

PAPER • OPEN ACCESS

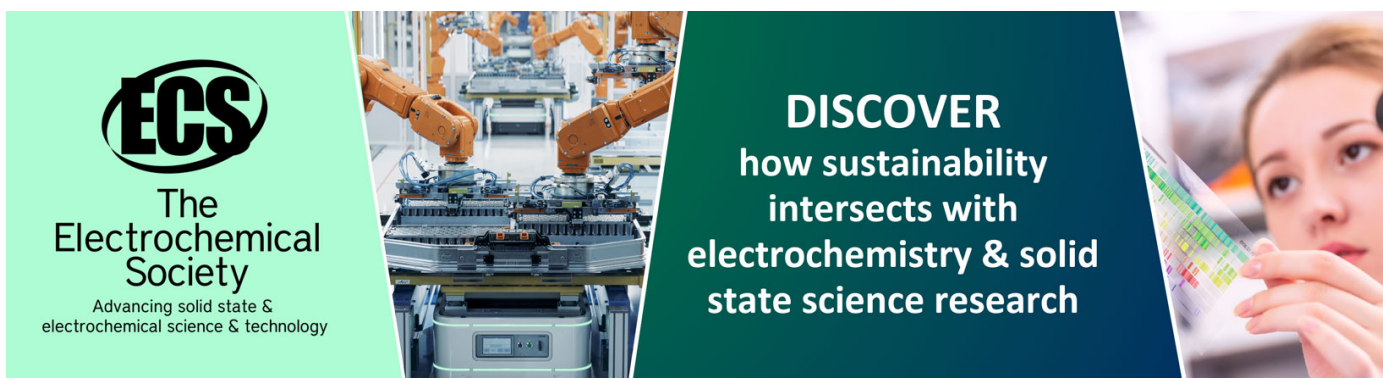
AGA-LSTM: An Optimized LSTM Neural Network Model Based on Adaptive Genetic Algorithm

To cite this article: Chenyao Bai 2020 *J. Phys.: Conf. Ser.* **1570** 012011

View the [article online](#) for updates and enhancements.

You may also like

- [A comparative study on agarose acetate and PDLLA scaffold for rabbit femur defect regeneration](#)
Ruifang Zhao, Zunkai Xu, Bing Li et al.
- [Operation stage division and RUL prediction of bearings based on 1DCNN-QN-LSTM](#)
Runxia Guo, Haonan Li and Chao Huang
- [Real-Time Prediction of Anode Potential in Li-Ion Batteries Using Long Short-Term Neural Networks for Lithium Plating Prevention](#)
Xianke Lin



ECS
The
Electrochemical
Society
Advancing solid state &
electrochemical science & technology

DISCOVER
how sustainability
intersects with
electrochemistry & solid
state science research

AGA-LSTM: An Optimized LSTM Neural Network Model Based on Adaptive Genetic Algorithm

Chenyao Bai^{1*}

¹ Department of Public Education, Shanghai Customs College, Shanghai, 201204, China

*Corresponding author's e-mail: baichenyao@shcc.edu.cn

Abstract. With the increase of the hidden layer, the weight update of the LSTM neural network model depends heavily on the gradient descent algorithm, and the convergence speed is slow, resulting in the local extremum of the weight adjustment, which affects the prediction performance of the model. Based on this, this paper proposes an optimized LSTM neural network model based on adaptive genetic algorithm (AGA-LSTM). In this model, the mean squared error is designed as the fitness function, and the adaptive genetic algorithm (AGA) is used to globally optimize the weights between neuron nodes of the LSTM model to improve the generalization ability. The experimental results show that, on the UCI dataset, the prediction accuracy of the AGA-LSTM model is greatly improved compared to the standard LSTM model, which verifies the rationality of the model.

