smaller weights between neurons. The effectiveness of the proposed method is demonstrated for real data of wind power generation.

Keywords: Renewable energy systems, Wind power generation, Forecasts, Time series analysis, Deep Neural networks, LSTM, Overfitting

1. INTRODUCTION

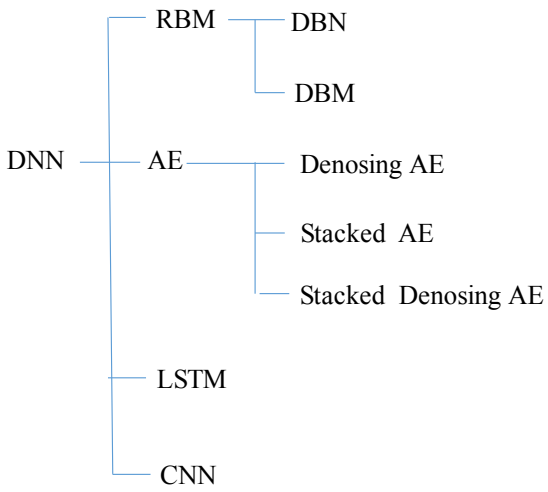
The penetration of wind power generation has been widely spread in the world. Recently, China, USA and Germany have taken the leadership to introduce wind power generation into power systems. Wind power generation has advantage that the generation is available for 24 hours if the wind blows and it has better generation efficiency of at most about 60% than other renewable energy. On the other hand, it has drawbacks that the generation output is significantly affected by weather conditions, and it generates noise. As a recent trend, offshore wind power generation photovoltaic generation is popular since it is hard to find appropriate places for wind power generation. As wind power generation has been introduced into power systems, power system operators are concerned with generation schedule due to the existence of variable energy sources with uncertainties. As a result, it is very important to forecast wind power generation output with higher accuracy. So far, a lot of the methods have been proposed to deal with wind power generation forecasting. They may be divided into two categories:

1) Statistical methods such as Autoregressive Integrated Moving Average (ARIMA) model (Box, *et al.*, 2015), Generalized Autoregressive Conditional Heteroscedastic ally (GARCH) model (Bollerslev, 1986), *etc.*

2) Machine learning such as Artificial Neural Network (ANN), Fuzzy Inference, Kernel Machines like Gaussian Process (Mori & Ohmi, 2005; Mori & Kurata, 2008), *etc.*

Item 1) is based on classical signal processing techniques and does not have the appropriate function of nonlinear approximations. On the other hand, item 2) provides better solutions than item 1) due to good nonlinear approximations. In this paper, the area of item 2) is discussed. A recurrent network model was proposed for short-term wind power generation forecasting (Kariniotakis, *et al.,* 1996). A Multi-Layer Perceptron (MLP)-based method was developed for wind power turbine output forecasting (Li, *et al.,* 2001). Mori and Okura presented a wind speed prediction method with Radial Basis Function Network (RBFN) (Mori & Okura, 2017).

In recent years, Deep Neural Network (DNN) (Hinton & Salakhutdinov, 2006; Goodfellow, *et al.*, 2016) is one of attractive research topics. Traditionally, the number of layers is limited to three due to the difficulty of learning, but the breakthroughs have brought about neural networks with four layers and more so that the model accuracy is improved. Fig. 1 shows a classification of DNN, four types of DNN are shown. Looking at applications of DNN to wind power generation or wind speed forecasting, the following works are found: A Deep Belief Network (DBN) based method was proposed to predict wind power generation output (Tao, *et al.,* 2014), where DBN (Hinton, Osindero, & Teh, 2006) was historically the first DNN model. It may be regarded as an extended model of Restricted Boltzmann Machine (RBM) with two layers in a way that RBM is modified to have three and more layers by the multistage decision. A Long Short Term Memory (LSTM) based method was developed for wind generation output forecasting (Wu, *et al.*, 2016). It is an



Note) AE: Autoencoder, CNN: Convolutional Neural Network, DBM: Deep Boltzmann Machine, DBN: Deep Belief Network, DNN: Deep Neural Network, LSTM: Long Short-Term Memory, RBM: Restricted Boltzmann Machine

