Research Article

A Novel Hybrid Model for Predicting Traffic Flow via Improved Ensemble Learning Combined with Deep Belief Networks

Yikang Rui [ID],[1,2,3] **Wenqi Lu** [ID],[1,2,3] **Ziwei Yi** [ID],[1,2,3] **Renfei Wu** [ID],[1,2,3] **and Bin Ran** [ID][1,2,3]

[1]*School of Transportation, Southeast University, Nanjing, Jiangsu 211189, China*
[2]*Joint Research Institute on Internet of Mobility, Southeast University and University of Wisconsin-Madison, Southeast University, Nanjing 211189, China*
[3]*Jiangsu Key Laboratory of Urban ITS, Southeast University, Nanjing 211189, China*

Correspondence should be addressed to Wenqi Lu; lplwq93@126.com

The intelligent transportation system (ITS) plays an irreplaceable role in alleviating urban traffic congestion and realizing sustainable urban development. Accurate and efficient short-term traffic state forecasting is a significant issue in ITS. This study proposes a novel hybrid model (ELM-IBF) to predict the traffic state on urban expressways by taking advantage of both deep learning models and ensemble learning framework. First, a developed bagging framework is introduced to combine several deep belief networks (DBNs) that are utilized to capture the complicated temporal characteristic of traffic flow. Then, a novel combination method named improved Bayesian fusion (IBF) is proposed to replace the averaging method in the bagging framework since it can better fuse the prediction results of the component DBNs by assigning the reasonable weights to DBNs at each prediction time interval. Finally, the proposed hybrid model is validated with ground-truth traffic flow data captured by the remote traffic microwave sensors installed on the multiple road sections of 2nd Ring Road in Beijing. The experimental results illustrate that the ELM-IBF method can effectively capture sharp fluctuations in the traffic flow. Compared with several benchmark models (e.g., artificial neural network, long short-term memory neural network, and DBN), the ELM-IBF model reveals better performance in forecasting single-step-ahead traffic volume and speed. Additionally, it is proved that the ELM-IBF model is capable of providing stable and high-quality results in multistep-ahead traffic flow prediction.

1. Introduction

With the rapid growth of vehicle ownership, the conflict between traffic demand and traffic supply becomes increasingly acute, and it causes more frequent traffic congestion. Although the construction and improvement of the expressway network alleviate the rapidly increasing traffic demand to some extent, the urban traffic problems cannot be solved only by building or expanding the expressway network because of the restrictions on urban land use and environmental factors [1]. Therefore, to alleviate traffic congestion on the urban expressways with limited space, ITS has gradually become an effective way to manage the traffic flow on expressways and to improve the mobility and safety of expressways.

Short-term traffic flow prediction technology is an essential issue in ITS. It describes the process of estimating the anticipated traffic conditions in the short-term given historical and current traffic information captured by several traffic detectors [2–4]. With the help of traffic information released by the advanced traffic information service system, travelers can understand road conditions and choose time-saving travel paths [5]. At present, the common detectors employed in the ITS include loop detectors, global positioning systems, and remote traffic microwave sensors (RTMS). As one of the most popular nonintrusive traffic detectors, RTMSs transmit microwave beams to both moving and stationary objects and receive the reflected signals as the background signals [6]. When a car enters the detection zone, the reflected signal will be strengthened to

exceed the background signal threshold, and the car will be detected. Meanwhile, RTMSs are installed on the side of the road without causing temporary lane closure or traffic flow interruption during the process of installation [7]. Hence, more traffic management departments regard this kind of non-invasive sensor as an automatic transportation data acquisition method. Due to the high measurement accuracy of RTMSs, which can be up to 95% [8], the continuous travel speed and volume captured by RTMS are used as the dataset in this study.

Because of the complexity and dynamic state of traffic flow, massive approaches including deep learning technologies [9–12] have been proposed to address traffic state estimation issues. However, high-efficiency and robust prediction of traffic conditions are difficult because of the following challenges and problems: (1) though traffic conditions on urban expressways exhibit recurrent patterns over time, recurrent traffic conditions are affected by planned incidents such as road construction and sports events, and unplanned incidents and accidents, resulting in a deviation from the recurrent patterns; (2) deep learning-based models such as convolutional neural network and long short-term memory (LSTM) neural network need big datasets to train thousands of parameters and learn the spatiotemporal characteristics of the traffic flow. It may cause huge time consumption and fail to meet the real-time requirements of forecasting traffic conditions; (3) most of the existing methods focus on predicting the traffic flow in the next time interval. In fact, accurate and stable methods are needed to deal with the task of multistep-ahead traffic flow forecasting [13].

By taking advantage of both deep learning models and ensemble learning framework, this paper builds a novel hybrid model (ELM-IBF) to forecast traffic state. It has been proven that deep belief networks (DBN) are valid in the prediction task [14]. Hence, we utilized DBNs as the subpredictors in this study. Then, a new bagging framework is introduced to decrease the size of the training data and reduce the training time of the DBNs. Finally, we developed the bagging framework by replacing the average combination methods with an improved Bayesian fusion method. Based on the aforementioned discussion, the improvements and contributions of this paper are summarized in the following four aspects:

(1) A novel hybrid model (ELM-IBF), which combines the ensemble learning theory and deep belief network, is proposed to make short-term traffic state prediction on urban expressways.

(2) Deep belief networks are utilized as the subpredictors in the Bagging algorithm to learn the complicated temporal characteristic of traffic flow.

