Article

Short-Term Traffic Flow Forecasting Based on Data-Driven Model

Su-qi Zhang ^{1,*} and Kuo-Ping Lin ² ¹ Department of computer science, Tianjin University of Commerce, Tianjin 300134, China² Department of Industrial Engineering and Enterprise Information, Tunghai University, Taichung 40704, Taiwan; kplin@thu.edu.tw or hal_linkimo@yahoo.com.tw

* Correspondence: zhangsuqilaoshi@126.com

Received: 20 November 2019; Accepted: 12 January 2020; Published: 21 January 2020

Abstract: Short-term traffic flow forecasting is the technical basis of the intelligent transportation system (ITS). Higher precision, short-term traffic flow forecasting plays an important role in alleviating road congestion and improving traffic management efficiency. In order to improve the accuracy of short-term traffic flow forecasting, an improved bird swarm optimizer (IBSA) is used to optimize the random parameters of the extreme learning machine (ELM). In addition, the improved bird swarm optimization extreme learning machine (IBSAELM) model is established to predict short-term traffic flow. The main researches in this paper are as follows: (1) The bird swarm optimizer (BSA) is prone to fall into the local optimum, so the distribution mechanism of the BSA optimizer is improved. The first five percent of the particles with better fitness values are selected as producers. The last ten percent of the particles with worse fitness values are selected as beggars. (2) The one-day and two-day traffic flows are predicted by the support vector machine (SVM), particle swarm optimization support vector machine (PSOSVM), bird swarm optimization extreme learning machine (BSAELM) and IBSAELM models, respectively. (3) The prediction results of the models are evaluated. For the one-day traffic flow sequence, the mean absolute percentage error (MAPE) values of the IBSAELM model are smaller than the SVM, PSOSVM and BSAELM models, respectively. The experimental analysis results show that the IBSAELM model proposed in this study can meet the actual engineering requirements.

Keywords: data-driven; short-term traffic flow; intelligent optimizer; extreme learning machine; traffic flow forecasting

1. Introduction

As a new type of technology, the intelligent transportation system combines theories of the sensor, data communication, data processing technology, artificial intelligence and automatic control [1,2]. The Intelligent Transportation System (ITS) has become a research hotspot all over the world. Japan is vigorously developing smart cars and autonomous driving technologies, and wants to build a world-class transportation system by 2020. In 2012, the number of intelligent transportation projects in China exceeded 230, with a total investment of more than 10 million yuan. Additionally, with the increase of time, the number of intelligent transportation projects and total investments are increasing [3]. Short-term traffic flow forecasting is the basic content of a traffic intelligence system. By predicting traffic flow, it can provide real-time data support for this intelligent traffic system, which is beneficial to the system to more efficiently plan the route and realize traffic congestion warnings. Therefore, it can achieve the purpose of alleviating traffic congestion, and also reduce air pollution and improve driving efficiency [4,5].

The Intelligent Transportation System is a huge and complex system that changes with time. Yang et al. [6] proposed a neural network method to predict traffic flow. This method adopts an

optimization framework of a traffic flow prediction model based on in-depth learning, and improves the efficiency of intelligent traffic management by improving the prediction accuracy of the model. Driving behavior analysis is an important part of the intelligent transportation system. He et al. [7] proposed a method to identify driving style and detect abnormal driving. The method uses phase space reconstruction technology to reconstruct the original vehicle data trajectory, and uses a convolution neural network to extract feature vectors. Traffic congestion prediction is a basic technology for implementing intelligent transportation systems. Zhang et al. [8] proposed a symmetric layer neural network model based on deep self-coding to predict traffic congestion. The capacity and efficiency of the transportation network are improved by predicting traffic congestion. With the gradual popularization of intelligent transportation projects, intelligent transportation systems are suffering from more complex, structured query language (SQL) injection attacks, and traditional detection methods have been unable to cope with this challenge. Li et al. [9] proposed a method for detecting SQL injection attacks. This method can cope with high-dimensional massive data, and solve the over-fitting problem caused by insufficient samples.

The main research contents of this thesis are as follows: (1) The performance of the BSA optimizer is improved, and the optimization performance of the IBSA optimizer is tested by the optimization functions; (2) the support vector machine (SVM), particle swarm optimization support vector machine (PSOSVM), bird swarm optimization extreme learning machine (BSAELM) and improved bird swarm optimization extreme learning machine (IBSAELM) are all used to predict the traffic flow for one day and two days, respectively. (3) The prediction results of the models are evaluated by using absolute error (AE), root mean square error (RMSE), mean absolute percentage error (MAPE) and decision coefficient (r^2) indicators. This research has the following three contributions: (1) This study proposes the IBSA optimizer with better search ability. The IBSA optimizer can not only realize extreme learning machine (ELM) parameters optimization, but can also be applied to various fields; (2) This study proposes the IBSAELM model to forecast short-term traffic flow, and test experiments show that the IBSAELM model has a higher predictive performance; (3) By improving the short-term traffic flow prediction accuracy of the model to provide reliable data support for the intelligent transportation system.

The remaining chapters of this paper are organized as follows. Section 2 introduces the literatures about short-term traffic flow methods and introduces the method proposed in this study. Section 3 introduces a short-term traffic flow prediction model. Section 4 analyzes the prediction results of the IBSAELM model for short-term traffic flow. Section 5 gives the findings and conclusions of this study.

2. Literature Review

This section describes the literatures on short-term traffic flow methods and the method presented in this paper.

2.1. Short-term Traffic Flow Forecasting Methods

The rapid development of the economy has led to an increasing demand for transportation, and the increasing demand for transportation has caused certain negative impacts, such as widespread traffic congestion and environmental pollution problems [10]. Short-term traffic flow prediction mainly refers to forecasting traffic flow within 15 minutes [11]. Traffic flow forecasting is an important part of the intelligent transportation system. By providing real-time traffic data information for traffic drivers, it is more efficient to select the optimal route and reduce the loss of time and money due to traffic congestion.

