

Applications of Recurrent Neural Networks in Environmental Factor Forecasting: A Review

Yingyi Chen

chenyingyi@cau.edu.cn

Qianqian Cheng

2367753738@qq.com

Yanjun Cheng

chengyanjun172@163.com

Hao Yang

804727068@qq.com

Huihui Yu

774754053@qq.com

College of Information and Electrical Engineering, China Agricultural University, Beijing 10083, China; Key Laboratory of Agricultural Information Acquisition Technology, Ministry of Agriculture Beijing 100125, China; and Beijing Engineering and Technology Research Centre for Internet of Things in Agriculture, Beijing 100083, China

Analysis and forecasting of sequential data, key problems in various domains of engineering and science, have attracted the attention of many researchers from different communities. When predicting the future probability of events using time series, recurrent neural networks (RNNs) are an effective tool that have the learning ability of feedforward neural networks and expand their expression ability using dynamic equations. Moreover, RNNs are able to model several computational structures. Researchers have developed various RNNs with different architectures and topologies. To summarize the work of RNNs in forecasting and provide guidelines for modeling and novel applications in future studies, this review focuses on applications of RNNs for time series forecasting in environmental factor forecasting. We present the structure, processing flow, and advantages of RNNs and analyze the applications of various RNNs in time series forecasting. In addition, we discuss limitations and challenges of applications based on RNNs and future research directions. Finally, we summarize applications of RNNs in forecasting.

1 Introduction ---

Along with the rapid development of information technology, a variety of information systems are widely used in people's daily life. These systems produce massive amounts of noisy time series data, which are unstable and

fluctuate (Singh, Basant, Malik, & Jain, 2009). In addition, the relationship between variables in these data is complex and nonlinear. Efficiently using these time series data to mine useful information is a hot topic in data processing. To the best of our knowledge, time series forecasting plays a critical role in many engineering and scientific applications (Zhang, 2003). Generally better forecasting is the key factor for providing better decision making and monitoring management. Furthermore, time series forecasting has attracted the attention of researchers from various fields. Recently, the literature on forecasting research has focused on the development of information technology and artificial intelligence (Sadaei, Guimaraes, Silva, Lee, & Eslami, 2017). Therefore, a variety of models and methodologies have been provided for sequential data prediction (Raza & Khosravi, 2015; Tasikaraoglu & Uzunoglu, 2014).

The methodologies for time series forecasting almost entirely rely on two different methods: traditional methods and artificial intelligence algorithms (Bontempi, Taieb, & Borgne, 2013). Traditional models, such as multiple sources linear regression (Mahmoud, 2008) and Fourier expansion methods (Sanz-Serna, 2009), are simple and easy to achieve. Most of these methods are based on mathematical theories. However, these methods have poor adaptability and unpredictable performance with increasing model complexity. In recent decades, artificial intelligence algorithms, including artificial neural networks (ANNs; Zhang, Patuwo, & Hu, 1998), support vector machines (SVM; Tay & Cao, 2007), and recurrent neural networks (RNNs; Garcia-Pedrero & Gomez-Gil, 2010), have attracted attention and have been successfully used for time series forecasting. Though these models have good learning ability and recognize the complexity and nonlinearity in the patterns of data sets, they have some shortcomings. The ANN can easily fall into a local optimum solution and has a complex learning process. The SVM requires a large storage space and longer training time when handling large amounts of data. These methods still need to be improved to achieve better forecasting accuracy. Fortunately, RNNs have proven to be suitable for time series forecasting due to their ability to capture sequence data relations in time, in contrast to feedforward neural networks (FFNNs; Bebis & Georgiopoulos, 2009). The RNN (Mandic & Chambers, 2001) is also a class of ANN in which connections between units form a directed cycle. This cycle establishes an internal state of the network that allows it to show dynamic temporal behavior. Due to advances in their architecture and training methods, various RNNs with different architectures and topologies have been widely and successfully applied for time series forecasting in various domains, such as electric power, environmental factors, finance, and economics (Motlagh & Khaloozadeh, 2016; Alzahrani, Shamsi, Dagli, & Ferdowsi, 2017; Zheng, Yuan, & Chen, 2017).

RNNs are considered to be an extremely promising category of methods for time series prediction and may be able to be used to compensate for the shortcomings of traditional forecasting models. RNNs were

developed from the Hopfield network in 1982 (Hopfield, 1982). Early RNNs were powerful dynamic systems that were able to correct errors through backpropagation and use of a gradient algorithm. However, RNNs were difficult to train and suffered from the vanishing or exploding gradient problem (Le & Zuidema, 2016), which cannot solve the long-term dependency problem. Therefore, this method was not widely used until the 1990s. To solve this problem, a major breakthrough was made by introducing a vastly improved LSTM-based architecture that used a gate mechanism to prevent backpropagated errors from vanishing or exploding to remember inputs for a long period of time. LSTM networks (Gers, Schmidhuber, & Cummins, 2000) were subsequently proven to be more effective and accurate than conventional RNNs and helped lead to the renaissance in AI.

As one of the most promising types of time series prediction models, RNNs have been widely studied in theory and applied in many fields. There is an extensive literature base regarding different applications of RNNs in time series forecasting, including electric power forecasting, environmental factor forecasting, and finance and economics forecasting. However, a single summary of applications on RNNs in time series forecasting has not been published. In addition, many novel fields need to be explored for applications of RNNs, and further research is required to improve the performance of models based on RNNs in dynamic real-time systems; these are important and challenging problems. In recent years, with the emergence of big data and deep learning, forecasting large-scale system has become feasible because of the abundant data and hierarchical representations in deep architectures. Therefore, novel deep learning models based on RNNs are becoming increasingly popular.

This review summarizes applications of RNNs in forecasting, mainly considering environmental factor forecasting. The structure, processing flow, and advantages of RNNs are introduced. In addition, the limitations and challenges of the current state of RNNs are discussed. Simultaneously, future research directions of RNNs and new areas are discussed. We believe that this review can be used as a guide for researchers regarding the applications of RNNs in data forecasting. The review is structured as follows. Section 2 explains the structure, processing flow, and advantages of RNNs in detail. In section 3, applications for time series forecasting based on RNNs are summarized and evaluated, including limitations and challenges. The research direction is explored in section 4, and section 5 summarizes the key conclusions.

2 Recurrent Neural Networks

This section focuses on the theory of RNNs and introduces their structure, processing flow, and advantages.

2.1 Structure of RNNs. The RNN is a class of ANN that allows nodes to be connected to directed loops; the RNN can be easily confused with the recursive neural network (Pollack, 1990). The formulation of the RNN ensures that it can show dynamic temporal behavior. The RNN can generate memory states of past data, process sequential data, and establish dependencies between data from different times. While FFNNs can be used only to establish mapping relations between data, they cannot be used to analyze the time dependence of past signals. In theory, RNNs can handle arbitrary input sequences, and weight sharing is adopted in a recursive manner. Not only can RNNs learn long-range temporal dependencies, but they can also efficiently simulate a universal Turing machine, which can perform almost any computation. In general, RNNs provide flexible machine learning tools that have the learning ability of FFNNs and can expand their expression ability depending on dynamic equations. Hence, RNNs can be used for tasks such as image processing, speech recognition, or time series prediction (Lee, Tseng, Wen, & Tsao, 2017; Sak, Senior, & Beaufays, 2014; Verdejo, Herreros, Luna, Ortuzar, & Ayuso, 1991).

RNNs were introduced in the 1980s (Rumelhart, Hinton, & Williams, 1988), and their use includes Hopfield networks, Elman networks, and Jordan networks (Chen, 2001; Cao, 2001; Kalinli & Sagiroglu, 2006; Turk, Barisci, Ciftci, & Ekmekci, 2015). Hopfield networks were developed by John Hopfield (1982), and all of the connections in RNNs are symmetrical. Moreover, the RNN can address temporal dependencies. Subsequently, in 1990, the Elman network was first proposed (Elman, 1990) for language processing, which has incomparable advantages for dealing with inertial input and output data. Therefore, the Elman network has been universally used for system modeling, time series prediction, and adaptive control. Jordan networks and Elman networks have similarities (Song, 2011). Their context units are fed from the output layer rather than the hidden layer, the context units in Jordan networks appear as the state layers, and their output can be passed directly to the hidden node. These networks are also called simple recurrent networks (SRNs; Cruse, 1996). The nonlinear autoregressive neural network with exogenous input (NARX; Chen, Billings, & Grant, 1990) is a mature dynamic forecasting model that uses a recurrent neural architecture. The NARX has limited feedback architectures that come only from output neurons instead of hidden neurons. It has been verified that this type of learning architecture with hidden states can produce more effective results in the NARX model than in other recurrent architectures. In recent years, these models have led to great achievements in natural language processing and sequence labeling (Collobert, Weston, Karlen, Kavukcuoglu, & Kuksa, 2011; Mikolov, Karafiát, Burget, Cernocký, & Khudanpur, 2010; Yao, Zweig, & Hwang, 2013).