(3) A combination method, named improved Bayesian fusion, is proposed to replace the averaging method in the Bagging framework. It can better fuse the prediction results of the component DBNs by assigning the reasonable weights to component DBNs at each prediction time intervals.

The remainder of this paper is organized as follows: a general overview of existing literature on traffic forecasting is provided in Section 2. The theory of DBN, the improved Bayesian fusion (IBF) methods, and the proposed ELM-IBF model are introduced in Section 3. The experimental dataset, the evaluation index, and the experimental environment are given in Section 4. The discussions of the prediction performance of different methods and relevant analyses are carried out in Section 5. Finally, we present the conclusions and future research efforts in Section 6.

2. Literature Review

Due to the significance and prospective applications of traffic flow prediction, considerable research efforts have been made to enrich traffic prediction approaches. In general, the methodology falls into three major categories: statistical methods, nonparametric methods, and hybrid methods.

2.1. Statistical Methods.
The statistical methods mainly include time-series models [15], Kalman filtering methods [16], and Auto-Regressive Integrated Moving Average (ARIMA) methods [17]. Among the various statistical methods, ARIMA [17] and its variants [18, 19] have been proved to be capable of getting promising forecasting results. Many researchers applied the data analysis techniques developed by Box and Jenkins [20] to predict freeway short-term traffic flow, and it is found that the ARIMA model can represent freeway time-series data in a highly accurate manner. It is also shown that the ARIMA model was adequate in reproducing time series of urban arterials [18]. Compared with typical ARIMA forecasting methods, Seasonal ARIMA (SARIMA) enables the extraction of the seasonal variations and reveals implicit periodical characteristics in the time-series data. Hence, SARIMA-based methods [21] have obtained better predictive performances than typical ARIMA methods.

2.2. Nonparametric Methods.
Though statistical methods offer explicit formulas to give valuable interpretations of traffic characteristics, they are inferior at predicting the traffic flow with irregular fluctuations considering their definite model structure [22]. During recent years, massive nonparametric methods for traffic flow prediction have emerged such as support vector regression [23, 24], k-nearest neighbor [25], random forest [26], and XGBoost [27]. Different from statistical methods, nonparametric methods have relaxed assumptions for inputs, and they are more capable of processing outliers and noisy data [28]. Artificial neural networks (ANN), which can capture the nonlinear correlations and mine the complex patterns of the measured historical data, are the most popular nonparametric methods. Since Hua and Faghri introduced ANN models to vehicle travel time estimation [29], many other advanced neural network models were shifted to the traffic forecast domain successively, such as feed-forward neural network [30], fuzzy neural network [31], echo state neural network [32], radial basis function neural network (RBFNN) [33], and recurrent neural network (RNN) [34].

With the development of big data, the nonparametric methods for traffic flow prediction are shifting from ANNs to deep learning methods [35–39]. For example, Huang et al. [14] proposed a deep architecture with a deep belief network (DBN) at the bottom and a multitask regression layer at the top. The DBN layer is employed for unsupervised feature learning, and a multitask regression layer is used above the DBN for supervised prediction. Yang et al. [40] established a novel stacked autoencoder model to learn the hierarchical representation of urban traffic flow. Ma et al. [41] introduced the LSTM neural network, which can overcome the vanishing gradient problem of the RNN, to forecast traffic speed based on the microwave data in Beijing. Li et al. [13] proposed a deep belief network optimized multiobjective particle swarm algorithm to forecast day-ahead traffic flow. Wang et al. [42] built a path-based deep learning framework that can produce better traffic speed prediction on a citywide scale. Peng et al. [43] provided a long-term traffic flow prediction method based on dynamic graphs to overcome data defects in traffic flow prediction. Although the deep learning method can process a large amount of data, its prediction results are strongly dependent on the amount of data, and training a deep learning model takes a long time and uses a large amount of storage space.

2.3. Hybrid Methods. As the statistical methods and nonparametric methods both have their flaws in traffic prediction, an effective combination of different approaches may be a better choice to solve the traffic flow forecasting problem. There are many existing works of literature demonstrating the advantage of hybrid methods, and consequently, numerous hybrid models [44–46] have been developed. Wei and Chen [47] combined empirical mode decomposition and back-propagation neural networks (BPNN) to predict the short-term passenger flow in metro systems. Wang et al. [48] proposed a short-term traffic speed forecasting hybrid model using chaos wavelet analysis and support vector machine. Gu et al. [49] established a Bayesian combination model with deep learning, which combines the ARIMA model, radial basis function neural network, and the gated recurrent unit neural networks to forecasting traffic volume in multiple scenarios. Li et al. [50] built a multimodal deep learning model using two parallel stacked autoencoders that can simultaneously consider the spatial and temporal dependencies of the traffic flow. Gu et al. [51] proposed a fusion model consisting of an entropy-based grey relation analysis and a double-layer RNN structure. Vlahogianni [52] proposed the surrogate model using three prediction methods for combination in short-term freeway traffic speed prediction. These studies have demonstrated that the combination of different predictors can improve the final accuracy of traffic prediction under normal traffic conditions on freeways and motorways. Qiu et al. [53] established an integrated precipitation-correction model for freeway traffic flow prediction using fusion techniques with four basic forecasting models. The advantages of the hybrid models lie in two points: (1) the superiority of each component model can be mined to improve the prediction accuracy and robustness of the whole model. (2) The

component models can be trained or calibrated parallelly, which saves time and is more efficient than some complicated single models.