At present, traffic flow prediction methods mainly include statistical theory prediction models, machine learning prediction models, and combined prediction models [12–14]. The autoregressive (AR) model describes the relationship between the current value and the historical value; the moving average (MA) model describes the error cumulant of the AR model, eliminating the random fluctuation of the prediction process; the ARMA model is the combination of the AR model and the MA model; the autoregressive integrated moving average (ARIMA) model is based on the autoregressive moving

average (ARMA) model. Some time series are periodic and trendy. The periodicity of the series is caused by seasonal changes or intrinsic factors. When the time series is periodic, this series is a seasonal time series. Therefore, seasonal differences can be used to smooth this seasonal time series. This time series can be predicted by the seasonal autoregressive integrated moving average (SARIMA) model. The SARIMA model is a short-term forecasting model formed by adding the seasonal variation of the time series on the basis of the original ARIMA model. It has strong linear modeling capabilities. When there are obvious time trends and seasonal changes in the time series, compared with the ARIMA model, the prediction effect of the SARIMA model is better. Moreover, the SARIMA model does not need to make prior assumptions about the development model of the time series. Therefore, compared with many other prediction model methods, this SARIMA model has a higher prediction accuracy. The introduction of differential operations to convert non-stationary data into stationary data is currently the most widely used time series prediction model. Williams and Hoel [15] proposed a seasonal ARIMA process to forecast traffic flow. The author compares the seasonal ARIMA process model with three heuristic methods, and the test results show that the prediction error of the seasonal ARIMA process model is the smallest. Smith et al. [16] used a nonparametric regression method to forecast traffic flow and compared the prediction effect of this same nonparametric regression method with the seasonal ARIMA method. The results show that the method based on the combination of heuristic and nonparametric regression has a greater application prospect. Tselentis et al. [17] compared the prediction effect of a single time series model and combined a model based on Bayes and statistics on short-term traffic flow. Compared with the combined model, the cost of the single model is lower, but the prediction risk of the combined model is lower. Karlaftis, and Vlahogianni [18] compared the neural network model with the statistical method in the field of traffic flow prediction. Neural network has advantages for data missing and nonlinear problems. Vlahogianni and Karlaftis [19] analyzed the effects of smooth weather and non-stationary weather on traffic flow and proposed a nonlinear dynamic method. The differences caused by weather factors need to be considered in short-term traffic flow forecasting. The statistical regression model is based on a time series, and when the nonlinearity of the time series increases, the prediction error of the model increases. Compared with the statistical regression method, the machine learning model has a stronger nonlinear mapping ability. Li et al. [20] proposed a method based on deep feature learning to predict short-term traffic flow. In this method, the particle swarm optimizer is improved, and the multi-objective particle swarm optimizer is proposed. The improved particle swarm optimizer is used to optimize the hyperparameters of the deep belief network. Compared with ELM, the learning speed of the deep belief network is slower and more sensitive to the selection of parameters. Zhu et al. [21] found that the traffic flow of two adjacent intersections affected each other by calculating the correlation coefficient and used the RBF neural network to predict the traffic flow. Through comparison, it is found that the RBF neural network considering the influence of adjacent intersections has the higher prediction accuracy. The gradient descent method is used in the artificial neural network, which is easy to cause the network to fall into the local minimum. Vlahogianni et al. [22] proposed a neural network optimization strategy which is based upon a genetic algorithm. This optimization strategy can improve the performance of the neural network. The neural network with optimized structure is used to predict the traffic, and the better prediction effect is achieved. Wang et al. [23] proposed a short-term memory regression neural network to predict short-term traffic flow. In this method, the traffic flow sequence is first decomposed into the residual sequence and the trend sequence, and the two traffic sequences decomposed are predicted by the model respectively, and the prediction results are synthesized to form the final result. This method can better realize traffic flow prediction on multiple time scales. By decomposing the dimension of the time series, then using the model to predict separately, and finally synthesizing the prediction results, this method can improve the prediction accuracy, but also increase the calculation amount. Diao et al. [24] proposed a hybrid approach to predict traffic flow. This method takes into account many factors, such as time, frequency, origin, destination, and so on. The wavelet time transformation is used to decompose the traffic time series, and the Gaussian model is used to predict the specific

components. This Gaussian model can fit adaptively, but its prediction efficiency is poor when dealing with high-dimensional problems.

2.2. The Method Proposed in This Study

The extreme learning machine is used in various fields due to its strong generalization and learning ability. Mariani et al. [25] applied ELM to the field of single cylinder engine pressure prediction. In order to improve the prediction accuracy of ELM, the biogeography-based optimizer (BBO) is improved, and the improved BBO optimizer is used to optimize the ELM prediction model. The experimental results show that the optimized ELM model can achieve an effective prediction of the average pressure. Li et al. [26] established the whale optimization extreme learning machine (WOA-ELM) model to evaluate the aging state of insulated gate transistors. In this method, the ELM model is optimized by the whale optimizer. The experimental results show that the WOA-ELM model is superior to the genetic algorithm optimization extreme learning machine (GA-ELM), the crow algorithm optimization extreme learning machine (CS-ELM) model and the dandelion algorithm optimization extreme learning machine (DA-ELM) model. Bui et al. [27] proposed the PSO-ELM model to predict flash floods. Firstly, the initial flood model is established by ELM, and then the PSO optimizer is used to optimize the ELM model. The analysis results show that the method is suitable for areas with frequent tropical typhoons, and an early warning of mountain torrents can be provided by forecasting. Kourehli et al. [28] used ELM to predict the location and depth of Timoshenko beam cracks at different motion masses. This method reduces the sensitivity to noise, and can still identify the severity of the crack in the case of noise.

In order to improve the prediction accuracy of the model, this paper improves the optimization performance of the BSA optimizer, and proposes the IBSAEM prediction model to forecast the short-term traffic flow. In the BSA optimizer, the positions of producers and beggars are randomly assigned, which causes birds with large food reserves in the flock to be randomly assigned to beggars, and birds with small food reserves are randomly assigned to beggars, thus reducing the speed of bird optimization. Aiming at the defect of the BSA optimizer, this study improves the particle allocation mechanism of the BSA optimizer. Firstly, the particles are sorted according to the fitness value. Then 5% of the particles with a good fitness value are selected as producers, 10% of the particles with poor fitness values are selected as beggars, and the remaining particles are selected randomly. In order to solve the problem that beggars are easy to fall into local optimal values in the BSA optimizer, this study improves the position rule of beggars to enhance the convergence ability of particles. Finally, the global and local convergence ability of the optimizer is enhanced by the nonlinear weight coefficient.