1. Introduction

With the updating and iteration of computer hardware, deep learning and artificial intelligence have become the research hotspots of scholars worldwide[1]. In deep learning, there are many neural network models, for example, the RBF neural network, the BP neural network, the deep belief network, the convolutional neural network, the recurrent neural network and so on. Among them, the recurrent neural network (RNN) and its variant-long-short time memory model have been widely recognized in industry and academia, and are widely used in speech recognition[2], natural language processing[3, 4], image recognition[5, 6] and time series prediction[7] and other various fields. The long-short time memory (LSTM) neural network is proposed by Hochreiter et al.[8], which is used to improve the standard RNN neural network, that is, by designing 3 gated units to achieve Selective memory of sequence information, and to some extent to inhibit the gradient disappearance.

However, when the number of hidden layers is large, the LSTM model still has the disadvantages of low learning efficiency, slow convergence speed, and easy gradient disappearance. Therefore, more and more improved LSTM models are proposed. For example, Gated Recurrent Unit (GRU) optimizes the network structure of LSTM, reduces 3 gating units to 2, increased training efficiency and improves the prediction accuracy. In 2016, Zhou et al. Proposed an improved bidirectional LSTM model for text classification[9]; in 2018, Li et al. Combined CNN and LSTM model, and also had significant results[10]; In 2020, Mirza et al. Introduced covariance into the gating structure to improve the learning performance of the LSTM network[11]. In the same year Liu et al. Combined the LSTM model with the ARIMA model to predict traffic flow[12].

The above mentioned improved LSTM models are all for the purpose of optimizing the network structure, but when the number of hidden layers is pretty large, the convergence speed of the LSTM model is slow, and the adjustment of its weights extremely depends on the gradient descent algorithm,



Content from this work may be used under the terms of the [Creative Commons Attribution 3.0 licence](https://creativecommons.org/licenses/by/3.0/). Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

which could only obtain locally optimal. Based on this, this paper proposes an optimized LSTM neural network model based on adaptive genetic algorithm (AGA-LSTM). The AGA-LSTM model uses the mean squared error of the output of the LSTM neural network, which is designed as an appropriate value function, and an adaptive genetic algorithm (AGA) is used to construct an optimization space, and the weights between the nodes of the LSTM model are globally optimized to improve the prediction performance. Finally, the experimental results on the classic UCI dataset also verify the rationality and effectiveness of the proposed AGA-LSTM neural network model.

2. Adaptive genetic algorithm and the LSTM neural network

2.1 Adaptive genetic algorithm

Genetic algorithm is a swarm intelligence algorithm proposed by Holland. It builds an optimization space by simulating the genetic mechanism of the biological world. It is widely used in solving various combinatorial optimization problems. Each chromosome in the genetic algorithm corresponds to an individual in the population, which represents a candidate solution for an optimization problem. However, the crossover probability and mutation probability of genetic algorithm are fixed, so the convergence speed is slow and premature phenomenon is easy to occur, and it cannot obtain the global optimal solution. And adaptive genetic algorithm (AGA) is an improved genetic algorithm proposed by Srinivas et al.[13] to overcome the randomness and blindness of traditional genetic algorithms in selection, crossover and mutation, and it is easier to obtain the global approximate optimal solution. AGA can make the crossover probability and mutation probability be dynamically adjusted with the change of the fitness function and the number of iterations, so as to retain the excellent individuals in the population and avoid the premature phenomenon.

The process of adaptive genetic algorithm includes the steps of initializing the population, calculating the individual fitness function of the population, selecting replication, adaptive crossover, adaptive mutation, and judging whether the iteration stops. The specific description is as follows:

- 1) Initialize the population space and set the relevant parameters of the algorithm.
- 2) Calculate the fitness function value of each individual(chromosome) in the population.
- 3) Select individuals to the crossover pool. Individuals (chromosomes) with the higher value of fitness function are selected and copied to the offspring to form new individuals. Individuals with a low fitness function are eliminated.
- 4) Adaptive crossover operation. A pair of chromosomes exchange information and recombine, that is, individuals in the population are randomly paired, and some chromosomes between the paired individuals are exchanged with adaptive cross probability.
- 5) Adaptive mutation operation. That is, each individual in the parent group will change some genes of an individual (chromosome) to other alleles with an adaptive probability.
- 6) Determine whether the iteration stops, and terminate the iteration if the conditions are met, otherwise skip to step (2).