Apart from the above models, there are also RNN variants (Dinarelli & Tellier, 2016), such as the bidirectional recurrent neural network (BRNN; Schuster & Paliwal, 1997). The state of SRNs in the t moment is only

related to the state of the past, while the state of the BRNNs at the t moment is related not only to the state of past but also to the state of future. The long short-term memory network (LSTM) is an improved RNN variant that Hochreiter and Schmidhuber (1997) proposed. The LSTM is characterized by its basic unit, which has a memory cell that can store a state of time and is protected while storing, writing, and reading information. The LSTM is also a deep learning system that efficiently avoids the vanishing gradient or explosion problem (Gers & Schraudolph, 2003). With the rapid development of deep learning, the LSTM network has played an increasingly important role because of the convergence of the learning process (Fernández, Graves, & Schmidhuber, 2007). Therefore, it has gradually replaced the classic RNN.

The structure of the recurrent neural network is closely related to the performance of the network (Awano et al., 2011). On the one hand, the internal dynamics of large-scale RNNs is complicated and confusing. However, the internal dynamics will greatly increase the storage of the network and calculation cost of the training algorithm. On the other hand, the dynamic characteristics of small-scale RNNs are relatively singular. The information contained in complex problems does not contain a learning ability and cannot meet the requirements of RNNs for processing information. The performance of RNNs is determined by their structure and training algorithm. Therefore, the activity and number of hidden neurons in RNNs are adjusted according to the object of study, which changes their topology. Improving the performance of RNNs has become a hot topic of recent research (Gil, Cardoso, & Palma, 2009).

2.2 Processing Flow of RNNs. A simple adaptation on the standard FFNNs enables RNNs to simulate continuous data. A multilayer neural network can only map input vectors and output vectors (Riedmiller, 1994), but RNNs can theoretically map an entire historical data set. At each time point, a node can accept an input, update the state of the hidden layer, and predict a result, as shown in Figure 1.

The forward propagation of RNNs is similar to the perceptron model that has only one hidden layer. The difference is that the hidden layer in the RNN not only receives an input from the outside but also accepts the value calculated from the activation function at the last moment. The output vector of the multilayer neural network is given by the activation function of the output layer. The input value of each output layer unit is the output value of all hidden layers connected to the unit. The number of output layer units and selection of the activation function mainly rely on application scenarios of neural networks. It should be emphasized that sigmoid, hyperbolic tangents, and the ReLU are widely used as activation functions in RNNs. Next, it is necessary to consider how to select the appropriate model parameters. In general, RNNs are trained with a common gradient-based

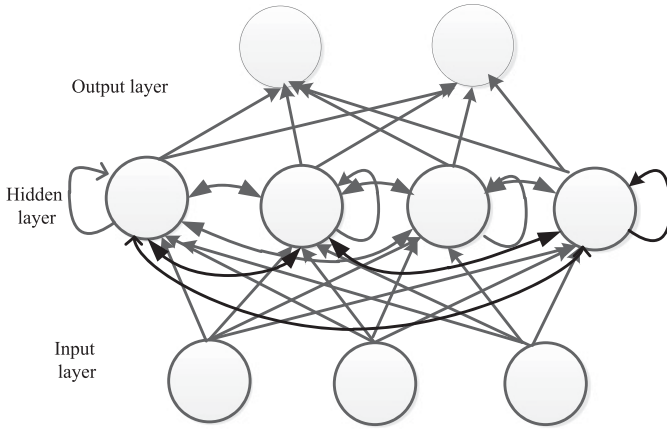


Figure 1: Simple recurrent neural network.

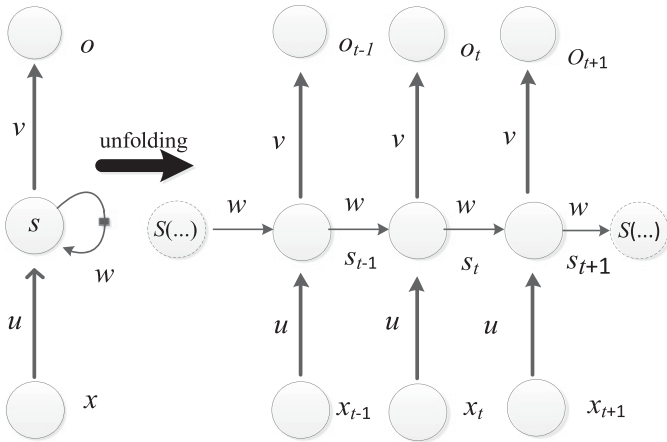


Figure 2: The RNN unfolds over time for the computation.

algorithm, such as real-time recurrent learning (RTRL) or backpropagation through time (BPTT), to derive the RNN parameters.

The RNN is shown in Figure 2 to unfold over time. The RNN can be considered to be a deep feedforward neural network with the same weights among all of its layers (LeCun, Bengio, & Hinton, 2015). RNNs include input units, output units, and hidden units, the vector sequences of which are marked as (x_1, \dots, x_T) , (o_1, \dots, o_T) , and (s_1, \dots, s_T) , respectively. The output and hidden states can be expressed by the following equations:

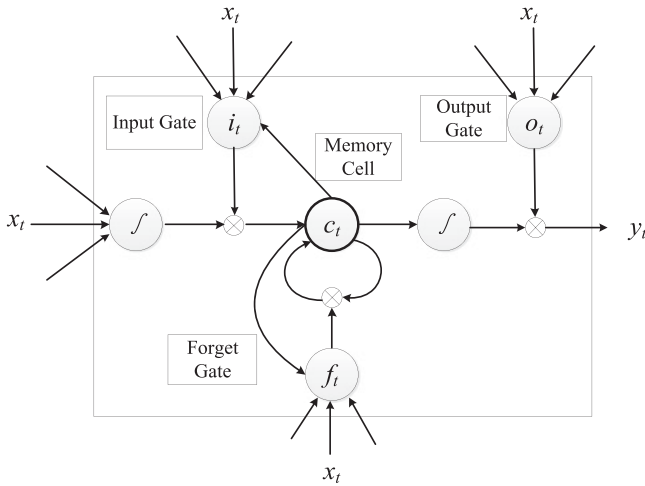


Figure 3: Structure of the LSTM cell.

$$s_t = f(w_{sx}x_t + w_{ss}s_{t-1} + b_s), \quad (2.1)$$

$$o_t = w_{os}s_t + b_o. \quad (2.2)$$

In these equations, w_{sx} is the input-to-hidden weight vector, w_{ss} is the hidden-to-hidden weight vector, w_{os} is the output-to-hidden weight vector, b_o and b_s are the biases, and u , v , and w are the network parameters. The same parameters are used at each time step, and f is an activation function such as tanh.

RNNs can easily process sequence data. However, traditional RNNs suffer from vanishing or exploding gradients when the number of time steps is large, and they fail to capture the long-term evolution. LSTMs effectively solve these problems by introducing a gate mechanism to prevent backpropagated errors from vanishing or exploding and were subsequently shown to be more effective than traditional RNNs. Figure 3 represents the structure of an LSTM cell that is mainly configured by three multiplicative gates and one memory cell. The data flow into the block is controlled by the input gate, and the decision of when to reset the memory cell state is made by the forget gate. In addition, the output data flow from the block is controlled by the output gate. The last memory cell is used to store the state of the block. The entire computation can be expressed as a series of equations as follows:

$$i_t = \sigma(w_{ix}x_t + w_{ih}h_{t-1} + w_{ic}c_{t-1} + b_i), \quad (2.3)$$

$$f_t = \sigma(w_{fx}x_t + w_{fh}h_{t-1} + w_{fc}c_{t-1} + b_f), \quad (2.4)$$

$$o_t = \sigma(w_{ox}x_t + w_{oh}h_{t-1} + w_{oc}c_{t-1} + b_o), \quad (2.5)$$

$$c_t = f_t c_{t-1} + i_t \tanh(w_{cx}x_t + w_{ch}h_{t-1} + b_o), \quad (2.6)$$

$$h_t = o_t \tanh(c_t), \quad (2.7)$$

where i_t , f_t and o_t are the calculation methods of the input gate, forget gate and output gate, respectively, at time t : c_t is the calculation method of the memory cell at time t . h_t is all of the output of the LSTM cell at time t ; σ and \tanh are the sigmoid and hyperbolic tangent activation functions, respectively; w are the weight matrices; and b are the bias vectors. The LSTM memory units easily catch the complex correlation features within the time series in both the short and long terms, a significant improvement compared with traditional RNNs via the function of the different gates.