Therefore, inspired by the powerful data mining capability of deep learning methods and the high robustness of ensemble learning technology, we propose a novel hybrid method (ELM-IBF) for traffic flow prediction through integrating the deep belief networks and an improved ensemble learning framework. Experimental results confirm that the ELM-IBF model enables us to make accurate and stable multistep-ahead traffic flow forecasting.

3. Methodology

In this section, we first introduce the DBNs, which are component predictors of the ELM-IBF model. Then, a description of an IBF method is given. Finally, we introduce the ELM-IBF model based on the bagging framework developed by DBNs and IBF.

3.1. Deep Belief Network (DBN). DBN is a deep artificial neural network with many hidden layers as shown in Figure 1(a). Each hidden layer has a large number of hidden units. The classical DBN is equivalent to the superposition of several Restricted Boltzmann Machines (RBMs) and an output layer. DBN usually utilizes a fast and greedy unsupervised learning algorithm to train RBMs. After the RBM layers training is completed, a supervised fine-tuning method is adopted to adjust the network by the training data [54]. Figure 1(b) illustrates the structure of an RBM, where **v** represents a visible layer, and **h** represents a hidden layer. In the stacked structure of the DBN shown in Figure 1(c), the hidden layer of one RBM is regarded as the visible layer of the next RBM. An energy function is defined by an RBM according to its parameter set $\theta = (w, b, a)$ as follows:

$$E(\mathbf{v}, \mathbf{h}|\boldsymbol{\theta}) = -\sum_i b_i v_i - \sum_j a_j h_j - \sum_i \sum_j w_{ij} v_i h_j, \quad (1)$$

where v_i and h_j are the ith visible layer unit and j^{th} hidden layer unit, respectively; w_{ij} is the weight between v_i and h_j; b_j and a_j are the biases of the layers.

The joint probability distribution of visible layer **v** and hidden layer **h** can be calculated as follows:

$$p(\mathbf{v}, \mathbf{h}|\boldsymbol{\theta}) = \frac{\exp(-E(\mathbf{v}, \mathbf{h}|\boldsymbol{\theta}))}{\sum_{\mathbf{v}, \mathbf{h}} \exp(-E(\mathbf{v}, \mathbf{h}|\boldsymbol{\theta}))}. \quad (2)$$

As shown in Figure 1(a), there are no interconnections between the neurons in the visible layer and the hidden layer. As the binary units are used, v_i and h_j belong to (0, 1). The activation probabilities of the neurons in the hidden layer and the neurons in the visible layer are given as follows:

$$p(h_j = 1|\mathbf{v}, \boldsymbol{\theta}) = \sigma\left(b_j + \sum_j w_{ij} h_j\right), \quad (3)$$

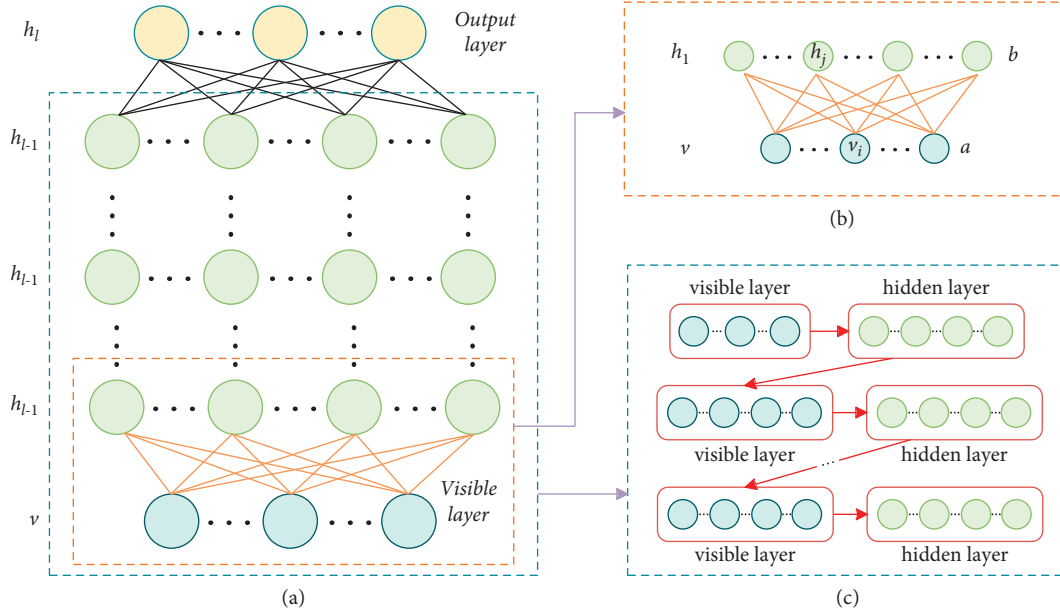$$p(v_i = 1|\mathbf{h}, \boldsymbol{\theta}) = \sigma\left(a_i + \sum_i w_{ij} v_i\right), \quad (4)$$

FIGURE 1: The deep belief network and restricted Boltzmann machines. (a) Deep belief network, (b) restricted Boltzmann machines, and (c) stacked procedure of RBMs.

where σ is a sigmoid activation function.

For the whole hidden layer and visible layer, they can be given by

$$p(\mathbf{h}|\mathbf{v}) = \prod_{j}(h_j|\mathbf{v}), \qquad (5)$$

$$p(\mathbf{v}|\mathbf{h}) = \prod_{i}(v_i|\mathbf{h}). \qquad (6)$$

The marginal probability distribution of input vector $\mathbf{v}$ over the hidden units is obtained as

$$p(\mathbf{v}|\boldsymbol{\theta}) = \frac{\sum_{\mathbf{h}}\exp(-E(\mathbf{v},\mathbf{h}|\boldsymbol{\theta}))}{\sum_{\mathbf{v},\mathbf{h}}\exp(-E(\mathbf{v},\mathbf{h}|\boldsymbol{\theta}))}. \qquad (7)$$

Therefore, the objective function can be given as

$$L(\boldsymbol{\theta},\mathbf{v}) = \sum_{v \in D}\log\,p(\mathbf{v},\boldsymbol{\theta}), \qquad (8)$$

where D represents the training dataset.