The crossover probability and mutation probability in the adaptive genetic algorithm can be dynamically adjusted in iterations to maintain the diversity of the population. And the appropriate value function is the main basis for evaluating the merits of individuals, so the crossover probability and mutation probability will change with the change of the appropriate value.

Formulas (1) and (2) show the adaptive calculation formulas of crossover probability and mutation probability in the AGA. Among them, f_{avg} is the average fitness value of all individuals in the population, f_{max} is the largest fitness value of the individual, f' is the larger fitness value of the two individuals to be crossed, f is the fitness value of the mutated individual; k_1, k_2, k_3 and k_4 are parameters between 0 and 1, which are used to adjust the crossover probability and mutation probability.

$$P_c = \begin{cases} \frac{k_1(f_{\max} - f')}{f_{\max} - f_{\text{avg}}} & f' \geq f_{\text{avg}} \\ k_2 & f' < f_{\text{avg}} \end{cases} \quad (1)$$

$$P_m = \begin{cases} \frac{k_3(f_{\max} - f)}{f_{\max} - f_{\text{avg}}} & f \geq f_{\text{avg}} \\ k_4 & f < f_{\text{avg}} \end{cases} \quad (2)$$

The adaptive genetic algorithm mainly reduces the crossover probability and mutation probability at the early stage of the iteration by setting the values of k_1, k_2, k_3 and k_4 , to ensure the survival of better individuals; at the late stage of the iteration, when all individuals in the population tend to stabilize or tend to the case of local optimal, the crossover probability and mutation probability are increased to overcome the local optimal and find an approximate optimal solution globally.

2.2 The LSTM neural network

The LSTM neural network is mainly composed of three gated units, such as the forget gate, input gate and output gate. The unique gated units are used to learn and memorize the sequence data to maintain long-distance time series information dependence and achieve high-precision prediction. The standard LSTM neuron structure is shown in Figure 1.

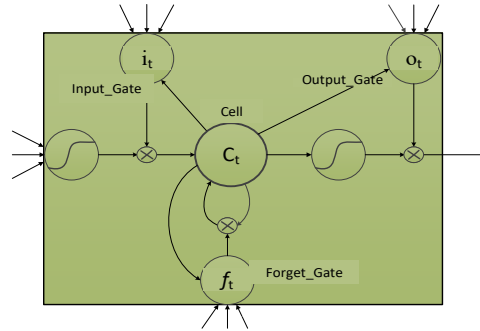


Figure 1. The structure of the LSTM neuron.

As shown in Figure 1, i_t , f_t , and o_t respectively represent the input gate, output gate, and forget gate. The input gate mainly processes input data. The forget gate determines the current neuron's retention of historical information. The output gate represents the output result of the neuron. Suppose the input sequence is $(x_1, x_2, \dots, x_t)$, then at time t , the calculation formula of each parameter of LSTM neuron is as follows:

$$i_t = S(W_i * [h_{t-1}, x_t] + b_i) \quad (3)$$

$$f_t = S(W_f * [h_{t-1}, x_t] + b_f) \quad (4)$$

$$o_t = S(W_o * [c_t, h_{t-1}, x_t] + b_o) \quad (5)$$

$$c_t = f_t * c_{t-1} + i_t * \tanh(W_c * [h_{t-1}, x_t]) \quad (6)$$

$$h_t = o_t * \tanh(c_t) \quad (7)$$

Where, i_t, f_t , and o_t are the input of the input gate, the forget gate, and the output gate at time t respectively; x_t is the input to the LSTM neuron at time t , h_{t-1} indicates the output state of the hidden layer at time $t-1$; W_i, W_f and W_o are respectively the weight matrix of the input and the input gate,

the forget gate and the output gate of the neuron at time t , b_i , b_f , and b_o are their corresponding offset vectors; W_c is the weight between the input and the cell unit; then h_t represents the output of the hidden layer at time t ; S is the Sigmoid function.

The deep LSTM neural network is based on the LSTM neural unit, and thus constructs the network model with multiple hidden layers, and continuously removes redundant information in the data set through the forget gates to maintain long-distance dependencies, so it has stronger predictive performance and better generalization ability.