3. Short-term Traffic Flow Forecasting Model

3.1. Bird Swarm Optimizer (BSA) and Improved Bird Swarm Optimizer (IBSA)

In recent years, bionic optimizers have been proposed, such as a bird flock optimizer and a chicken flock optimizer [29]. Meng et al. [30] proposed the Bird Swarm Optimizer (BSA) in 2016. BSA is a new type of optimizer. BSA imitates the foraging, migrating and alert behavior of birds.

Compared with particle swarm optimization (PSO) and differential evolution (DE), BSA has faster convergence speed and higher convergence accuracy. The BSA optimizer follows the following rules [31–33].

Rule 1: Each particle in a flock can randomly select alert behavior and foraging behavior.

Rule 2: Information in the flock can be shared, that is, the best location information of each bird can be shared in the flock. Birds update the optimal position of the population according to the information of individual particles.

Rule 3: Birds with alert information are more likely to fly to the center of the group, and individuals with better fitness are more likely to fly to the center of the group. In the process of flying to the center of the group, every particle has competitive behavior.

Rule 4: Birds migrate regularly. When birds migrate to their destination, each particle in the group chooses between the roles of producer and beggar.

Rule 5: The particles with the best fitness value are selected as producers, the particles with the lowest fitness value are selected as beggars, and the remaining particles are randomly selected between producers and beggars.

Rule 6: Producers dominate the flock of birds, and the beggars follow the producers to find food.

Rule 1 is a process of random decision. Set the random number y , when y is less than $rand(0, 1)$, the bird group particles choose the foraging behavior; and then when y is greater than $rand(0, 1)$, the bird group particles choose the warning behavior.

Rule 2 can be described by the following mathematical formulas:

$$pos_i^j(t+1) = pos_i^j(t) + W \times rand \times (k_i^j - pos_i^j(t)) + \dots + C \times rand \times (bestpos - pos_i^j(t)) \quad (1)$$

where $pos_i^j(t)$ is the position of the t -th iteration of the bird group particles; W is the local learning coefficient; C is the global cognitive coefficient; $rand$ is the random number between 0 and 1; k_i^j is the optimal individual position in the j -th dimensional space; $bestpos$ represents the optimal position in the population.

The competitive behavior of the birds flying to the center of the population in Rule 3 is simulated by the following mathematical formulas:

$$\begin{aligned} pos_i^j(t+1) &= pos_i^j(t) + S1 \times rand \times (pos_{mean}^j - pos_i^j(t)) + \dots \\ &\quad + S2 \times randn \times (pos_u^j - pos_i^j(t)) \\ S1 &= s1 \times \exp\left(-\frac{L \times posfit_i}{sumposfit + \varepsilon}\right) \\ S2 &= s2 \times \exp\left(-\frac{posfit_i - posfit_u}{|posfit_i - posfit_u| + \varepsilon} \times \frac{L \times posfit_i}{sumposfit + \varepsilon}\right) \end{aligned} \quad (2)$$

where $S1$ represents the indirect effect of the individual particles in the bird population flying towards the center of the flock; pos_{mean}^j represents the average of the j -th position; $S2$ indicates the direct impact of specific competition in the process of individual particles flying to the center of a flock; L represents the number of particles in the flock; $posfit_i$ represents the fitness value of the i -th particle; $posfit_u$ ($u \in [1, L], u \neq i$) represents the fitness value of the u -th particle; $sumposfit$ represents the sum of the fitness values of the flock; ε represents an infinitesimal amount.

Simulate the positions of producers and beggars in rules 4, 5 and 6 using Equations (3) and (4):

$$pos_i^j(t+1) = pos_i^j(t) + rand \times pos_i^j(t) \quad (3)$$

$$pos_i^j(t+1) = pos_i^j(t) + Fol \times rand \times (pos_u^j(t) - pos_i^j(t)) \quad (4)$$

where $Fol \in [0, 1]$ represents the probability that the beggar follows the producer for foraging.

In the BSA optimizer, the producer is the main force to find food in the bird flock, which determines the global optimization ability of this same BSA optimizer. The beggar keeps the diversity of the bird flock, which determines the local optimization ability of BSA optimizer.

Beggars and producers are basically randomly selected, and there are some problems in such a distribution mechanism. Because of random distribution, particles with higher food reserves may be selected as beggars, and particles with lower food reserves may be selected as producers, which will increase the number of iterations of the BSA optimizer and increase the computational cost of the BSA optimizer.

In this study, the BSA optimizer is improved to solve the above problems. When the birds migrate to their destination, the bird population particles are classified. The first 5% of the particles with high food reserves are selected as producers. The last 10% of the particles with lower food reserves are

selected as beggars. The remaining particles randomly become producers and beggars. In this way, the population diversity of the BSA optimizer can be guaranteed and the optimization speed of the BSA optimizer can be accelerated. Beggars have poor productivity in the BSA optimizer and are prone to fall into the minimum in the process of optimization. In order to improve the productive capacity of beggars, the part of learning from the best particle of birds is added to the position of beggars. In addition, in order to accelerate the convergence speed of the BSA optimizer, the nonlinear weight coefficient d is introduced into the producer update formula.

The above behavior is described by Equation (5):

$$\begin{cases} \text{Producer } i > 5\% \times L \& i \leq 90\% \times L \& a \leq rand \text{ or } i \leq 5\% \times L \\ \quad pos_i^j(t+1) = d \times pos_i^j(t) + rand \times pos_i^j(t) \\ \text{Beggari } > 5\% \times L \& i \leq 90\% \times L \& a > rand \text{ or } i > 90\% \times L \\ \quad pos_i^j(t+1) = pos_i^j(t) + Fol \times rand \times (pos_u^j(t) - pos_i^j(t)) + \dots \\ \quad ptLn \times (bestpos - pos_i^j(t)) \end{cases} \quad (5)$$

$$Ln = e^{(bestposfit - posfit_i)} \quad (6)$$

$$d = d_{\max} + (d_{\max} - d_{\min}) \times \exp\left(-\frac{t}{T}\right)^3 \quad (7)$$

where d ($d_{\max} = 0.9, d_{\min} = 0.2$) is the weight coefficient; t is the current iteration number; T is the maximum number of iterations; Ln is the adaptive learning coefficient; a is a random number.

By enhancing the convergence ability of the IBSA optimizer, the optimal parameters of the ELM model can be better mined, thereby improving the prediction accuracy of the model for traffic flow.