To obtain the optimal θ for a single data vector $\mathbf{v}$, the gradient of log-likelihood estimation can be calculated based on the following equations:

$$\frac{\partial\log p(\mathbf{v}|\boldsymbol{\theta})}{\partial w_{ij}} = E_p\langle h_j v_i\rangle - E_{\widehat{p}}\langle h_j v_i\rangle, \qquad (9)$$

$$\frac{\partial\log p(\mathbf{v}|\boldsymbol{\theta})}{\partial a_j} = E_p\langle h_j\rangle - E_{\widehat{p}}\langle h_j\rangle, \qquad (10)$$

$$\frac{\partial\log p(\mathbf{v}|\boldsymbol{\theta})}{\partial b_i} = E_p\langle v_i\rangle - E_{\widehat{p}}\langle v_i\rangle, \qquad (11)$$

where $E_p\langle.\rangle$ and $E_{\widehat{p}}\langle.\rangle$ are the expectations of the probability over v under the empirical distribution P and model's distribution $\widehat{P}$.

The contrastive divergence (CD) learning method [54–56] can be utilized by reconstruction to minimize the difference of two Kullback-Leibler divergences (KL) to gain $E_{\widehat{p}}\langle.\rangle$. CD learning is proved to be efficiently practical and also can reduce computational cost compared with the typical Gibbs sampling method. At the beginning of the CD algorithm, the visible layer is initialized with training data; then, the hidden layer is calculated with the conditional distribution. Afterward, the visible layer is calculated with conditional distribution according to the hidden layer. The result is a reconstruction of the input. This algorithm only needs to iterate for k times to obtain the estimation of the model, and k usually takes the value of 1. The pseudocode of training an RBM is presented in Algorithm 1.

After obtaining a trained RBM, a regression layer is introduced to form the DBN model, which enables it to predict traffic flow data. The whole structure of the DBN model is shown in Figure 1(c). Afterward, the complete training process of DBN can be divided into two phases including a pretraining phase and a fine-tuning phase.

During the pretraining phase, a greedy layer-by-layer is used to gain the weights. First, training data is employed to fully train the first RBM as the code in Algorithm 1. Second, the output layer of the first RBM is taken as the input of the second RBM. Then, the above steps can be repeated several times.

After the pretraining phase, the backpropagation (BP) algorithm, a supervised learning method, is adopted to adjust the parameters to finish the fine-tuning phase.

Input: Training dataset $X = \{\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_m}\}$
Output: An RBM with the trained parameters
(1) **for** all sample $\mathbf{x}$ in X **do**
(2) $\mathbf{v}^{(0)} \longleftarrow \mathbf{x}$
(3) compute $p(h_j^{(0)} = 1|\mathbf{v}^{(0)}) = \sigma(b_j + \sum_j w_{ij} h_j)$
(4) select a sample from the hidden layer
(5) $\mathbf{h}^{(0)} \sim p(\mathbf{h}^{(0)}|\mathbf{v}^{(0)})$
(6) reconstruct the visible layer $p(v_i^{(0)} = 1|\mathbf{h}^{(0)}) = \sigma(a_i + \sum_i w_{ij} v_i)$
(7) select a sample from the visible layer
(8) compute $p(h_j^{(1)} = 1|\mathbf{v}^{(1)}) = \sigma(b_j + \sum_j w_{ij} h_j)$
(9) update: $\mathbf{w} = \mathbf{w} + \eta[[p(\mathbf{h}^{(0)} = 1|\mathbf{v}^{(0)})\mathbf{v}^{(0)T} - p(\mathbf{h}^{(1)} = 1|\mathbf{v}^{(1)})\mathbf{v}^{(1)T}]]$
$\mathbf{a} = \mathbf{a} + \eta(\mathbf{v}^{(0)} - \mathbf{v}^{(1)})$
$\mathbf{b} = \mathbf{b} + \eta[p(\mathbf{h}^{(0)} = 1|\mathbf{v}^{(0)}) - p(\mathbf{h}^{(1)} = 1|\mathbf{v}^{(1)})]$
(10) **end**

ALGORITHM 1: Training an RBM ($\mathbf{a}$, $\mathbf{b}$, $\mathbf{w}$).

Compared with the random initialization, the initial weights of each layer will be in a better position in the parameter space. Empirically speaking, starting from these positions, the gradient descent is more likely to converge to a better local extreme point because the unlabeled data has provided prior information about the patterns contained in a large amount of input data.

3.2. Improved Bayesian Fusion (IBF). Bayesian fusion is a linear fusion method with dynamic weights based on conditional probability and Bayesian rules to combine the prediction results of several subpredictors to form a better one. The principle of Bayesian fusion is to assign dynamic weights to different subpredictors based on their historical performance. In this paper, we intend to present an improved Bayesian fusion approach for developing the framework of ensemble learning.