2.2.1 An example. In this example we can see that there are footnotes after each author name and only 5 addresses; the 6th footnote might say, for example, ‘Author to whom any correspondence should be addressed.’ In addition, acknowledgment of grants or funding, temporary addresses etc might also be indicated by footnotes[5].

3. The AGA-LSTM neural network model

In this paper, the adaptive genetic algorithm is used to optimize the weights between the nodes of the LSTM neural network, which can make the weights between neurons more reasonable, and improve the generalization ability and prediction performance of the model. This chapter mainly describes the AGA-LSTM model in detail.

3.1 Chromosome coding

In this paper, the adaptive genetic algorithm is used to optimize the weights between the nodes of the LSTM neural network, which can make the weights between neurons more reasonable, and improve the generalization ability and prediction performance of the model. This chapter mainly describes the AGA-LSTM model in detail.

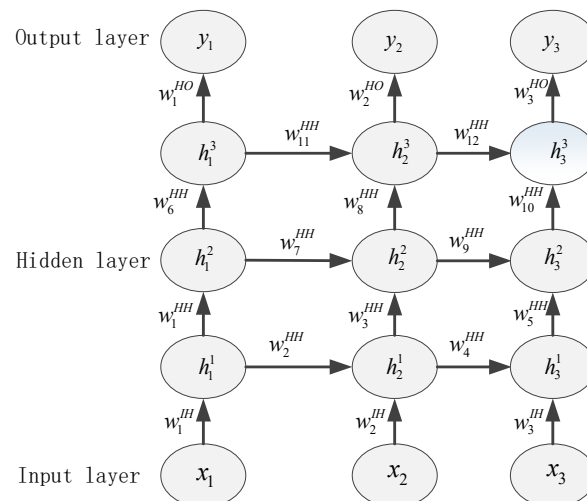


Figure 2. The structure of LSTM neural network with 3 hidden layers

A chromosome in the AGA-LSTM model corresponds to an individual in the population, and also represents a set of weight values between nodes in the LSTM model. The chromosomes of all individual are encoded with real numbers. Figure 2 shows the structure of LSTM model with 3 hidden layers, Figure 3 shows the coding method of a certain chromosome. Among them, w_1^{IH} represents the weight between the first neuron of the input layer and the first neuron of the hidden layer of the first layer, and so on. The coding method of chromosomes in this paper includes the weights between all neuron nodes, all of which are real numbers.

Chromosome	w_1^{IH}	...	w_3^{IH}	w_1^{HH}	...	w_{12}^{HH}	w_1^{HO}	...	w_3^{HO}
------------	------------	-----	------------	------------	-----	---------------	------------	-----	------------

Figure 3. The coding method of chromosomes

3.2 Fitness value function

In this paper, the adaptive genetic algorithm is used to optimize the weights between the nodes of the LSTM neural network, which can make the weights between neurons more reasonable, and improve the generalization ability and prediction performance of the model. This chapter mainly describes the AGA-LSTM model in detail.

The mean squared error, usually expressed by, is the mean squared error value on the test set of the global optimal solution obtained by the chromosome during the iteration process, and is also a suitable value function in the AGA-LSTM model. The calculation process is shown in (8).

$$MSE_i = \frac{1}{T_{num}} \sum_{j=1}^{T_{num}} (d_{ij} - y_{ij})^2 \quad (8)$$

Among them, num is the number of population; T_{num} is the number of test set; d_{ij} is the prediction result of LSTM model corresponding to chromosome i ; y_{ij} is the actual value.

3.3 The flow of AGA-LSTM model

In this paper, the adaptive genetic algorithm is used to optimize the weights between the nodes of the LSTM neural network, which can make the weights between neurons more reasonable, and improve the generalization ability and prediction performance of the model. This chapter mainly describes the AGA-LSTM model in detail.