3.2. Basic Principles of Extreme Learning Machine

The single hidden layer feedforward neural network (SLFN) has a simple structure and strong learning ability, so SLFN has been widely used in many fields. The learning method adopted by most SLFNs is the gradient descent method. However, the gradient descent method requires multiple iterations, resulting in high computational cost and low learning efficiency, which has a great influence on the convergence speed of the SLFN. A new predictive model, the Extreme Learning Machine (ELM), is proposed for the shortcomings of SLFN. Compared with traditional SLFN, ELM learns faster and converges faster [25,26].

The network topology of ELM is similar to that of SLFN. It adopts a unidirectional three-layer structure, each layer consists of multiple neurons and there is no connection between the same layer of neurons. The first layer of the network topology is the input layer, the last layer is the output layer, and the middle layer is the hidden layer. Let the number of three layers of ELM neurons be o , p and q [28]. The connection weights between the first layer and the middle layer is $\mathbf{e}$:

$$\mathbf{e} = \begin{bmatrix} e_{11} & e_{12} & \cdots & e_{1o} \\ e_{21} & e_{22} & \cdots & e_{2o} \\ \vdots & \vdots & \vdots & \vdots \\ e_{p1} & e_{p2} & \cdots & e_{po} \end{bmatrix}_{p \times o} \quad (8)$$

where $e_{i,j}$ ($i = 1, 2, \dots, p; j = 1, 2, \dots, o$) is the connection weight between the i -th neuron of the first layer and the j -th neuron of the intermediate layer.

The connection weight between the second and third layers is r [34,35]:

$$r = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1q} \\ r_{21} & r_{22} & \cdots & r_{2q} \\ \vdots & \vdots & \vdots & \vdots \\ r_{p1} & r_{p2} & \cdots & r_{pq} \end{bmatrix}_{p \times q} \quad (9)$$

where r_{ij} ($i = 1, 2, \dots, p; j = 1, 2, \dots, q$) is the connection weight between the i -th neuron of the middle layer and the j -th neuron of the last layer.

Let the threshold between the middle layer and the last layer be $v = [v_1, v_2, \dots, v_o]^T$. The input and output matrices of b sample sets are M and N , respectively.

$$M = \begin{bmatrix} m_{11} & m_{12} & \cdots & m_{1b} \\ m_{21} & m_{22} & \cdots & m_{2b} \\ \vdots & \vdots & \vdots & \vdots \\ m_{o1} & m_{o2} & \cdots & m_{ob} \end{bmatrix}_{o \times b} \quad N = \begin{bmatrix} n_{11} & n_{12} & \cdots & n_{1b} \\ n_{21} & n_{22} & \cdots & n_{2b} \\ \vdots & \vdots & \vdots & \vdots \\ n_{q1} & n_{q2} & \cdots & n_{qb} \end{bmatrix}_{q \times b} \quad (10)$$

where m is the input quantity and n is the output quantity.

The activation function enables the ELM to solve nonlinear problems. The activation function used in this paper is the Sigmoid function:

$$Z(m) = \frac{1}{1 + e^m} \quad (11)$$

Thus the output of the ELM network is obtained.

$$h_j = \begin{bmatrix} h_{1j} \\ h_{2j} \\ \vdots \\ h_{qj} \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^p r_{i1} Z(e_i m_j + v_i) \\ \sum_{i=1}^p r_{i2} Z(e_i m_j + v_i) \\ \vdots \\ \sum_{i=1}^p r_{iq} Z(e_i m_j + v_i) \end{bmatrix} \quad (j = 1, 2, \dots, b) \quad (12)$$

The ELM network output H is the product of the hidden layer output matrix G and the connection weight r :

$$H^T = Gr, G = \begin{bmatrix} Z(e_1 \cdot m_1 + v_1) & Z(e_2 \cdot m_1 + v_2) & \cdots & Z(e_p \cdot m_1 + v_p) \\ Z(e_1 \cdot m_2 + v_1) & Z(e_2 \cdot m_2 + v_2) & \cdots & Z(e_p \cdot m_2 + v_p) \\ \vdots & \vdots & \cdots & \vdots \\ Z(e_1 \cdot m_b + v_1) & Z(e_2 \cdot m_b + v_2) & \cdots & Z(e_p \cdot m_b + v_p) \end{bmatrix}_{b \times p} \quad (13)$$

where H^T is the transposed matrix of H .

When the activation function Z of the ELM is infinitely divisible, it is not necessary to adjust all of the parameters of the ELM. The weight e and the threshold v in the ELM model can be randomly initialized, and remain unchanged during the training. The weight r between the intermediate layer and the output layer can be obtained by the following least squares solution.

$$\min_r \|Gr - H^T\| \quad (14)$$

where $r = G^+ H^T$ can be obtained by finding the least squares solution; G^+ is the generalized inverse matrix.

In this study, the ELM model is used as the basic model of traffic flow prediction. Through the above analysis of the principle of this ELM model, it is found that the super parameters of the ELM model are randomly selected. The choice of super parameters has a great influence on the prediction effect of traffic flow. Therefore, this paper uses the IBSA optimizer to optimize the super parameters of the ELM model to achieve a good effect on traffic flow prediction.

3.3. IBSA Optimization Effect Analysis

In order to analyze the optimization performance of the IBSA optimizer, the BSA and IBSA optimizers are used to optimize the standard functions, and the optimization performance of the two optimizers is compared. In order to make the optimization results more convincing, two optimizers respectively search for the standard function 50 times, and the optimization dimension is set to 30. The optimization performance test of the two optimizers uses a unified simulation platform.

The BSA and IBSA optimizer parameters are set as follows: the number of populations L is 20, the number of cycles T is 500, the migration frequency Q is 10, the local learning coefficient W and the global cognitive coefficient S are 2. The standard functions are shown in Table 1.

Table 1. Standard optimization functions.

Equation	Minimum	Optimizing Boundary
$F1 = \sum_{k=1}^L l_k^2$	0	$-100 \leq l \leq 100$
$F2 = \sum_{k=1}^L l_i + \prod_{k=1}^L l_i$	0	$-10 \leq l \leq 10$
$F3 = \sum_{k=1}^L (l_k^2 - 10 \times \cos(2 \times \pi \times l_k)) + 10 \times T$	0	$-5.12 \leq l \leq 5.12$
$F4 = \sum_{k=1}^L \frac{l_k^2}{4000} - \prod_{k=1}^L \cos(\frac{l_k}{\sqrt{k}}) + 1$	0	$-600 \leq l \leq 600$

The optimization results of the BSA and IBSA optimizers are as follows.