Let p_t^n denote the weight of the subpredictor n at time interval t. Traditional Bayesian fusion (TBF) [57] assumes that the weight p_t^n depends on the prediction errors of all past intervals rather than only depending on the prediction error at prediction interval t. This assumption makes the TBF very insensitive to the fluctuating accuracy of component predictors, such that if the dominant subpredictor m is no longer the most accurate, it will spend many intervals reducing the dominant status of that subpredictor, thus imposing a negative impact on the predictions of the TBF. Therefore, Wang et al. [46] proposed a new Bayesian fusion (BF) to select a few traffic flows that may have comparatively higher relevance to the traffic flow at the prediction interval and neglect less relevant traffic flows when calculating the weights of the subpredictors.

According to Wang's research, the weight of the subpredictor n at the time interval t can be calculated as follows:

$$p_t^n = \frac{\left(1/\sqrt{2\pi}\,\sigma_n\right)^{R(P)} \exp\left(\sum_{i\in P} -\left[(y_i - y_i^n)/\sigma_n\right]^2\right)}{\sum_{m=1}^{N} \left(1/\sqrt{2\pi}\,\sigma_m\right)^{R(P)} \exp\left(\sum_{i\in P}(y_i - y_i^m)/\sigma_m\right)}, \quad (12)$$

where N is the number of the subpredictors; P is the set that contains the past time intervals such as $P = \{t-1, t-2, \ldots, t-\lambda\}$ and λ is the length of the fusion step; $R(P)$ is the

dimension of the set P and $R(P) = \lambda$; y_i^i is the measured traffic flow data at the time interval i. y_i^n is the predicted value of the traffic flow data of the subpredictor n at the time interval i; σ_n is the deviation of e_t^n which represents the prediction error of the subpredictor n at the time interval t, and $e_t^n = y_i - y_i^n$.

The prediction result of the BF is written as the linear combination of the output of all the subpredictors, which is formulated as

$$\hat{y}_{t+1} = \sum_{m=1}^{N} p_t^m y_{t+1}^m, \quad (13)$$

where $\hat{y}_{t+1}$ is the predicted value of the BF at time interval $t+1$; p_t^m is the weight of the subpredictor m at the time interval $t+1$; y_{t+1}^m is the predicted value of the subpredictor m at the time interval $t+1$.

However, the BF still has a drawback in the linear combination equation, because the sum of the weights of all subpredictors is equal to one. Hence, if the prediction results of the subpredictors are all larger or less than the actual value, the fusion result will still be larger or less than the actual value, and the prediction error of the BF could be even larger than those of some of the subpredictors. To deal with the problem above, this study replaces equation (13) with a nonlinear equation illustrated as follows:

$$\hat{y}_{t+1} = \tau_t \sum_{m=1}^{N} p_t^m y_{t+1}^m, \quad (14)$$

where τ_t is an error compensation factor, which can be calculated in the following equation:

$$\tau_t = \begin{cases} 1 + \dfrac{\left|\sum_{m=1}^{N} \sum_{i=t-\lambda}^{t} e_i^m\right|}{\sum_{m=1}^{N} \sum_{i=t-\lambda}^{t} y_i^m}, & \sum_{i=t-\lambda}^{t} e_i^m \geq 0, \forall m \in [1, N], \\[4mm] 1, & \text{otherwise}, \\[4mm] 1 - \dfrac{\left|\sum_{m=1}^{N} \sum_{i=t-\lambda}^{t} e_i^m\right|}{\sum_{m=1}^{N} \sum_{i=t-\lambda}^{t} y_i^m}, & \sum_{i=t-\lambda}^{t} e_i^m \leq 0, \forall m \in [1, N], \end{cases}$$

$$(15)$$

where e_i^m is the prediction error of the subpredictor m at the time interval i and $e_i^m = y_i - y_i^m$.

From equation (15), it is illustrated that τ_t is related to the length of the fusion step λ, the prediction error of the subpredictors e_i^m, and the prediction value of the subpredictors y_i^m. Based on Wang's study [46], an improved Bayesian fusion (IBF) can be proposed by combining equations (12), (14) and (15).

3.3. Proposed Hybrid Method (ELM-IBF).

Ensemble learning, which is also referred to as the multiclassifier system, completes the learning task by building and combining multiple learning devices [58]. Common ensemble learning strategies include AdaBoost [59], bagging [60], and stacking.

The Bagging method is a representative parallel ensemble learning algorithm that mainly uses the Bootstrap sampling approach to create subdatasets. During the training process, a Bagging method takes a certain number of samples from the original samples, and repeated sampling is allowed during training process. In this case, some samples in the original sample may be selected as training set samples many times, while other samples may not appear in the process of Bootstrap sampling. Through this process, the diversity of individual learners can be increased, and the overall generalization ability of the entire model can be improved.

For a given training dataset $D = \{(x^{(i)}, y^{(i)}), i = 1, 2, \ldots, l\}$, the number of iterations is $t_s = 1, 2, \ldots, T_s$ and the specific process of a general Bagging method for regression problem is shown as follows:

Step 1: Use Bootstrap to sample subdatasets $D(t_s)$ with the size of k from the dataset D for T_s times. T_s training subdatasets with the size of k can be obtained.

Step 2: Each training set is learned with a subpredictor, and the corresponding subpredictor h_t can be obtained.

Step 3: Use each trained subpredictor $h_{t_s}, t_s = 1, 2, \ldots, T_s$ to make a prediction, and use the averaging method to calculate the final output.