Fig. 1. Classification of DNN.

extension of Recurrent Neural Network (RNN) that has a feedback loop in feedforward neural networks. It improves the performance of RNN by replacing hidden layer with a new unit called LSTM block (Hochereiter & Schmidhuber, 1998; Gers, *et al.*, 1999; Baytas, *et al.*, 2017). An Autoencoder-based method was proposed for wind speed forecasting (Mezaache & Bouzgou, 2018). The method combined Autoencoder of pre-training with Extreme Leaning Machine (ELM) of predictor, where ELM is feedforward neural network with the random weights between input and hidden layers and the analytical weights between hidden and output layers (Huang, *et al.,* 2006). Afterward, Stacked Denosing Autoencoder was proposed for wind power forecasting (Yan, *et al.*, 2018), where it is an extension of Autoencoder to increase the number of layers and make Autoencoder more robust by adding noises to learning data.

In this paper, an efficient LSTM method is proposed for wind power generation forecasting. LSTM is attractive in a way that unlike the conventional RNNs, long term memory is maintained. The difference between the proposed LSTM and the conventional one is that two strategies are presented to suppress model overfitting for learning data. One is Weight Decay (Plaut, Nowlan & Hinton, 1986) that evaluates a few parameters to suppress overfitting while the other is Dropout (Srivastava, *et al.*, 2014) that excludes some nodes in neural networks to make the model simple and prevent them from overfitting. It is expected that the proposed method is more robust for unknown data. The proposed method is successfully applied to real data of wind power generation.

2. Recurrent Neural Networks

This section describes recurrent neural networks (RNNs) with feedback loop that give information on historical data of time series. The use of ANNs has been widely spread in engineering areas due to the good approximation of nonlinear systems. The applications of ANNs to power systems may be divided into the following categories (Mori, 1996):

1) Multilayer Perceptron (MLP)
2) Hopfield Net
3) Kohonen Net
4) Others

In item 1)-4), MLP is the most popular as a pattern recognition technique. The types of MLP may be divided into two categories:

i) Feedforward MLP
ii) Feedback MLP

So far, type i) has been applied to load forecasting, static and dynamic security assessment, voltage stability assessment, fault detection, voltage and reactive power control, *etc.* On the other hand, type ii) is referred to as RNN and has been applied to problems of image processing, speech recognition, machine translation, time-series forecasting, *etc.* It has a feature to make use of information on back and forth states by adding feedback loop between layers through a new context layer. In other words, it is easy to grasp the dynamics of time series with memory. Now, let us consider the predication problem of time-series as follows:

$$y_{t+1} = f(x_t) \tag{1}$$

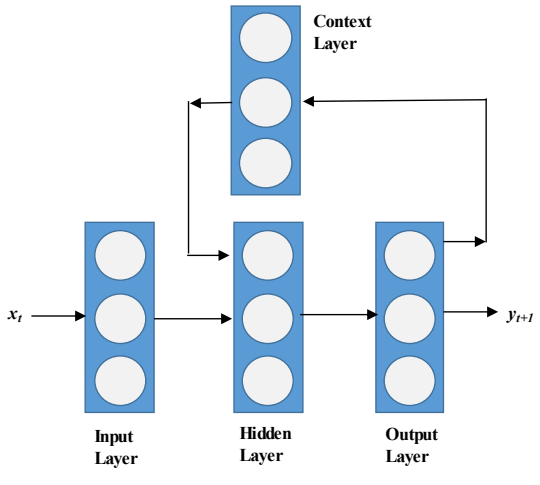
where

y_{t+1}: output variable at time $t+1$

$f(\cdot)$: nonlinear function of $\cdot$ corresponding to RNN

x_t : input variable vector at time t

There is some cases in simple RNNs where the new context layer should be placed in the network. The Jordan recurrent neural network (Jordan, 1986) (see Fig. 2) keeps the context layer between hidden and input layer while the Elman model (Elman, 1989) has the hidden layer to return the output variables to it through the context layer (see Fig. 3). in speech recognition, the Elman recurrent neural network has been widely spread due the good performance. As far as power systems are concerned, RNNs have been developed for time-series prediction such load forecasting (Mori & Ogasawara, 1993). However, RNNs have a problem called *vanishing gradient problem* that the leaning is difficult to carry out because error signals are not transmitted in the algorithm of the back rogation through time and the gradients become much smaller. For this reason, studies on RNNs have not been done for a while positively.



i

Fig. 2. Jordan recurrent neural network.