Table 2 shows the optimization results of the BSA and IBSA optimizers from four aspects: the optimal value, the occurrence times of optimal value for 50 iterations, the running time of 50 iterations and the average running time of each iteration. For $F1$ and $F2$, the BSA optimizer cannot find the optimal value of 0, and the local optimal values are $1.2394\text{e-}284$ and $3.9898\text{e-}143$, respectively. For the BSA optimizer, the number of times to find the local optimal value are 9 times and 10 times in the 50 iterations, respectively.

Table 2. Optimization results.

F	Name	Optimum Value of 50 Searches	Number of Occurrences of Optimal Values	Average Value	Time of 50 Iterations/s	Average Running Time per Iteration
F1	BSA	$1.2394\text{e-}284$	9	$1.1448\text{e-}253$	16.9573	0.3391
	IBSA	0	28	$7.6166\text{e-}298$	15.3817	0.3076
F2	BSA	$3.9898\text{e-}143$	10	$2.1716\text{e-}127$	17.5345	0.3507
	IBSA	$1.9202\text{e-}180$	17	$5.8248\text{e-}155$	13.5409	0.2708
F3	BSA	0	50	0	17.2381	0.3447
	IBSA	0	50	0	13.0729	0.2615
F4	BSA	0	50	0	18.1897	0.3638
	IBSA	0	50	0	16.1929	0.3239

For $F1$, the IBSA optimizer converges to the optimal value 0 and 28 times to the global optimal value in 50 iterations. The average convergence accuracy of IBSA optimizer is $7.6166\text{e-}298$, and the search accuracy is higher than that of BSA optimizer. For $F2$, the optimal convergence value of the IBSA optimizer is $1.9202\text{e-}180$, and the average convergence value is $5.8248\text{e-}155$. The search accuracy of the IBSA optimizer is also higher than that of the BSA optimizer.

For F3 and F4, the BSA optimizer and IBSA optimizer converge to global optimal value 0, and both optimizers converge to global optimal value 0 in each iteration process. For the four optimization functions, the IBSA optimizer consumes 1.5756 s, 3.9936 s, 4.1652 s and 1.2607 s less time than the BSA optimizer in 50 iterations. Similarly, the running time of each iteration of the IBSA optimizer are 0.0315 s, 0.0799 s, 0.0832 s and 0.0399 s less than that of BSA optimizer, respectively.

Through the analysis of the optimization data of both BSA and IBSA optimizers, it is found that IBSA has better optimization accuracy and speed than the BSA optimizer. By improving the distribution mechanism of birds, the searching ability of the whole bird group is improved, and the nonlinear weight coefficient makes the bird group approach the optimal value faster. For four optimization functions, the optimization curves of BSA and IBSA optimizers are shown in Figure 1.

Figure 1. The optimization curves of BSA and IBSA optimizers. (a) Optimization curves for F1; (b) Optimization curves for F2; (c) Optimization curves for F3; (d) Optimization curves for F4.

From Figure 1c,d, we can see that both the BSA optimizer and IBSA optimizer converge to the global optimal value 0, but the IBSA optimizer searches the optimal value at the 21th and 34th, respectively, and the BSA optimizer searches the optimal value at the 25th and 44th, respectively. By comparing the convergence curves in the graph, it is found that the search speed of the IBSA optimizer is faster than that of the BSA optimizer.

Through the above analysis, it is found that compared with the BSA optimizer, the convergence accuracy and convergence time of the IBSA optimizer are significantly improved. Therefore, the IBSA optimizer can achieve better optimization of the model hyperparameters, thereby improving the model's prediction effect on traffic flow.

4. Short-term Traffic Flow Forecasting and Prediction Effect Analysis

4.1. IBSAELM Short-term Traffic Flow Forecasting Model

In the training process of the ELM model, the connection weight between the input layer and the hidden layer and the threshold between the hidden layer and the output layer are randomly determined. During the training process, the number of neurons in the hidden layer of the ELM is increased, resulting in an increase in the computational cost of the ELM model. In order to reduce the influence of random parameters on the prediction effect of ELM model, this study uses IBSA to optimize the random parameters of the ELM model to improve the prediction effect of this ELM model.

The specific process of the IBSAELM model for predicting short-term traffic flow is as follows:

- (1) Divide the short-term traffic flow samples into model test samples and model training samples.
- (2) Normalize the traffic flow samples.
- (3) Set the parameters of the BSA optimizer, such as the number of cycles T , the number of birds group L , and so on.
- (4) The bird group position is updated according to each rule, and the fitness value is calculated.
- (5) Start the iteration and use the IBSA optimizer to optimize the random parameters of the ELM model.
- (6) Input the parameters of the IBSA optimization into the ELM model.
- (7) The short-term traffic volume is predicted using the IBSAELM model.
- (8) Analyze the prediction results of the IBSAELM model.

The IBSAELM model prediction flow chart is shown in Figure 2.

Figure 2. Short-term traffic flow prediction.

4.2. Short-term Traffic Flow Forecasting Based on IBSAELM Model

The traffic flow data in this study were collected from an intersection in Tianjin, China. The traffic flow of the intersection was recorded every 15 minutes. For five days, 480 traffic flow data of the intersection were collected. The 5-day traffic flow sequence collected is shown in Figure 3.

Figure 3. Traffic flow sequence.

As can be seen from Figure 3, the traffic flow sequence exhibits a certain periodicity. Traffic flow is small before 6 o'clock in the morning, and traffic flow increases significantly after 6 o'clock. Since the traffic flow at a certain time at the intersection is related to the traffic flow in the first few hours of the time, the traffic flow data for the first five moments of this moment is used as the input of the prediction model.

In this study, two sets of prediction simulation experiments were carried out. Firstly, the traffic flow data of the first 4 days was used as training samples for the IBSAELM model, and the last day traffic flow data was used as test samples for the IBSAELM prediction model for testing; secondly, the traffic flow data of the first 3 days was used as training samples for the IBSAELM model, while the last two days of the traffic flow data were used as test samples for the IBSAELM prediction model.