2.3 Advantages of Recurrent Neural Networks. RNNs provide flexible machine learning tools that not only have the learning abilities of FFNNs but also expand their expression abilities based on dynamic equations. Therefore, RNNs can directly handle complex spatiotemporal data and build complicated dynamic systems. Due to temporal and spatial data being used in many fields, such as modeling electric power, finance and economics, and processing environment time series, RNNs are promising candidates for a variety of applications (Wu, Wang, Jiang, Ye, & Xue, 2015; Giles, Kuhn, & Williams, 1994). In addition, RNNs have undergone significant performance improvements in time series forecasting.

Traditional neural networks applied in time series prediction easily sink into a local optimum and have a complex learning process, resulting in a relatively slow computation speed. RNNs are tools that can obtain high precision and good performance when processing time series predictions based on a large number of data sets. The most fundamental characteristic of RNNs is their short-term memory. The understanding of short-term memory directly affects the design of the RNN structure and indirectly influences the weights of the training methods. RNNs require more virtual connections and much more memory for simulations than the conventional BPNN. RNNs achieve a better effect due to the rough repetition of similar patterns present in sequence data. These regular but subtle time series are important to make useful predictions.

RNNs are known to be local feedback networks, in which only local connections are active. The network's generalization capability is remarkably enhanced by not learning complex and fully connected recurrent architectures. In addition, redundant connections are eliminated (Tsoi, 1998). It is also important for different configurations of RNNs to choose appropriate learning parameters, an appropriate number of hidden nodes, and appropriate activation functions. Compared with the traditional multilayer perceptron, RNNs have a feedback connection and memory storage. They can also process sequence data at each time step while accepting the input,

updating the hidden state, and predicting the next value. Though RNNs have the computational capability to process sequential data, learning long-range temporal dependencies is difficult when a decreasing stochastic gradient is used for training. To solve this problem, LSTMs were proposed to introduce memory units to decide whether to forget and update hidden states. As a result, LSTMs have been proven to be more effective than traditional RNNs (Gers & Schmidhuber, 2001).

RNNs can directly and vividly reflect the dynamic characteristics of the system as well as represent the direction of modeling and identify neural networks. They also contain feedback information in the internal state of the networks. The nonlinear dynamic behavior of the system is described using network internal feedback. Therefore, the prediction performance of RNNs is better than that of feedforward networks when dealing with time series data. Theoretically, RNNs can handle any length of sequence data (Bodén, 2002). In practice, however, to reduce computational complexity, it is assumed that the recurrent state is only related to the previous states. Similar to the standard forward propagation, the BPTT contains important chain rules. The difference of RNNs is that the activation of the hidden layer depends on the loss function and affects the output layer and hidden layer at the next moment. In summary, RNNs have strong computing power and are the most widely used neural network models.

3 Environmental Factor Forecasting

The environment encompasses many aspects, such as the atmosphere, water, soil, and weather. According to the current situation and developmental trend, the quality of the environment is predicted scientifically, which contributes to human health and the normal production and life. As an important subject, environmental forecasting models have been given increasing attention by decision makers of environmental planning. We mainly introduce environment factor forecasting based on RNNs in this section. In addition, Table 1 shows the application of RNNs in environmental factor forecasting and describes the prediction objects, data source, current methods, and results.

3.1 Water Factor Forecasting. Accurate prediction of water, including water level predictions, runoff predictions, and flood forecasting, is necessary and plays an important role in the regulation and protection of water quality. To date, ANNs, especially RNNs, have attracted attention and have been accepted as powerful tools for water factor forecasting. Many researchers have made great progress in forecasting the water factor.

Accurate forecasting of runoff, including rivers and stormwater, is important in water resource planning and management. Over the years, considerable research has been carried out in this area, and numerous

Table 1: Summary of RNNs Applied to Environment Factor Forecasting.

Prediction Object	Source of Data	Methods	Results	Reference
Urban stormwater runoff prediction	U.S. Geological Survey's national water information system	RNN with the Levenberg-Marquardt backpropagation training algorithm	Levenberg-Marquardt backpropagation training algorithm proved to be successful in training the RNN for stormwater runoff prediction.	Zhang (2011)
Runoff forecasting	Rainfall-runoff records for the Dikrong catchment	Geomorphology-based time-lagged recurrent neural networks	The proposed model can be a reliable and effective prediction tool in runoff forecasting.	Saharia and Bhattacharya (2012)
Runoff prediction	The Climate data and daily river runoff data collected over 22 years	Recurrent fuzzy neural network (RFNN)	The relative error of RFNN is approximately 0.35, and the relative error of SWAT is 0.44, indicating that the RFNN outperforms SWAT.	Duong et al. (2014)
Hourly water flow rate forecasting	Experimental data were collected from a photovoltaic water pumping system installed at the Madinah site	Nonlinear autoregressive with exogenous input-recurrent neural network (NARX RNN)	Results showed that the developed NARX-based model is able to reach acceptable accuracy for predictions 1 to 12 hours (next-day) ahead	Haddad et al. (2016)
Environmental monitoring data prediction	in Japan using openly available sensor data.	A deep RNN with a new pretraining method, DynAE	Improves the accuracy of PM _{2.5} concentration-level predictions reported in Japan.	Ong et al. (2014)
Weather forecasting	ESNO data set provided by an international institution and weather data set in the Aceh area from 1973 to 2009, provided by BMKG	RNN, conditional Boltzmann machine (CRBM), and convolutional network (CN) models	The result is that RNN can be applied in predicting rainfall with an adequate accuracy level.	Salman et al. (2015)

Table 1: Continued.

Prediction Object	Source of Data	Methods	Results	Reference
PM _{2.5} air pollutant forecasting	The data used contain hourly PM _{2.5} concentrations from an urban traffic air quality monitoring station.	Feedforward neural networks and recurrent neural networks	The criteria used to choose the best configuration were the smallest RMSE and the biggest values of IA, R ₂ , and R	Oprea et al. (2016)
Traffic speed prediction	Travel speed data from traffic microwave detectors in Beijing	Long short-term memory neural network (LSTM)	LSTM NN can achieve the best prediction performance in terms of accuracy and stability.	Ma et al. (2015)
Traffic flow prediction	The data used are collected from the PeMS data set, which has over 15,000 sensors deployed statewide in California.	LSTM and gated recurrent units (GRU) neural network	The RNN-based deep learning methods such LSTM and GRU perform better than the ARIMA model.	Fu et al. (2016)
Traffic congestion prediction	The traffic condition data were collected every 5 minutes covering 1649 segments of arterial roads in Beijing, China	LSTM	Experimental results show that the proposed model for predicting traffic has superior performance over the multilayer perceptron model, decision tree model, and support vector machine model.	Chen et al. (2016)
Short-term traffic forecast	The traffic data are collected from over 500 observation stations with a frequency of 5 minutes, which are mostly deployed within the fifth ring road of Beijing.	LSTM network is composed of many memory units	A comparison with other representative forecast models validates that the proposed LSTM network can achieve a better performance.	Zhao et al. (2017)
Traffic prediction	Data were collected for 92 days; the traffic network is located between the second ring road and third ring road in Beijing.	Spatiotemporal recurrent convolutional networks	SRCNs outperform other deep learning-based algorithms in both short-term and long-term traffic prediction.	Yu et al. (2017)

runoff forecast models based on RNNs have been proposed. Zhang (2011) developed a model based on RNNs with the Levenberg-Marquardt back-propagation training algorithm to predict stormwater runoff. The experimental results indicated that the improved model was successful for making stormwater runoff predictions in which the best number of hidden neurons and delays in the tapped delay lines were 50 and 11, respectively. Saharia and Bhattacharjya (2012) presented the distributed time-lagged recurrent neural network (TRLNN)-based runoff prediction model, which had the advantages of a dynamic neural network, integration of morphometric properties, and adaptation of a semidistributed modeling approach. The TRLNN was used in conjunction with geomorphologic information to achieve better forecasting results, and the TRLNN was shown to be a significant improvement over all of the other models in this review. Duong, Nguyen, Bui, Nguyen, and Snasel (2014) employed the RFNN, a hybrid of the RNN and fuzzy theory, to predict the Srepok runoff in Vietnam with changes in climate. The RFNN can make accurate predictions in comparison to an environmental model called SWAT on the same data set because the relative error of the RFNN is low. Additionally, the RFNN does not require as many data as the SWAT. Subsequently, the NARX-RNN was suggested for predicting the rate of water flow (WFT); the NARX-RNN is based on the most relevant parameters, solar irradiance and air temperature. The results indicated that the presented model had an acceptable accuracy for next-day forecasting, which provided valuable information to the photovoltaic water pumping system (Haddad, Mellit, Benghanem, & Daffallah, 2016). In the same year, Shoaib, Shamseldin, Melville, and Khan (2016) explored the potential of wavelet-coupled time-lagged recurrent neural network (TLRNN) models to accurately predict runoff. Wavelet-coupled TLRNN models with large depths were proven to be insensitive to the selection of the wavelet function because all wavelet functions have similar performance, while the db8 wavelet function was shown to have the best performance with the static MLP.