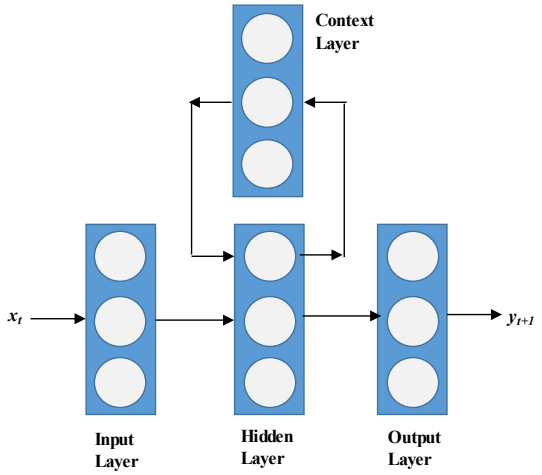


Fig. 3. Elman recurrent neural network.

3. LSTM

In this section, LSTM is explained (Hochereiter & Schmidhuber, 1998; Gers, *et al.*, 1999; Baytas, *et al.*, 2017). As mentioned in the previous section, the conventional RNNs have a drawback on the learning process that they do not deal with long term memory. To overcome it, LSTM was developed to make use of the LSTM Block with memory and three gates of input, output and forget. That allows one to deal with the dynamics of complicated nonlinear time-series appropriately. Fig. 4 shows the structure of LSTM, where three layered networks are connected through the hidden layer and the horizontal axes is shows time evolution. Compared with the conventional RNNs, LSTM has the following features:

1) To possess both long and short term memory that the conventional RNNs do not keep

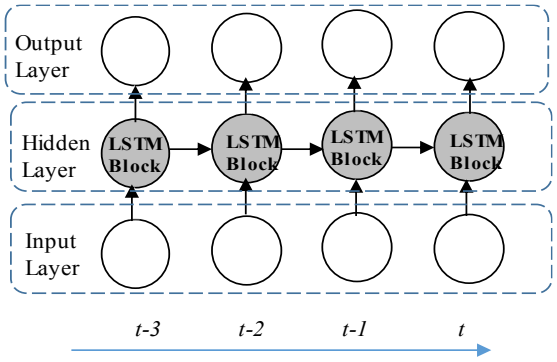


Fig. 4. Structure of LSTM.

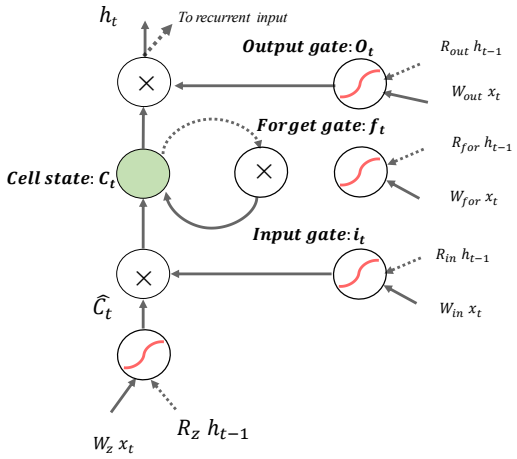


Fig. 5. LSTM Block.

2) To develop special hidden nodes called LSTM Block that consists of a cell including output, input, forget gates, *etc.*(Input and output gates play a key role to determine whether input and output of one-step back should be accepted or not, respectively. Also, forget one is needed to update the cell state in case where input pattern changes,)

3) To set the weight matrix for regressive input as the unit matrix to avoid vanishing gradient problems

The output of LSTM may be written as

$$m_{t,i,u} = m_{t-1,i,u} + z_{t-1,i,u} \tag{2}$$

where

$m_{t,i,u}$: output of *i*t memory at layer *u* at time *t*

$z_{t-1,i,u}$: input from neighbor layer at time *t*

It should be noted that the partial derivative coefficient for $m_{t-1,i,u}$ is the unity, which implies that LSTM does cause vanishing gradient problems. However, some modifications are required due to the simple equation in (2) so that LSTM Block is implemented as shown in Fig. 5.