In order to evaluate the prediction results of the model, the absolute error (AE), mean absolute percentage error (MAPE), root mean square error (RMSE) and decision coefficient (r^2) are used as evaluation indexes of the IBSAELM model.

$$AE = x^* - x \quad (15)$$

$$MAPE = \frac{1}{m} \left(\sum_{j=1}^m \left(\frac{|x_j^* - x|}{x} \times 100 \right) \right) \quad (16)$$

$$RMSE = \sqrt{\frac{1}{m} \sum_{j=1}^m (x_j^* - x)^2} \quad (17)$$

$$r^2 = \frac{(\sum x \sum x^* - m \sum xx^*)^2}{(m \sum (x)^2 - \sum (x)^2) \times (m \sum (x^*)^2 - \sum (x^*)^2)} \quad (18)$$

where x^* is the fitting value of the prediction model; m is the number of traffic flow data; x is the actual value of the sample.

AE , $MAPE$ and $RMSE$ are mainly used to evaluate the gap between the predicted value and the actual value of the prediction model. r^2 ($r^2 \in [0, 1]$) is mainly used to evaluate the fitting degree between the predicted value and the actual value. When the r^2 value tends to 1, it shows that the predicted value of the model fits well with the actual value; when the r^2 value approaches 0, it shows that the fitting effect of the predicted value and the real value is poor.

Firstly, the IBSAELM, BSAELM, SVM and PSOSVM models are used to predict traffic flow on the fifth day, respectively. The results of each model are shown in Figure 4.

Figure 4. Forecast results of traffic flow sequence for one-day.

The prediction curves of the five prediction models in Figure 4 can basically reflect the trend of one-day traffic flow series, but the prediction results of the BSAELM model and IBSAELM model are closer to the real trend of traffic flow. In order to analyze the errors of these five prediction models for one-day traffic flow prediction, the absolute error curves of each model are given in Figure 5, and the evaluation results of each model are given in Table 3.

Figure 5. AE error curves for each model.

Table 3. Assessment results for one-day traffic flow forecasting.

Model	AE interval	MAPE	RMSE	r2
SVM	[−52.9172, 56.0597]	36.5020%	23.5329	0.8534
PSOSVM	[−59.2102, 53.2237]	34.2059%	23.4623	0.8559
BSAELM	[−24.6309, 29.6506]	14.8894%	10.8332	0.9689
IBSAELM	[−21.6707, 27.7891]	10.1507%	8.8253	0.9793

Compared with the error curves in Figure 5, it is found that the prediction errors of the SVM model and PSOSVM model are higher than those of the BSAELM model and IBSAELM model. In Table 3, the AE intervals of the SVM model and the PSOSVM model are [−52.9172, 56.0597] and [−59.2102, 53.2237], respectively. The error intervals of the SVM and PSOSVM models are twice as large as those of the BSAELM and IBSAELM models. The MAPE and RMSE of the SVM model and PSOSVM model are significantly higher than those of the BSAELM and IBSAELM models. The r^2 values of the SVM and PSOSVM models are 0.8534 and 0.8559, respectively, which indicates that the two models have poor interpretation ability for traffic flow data.

By analyzing the four evaluation indicators in Table 3, it is found that the MAPE and RMSE values of the IBSAELM model are the smallest. The MAPE of IBSAELM model is 26.4513%, 24.0552% and 4.7387% smaller than that of SVM, PSOSVM and BSAELM, respectively. The RMSE of the IBSAELM model is 14.7076, 14.6370 and 2.0079 smaller than that of the SVM, PSOSVM and BSAELM, respectively. The maximum r^2 of the IBSAELM model is 0.9793, which indicates that the predicted value fits well with the actual value.

Secondly, the traffic flow data of the first three days are used as training samples of the IBSAELM, BSAELM, SVM, and PSOSVM prediction models, and the traffic flow data of 192 moments in the last two days are used as test samples. The predicted results of the IBSAELM, BSAELM, SVM and PSOSVM models are shown in Figure 6.

Figure 6. Prediction results of traffic flow series for two days.

Four prediction models can still reflect the changing trend of the actual value curve, and with the increase of traffic flow samples, the predicted values of the four models do not fluctuate significantly, which shows that the four prediction models have high prediction stability. The prediction error curves of the IBSAELM, BSAELM, SVM and PSO SVM models are shown in Figure 7. Table 4 gives the evaluation results of four evaluation indicators for the prediction effect of each model.

Figure 7. AE error of two-day traffic flow sequence.

Table 4. Evaluation results of two-day traffic flow forecasting.

Model	AE interval	MAPE	RMSE	r2
SVM	[−74.3384, 58.8470]	48.7763%	24.8414	0.8325
PSOSVM	[−71.2117, 58.6361]	51.0760%	23.9641	0.8439
BSAELM	[−46.2912, 61.0277]	23.0289%	18.2291	0.9089
IBSAELM	[−44.7104, 40.1863]	28.1498%	16.0354	0.9295

By analyzing the error curves in Figures 5 and 7, it is found that the fluctuation range of the error curves in Figure 7 is increased. Comparing the data in Table 3, the AE interval, MAPE and RMSE of the four models in Table 4 have increased significantly.

The *MAPE* values of the IBSAELM, BSAELM, SVM and PSOSVM models increase by 12.2743%, 16.6811%, 8.1935% and 17.9991%, respectively; the *RMSE* of the IBSAELM, BSAELM, SVM and PSOSVM models increases by 1.3085, 0.5018, 7.3959 and 7.2101, respectively. As the number of traffic flow samples increases, the prediction errors of the four models increase, and the fitting effect of the prediction model is also deteriorating. Compared with r^2 in Table 3, the r^2 of the BSAELM and IBSAELM models are reduced by 0.06 and 0.0498, respectively.

Comprehensive analysis of the data in Table 4 shows that the prediction results of the IBSAELM model are better than those of the other three models. The *AE* interval of the IBSAELM model is $[-44.7104, 40.1863]$, controlled in interval $[-50, 50]$, and the *RMSE* of IBSAELM model is 8.806, 7.9287 and 2.1937 smaller than that of SVM, PSOSVM and BSAELM respectively. For two-day traffic flow, the r^2 value of IBSAELM is 0.9295 higher than that of the other three prediction models, which shows that IBSAELM model is better than the other three models in explaining traffic flow sequence.

5. Conclusions

With the rapid development of economy, people's demand for automobiles is increasing. Too many automobiles are the root cause of urban traffic congestion. By forecasting the short-term traffic flow, traffic control and efficient vehicle routing planning can be achieved, thus alleviating traffic congestion and reducing the waste of resources caused by traffic congestion. In this study, the IBSA optimizer is proposed to optimize the stochastic parameters of the ELM model, and the IBSAELM model is proposed to predict one-day traffic flow and two-day traffic flow, respectively. The prediction results of the model are evaluated by *AE*, *MAPE*, *RMSE* and r^2 , respectively.