Flood forecasting is necessary and plays a vital role in planning flood regulations and protection measures. Deshmukh and Ghatol (2010) compared the Jordan and Elman networks for rainfall-runoff modeling. They collected data from the upper area of the Wardha River in India and used context units to expand the multilayer perceptron, a processing element that can remember past events. We found that the MSE and NMSE for the Jordan network were 0.0187 and 0.0357, respectively, which were lower than those of the Elman neural network, demonstrating that the Jordan network was more versatile and outperformed the Elman neural network. Roy, Choudhury, and Saharia (2010) noted a flood forecasting model using a focused time-Lagaed recurrent neural network (TLRN) with three memories: TDNN, gamma memory, and Laguerre. The model performance results indicated that a TLRN with gamma memory had better applicability, followed by a TDNN with Laguerre memory.

Subsequently, Chang, Chen, and Chang (2012) proposed a reinforced real-time recurrent learning algorithm (RTRL) for two-step-ahead (2SA) forecasting using RNNs to investigate two well-known benchmark sequence data sets and runoff in Taiwan flood events. For comparison, the original RTRL algorithms—the RNN, ESN, BPNNI, and BPNNII—were also used. It was shown that the novel reinforced 2SA weight adjustment technique had excellent feasibility and good precision for real-time 2SA forecasting by combining preliminary forecasting information with an online learning process. Future work will focus on the development of novel model forecasting on ungauged basins. Chen, Chang, and Chang (2013) incorporated the closest antecedent messages into an online learning proceeding, in which the authors considered multistep-ahead (MSA) forecasts for water factor research. A MAS-reinforced RTRL algorithm for RNNs (R-RTRLNN) was proposed that adequately adjusted the model parameters repeatedly according to the current information to improve reliability and forecast accuracy. The results showed that the presented R-PTLNN had good practicability and much better capability than comparative methods for MSA flood forecasts.

In recent years, significant water-level fluctuations have taken place and may be related to climate change. Therefore, water-level forecasting is a method to ensure sustainable water use. Guldal and Tongal (2010) discussed and compared the RNN, neurofuzzy approach, adaptive network-based fuzzy inference system (ANFIS), and classical stochastic models, such as the autoregressive (AR) and autoregressive moving average (ARMA) approaches. The results showed that the use of the generated RNN and ANFIS models had a good ability to learn and predict lake level changes. Afterward, Chang, Chen, Lu, Huang, and Chen (2014) performed a similar study in terms of flood forecasting using reinforced RNNs. They used three models—the BPNN, the Elman neural network, and the NARX network—to construct floodwater storage pond (FSP) water-level forecast models in two scenarios as model inputs. A gamma test was used to obtain effective factors that remarkably influenced the FSP water level. The experimental results showed that the NARX network had a higher applicability than the BPNN and Elman network. The method provided effective coefficients within 0.9 to 0.7 (scenario I) and 0.7 to 0.5 (scenario II) in the testing phase. They found that the presented NARX models were valuable and beneficial for urban flood control. We conclude with comments on possible future research directions in this field.

3.2 Weather Factor Forecasting. Weather forecasting has attracted considerable attention from various research teams due to its effort to sustain global human life. The main goal of weather forecasting is to predict the temperature, rain, wind, and special weather disasters of a local area. These predictions are important to many fields, including flight navigation, agriculture, tourism, and transportation. We generally depend on weather

forecasters to guide the planning of our daily routines. Numerous significant developments in weather forecasting have been proposed that make use of statistical modeling techniques and machine learning with remarkable success.

To achieve 24 hour weather forecasting of southern Saskatchewan, Maqsood, Khan, Huang, and Abdallah (2005) developed a soft computing model based on a radial basis function network (RBFN). Compared with the multilayered perceptron (MLP), ERNN, and Hopfield model (HFM), the RBFN was faster and more reliable than the other weather forecasting methods, exhibited good approximation and learning abilities, and was easier to train for faster convergence. It is important to consider other significant seasonal factors, such as rainfall and snowfall. Subsequently, these researchers contrasted the performance of the MLP, ERNN, and RBFN using several statistical measures to predict the weather of Vancouver, Canada. The empirical results clearly demonstrated the RBFN was much faster and more reliable for weather forecasting than other network models (Maqsood & Abraham, 2007). Over the past decade, coupled with the development of data and GPU-accelerated computing, deep learning has been widely used in many fields, such as in speech recognition and computer vision. Salman, Kanigoro, Heryadi, and IEEE (2015) investigated deep learning for weather forecasting. Particularly, they compared the forecasting performance of the RNN, conditional restricted Boltzmann machine (CRBM), and convolutional network (CN). The experimental results showed that the RNN had good performance for rainfall prediction. In the future, other deep learning algorithms will be used to accurately represent, classify, and predict time issues.

Recently, there has been much work on air quality forecasting using RNNs. Prakash, Kumar, Kumar, and Jain (2011) employed a wavelet-based Elman model to predict air pollution. The model results underlined that the high efficiency of the RNN was greater than those of the network models used in previous studies. In the same year, Wu, Feng, Du, and Li (2011) described an improved Elman neural network relying on a new activation function to predict the peak values of PM₁₀ air pollutants in the area of Wuhan, China. The improved Elman model provided low RMSE values and MAE values compared with the Elman model. Then a study was carried out to develop and compare different soft computing intelligence methodologies, such as the FFNN, NARX, and an adaptive neurofuzzy inference system (ANFIS), to forecast the emissions of CO₂ in the city of Nis. The data sets included air temperature, wind direction, traffic frequency, time of day, atmospheric stability, and CO₂ concentration. The study showed that the presented models offered more effective and accurate assessments using available expert knowledge. Simultaneously the NARX network performed the best in terms of evaluation because it considered both previous states and inputs, but it required a more advanced training method, and its computational time was significant. In addition, computational

intelligence methodologies can be used in other interesting applications in future research (Ciric, Cojbasic, Nikolic, Zivkovic, & Tomic, 2012).

Ong et al. (2014) introduced a deep recurrent neural network (DRNN) that was trained by exploiting a new autoencoder (AE) pretraining method called DynAE that was especially developed for the $PM_{2.5}$ concentration predictions in Japan. The main advantage of this method was that deep learning techniques on temporal predictions were improved. The experiments demonstrated that the presented method performed $PM_{2.5}$ concentration predictions with enhanced accuracy compared to the FFNN, RNN, and DFNN. The latest research, by Oprea, Popescu, and Mihalache (2016), presented a short-time $PM_{2.5}$ air prediction model based on the FFNN and RNN. The accuracy of each model was evaluated using the RMSE, IA, R^2 , and R . The major contribution of this work was verifying that the best neural network had a feedforward architecture. In future work, the $PM_{2.5}$ predictive model can be put into practical use, such as in the $PM_{2.5}$ monitoring station.

Apart from air quality, studies on drought, hurricanes, and other natural disaster predictions have also gradually increased, such as analysis of the surface of the ozone layer based on the RNN (Biancofiore et al., 2015). Mohammadinezhad and Jalili (2013) developed a prediction model using echo state networks that is a class of RNN based on remote sensing data to predict drought conditions. They used the Kronecker product to reduce the number of parameters to be optimized and supplied three evolution methods: a genetic algorithm, simulated annealing, and differential evolution. The method based on different optimization technologies achieved an average accuracy of 74.25%, which outperformed the other methods. Future work will compare the performance of this method with those of other classic techniques. Fang, Wang, Murphey, Weber, and MacNeille (2014) studied the MLP and ERNN algorithms to forecast specific humidity from three weather stations. The results showed that the ERNN is a promising alternative to for this forecast. In the future, we should focus on other state-of-the-art time series forecasting models.