In this paper, an improved hybrid model (ELM-IBF) based on an improved ensemble learning method and DBNs is established to make short-term traffic flow predictions for the expressways. Figure 2 illustrates the structure of the ELM-IBF model. After applying the bagging method as its basic framework, the proposed method introduces DBNs as the subpredictors of the bagging framework, which are trained based on the different subdatasets and used to forecast multistep-ahead traffic flow, respectively. Then, the IBF method, which can dynamically adjust the weights of the subpredictors and adopt the error compensation factor changing the sum of the weights at each prediction time interval, is employed as the combination strategy of the bagging framework. Finally, the prediction results of the ELM-IBF model can be calculated by equation (14). Note that the implementation process of the ELM-IBF model can be divided into two phases, including Phase I for training the DBN predictors and Phase II for forecasting the traffic flow.

Algorithm 2 reveals a pseudocode of implementing an ELM-IBF model. As indicated in the tables, the main parameters in the ELM-IBF model are the number of the DBNs T_s and the lengths of the fusion steps λ, which are determined before the fusion. At each time step, T_s DBNs are assigned appropriate weights automatically according to their performance according to the previous λ time steps. The hyperparameters in the DBNs can be set according to previous studies [13].

The input and output of the ELM-IBF model can be written as follows:

$$x_{in} = \left[x_{t-t_b+1}, x_{t-t_b+2}, \ldots, x_t \right]^T, \qquad (16)$$

$$x_{out} = \left[\widehat{x}_{t+1}, \widehat{x}_{t+2}, \ldots, \widehat{x}_{t+t_a} \right]^T, \qquad (17)$$

where t_b is the look-back time step of the input vectors; t_a is the look-ahead prediction time step; x_t is the measured traffic flow data (volume or speed) at time interval t; $\widehat{x}_{t+1}$ is the predicted traffic flow data (volume or speed) at time interval $t + 1$.

4. Experiment

4.1. Data Description.

To demonstrate the performance of the proposed model, we conduct numerical experiments based on the real-world traffic flow data containing volume and speed data. As revealed in Figure 3, the ground-truth data for measuring varied model performance were captured by multiple remote traffic microwave sensors (RTMS) located at six road sections on the 2nd ring roads of Beijing. These six road sections include the East of Jishuitan Bridge (P1), the North of Fuxingmen Bridge (P2), the West of You'anmen Bridge (P3), the West of Zuo'anmen Bridge (P4), the South of Dongbianmen Bridge (P5), and the North of Chaoyangmen Bridge (P6). The length of the sampling time is two weeks (from January 6, 2014 to January 19, 2014) with a 2 min sampling time interval (720 time intervals in a day). Thus, the total number of records captured by each detector is 10080 with less than 5% missing and error rate. In addition, the whole dataset is divided into two parts including the first nine days for training (from Jan 6, 2014, to Jan 14, 2014) and the next five days for testing (from Jan. 15, 2014, to Jan. 19, 2014). Besides, the raw traffic flow data may contain little noises or abnormal data, and the threshold value method was employed to remove the outliers. To ensure a more reliable result, missing and erroneous records were properly remedied by using the temporally adjacent records. For the purpose of examining the performance of the models, especially the stability and robustness with the limited dataset, volume and speed data are both treated as the prediction target during the experiments.

The experimental platform of our research is a Lenovo computer with Intel(R) Core(TM) i7-8700 CPU@3.20 GHz and 8 GB memory. Python 3.6 with TensorFlow 1.0, Scikit-learn, and Keras 0.9 is exploited to implement the relevant models.
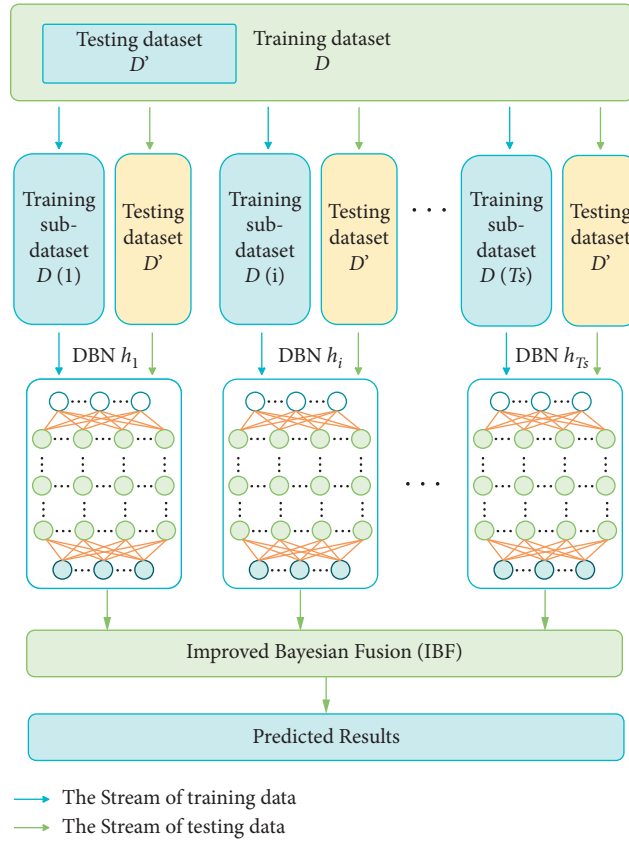
FIGURE 2: The structure of the ELM-IBF model method.