The mathematical formulation of LSTM Block may be written as

$$LSTM\ Block\ output:\ h_t = O_t * tanh(C_t) \quad (3)$$

$$Output\ gate: O_t = \sigma(W_{out} * x_t + R_{out}\ h_{t-1} + b_{out}) \quad (4)$$

$$\text{Cell State: } C_t = f_t * C_{t-1} + i_t * \widehat{C_t} \quad (5)$$

$$where\ \widehat{C_t} = tanh(W_z * x_t + R_z\ h_{t-1} + b_z) \quad (6)$$

$$Forget\ gate: f_t = \sigma\left(W_{for} * x_t + R_{for}\ h_{t-1} b_{for}\right) \quad (7)$$

$$Input\ gate: i_t = \sigma(W_{in} * x_t + R_{in}\ h_{t-1} + b_{in}) \quad (8)$$

4. PROPOSED METHOD

In this paper, a modified LSTM is proposed for wind power generation forecasting. Time series of wind power generation has high nonlinearity so that the volatility is higher than others in power systems. If ANN-based models are employed to predict one-step ahead wind power generation output, the constructed models are inclined to overfit learning data so that they do not work so well at the implementation phase. The proposed method introduces a couple of strategies into LSTM to avoid overfitting: One is Weight Decay while the other is Dropout.

Regularization

This paragraph describes regularization that was developed to prevent artificial neural networks (ANNs) from overfitting learning data. The reasons why ANNs encounter it may be given as follows:

a) The weights between neurons are large.

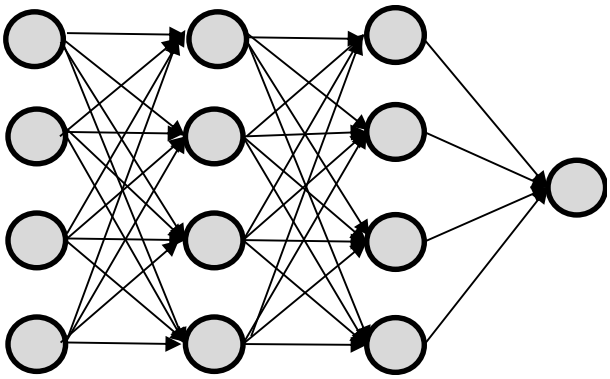
b) The number of weights is large.

c) The number of learning data is not sufficient

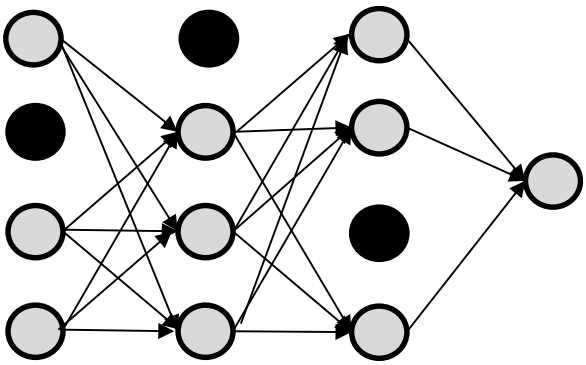
Regularization is related to item1) in a way that the weights are maintained to be small and plays an important role to keep balance between the model fitting for learning data and the model complexity. In other words, it allows one to construct the simple reasonable model like AIC (Akaike's Information Criterion) (Akaike, 1974). Regularization may be expressed as the sum of the model errors and the L_1 or L_2 regularization term that corresponds to the penalty term, where the L_1 and L_2 regularization mean the L_1 and L_2 norm of the weight vector, respectively. As one of the regularization techniques, Weight Decay plays a key role to prevent the model from overfitting (Reed, 1993). The cost function with Weight Decay may be written as

$$F = \frac{1}{2}\sum_{j=1}^{J}(y_j - t_j)^2 + \lambda \sum_k^K p_k^2 \quad (9)$$

where
F : cost function
J : number of learning data
y_j: *j-th* output
t_j : teaching signal for data y_j
λ: Penalty coefficient
K: Number of parameters
p_k: parameter k



(a) Fully-connected network without Dropout.