The main findings of this study are as follows: (1) In order to improve the optimization performance of the BSA optimizer, the BSA optimizer was improved in this study. (2) By optimizing four standard functions, it is found that the time consumed by 50 iterations of the IBSA optimizer takes 1.5756 s, 3.9936 s, 4.1652 s and 1.2607 s less time than the BSA optimizer, respectively. Similarly, the running time of each iteration of IBSA optimizer is 0.0315 s, 0.0799 s, 0.0832 s and 0.0399 s less than that of the BSA optimizer, respectively. (3) For one-day traffic flow forecasting, the *MAPE* of IBSAELM model is 26.4513%, 24.0552% and 4.7387% smaller than that of SVM, PSOSVM and BSAELM, respectively. The *RMSE* of the IBSAELM model is 14.7076, 14.6370 and 2.0079 smaller than that of SVM, PSOSVM and BSAELM, respectively; and the IBSAELM model has a higher r^2 value.

This study has three contributions: (1) The IBSA optimizer proposed in this study has better optimization effect; (2) The IBSAELM model is proposed to predict short-term traffic flow, and the feasibility of the model is verified by simulation experiments; (3) The accurate prediction of short-term traffic flow lays a solid foundation for the realization of intelligent transportation system. It has positive significance to alleviate urban traffic congestion and improve traffic management efficiency.

Although this study proposes a traffic flow prediction method based on machine learning theory. But time series statistical models (such as ARIMA and SARIMA models) are not studied in depth. In future research, the author will use intelligent optimizer to optimize the super parameters of SARIMA model to achieve the best parameter setting of SARIMA model. In the future, the improved SARIMA model will be applied to the field of traffic flow prediction.

Author Contributions: K.-P.L. designed the experiment and got the experimental data; S.-q.Z. analyzed the experimental data and wrote the paper. All authors contributed to discussing and revising the manuscript. All authors have read and agreed to the published version of the manuscript.

Acknowledgments: The authors would like to thank the editors and reviewers for their constructive comments on this study.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhang, X.J.; Mu, L.M.; Zhao, J.; Xu, C.X. An Efficient Anonymous Authentication Scheme with Secure Communication in Intelligent Vehicular Ad-Hoc Networks. *Ksii Trans. Internet Inf. Syst.* **2019**, *13*, 3280–3298.
2. Senouci, O.; Aliouat, Z.; Harous, S. Mca-V2i: A Multi-Hop Clustering Approach over Vehicle-to-Internet Communication for Improving Vanets Performances. *Future Gener. Comput. Syst. Int. J. Esci.* **2019**, *96*, 309–323. [\[CrossRef\]](#)
3. Hu, X.; Deng, Z.H. Research on Perception Bias of Implementation Benefits of Urban Intelligent Transportation System Based on Big Data. *Eurasip J. Wirel. Commun. Netw.* **2019**, *7*. [\[CrossRef\]](#)
4. Xie, J.; Choi, Y.K. Hybrid Traffic Prediction Scheme for Intelligent Transportation Systems Based on Historical and Real-Time Data. *Int. J. Distrib. Sens. Netw.* **2017**, *13*, 11. [\[CrossRef\]](#)
5. Shen, Z.G.; Wang, W.L.; Shen, Q.; Li, Z.C. Hybrid Csa Optimization with Seasonal Rvr in Traffic Flow Forecasting. *Ksii Trans. Internet Inf. Syst.* **2017**, *11*, 4867–4887.
6. Yang, H.F.; Dillon, T.S.; Chen, Y.P.P. Optimized Structure of the Traffic Flow Forecasting Model with a Deep Learning Approach. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *28*, 2371–2381. [\[CrossRef\]](#)
7. He, X.; Xu, L.; Zhang, Z. Driving Behaviour Characterisation by Using Phase-Space Reconstruction and Pre-Trained Convolutional Neural Network. *Iet Intell. Transp. Syst.* **2019**, *13*, 1173–1180. [\[CrossRef\]](#)
8. Zhang, S.; Yao, Y.; Hu, J.; Zhao, Y.; Li, S.B.; Hu, J.J. Deep Autoencoder Neural Networks for Short-Term Traffic Congestion Prediction of Transportation Networks. *Sensors* **2019**, *19*, 19. [\[CrossRef\]](#)
9. Li, Q.; Wang, F.; Wang, J.F.; Li, W.S. Lstm-Based Sql Injection Detection Method for Intelligent Transportation System. *IEEE Trans. Veh. Technol.* **2019**, *68*, 4182–4191. [\[CrossRef\]](#)
10. Cheng, A.Y.; Jiang, X.; Li, Y.F.; Zhang, C.; Zhu, H. Multiple Sources and Multiple Measures Based Traffic Flow Prediction Using the Chaos Theory and Support Vector Regression Method. *Phys. Stat. Mech. Appl.* **2017**, *466*, 422–434. [\[CrossRef\]](#)
11. Hui, M.; Bai, L.; Li, Y.B.; Wu, Q.S. Highway Traffic Flow Nonlinear Character Analysis and Prediction. *Math. Probl. Eng.* **2015**, *7*. [\[CrossRef\]](#)
12. Li, Y.F.; Jiang, X.; Zhu, H.; He, X.Z.; Peeta, S.; Zheng, T.X.; Li, Y.G. Multiple Measures-Based Chaotic Time Series for Traffic Flow Prediction Based on Bayesian Theory. *Nonlinear Dyn.* **2016**, *85*, 179–194. [\[CrossRef\]](#)
13. Zhang, Y.R.; Zhang, Y.L.; Haghani, A.A. Hybrid Short-Term Traffic Flow Forecasting Method Based on Spectral Analysis and Statistical Volatility Model. *Transp. Res. Part C Emerg. Technol.* **2014**, *43*, 65–78. [\[CrossRef\]](#)
14. Shang, Q.; Lin, C.Y.; Yang, Z.S.; Bing, Q.C.; Zhou, X.Y. Short-Term Traffic Flow Prediction Model Using Particle Swarm Optimization-Based Combined Kernel Function-Least Squares Support Vector Machine Combined with Chaos Theory. *Adv. Mech. Eng.* **2016**, *8*, 12. [\[CrossRef\]](#)
15. Williams, B.M.; Hoel, L.A. Modeling and Forecasting Vehicular Traffic Flow as a Seasonal ARIMA Process: Theoretical Basis and Empirical Results. *J. Transp. Eng.* **2003**, *129*. [\[CrossRef\]](#)
16. Smith, B.L.; Williams, B.M.; Keith Oswald, R. Comparison of parametric and nonparametric models for traffic flow forecasting. *Transp. Res. Part C Emerg. Technol.* **2002**, *10*, 303–321. [\[CrossRef\]](#)
17. Tselentis, D.I.; Vlahogianni, E.I.; Karlaftis, M.G. Improving short-term traffic forecasts: To combine models or not to combine? *Iet Intell. Transp. Syst.* **2015**, *9*, 193–201. [\[CrossRef\]](#)
18. Karlaftis, M.G.; Vlahogianni, E.I. Statistical methods versus neural networks in transportation research: Differences, similarities and some insights. *Transp. Res. Part C Emerg. Technol.* **2011**, *19*, 387–399. [\[CrossRef\]](#)
19. Vlahogianni, E.; Karlaftis, M. Comparing traffic flow time-series under fine and adverse weather conditions using recurrence-based complexity measures. *Nonlinear Dyn.* **2012**, *69*, 1949–1963. [\[CrossRef\]](#)
20. Li, L.C.; Qin, L.Q.; Qu, X.; Zhang, J.; Wang, Y.G.; Ran, B. Day-Ahead Traffic Flow Forecasting Based on a Deep Belief Network Optimized by the Multi-Objective Particle Swarm Algorithm. *Knowl. Based Syst.* **2019**, *172*, 1–14. [\[CrossRef\]](#)
21. Zhu, J.Z.; Cao, J.X.; Zhu, Y. Traffic volume forecasting based on radial basis function neural network with the consideration of traffic flows at the adjacent intersections. *Transp. Res. Part C Emerg. Technol.* **2014**, *47*, 139–154. [\[CrossRef\]](#)
22. Vlahogianni, E.I.; Karlaftis, M.G.; Golias, J.C. Optimized and meta-optimized neural networks for short-term traffic flow prediction: A genetic approach. *Transp. Res. Part C Emerg. Technol.* **2005**, *13*, 211–234. [\[CrossRef\]](#)