The AGA-LSTM model maps the weight values between the nodes of the LSTM neural network through the adaptive genetic algorithm, and maps each weight value to a certain dimension of the chromosome, making the chromosome a solution set of candidate weights for the LSTM neural network. Afterwards, the adaptive genetic algorithm is used to establish the optimization space and iterate continuously to make the weights between the neurons more reasonable, improving the prediction performance and accuracy of the LSTM model.

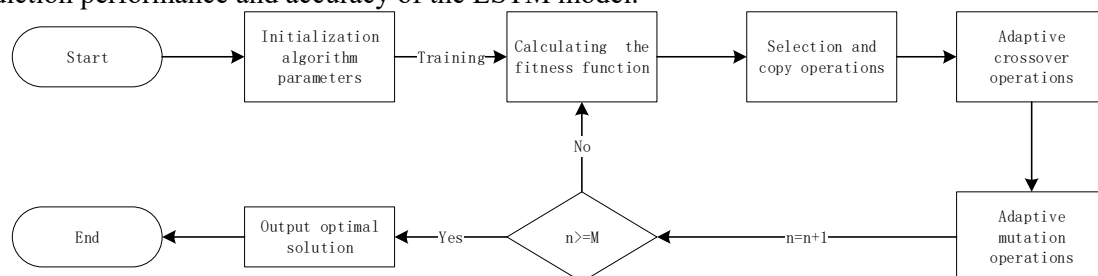


Figure 4. the flow of AGA-LSTM model

Figure 4 is the flowchart of the AGA-LSTM model optimized based on adaptive genetic algorithm. Let n be the number of iterations, M be the maximum number of iterations, and N be the number of chromosomes in the population. The main steps are as follows:

(1) Initialize the algorithms of the LSTM model and AGA. Such as the structure of LSTM model, the population of AGA, the number of iterations and other parameters.

(2) Train the LSTM neural network to get the default optimal weights.

(3) Use the formula (8) to calculate the fitness value function of each chromosome in the population, and use the adaptive genetic algorithm to construct the optimization space.

(4) Select the chromosomes to the crossover pool. The chromosomes with the fitness value ranking in the top $0.1 \cdot N$ are completely copied to the offspring to form new individuals, and the chromosomes with lower value of the fitness function are eliminated.

(5) According to formula (1), perform an adaptive crossover operation.

(6) According to formula (2), perform an adaptive mutation operation.

(7) Add 1 to the number of iterations n to determine whether n is greater than the maximum value M of iterations. If yes, terminate the iteration, otherwise skip to step (3).

(8) Output the globally optimal chromosome, which corresponds to the optimal weight distribution of the LSTM network model.

It should be noted that when using adaptive genetic algorithm to optimize the weights of the LSTM model, the selection operation adopts the elite selection method: the basic idea is to use the roulette selection method for selection operation, and at the same time The chromosomes with ranged the top $0.1 * N$ are completely copied to the next generation according to the original coding, N is the population number; in the crossover operation, that is, two new chromosomes are linearly combined after two paired chromosomes are generated. See the formula (1) for the crossover probability; In the adaptive mutation operation, non-uniform mutation is used. For the mutation probability, see the formula (2).

4. Experimental results and analysis

In this paper, the models based on deep GRU neural network, deep LSTM neural network and deep AGA-LSTM neural network are established respectively, and the performance of AGA-LSTM model is verified by comparing the prediction accuracy of each model. The experimental environment configuration is as follows: deep learning framework tensorflow1.10, the language is Python3.

4.1 Experimental data set

This article selects Air Quality (AQ), EEG, Dow Jones Index (DJI), Ozone Level Detection (OLD) in UCI database. Among them, the AQ data set records the air quality data of a region in Italy from 2014 to 2015, which is used to predict the air quality of the region; the DJI data set records the stock price data of the Dow Jones Index, which can be used to predict the stock price trend of a certain period in the future; The OLD records the ozone concentration on the ground, which can be used to predict the ozone concentration of the day, which belongs to the problem of time series prediction.