(b) Partially-connected network with Dropout.

Fig. 6. Structure of LSTM.

Dropout

The idea of Dropout (Srivastava, *et al*, 2014) was developed to deal with overfitting of ANNs. It prevents them from overfitting by excluding some nodes and cutting weights in a way that the degree of freedom is decreased to improve the generalization ability as shown in Fig. 6, where the grey and black circles show nodes with and without the node connections, respectively. Fig. 6(a) and (b) depict connected networks with and without Dropout, respectively. The method has the following features:

-It may be regarded as another method for regularization mentioned in the previous paragraph.

-It is different from other methods since it does not rely on features by dropping out some nodes.

- ANNs are learned by the conditions where some neuron units at input and hidden layers are randomly excluded from fully-connected ANN with the Bernoulli distribution, which means that different ANNs are constructed at each learning step and computational effort becomes large.

- The independent random noises are added to input variables at the learning so that robust ANNs are obtained under partially connected network configurations.

- The learning scheme may be considered averaging over a lot of ANNs, which may be regarded as Ensemble Learning in Machine Learning (Zhou, 2012).

5. SIMULATION

5.1 Simulation Conditions

a) This paper dealt with 10-minute ahead The proposed method was tested for real data of wind firm, Calgary, Canada. The number of learning and test data are given as follows:

No. of learning data: 1008, No. of test data: 432

The sampling time was 10 [min]. The proposed model used the following input and output variables:

Input variables:

x_1^t –x_t-k^t: wind power generation output at time *t-k (k=1,2, ...20)*

y_1^{t+1}: 10minute ahead wind power generation output at time *t+1*

b) The proposed method was compared with the conventional methods. For convineience, the following methods were defined

Method A: MLP

Method B: RNN (Elman Model)

Method C: LSTM

Method D: LSTM +Dropout

Method E: LSTM +Weight Decay

Method F: LSTM + Dropout+ Weight Decay

c) Table 1 shows parameters of Methods A-F that were tuned up by preminarily simulation.

5.2 Simulation Results

Table 2 gives the average, maximum forecasting errors and standard deviation (SD) of errors. It should be noted that values in parentheses show the errors normalized by the error of MLP. Figs. 7-9 give the average, maximum errors, and SD, respectively. A comparison between Methods A and B shows that RNN is better than MLP. Looking over Method C, it can be seen that LSTM provides better results than RNN. Compared with D, Method E outperformed it because Dropout has better performance than Weight Decay as the overfitting prevention method. Method F of the proposed method gave better results in the average, maximum forecasting errors and the standard deviation (SD), respectively. Method F reduced the average, maximum forecasting errors and the standard deviation of method by 56%, 64%, and 56%, respectively. It is concluded that the proposed has better results than others.

Table 1. Parameters of each method

Methods	Learning rate	# of hidden units	Dropout rate	Penalty parameter λ	# of iterations
A	0.9	30			20000
B	0.01	30			10000
C	0.01	30			10000
D	0.01	50	0.1	0.01	15000
E	0.01	50		0.01	15000
F	0.01	50	0.1	0.01	15000

Table 2. Forecasting errors of each method

Methods	Ave.[%]	Max.[%]	SD
A	5.78 （1）	54.24 （1）	6.03 （1）
B	3.52 （0.6）	28.7 （0.56）	3.61 （0.59）
C	3.28 （0.56）	24.2 （0.47）	3.34 （0.55）
D	2.76 （0.47）	21.3 （0.41）	2.9 （0.48）
E	2.85 （0.49）	21.69 （0.42）	3.04 （0.50）
F	2.55 （.44）	18.87 （0.36）	2.71 （0.44）

Note) Values in parentheses indicate data normalized by that of MLP.