The accurate prediction of hurricane occurrences is important and can directly reduce economic loss and save human lives. Kordmahalleh, Sefidmazgi, Homaifar, and ACM (2016) introduced a sparse RNN with an agile topology for trajectory forecasting of Atlantic hurricanes. The topology of the RNN was optimized through a customized genetic algorithm. The proposed approach had a high degree of correlation and accuracy for 6 and 12 hours ahead of the trajectory forecast of four catastrophic Atlantic hurricanes. In the future, exploring the proposed approach for tracking other Atlantic hurricanes and comparing the results with different techniques should be the main focus. In a recent study, Le, El-Askary, Allali, and Struppa (2017) applied RNNs for drought prediction in California. The correlation coefficients varied by approximately 0.7, which was quite similar to the current observed precipitation levels and PZI values for 2016 compared

to those of the 1997–1998 season. The result of this study contributed to the prediction that drought conditions will continue to persist and showed that precipitation associated with the 2015–2016 El Niño season continued to weaken compared with the historic 1997–1998 El Niño season.

3.3 Artificial Environment Factor Forecasting. The scope of the artificial environment gradually increased, along with the development of social economies, such as supporting facilities, public service facilities, and traffic. This section primarily discusses traffic forecasting based on RNNs, which has been vital and critical for traffic control with the rapid increase of vehicles.

Many researchers have used RNNs for traffic forecasting, an area that requires highly accurate information on traffic congestion management in intelligent transportation systems. A novel dynamic time-delay recurrent wavelet neural network (WNN) model was discussed by Jiang and Adeli (2005) that incorporated the self-similar, singular, and fractal properties discovered in the traffic flow and achieved high prediction accuracy. Sheu, Lan, and Huang (2009) presented a novel real-time recurrent (RTRL) algorithm to train nonlinear traffic dynamics measured in different aspects. Thus, the goal of accurate prediction which is affected by time intervals, time lags, and time periods, can be achieved.

An, Song, and Zhao (2011) used a model for traffic flow forecasting based on echo state neural networks (ESNs). ESNs can effectively avoid the troublesome problem of using random network structure generation and training using least squares algorithms, as well as comparisons with the FFNN. The experimental results revealed that the ESN method had better forecasting performance, which proved its validity. To predict traffic speeds, Ma, Tao, Wang, Yu, and Wang (2015) designed a scheme based on the LSTM using remote microwave sensor data. The LSTM was compared with three typologies of RNNs (the Elman NN, TDNN, and NARX) and other classical statistical models, such as the SVM and a Kalman filter based on the same data set. The experiments demonstrated that the LSTM, an effective approach for learning time series with a long time dependency and automatically identifying the optimal time lags, outperformed the other algorithms in terms of accuracy and stability. Future studies should consider inputting spatial and temporal information into the LSTM and studying forecasting performance with different data aggregation levels. Adding multiple layers to the LSTM to enhance the learning capability of neural network is a potential path for improving the method. A similar study was carried out by Tian and Pan (2015). They used the LSTM RNN model to forecast short-term traffic flows, and the data set was collected from the Caltrans Performance Measurement system. Four classical prediction methods were selected for comparison: the random walk (RW), SVM, feedforward vector machine, and stacked autoencoder (SAE). Three aspects of those models were tested and compared: forecast accuracy, memory capability of long

historical data, and generalization ability with different intervals of prediction. The experimental results showed that both the MAPE and RMSE were lowest, with the different prediction intervals, which proved that the LSTM was able to effectively capture the nonlinearity and randomness of traffic flow and achieve good prediction accuracy and generalization.

Other research based on the LSTM NN for traffic prediction includes that by Chen, Lv, Li, and Wang (2016), Fu, Zhang, and Li (2016), and Zhao, Chen, Wu, Chen, and Liu (2017). Future work should focus on comprehensive traffic forecasts, and other machine learning techniques should be used to enhance performance, while inputting social media, weather information, and other factors into the model. In addition, the LSTM can also be used for predicting dynamic origin-destination (OD) matrices in a subway network, which also requires additional factors to be input into the OD matrices of other transport systems. The results validated that the presented model achieved better performance in comparison with traditional tools, such as the calendar methodology and vector autoregression. Future investigations will be carried out to reduce the computational time and achieve better performance (Toque, Come, Mahrsi, & Oukhellou, 2016).

There are RNNs that use other deep learning algorithms, such as the convolutional neural network (CNN), for time series forecasting. Lai, Wei-Cheng, Yiming, and Hanxiao (2017) presented a novel deep learning framework, the long- and short time-series network (LSTNet), which used the CNN to extract short-term load dependency patterns among variables and the RNN to discover long-term patterns and trends. The experimental results showed that the LSTNet indeed achieved significant performance improvements over those of several baseline methods. In this period, a spatiotemporal image-based approach was adopted by Yu, Wu, Wang, Wang, and Ma (2017) to predict large-scale transportation network traffic, in which the deep convolutional neural network (DCNN) was used to obtain spatial dependencies in different links and the LSTM was presented to learn the long-term temporal dependencies of each link. The authors compared the novel model with the LSTM, DCNN, SAE, and SVM using the same data set. The numerical tests revealed that the SRCN outperformed other methods with respect to accuracy and stability. However, additional factors, such as weather, social events, and traffic control, should be considered, as well as the pretraining methods to enhance the model's performance. Li, Yu, Shahabi, and Yan (2017) designed experiments to present a deep learning framework for traffic flow prediction, which included both spatial and temporal dependencies. The methods significantly outperformed conventional approaches when evaluating two large-scale, real-world traffic data sets. In the future, applying the presented method to other spatiotemporal prediction tasks will be studied.

This review mainly introduces current applications of RNNs in time series forecasting. RNNs are being applied in various fields, such as environmental factor forecasting. Moreover, the methods used, including the

classic RNNs, improved RNNs, hybrid models of RNNs, and other models, enhance prediction accuracy and obtain good prediction effects. Numerous theoretical studies and practical implementations have been carried out by researchers in many fields. In the future, RNNs can be extended to other fields, such as agricultural applications, while seeking better optimization algorithms and using more advanced RNN technology.

4 Limitations and Challenges

In nearly all scientific and industrial fields, such as finance, economics, hydrology, and telecommunication and electrical systems, research on time series forecasting has attracted attention in recent years (Nogales, Contreras, Conejo, & Espinola, 2002; Kim, 2003; Chen & Chen, 2015). Forecasting time series events in these fields is challenging because of the high volatility and multiple influencing factors. It is also difficult to process data that have nonlinearities, low reliability, and high heterogeneity. However, accurate predictions and evaluations of time series data in these fields are extremely important for effective decision making, intelligent monitoring, scientific management, and risk assessment for future events.

To obtain high precision, widely known work on RNNs has been carried out that proposes new forecasting techniques and architectures from multiple models. The problem of traditional networks is that they have slow convergence and poor stability, which lead to overfitting. Overall, traditional methods and techniques have poor prediction accuracy on sequential data. Recurrent neural networks are widely used because of their powerful dynamic characteristics, excellent architecture, and training methods. RNNs have many advantages, including associative memory, adaptive learning, fast optimization, and strong robustness. They have been shown to be feasible and to obtain good predictive performance with generalized capability, but they need to be improved regarding time sequence predictions.

Some shortcomings and challenges need to be improved. On the one hand, conventional methods involve few influencing factors and do not take other complicated factors into account, such as social media, local events, and weather conditions. RNNs as a class of data-driven models are different from traditional models and require a large number of data sets to obtain accurate predictions. On the other hand, it is necessary to consider how to simultaneously establish optimal model parameters and structures, as well as to reduce the computational time for training. RNNs mainly focus on short-term prognostics, while applications of RNNs for long-term prediction are relatively scarce and have poor prediction accuracy, as shown in the literature. Most research compares the performance of RNNs only with that of traditional neural networks. However, several studies have been devoted to combining RNNs with other deep learning algorithms (LeCun et al., 2015; Zuo, Fan, Blasch, & Ling, 2017), such as the recurrent temporal

restricted Boltzmann machine (RTRBM; Sutskever, Hinton, & Taylor, 2008; Wang, Wang, Zhao, & Wang, 2017) and convolutional LSTM (ConvLSTM; Shi et al., 2015; Zhu, Zhang, Shen, & Song, 2017). In addition, the combination of RNNs and other deep learning methods applied to time series forecasting in the fields noted will be key in future studies.