Phase I: Training the DBNs
Input: Training dataset D;
Sampling size k;
Number of DBN predictor T_s
Output: DBNs with the trained parameters
(1) **for** $i = 1$ to T_s **do**
(2) Sample the subdataset $D(i)$ with size k from the dataset D by Bootstrap
(3) Train the DBN h_i with subdataset $D(i)$ by Algorithm 1 and BP Algorithm
(4) **end**
Phase II: Forecasting the traffic flow with trained DBNs
Input: Testing dataset D';
Length of the testing dataset D': l;
Trained DBNs;
Length of the fusion step: λ
Output: Predicted values of the ELM-IBF method
(5) **for** $i = 1$ to T_s **do**
(6) Predict the traffic flow data with testing dataset D' by using DBN h_i
(7) **end**
(8) **for** i from 1 to l **do**
(9) Calculate the weights of DBNs h_j, $j = 1, 2, \ldots, T_s$ at each prediction time interval by using equation (12)
(10) **end**
(11) Calculate the predicted values with equations (14) and (15) in IBF
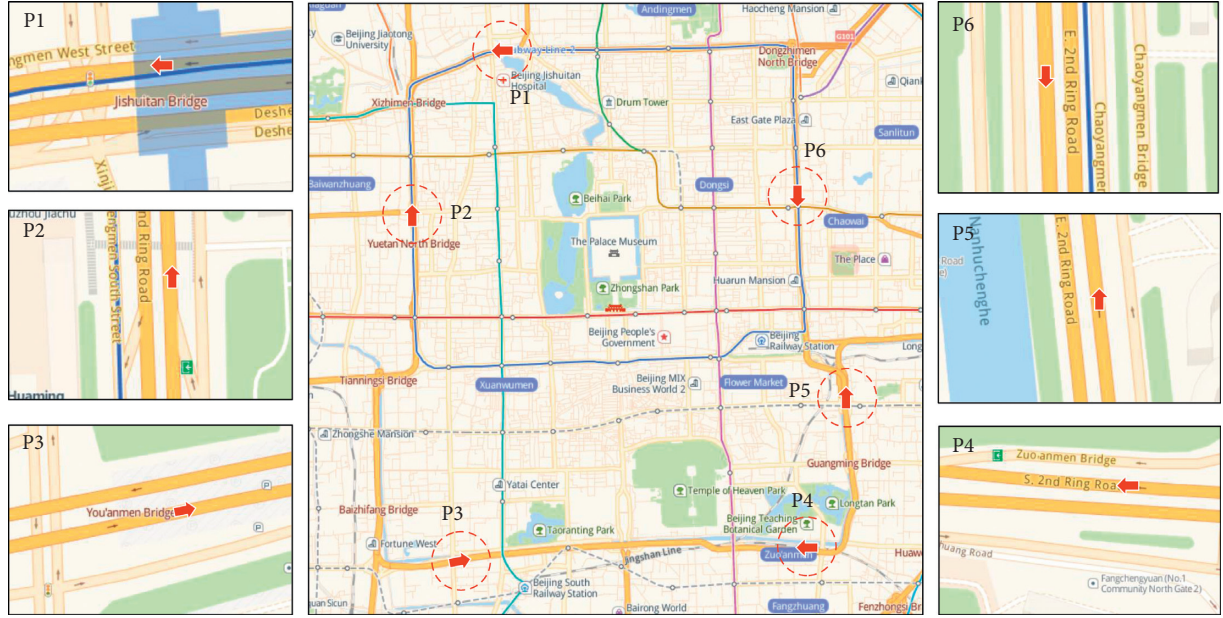
ALGORITHM 2: Implementing an ELM-IBF model.

FIGURE 3: The observation of the experimental road sections on 2nd Ring Road of Beijing.

4.2. Baseline Methods. To examine the practicability and effectiveness of the proposed method, ARIMA, radial basis function neural network (RBFNN), BPNN, LSTM, and DBN are chosen as the single benchmark models. Also, the ensemble learning methods ELM-AM and ELM-BF are employed as the hybrid benchmarks.

For the ARIMA, the Akaike information criterion (AIC) is considered to determine the best order (p, d, q) of the model, and the parameters p, d, q are chosen as 2, 1, and 1, respectively. For the RBFNN and BPNN, they both have an input layer with 20 neurons, a hidden layer with 50 neurons, and an output layer with one neuron. The sigmoid function and Gaussian radial basis function are employed as the activation function in the BPNN and the RBFNN, respectively. Meanwhile, the LSTM, which has the superior capability for time-series prediction with long temporal dependency, is composed of one input layer, one LSTM layer with memory blocks, and one output layer. The number of the hidden units in the LSTM layer is 200, and the largest time lag is 20. The relevant parameters of the benchmark DBN model are shown in Table 1, and the parameters of the DBNs in the ELM-IBF method are set the same as those of the DBN-based benchmark models. It is necessary to notice that the ELM-AM and the ELM-BF share the same DBN structures with the ELM-IBF model. To be specific, the ELM-AM and ELM-BF use the average method combination strategy and the Bayesian fusion strategy, respectively.

Note that the look-back time step t_b of the NN-based models (RBFNN, BPNN, LSTM, DBN, ELM-AM, ELM-BF, and ELM-IBF) is set as 20, and it means that the traffic condition during the previous 40 min will be considered in the input of the NN-based models. In the proposed method, the sampling size k is set as the 2/3 size of the entire training dataset. The number of DBN predictors T_s is set as 3, and the

TABLE 1: Relevant parameters of the DBNs.

Description	Value
Number of nodes in the input layer	20
Number of RBM layers	2
Number of nodes in the first hidden layer	400
Number of nodes in the second hidden layer	400
Iteration number of each RBM	200
Activation function (RBM)	Sigm
Learning rate (RBM)	1
Batch size (RBM)	100
Number of epochs (RBM)	200
Batch size (NN)	100
Number of epochs (NN)	500
Activation function (NN)	Sigm
Learning rate (NN)	1
Input Zero Masked Fraction (RBM)	0.5

fusion step λ in IBF is set as 1. The prediction target of this study is the traffic volume and speed in the testing dataset.