4.2 Establishing prediction models

The modeling steps of the neural network model include data preprocessing, training data set and verification model. Based on the above steps, this paper establishes prediction models based on GRU, LSTM and AGA-LSTM neural networks on three UCI data sets, such as AQ, DJI and OLD, to verify the effectiveness of the AGS-LSTM model. For the pre-processed data set, 70% of them is taken as the training set, 20% is taken as the verification set, and 10% is taken as the test set. The structure of the three network models established in this paper is the same, the input layer and the output layer are set to 1 layer, and the hidden layer is set to 6 layers. And, the error square sum SSE is used as the index to predict the performance of the test model, as shown in formula (9):

$$SSE = \sum_{i=1}^{n_{test}} (y'_i - y_i)^2 \quad (9)$$

Among them, n_{test} is the number of test sets, y'_i is the prediction result of the model, y_i is the true value. The smaller the SSE value, the higher the prediction accuracy of the model.

4.3 Analysis of experimental results

Table 1. SSE values of three models under different data sets.

Data set	The neural network model		
	GRU	LSTM	AGA-LSTM
AQ	287.3	322.8	245.6
DJI	276.1	302.1	251.2
OLD	153.4	192.5	134.4
Average SSE value	238.9	272.5	210.4

After the training of the neural network model is completed, the test sets of the above three data sets are sequentially input to each model, and the comparison results of the experiment can be obtained. Table 1 shows the SSE values of the three models of GRU, LSTM and AGA-LSTM under SSD, MDTD, OLD and other data sets. Figure 5 is a histogram drawn according to Table 1.

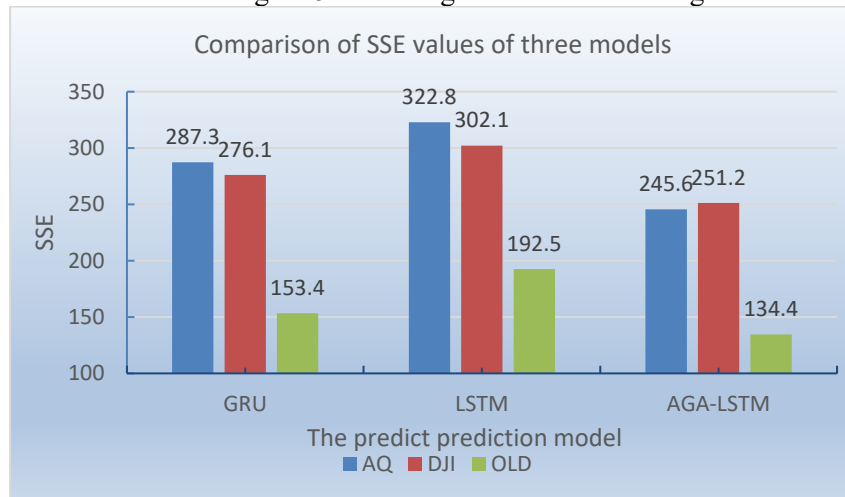


Figure 5. Comparison of SSE values of three models

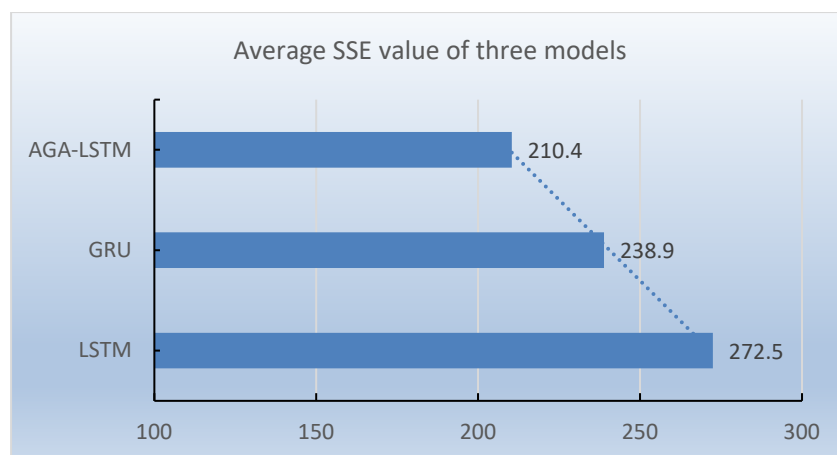


Figure 6. The average SSE values of the three models

As can be seen from Table 1 and Figure 5, the SSE values of the LSTM model under the three different data sets are the largest, followed by the GRU model, and the SSE values of the AGA-LSTM model are the lowest on each data set, indicating that the model has the highest prediction accuracy, and its generalization ability and prediction performance are best.