These problems are not easily dealt with. In future work, we will take other complex factors into account in models to obtain higher prediction accuracy. As many sequence data as possible have been obtained and used to train and test the models. In addition, further investigations will be carried out to enhance the learning capability of neural networks and reduce computational time, leading to faster training. It is also important for RNNs to store information for a long time. Moreover, long-term predictions using RNNs and their hybrid models will lead to new breakthroughs. More researchers are needed to study long-term forecasts in various fields. Such studies can increase scientific and forecasting accuracy. Simultaneously, we expect that deep learning methods will be combined with RNNs to provide precise predictions for select time series problems and compare the predictive performance with the shallow models. Finally, expanding into new areas for forecasting research is of great value to solve real-world problems.

The aim of this review is to describe the applications of RNNs in forecasting that have great practical and economic value in many fields. The areas of application are mainly environmental factors based on current work. Of course, many novel fields need to be explored. For example, accurate and efficient predictions of water quality changes play a significant role in environmental planning and can prevent water deterioration and disease outbreak in aquaculture, as well as guide scientific breeding. Moreover, environmental factor predictions in solar greenhouse or aquaponics are also worthwhile research directions. With the development of the Internet of things and computer technology, real-time, multivariate, and high-dimensional water quality data are obtained quickly and accurately. It is of great significance and value to construct models using these time series data to lead to the healthy growth of aquatic products and scientific management of aquatic works. There are some conventional methods for constructing water quality models (Chau, 2006; Faruk, 2010; Chang, Tsai, Chen, Coynel, & Vachaud, 2015; Gazzaz, Yusoff, Aris, Juahir, & Ramli, 2012; Najah, El-Shafie, Karim, & El-Shafie, 2013). Liu et al. (2013) presented a hybrid approach that combines support vector regression with a genetic algorithm to solve the aquaculture water quality prediction problem. Liu, Wei, and Chen (2013) proposed a fuzzy neural network to solve the problem of dissolved oxygen forecasting. These models have had an effect on helping to facilitate early warning and reduce losses. However, there are still many disadvantages. To solve the problem of poor robustness and low precision of traditional forecasting models, RNNs are a promising tool for making high-precision water quality predictions. We believe that future applications of

RNNs in forecasting will expanded to other areas that few people have considered to date, such as aquaculture.

5 Conclusion

In this review, we have presented a number of preliminary publications on the applications of RNNs in the time series analysis and forecasting. As we have summarized, RNNs have been applied for forecasting time series data in most scientific and industrial fields, but mainly in environmental factor forecasting. In addition, we present the structure, processing flow, and advantages of RNNs in this review. Furthermore, RNNs, such as the Elman neural network, LSTM neural network, and improvement models can be powerful prediction alternatives to traditional neural networks and can obtain better prediction results for some problems. We also present the limitations and challenges of prediction models based on RNNs and discuss the future development of RNNs to make predictions from sequential data. This review provides useful guidance for RNN modeling and novel research fields in subsequent studies.

Acknowledgments

We thank American Journal Experts for providing English-language editing of this review. The review is supported by the Science & Technology Program of Beijing Research and Demonstration of Technologies Equipment Capable of Intelligent Control for Large-Scale Healthy Cultivation of Freshwater Fish (Z171100001517016); and the Shandong Province key Research and Development Program "Research and Demonstration of Accurate Monitoring and Controlling Technologies for Environment of Vegetable in Facility" (2017CXGC0201).

References

- Alzahrani, A., Shamsi, P., Dagli, C., & Ferdowsi, M. (2017). Solar irradiance forecasting using deep neural networks. In C. H. Dagli (Ed.), *Complex adaptive systems conference with theme: Engineering cyber physical systems, Cas* (vol. 114, pp. 304–313). Amsterdam: Elsevier.
- An, Y., Song, Q., & Zhao, X. (2011). Short-term traffic flow forecasting via echo state neural networks. In *Proceedings of the International Conference on Natural Computation* (pp. 844–847). Piscataway, NJ: IEEE.
- Awano, H., Nishide, S., Arie, H., Tani, J., Takahashi, T., Okuno, H. G., & Ogata, T. (2011). *Use of a sparse structure to improve learning performance of recurrent neural networks*. Berlin: Springer.
- Bebis, G., & Georgiopoulos, M. (2009). Feed-forward neural networks. *IEEE Potentials*, 13(4), 571–586.

- Biancofiore, F., Verdecchia, M., Di Carlo, P., Tomassetti, B., Aruffo, E., Busilacchio, M., & Colangeli, C. (2015). Analysis of surface ozone using a recurrent neural network. *Science of the Total Environment*, 514, 379–387. doi:10.1016/j.scitotenv.2015.01.106
- Bodén, M. (2002). *A guide to recurrent neural networks and backpropagation*. Halmstad Sweden: Halmstad University.
- Bontempi, G., Taieb, S. B., & Borgne, Y. A. L. (2013). *Machine learning strategies for time series forecasting*. Berlin: Springer.
- Cao, J. D. (2001). Global exponential stability of Hopfield neural networks. *International Journal of Systems Science*, 32(2), 233–236. doi:10.1080/00207720117783
- Chang, F.-J., Chen, P.-A., Lu, Y.-R., Huang, E., & Chang, K.-Y. (2014). Real-time multi-step-ahead water level forecasting by recurrent neural networks for urban flood control. *Journal of Hydrology*, 517, 836–846. doi: 10.1016/j.jhydrol.2014.06.013
- Chang, F. J., Tsai, Y. H., Chen, P. A., Coynel, A., & Vachaud, G. (2015). Modeling water quality in an urban river using hydrological factors: Data driven approaches. *Journal of Environmental Management*, 151, 87–96. doi: 10.1016/j.jenvman.2014.12.014
- Chang, L.-C., Chen, P.-A., & Chang, F.-J. (2012). Reinforced two-step-ahead weight adjustment technique for online training of recurrent neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 23(8), 1269–1278. doi:10.1109/tnnls.2012.2200695
- Chau, K.-W. (2006). A review on integration of artificial intelligence into water quality modelling. *Marine Pollution Bulletin*, 52(7), 726–733. doi:10.1016/j.marpolbul.2006.04.003
- Chen, M. Y., & Chen, B. T. (2015). A hybrid fuzzy time series model based on granular computing for stock price forecasting. *Information Sciences*, 294, 227–241. doi:10.1016/j.ins.2014.09.038
- Chen, P.-A., Chang, L.-C., & Chang, F.-J. (2013). Reinforced recurrent neural networks for multi-step-ahead flood forecasts. *Journal of Hydrology*, 497, 71–79. doi:10.1016/j.jhydrol.2013.05.038
- Chen, S., Billings, S. A., & Grant, P. M. (1990). Nonlinear-system identification using neural networks. *International Journal of Control*, 51(6), 1191–1214. doi:10.1080/00207179008934126
- Chen, T. P. (2001). Global exponential stability of delayed Hopfield neural networks. *Neural Networks*, 14(8), 977–980. doi:10.1016/s0893-6080(01)00059-4
- Chen, Y. Y., Lv, Y., Li, Z., & Wang, F. Y. (2016). Long short-term memory model for traffic congestion prediction with online open data. In *Proceedings of the IEEE International Conference on Intelligent Transportation Systems*. Piscataway, NJ: IEEE.
- Ciric, I. T., Cojbasic, Z. M., Nikolic, V. D., Zivkovic, P. M., & Tomic, M. A. (2012). Air quality estimation by computational intelligence methodologies. *Thermal Science*, 16, S493–S504. doi:10.2298/tsci120503186c
- Collobert, R., Weston, J., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(1), 2493–2537.
- Cruise, H. (1996). *Neural networks as cybernetic systems*. Stuttgart: Thieme Medical Publishers.