23. Wang, X.X.; Xu, L.H.; Chen, K.X. Data-Driven Short-Term Forecasting for Urban Road Network Traffic Based on Data Processing and Lstm-Rnn. *Arab. J. Sci. Eng.* **2019**, *44*, 3043–3060.
24. Diao, Z.L.; Zhang, D.F.; Wang, X.; Xie, K.; He, S.Y.; Lu, X.; Li, Y.B. A Hybrid Model for Short-Term Traffic Volume Prediction in Massive Transportation Systems. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 935–946. [[CrossRef](#)]
25. Mariani, V.C.; Och, S.H.; Coelho, L.D.; Domingues, E. Pressure Prediction of a Spark Ignition Single Cylinder Engine Using Optimized Extreme Learning Machine Models. *Appl. Energy* **2019**, *249*, 204–221. [[CrossRef](#)]
26. Li, L.L.; Sun, J.; Tseng, M.L.; Li, Z.G. Extreme Learning Machine Optimized by Whale Optimization Algorithm Using Insulated Gate Bipolar Transistor Module Aging Degree Evaluation. *Expert Syst. Appl.* **2019**, *127*, 58–67. [[CrossRef](#)]
27. Bui, D.T.; Ngo, P.T.T.; Pham, T.D.; Jaafari, A.; Minh, N.Q.; Hoa, P.V.; Samui, P. A Novel Hybrid Approach Based on a Swarm Intelligence Optimized Extreme Learning Machine for Flash Flood Susceptibility Mapping. *Catena* **2019**, *179*, 184–196. [[CrossRef](#)]
28. Kourehli, S.; Ghadimi, S.; Ghadimi, R. Vibration Analysis and Identification of Breathing Cracks in Beams Subjected to Single or Multiple Moving Mass Using Online Sequential Extreme Learning Machine. *Inverse Probl. Sci. Eng.* **2019**, *27*, 1057–1080. [[CrossRef](#)]
29. Li, L.L.; Lv, C.M.; Tseng, M.L.; Song, M.L. Renewable energy utilization method: A novel Insulated Gate Bipolar Transistor switching losses prediction model. *J. Clean. Prod.* **2018**, *176*, 852–863. [[CrossRef](#)]
30. Meng, X.B.; Gao, X.Z.; Lu, L.H.; Liu, Y.; Zhang, H.Z. A new bio-inspired optimisation algorithm: Bird Swarm Algorithm. *J. Exp. Theor. Artif. Intell.* **2016**, *28*, 673–687. [[CrossRef](#)]
31. Xiang, L.; Deng, Z.Q.; Hu, A.J. Forecasting Short-Term Wind Speed Based on IEWT-LSSVM Model Optimized by Bird Swarm Algorithm. *IEEE Access* **2019**, *7*, 59333–59345. [[CrossRef](#)]
32. Wu, K.H.; Li, K.; Liang, R.; Ma, R.Z.; Zhao, Y.X.; Wang, J.; Qi, L.J.; Liu, S.Y.; Han, C.; Yang, L.; et al. A Joint Planning Method for Substations and Lines in Distribution Systems Based on the Parallel Bird Swarm Algorithm. *Energies* **2018**, *11*, 14. [[CrossRef](#)]
33. Li, L.L.; Liu, Z.F.; Tseng, M.L.; Chiu, A.S.F. Enhancing the Lithium-Ion Battery Life Predictability Using a Hybrid Method. *Appl. Soft Comput.* **2019**, *74*, 110–121. [[CrossRef](#)]
34. Yang, Y.L.; Hou, M.Z.; Luo, J.S. A Novel Improved Extreme Learning Machine Algorithm in Solving Ordinary Differential Equations by Legendre Neural Network Methods. *Adv. Differ. Equ.* **2018**, *24*. [[CrossRef](#)]
35. Zhao, W.T.; Li, P.; Liu, Q.; Liu, D.; Liu, X.W. Selecting the Optimal Hidden Layer of Extreme Learning Machine Using Multiple Kernel Learning. *Ksii Trans. Internet Inf. Syst.* **2018**, *12*, 5765–5781.

© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).