Figure 6 shows the trend change graph of the average SSE value of the three models under each data set. The average SSE value of the GRU model is 12.3% lower than that of the LSTM model, while the average SSE value of the AGA-LSTM model is 11.9% lower than that of the LSTM model and 22.8% lower than that of the LSTM model. Obviously, the AGA-LSTM model has the smallest error and the best prediction accuracy. This proves the effectiveness of the AGA-LSTM model.

5. Conclusion

In this paper, an LSTM neural network model based on adaptive genetic algorithm optimization is proposed to solve the problems of slow convergence speed and the emerging of local extremum in the weights adjustment of the LSTM neural network. The experimental results show that, compared with the traditional LSTM and GRU models, the AGA-LSTM model has the lowest average SSE value and the smallest prediction errors under the three UCI data sets. Therefore, the prediction performance and

generalization ability of the AGA-LSTM model are superior to the LSTM model and the GRU model, which not only verifies the effectiveness of the model, but also provides an optimization idea of neural networks.

Acknowledgments

This work was supported by the Shanghai Customs College scientific research start-up project on Nano Sensor in Anti-Smuggling of Inland River (No. 2314050A17).

References

- [1] Wu J, Long J, Liu M. Evolving RBF neural networks for rainfall prediction using hybrid particle swarm optimization and genetic algorithm[J]. *Neurocomputing*, 2015, 148: 136-142.
- [2] Podani, J. (1994) *Multivariate Data Analysis in Ecology and Systematics*. SPB Publishing, The Hague.
- [3] Park S H, Kim B D, Kang C M, et al. Sequence-to-sequence prediction of vehicle trajectory via LSTM encoder-decoder architecture[C]//2018 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2018: 1672-1678.
- [4] Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks[C]//Advances in neural information processing systems. 2014: 3104-3112.
- [5] Lev G, Sadeh G, Klein B, et al. Rnn fisher vectors for action recognition and image annotation[C]//European Conference on Computer Vision. Springer, Cham, 2016: 833-850.
- [6] Qi M, Qin J, Li A, et al. stagnet: An attentive semantic RNN for group activity recognition[C]//Proceedings of the European Conference on Computer Vision (ECCV). 2018: 101-117.
- [7] Selvin S, Vinayakumar R, Gopalakrishnan E A, et al. Stock price prediction using LSTM, RNN and CNN-sliding window model[C]//2017 international conference on advances in computing, communications and informatics (icacci). IEEE, 2017: 1643-1647.
- [8] Hochreiter S, Schmidhuber J. Long short-term memory[J]. *Neural computation*, 1997, 9(8): 1735-1780.
- [9] Zhou P, Qi Z, Zheng S, et al. Text classification improved by integrating bidirectional LSTM with two-dimensional max pooling[J]. *arXiv preprint arXiv:1611.06639*, 2016.
- [10] Li C , Zhan G , Li Z . News Text Classification Based on Improved Bi-LSTM-CNN[C]// 2018 9th International Conference on Information Technology in Medicine and Education (ITME). IEEE Computer Society, 2018.
- [11] Mirza A H, Kerplici M, Kozat S S. Efficient online learning with improved LSTM neural networks[J]. *Digital Signal Processing*, 2020: 102742.
- [12] Liu B, Tang X, Cheng J, et al. Traffic flow combination forecasting method based on improved LSTM and ARIMA[J]. *International Journal of Embedded Systems*, 2020, 12(1): 22-30.
- [13] Srinivas M, Patnaik L M. Adaptive probabilities of crossover and mutation in genetic algorithms[J]. *IEEE Transactions on Systems, Man, and Cybernetics*, 1994, 24(4): 656-667.