- Deshmukh, R. P., & Ghatol, A. A. (2010). Comparative study of Jordan and Elman model of neural network for short term flood forecasting. In *Proceedings of the 2010 3rd IEEE International Conference on Computer Science and Information Technology* (vol. 9, 400–404). Piscataway, NJ: IEEE.
- Dinarelli, M., & Tellier, I. (2016). *Improving recurrent neural networks for sequence labelling*. arXiv:1606.02555v1.
- Duong, H. N., Nguyen, Q. N. T., Bui, L. T., Nguyen, H. T., & Snaes, V. (2014). Applying recurrent fuzzy neural network to predict the runoff of Srepok River. In K. Saeod & V. Snaes (Eds.), *Computer information systems and industrial management* (vol. 8838, pp. 55–66). Berlin: Springer-Verlag.
- Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, 14(2), 179–211.
- Fang, C., Wang, X., Murphey, Y. L., Weber, D., & MacNeille, P. (2014). Specific humidity forecasting using recurrent neural network. In *Proceedings of the 2014 International Joint Conference on Neural Networks* (pp. 955–960). Piscataway, NJ: IEEE.
- Faruk, D. O. (2010). A hybrid neural network and ARIMA model for water quality time series prediction. *Engineering Applications of Artificial Intelligence*, 23(4), 586–594. doi:10.1016/j.engappai.2009.09.015
- Fernández, S., Graves, A., & Schmidhuber, J. (2007). Sequence labelling in structured domains with hierarchical recurrent neural networks. In *Proceedings of the International Joint Conference on Artificial Intelligence*. New York: ACM.
- Fu, R., Zhang, Z., & Li, L. (2016). Using LSTM and GRU neural network methods for traffic flow prediction. In *Proceedings of the 2016 31st Youth Academic Annual Conference of Chinese Association of Automation* (pp. 324–328).
- Garcia-Pedrero, A., & Gomez-Gil, P. (2010). Time series forecasting using recurrent neural networks and wavelet reconstructed signals. In *Proceedings of the International Conference on Electronics, Communications and Computer*. Piscataway, NJ: IEEE.
- Gazzaz, N. M., Yusoff, M. K., Aris, A. Z., Juahir, H., & Ramli, M. F. (2012). Artificial neural network modeling of the water quality index for Kinta River (Malaysia) using water quality variables as predictors. *Marine Pollution Bulletin*, 64(11), 2409–2420. doi:10.1016/j.marpolbul.2012.08.005
- Gers, F. A., Schmidhuber, J., & Cummins, F. (2000). Learning to forget: Continual prediction with LSTM. *Neural Computation*, 12(10), 2451–2471.
- Gers, F. A., & Schmidhuber, E. (2001). LSTM recurrent networks learn simple context-free and context-sensitive languages. *IEEE Trans. Neural Netw.*, 12(6), 1333–1340.
- Gers, F. A., & Schraudolph, N. N. (2003). *Learning precise timing with LSTM recurrent networks*. JMLR.org.
- Gil, P., Cardoso, A., & Palma, L. (2009). Estimating the number of hidden neurons in recurrent neural networks for nonlinear system identification. In *Proceedings of the IEEE International Symposium on Industrial Electronics*. Piscataway, NJ: IEEE.
- Giles, C. L., Kuhn, G. M., & Williams, R. J. (1994). Dynamic recurrent neural networks: Theory and applications. *IEEE Transactions on Neural Networks*, 5(2), 153–156.
- Guldal, V., & Tongal, H. (2010). Comparison of recurrent neural network, adaptive neuro-fuzzy inference system and stochastic models in Egirdir Lake level forecasting. *Water Resources Management*, 24(1), 105–128. doi:10.1007/s11269-009-9439-9

- Haddad, S., Mellit, A., Benghanem, M., & Daffallah, K. O. (2016). NARX-based short-term forecasting of water flow rate of a photovoltaic pumping system: A case study. *Journal of Solar Energy Engineering-Transactions of the ASME*, 138(1), 6. doi:10.1115/1.4031970
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780.
- Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. USA*, 79(8), 2554.
- Jiang, X. M., & Adeli, H. (2005). Dynamic wavelet neural network model for traffic flow forecasting. *Journal of Transportation Engineering—ASCE*, 131(10), 771–779. doi:10.1061/(asce)0733-947x(2005)131:10(771)
- Kalinli, A., & Sagirolu, S. (2006). Elman network with embedded memory for system identification. *Journal of Information Science and Engineering*, 22(6), 1555–1568.
- Kim, K. J. (2003). Financial time series forecasting using support vector machines. *Neurocomputing*, 55(1–2), 307–319. doi:10.1016/s0925-2312(03)00372-2
- Kordmahalleh, M. M., Sefidmazgi, M. G., Homaifar, A., & ACM (2016). A sparse recurrent neural network for trajectory prediction of Atlantic hurricanes. In *Gecco'16: Proceedings of the 2016 Genetic and Evolutionary Computation Conference* (pp. 957–964). New York: ACM. doi:10.1145/2908812.2908834
- Lai, G., Wei-Cheng, C., Yiming, Y., & Hanxiao, L. (2017). *Modeling long- and short-term temporal patterns with deep neural networks*. arXiv:1703.07015.
- Le, J. A., El-Askary, H. M., Allali, M., & Struppa, D. C. (2017). Application of recurrent neural networks for drought projections in California. *Atmospheric Research*, 188, 100–106. doi: 10.1016/j.atmosres.2017.01.002
- Le, P., & Zuidema, W. (2016). Quantifying the vanishing gradient and long distance dependency problem in recursive neural networks and recursive LSTMs. In *Workshop on Representation Learning for NLP* (pp. 87–93).
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436–444. doi:10.1038/nature14539
- Lee, H.-Y., Tseng, B.-H., Wen, T.-H., & Tsao, Y. (2017). Personalizing recurrent-neural-network-based language model by social network. *IEEE-ACM Transactions on Audio Speech and Language Processing*, 25(3), 519–530. doi:10.1109/taslp.2016.2635445
- Li, Y., Yu, R., Shahabi, C., & Yan, L. (2017). *Graph convolutional recurrent neural network: Data-driven traffic forecasting*. arXiv:1707.01926v3.
- Liu, S., Tai, H., Ding, Q., Li, D., Xu, L., & Wei, Y. (2013). A hybrid approach of support vector regression with genetic algorithm optimization for aquaculture water quality prediction. *Mathematical and Computer Modelling*, 58(3–4), 458–465.
- Liu, Y., Wei, Y., & Chen, Y. (2013). Dissolved oxygen prediction model which based on fuzzy neural network. *IFIP Advances in Information and Communication Technology*, 420(1), 49–54.
- Ma, X. L., Tao, Z. M., Wang, Y. H., Yu, H. Y., & Wang, Y. P. (2015). Long short-term memory neural network for traffic speed prediction using remote microwave sensor data. *Transportation Research Part C—Emerging Technologies*, 54, 187–197. doi:10.1016/j.trc.2015.03.014
- Mahmoud, M. A. (2008). Phase I analysis of multiple linear regression profiles. *Communications in Statistics—Simulation and Computation*, 37(10), 2106–2130. doi:10.1080/03610910802305017

- Mandic, D. P., & Chambers, J. (2001). Recurrent neural networks for prediction: Learning algorithms, architectures and stability. In K. S. Narendra (Ed.), *Adaptive and learning systems for signal processing communications and control*. Boston: Springer.
- Maqsood, I., & Abraham, A. (2007). Weather analysis using ensemble of connectionist learning paradigms. *Applied Soft Computing*, 7(3), 995–1004. doi:10.1016/j.asoc.2006.06.005
- Maqsood, M., Khan, M. R., Huang, G. H., & Abdallah, R. (2005). Application of soft computing models to hourly weather analysis in southern Saskatchewan, Canada. *Engineering Applications of Artificial Intelligence*, 18(1), 115–125. doi:10.1016/j.engappai.2004.08.019
- Mikolov, T., Karafiát, M., Burget, L., Cernocký, J., & Khudanpur, S. (2010). *Recurrent neural network based language model*. Paper presented at the INTERSPEECH 2010, Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September.
- Mohammadinezhad, A., & Jalili, M. (2013). Optimization of echo state networks for drought prediction based on remote sensing data. In *Proceedings of the 2013 IEEE 8th Conference on Industrial Electronics and Applications* (pp. 126–130). Piscataway, NJ: IEEE.
- Motlagh, M. T., Khaloozadeh, H., & IEEE. (2016). A new architecture for modeling and prediction of dynamic systems using neural networks: Application in Tehran stock exchange. In *Proceedings of the 2016 4th International Conference on Control, Instrumentation, and Automation (ICCIA)* (pp. 196–201). Piscataway, NJ: IEEE.
- Najah, A., El-Shafie, A., Karim, O. A., & El-Shafie, A. H. (2013). Application of artificial neural networks for water quality prediction. *Neural Computing and Applications*, 22, S187–S201. doi:10.1007/s00521-012-0940-3
- Nogales, F. J., Contreras, J., Conejo, A. J., & Espinola, R. (2002). Forecasting next-day electricity prices by time series models. *IEEE Transactions on Power Systems*, 17(2), 342–348. doi:10.1109/tpwrs.2002.1007902
- Ong, B. T., Sugiura, K., & Zettsu, K. (2014). Dynamic pre-training of deep recurrent neural networks for predicting environmental monitoring data. In *Proceedings of the 2014 IEEE International Conference on Big Data (Big Data)* (pp. 760–765). Piscataway, NJ: IEEE.
- Oprea, M., Popescu, M., & Mihalache, S. F. (2016). A neural network based model for PM2.5 air pollutant forecasting. In *Proceedings of the 2016 20th International Conference on System Theory, Control and Computing* (pp. 776–781). Piscataway, NJ: IEEE.
- Pollack, J. B. (1990). Recursive distributed representations. *Artificial Intelligence*, 46(1), 77–105.
- Prakash, A., Kumar, U., Kumar, K., & Jain, V. K. (2011). A wavelet-based neural network model to predict ambient air pollutants' concentration. *Environmental Modelling and Assessment*, 16(5), 503–517.
- Raza, M. Q., & Khosravi, A. (2015). A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings. *Renewable and Sustainable Energy Reviews*, 50, 1352–1372. doi:10.1016/j.rser.2015.04.065