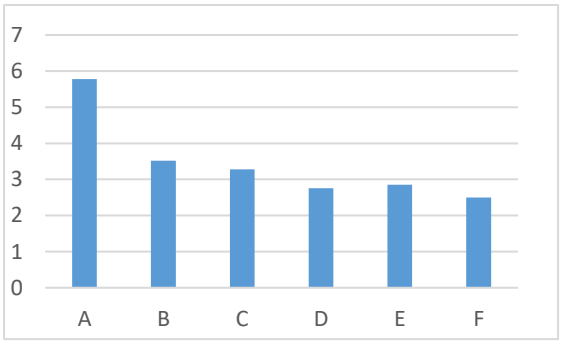


Fig. 7. Average errors of each method.

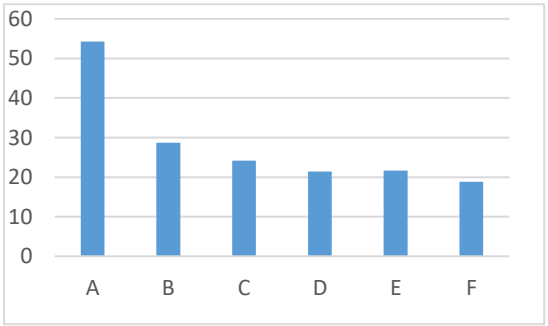


Fig. 8. Maximum errors of each method.

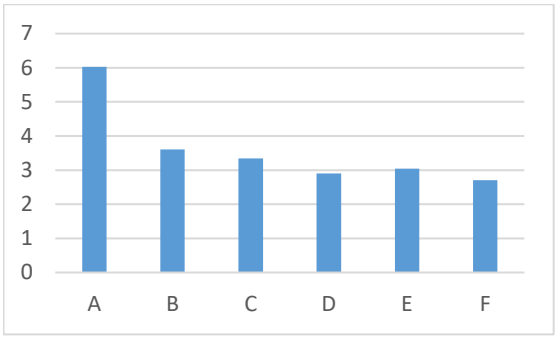


Fig. 9. Standard deviation of errors of each method.

6. CONCLUSIONS

In this paper, a new method has been proposed for wind power generation forecasting. The proposed method makes of Long Short Term Memory (LSTM) of Deep Neural Network (DNN) has better performance for time-series forecasting. To improve the performance of LSTM, this paper presented a couple of strategies to improve the performance of LSTM in terms of preventing the forecasting model from overfitting. One is Weight Decay for preventing the model from overfitting for unknown data by evaluating smaller parameters. The other is Dropout for improving the forecasting model by excluding some neurons at input and hidden layers at the learning process randomly. The effectiveness of the proposed method was tested for real data of the wind firm. The simulation results have shown that the proposed LSTM with two strategies provides better results than other LSTM in terms of the average, maximum errors as well as the standard deviation of the forecasting errors.

REFERENCES

Akaike, H. (1974) : A New Look at theStatistical Model Identification. IEEE Trans. on Automatic Control, Vol. AC-19, pp. 716-723.

Baytas, I. M., Xio, C., Zhang, X., Wang, F., Jain, A.K., J. and Zhou, J. (2017): Patient Subtyping via Time-Aware LSTM Networks. Proc. of ACM KDD'17, pp. 65-74.

Bollerslev, T. (1986): Generalized Autoregressive Conditional Heteroskedasticity. Journal of Econometrics, Vol. 31, pp. 307-327.

Box, G.E.P Jenkins. G.M., Reinsel, G.C. and Lijung, G. M. (2015): Time Series Analysis : Forecasting and Control (5th Edition). Wiely, New York, USA.

Elman, J.L. (1989) : Finding Structure in Time. CRL technical Report 8901, Institute of Cognitive Science, US San Diego

Gers, F., Schimidhuber, J., and Cummins, F.(1999) :Learning to Forget : Continual Prediction with LSTM. Proc. ICANN'99, pp. 850-855.

Goodfellow, I., Bengio, Y. and Courville, A. (2016): Deep Learning. MIT Press, Cambrige, MA, USA.