- Riedmiller, M. (1994). Advanced supervised learning in multilayer perceptrons: From backpropagation to adaptive learning algorithms. *Computer Standards and Interfaces*, 16(3), 265–278. doi:10.1016/0920-5489(94)90017-5
- Roy, P., Choudhury, P. S., & Saharia, M. (2010). Dynamic ANN modeling for flood forecasting in a river network. In *Proceedings of the International Conference on Modeling, Optimization, and Computing* (vol. 1298, pp. 219–225). College Park, MD: American Institute of Physics.
- Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1988). *Learning internal representations by error propagation*. Cambridge, MA: MIT Press.
- Sadaei, H. J., Guimaraes, F. G., da Silva, C. J., Lee, M. H., & Eslami, T. (2017). Short-term load forecasting method based on fuzzy time series, seasonality and long memory process. *International Journal of Approximate Reasoning*, 83, 196–217. doi:10.1016/j.ijar.2017.01.006
- Saharia, M., & Bhattacharjya, R. K. (2012). Geomorphology-based time-lagged recurrent neural networks for runoff forecasting. *KSCE Journal of Civil Engineering*, 16(5), 862–869. doi:10.1007/s12205-012-1463-2
- Sak, H., Senior, A., & Beaufays, F. (2014). Long short-term memory recurrent neural network architectures for large scale acoustic modeling. *Computer Science*, 24, 338–342.
- Salman, M. G., Kanigoro, B., Heryadi, Y., & IEEE. (2015). Weather forecasting using deep learning techniques. In *Proceedings of the 2015 International Conference on Advanced Computer Science and Information Systems* (pp. 281–285). Piscataway, NJ: IEEE.
- Sanz-Serna, J. M. (2009). Modulated Fourier expansions and heterogeneous multiscale methods. *IMA Journal of Numerical Analysis*, 29(3), 595–605. doi:10.1093/imanum/drn031
- Schuster, M., & Paliwal, K. K. (1997). *Bidirectional recurrent neural networks*. Piscataway, NJ: IEEE Press.
- Sheu, J.-B., Lan, L. W., & Huang, Y.-S. (2009). Short-term prediction of traffic dynamics with real-time recurrent learning algorithms. *Transportmetrica*, 5(1), 59–83. doi:10.1080/18128600802591681
- Shi, X., Chen, Z., Wang, H., Yeung, D. Y., Wong, W., & Woo, W. (2015). *Convolutional LSTM network: A machine learning approach for precipitation nowcasting*. arXiv:1506.04214v1
- Shoab, M., Shamseldin, A. Y., Melville, B. W., & Khan, M. M. (2016). A comparison between wavelet based static and dynamic neural network approaches for runoff prediction. *Journal of Hydrology*, 535, 211–225. doi:10.1016/j.jhydrol.2016.01.076
- Singh, K. P., Basant, A., Malik, A., & Jain, G. (2009). Artificial neural network modeling of the river water quality: A case study. *Ecological Modelling*, 220 (6), 888–895.
- Song, Q. (2011). Robust initialization of a Jordan network with recurrent constrained learning. *IEEE Transactions on Neural Networks*, 22(12), 2460–2473.
- Sutskever, I., Hinton, G., & Taylor, G. (2008). The recurrent temporal restricted Boltzmann machine. In D. Koller, D. Schuurmans, Y. Bengio, & L. Bottou (Eds.), *Advances in neural information processing systems*, 21. Red Hook, NY: Curran.
- Tascikaraoglu, A., & Uzunoglu, M. (2014). A review of combined approaches for prediction of short-term wind speed and power. *Renewable and Sustainable Energy Reviews*, 34, 243–254. doi:10.1016/j.rser.2014.03.033

- Tay, F. E. H., & Cao, L. (2001). Application of support vector machines in financial time series forecasting. *Journal of University of Electronic Science and Technology of China*, 29(4), 309–317.
- Tian, Y. X., & Pan, L. (2015). Predicting short-term traffic flow by long short-term memory recurrent neural network. In *Proceedings of the 2015 IEEE International Conference on Smart City/Socialcom/Sustaincom* (pp. 153–158). Piscataway, NJ: IEEE. doi:10.1109/SmartCity.2015.63
- Toque, F., Come, E., El Mahrsi, M. K., & Oukhellou, L. (2016). Forecasting dynamic public transport origin-destination matrices with long-short term memory recurrent neural networks. In *Proceedings of the 2016 IEEE 19th International Conference on Intelligent Transportation Systems* (pp. 1071–1076). Piscataway, NJ: IEEE.
- Tsoi, A. C. (1998). Recurrent neural network architectures: An overview. In *The Adaptive Processing of Sequences and Data Structures*, International Summer School on Neural Networks, Er Caianiello Tutorial Lectures.
- Turk, F., Barisci, N., Ciftci, A., & Ekmekci, Y. (2015). Comparison of multi layer perceptron and Jordan Elman neural networks for diagnosis of hypertension. *Intelligent Automation and Soft Computing*, 21(1), 123–134. doi:10.1080/10798587.2014.959312
- Verdejo, J. E. D., Herreros, A. P., Luna, J. C. S., Ortuzar, M. C. B., & Ayuso, A. R. (1991). Recurrent neural networks for speech recognition. *Lecture Notes in Computer Science* (pp. 1–26). Berlin-Springer-Verlag.
- Wang, Q., Wang, L., Zhao, J., & Wang, W. (2017). Long-term time series prediction based on deep denoising recurrent temporal restricted Boltzmann machine network. In *Proceedings of the Chinese Automation Congress*. Piscataway, NJ: IEEE.
- Wu, S., Feng, Q., Du, Y., & Li, X. (2011). Artificial neural network models for daily PM10 air pollution index prediction in the urban area of Wuhan, China. *Environmental Engineering Science*, 28(5), 357–363. doi:10.1089/ees.2010.0219
- Wu, Z., Wang, X., Jiang, Y. G., Ye, H., & Xue, X. (2015). Modeling spatial-temporal clues in a hybrid deep learning framework for video classification. In *Proceedings of the ACM International Conference on Multimedia*. New York: ACM.
- Yao, K., Zweig, G., & Hwang, M. (2013). Recurrent neural networks for language understanding. In *Proceedings of Interspeech*. doi:10.13140/2.1.2755.3285
- Yu, H., Wu, Z., Wang, S., Wang, Y., & Ma, X. (2017). Spatiotemporal recurrent convolutional networks for traffic prediction in transportation networks. *Sensors (Basel, Switzerland)*, 17(7). doi:10.3390/s17071501
- Zhang, G. P. (2003). Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing*, 50, 159–175. doi:10.1016/s0925-2312(01)00702-0
- Zhang, G. Q., Patuwo, B. E., & Hu, M. Y. (1998). Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting*, 14(1), 35–62. doi:10.1016/s0169-2070(97)00044-7
- Zhang, N. (2011). Urban stormwater runoff prediction using recurrent neural networks. *Advances In Neural Networks - Isnn 2011, Pt I* (vol. 6675, pp. 610–619).
- Zhao, Z., Chen, W. H., Wu, X. M., Chen, P. C. Y., & Liu, J. M. (2017). LSTM network: A deep learning approach for short-term traffic forecast. *IET Intelligent Transport Systems*, 11(2), 68–75. doi:10.1049/iet-its.2016.0208

- Zheng, H., Yuan, J., & Chen, L. (2017). Short-term load forecasting using EMD-LSTM neural networks with a xgboost algorithm for feature importance evaluation. *Energies*, *10*(8). doi: 10.3390/en10081168
- Zhu, G. M., Zhang, L., Shen, P. Y., & Song, J. (2017). *Multimodal gesture recognition using 3-D and convolutional LSTM*. Piscataway, NJ: IEEE.
- Zuo, H., Fan, H., Blasch, E., & Ling, H. (2017). Combining convolutional and recurrent neural networks for human skin detection. *IEEE Signal Processing Letters*, *24*(3).

Received October 5, 2017; accepted July 18, 2018.