Kariniotakis, G. N., Stavrakakis, G. S. and Nogaret, E. F. (1996): Wind Power Forecasting Using Advanced Neural Networks Models. *IEEE Trans. on Energy Conversion*, Vol. 11, No. 4, pp. 762-767.

Hochereiter, S. and Schmidhuber, J. (1997): Long Short-Term Memory. Neural Computation, Vol. 9, No. 8, pp. 1735-1780.

Hinton, G.E. and Osindero, S., and Teh, Y.W. (2006): A Fast Learning Algorithm for Deep Belief Net. Neural Computation, Vol. 18, No. 7, pp. 1527-1554.

Hinton, G.E. and Salakhutdinov, S.S. (2006): Reducing the Dimensionality of Data with Neural Networks. Science, Vol. 313, No. 5786, pp. 504-507.

Huang, G.-B., Zhu, Q.Y., and Siew, C.K. (2006): Extreme Learning Machine: Theory and Applications. Neurocomputing, Vol. 70, No 1, pp. 489- 501.

Jordan, M.I. (1986) : Serial Order a Parallel Distributed Processing Approach. CRL technical Report 8604, Institute of Cognitive Science, UC San Diego

Li, S., Wunsch, D. C., O'Hair, E. A. and Giesselmann, M. G. (2001): Using Neural Networks to Estimate Wind Turbine Power Generation. *IEEE Trans. on Energy Conversion*, Vol. 16, No. 3, pp. 276-282.

Mezaache, H. and Bouzgou, H. (2018): Auto-Encoder with Neural Networks for Wind Speed Forecasting. Proc. of IEEE ICCEE, 5 pages.

Mori, H. and Ogasawara, T. (1993): A Recurrent Neural Network for Short-Term Load Forecasting. Proc. of IEEE ANNPS'93, pp. 395-340, Yokohama, Japan.

Mori, H. (1996): State-of-the-Art Overview on Artificial Neural Networks in Power Systems, in M.A. El-Sharkawi and D. Nibuer (Eds.), A Tutorial Course on Artificial Neural Networks with Applications to Power Systems. pp. 51-70, IEEE Catalog No. 96 TP 112-0.

Mori, H. and Ohmi, M. (2005): Probabilistic Short-term Load Forecasting with Gaussian Processes. Proc. of IEEE ISAP2005, pp. 452-457, Washington, D.C., USA.

Mori, H. and Kurata, E. (2008): Application of Gaussian Process to Wind Speed Forecasting for Wind Power Generation. Proc. of IEEE ICSET2008, 4 pages, Singapore.

Mori, H. and Okura, S. (2017): Application of S-Transform-Based Artificial Neural Network to Wind Speed Forecasting. Proc. of IEEE PowerTech2017, 6 pages, Manchester, UK.

Plaut, D.C., Nowlan, S.J., and Hinton, G.E. (1986): Experiments on Learning by Backpropagation. Proc. of Tech. Rep. CMU-CS-86-126, Carnegie Melon University, Pittsburgh, PA, USA.

Reed, R. (1993): Pruning Algorithms-A Survey. *IEEE Trans. on Neural Networks*, Vol. 46, No. 5, pp. 740-747.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskerver, and Salakhutdinov, R. (2014): Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research, Vol. 15, pp. 1928-1958.

Tao, Y., Chen H., and Qiu, C. (2014): Wind Power Prediction and Pattern Future Based on Deep Learning Method. Proc. of IEEE APPEEC2014, 4 pages.

Wu, W., Chen, K., Qio, Y., and Lu, Z. (2016): Probabilistic Short-term Wind Power Forecasting. Proc. of IEEE PMAPS2016, 8 pages.

Yan, J., Zhang, H. Liu,Y. Han, S., Li, L. and Lu, Z. (2018): Forecasting the High Penetration of Wind Power on Multiple Scales Using Multi-to-Multi Mapping. *IEEE Trans. on Power Systems*, Vol. 33, No. 3, pp. 3276-3284.

Zhou, Z.-H. (2012): Ensemble Methods: Foundations and Algorithms. Chapman & Hall/CRC, Boca Raton, FL, USA.