4.3. Evaluation Indicators. To evaluate the performance of the proposed model and benchmark models, mean absolute error (MAE), root mean square error (RMSE), mean absolute percentage error (MAPE), and Theil inequality coefficient (TIC) are selected as the evaluation index.

$$\text{MAE} = \frac{1}{s} \sum_{i=1}^{s} |\widehat{y}_i - y_i|, \tag{18}$$

$$\text{RMSE} = \sqrt{\frac{1}{s} \sum_{i=1}^{s} (\widehat{y}_i - y_i)^2}, \tag{19}$$

$$\text{MAPE} = \frac{1}{s} \sum_{i=1}^{s} \left| \frac{\widehat{y}_i - y_i}{y_i} \right| \times 100\%, \tag{20}$$

$$\text{TIC} = \frac{\sqrt{(1/s) \sum_{i=1}^{s} (\widehat{y}_i - y_i)^2}}{\sqrt{(1/s) \sum_{i=1}^{s} (\widehat{y}_i)^2} + \sqrt{(1/s) \sum_{i=1}^{s} (y_i)^2}}, \tag{21}$$

where $\widehat{y}_i$ is the predicted value, and y_i is the observed value. s is the length of the prediction time interval.

Note that MAE, RMSE, and MAPE are utilized to represent the accuracy of the prediction models. TIC reflects the fitting degree between the predicted value and the observed value. The numerical distribution of TIC is between 0 and 1. The smaller the TIC is, the higher the fitting degree is.

5. Results and Discussion

5.1. Overall Prediction Performance of Different Methods. With the look-ahead prediction time step t_a of the ELM-IBF model set as 1, Tables 2 and 3 compare the overall prediction performance of the ELM-IBF model and its benchmarks in terms of traffic volume and traffic speed prediction by taking the prediction results of the separated six scenarios as a whole.

Generally, the ELM-IBF model demonstrates better predictive performance compared with the single models. In terms of speed prediction illustrated in Table 2, the ELM-IBF model outperforms the DBN, which is the second-best single model with improvements of 28.98%, 30.66%, and 28.66% on MAE, MAPE, and RMSE, respectively. In terms of the volume prediction revealed in Table 3, the MAE and MAPE of the ELM-IBF model are 33.69% and 36.17% less than those of the DBN. Among the hybrid models, the performance advantages of the ELM-IBF model are more obvious under the condition that the ELM-AM, the ELM-BF, and the ELM-IBF share the same component predictors. Meanwhile, the robustness of the ELM-IBF model can be guaranteed since its ensemble learning structure can keep the model robust, and its improved Bayesian fusion mechanism can benefit the accuracy.

It can be observed that the prediction performance of the ELM-AM and the ELM-BF is slightly worse than that of their component model DBN. The possible reasons can be summarized as follows: (1) the training data of DBN in the ensemble learning frameworks are not large enough, which leads to insufficient training. (2) The fusion algorithms, such as AM and BF, are not efficient enough to assign larger weights to the DBN predictors, which illustrate better performance.

Figure 4 gives the overall prediction errors produced by these different methods in several road sections. As indicated in Figure 4, the proposed ELM-IBF method reveals better predictive performance than other models in terms of the maximum, minimum, and median of errors. Besides, it can be found that the ELM-IBF model has a smaller distance between $Q1$ and $Q3$, and the error distribution of the ELM-IBF model is more concentrated than that of other models.

Figure 5 illustrates the prediction results of the ELM-IBF model on a weekday (2014.1.16) and a weekend (2014.1.18)

TABLE 2: Comparison of the speed prediction performance of different methods.

Methods	MAE	MAPE (%)	RMSE	TIC
ARIMA	4.75	8.86	6.82	0.0522
RBFNN	5.30	10.57	7.68	0.0588
BPNN	4.79	9.47	6.80	0.0520
DBN	4.52	8.84	6.49	0.0496
LSTM	4.55	8.89	6.50	0.0496
ELM-AM	4.66	8.96	6.64	0.0512
ELM-BF	4.59	8.80	6.52	0.0503
ELM-IBF	3.21	6.13	4.63	0.0354

TABLE 3: Comparison of the volume prediction performance of different methods.

Methods	MAE	MAPE (%)	RMSE	TIC
ARIMA	10.50	11.40	14.15	0.0543
RBFNN	12.36	14.02	16.64	0.0639
BPNN	10.75	12.93	14.30	0.0549
DBN	10.42	11.39	14.05	0.0540
LSTM	10.49	11.21	14.10	0.0546
ELM-AM	10.90	12.34	14.59	0.0558
ELM-BF	10.78	12.23	14.41	0.0551
ELM-IBF	6.91	7.27	9.21	0.0359

on the road section P1. In Figure 5, the ELM-IBF model is capable of capturing the tendency and volatility of the traffic flow for the entire day on the weekdays and the weekends. Even at the morning peak (8:00–11:00) and the evening peak (16:00–20:00) of the weekday when the traffic speeds fluctuate sharply, the proposed method is still able to fit the fierce fluctuation of traffic speeds and make a precise single-step-ahead prediction.

Figures 6 and 7 display the correlation between actual values and predicted values from six models including one statistical method (ARIMA), two traditional nonparametric methods (BPNN and RBFNN), two deep learning methods (DBN and LSTM), and the proposed ELM-IBF model in five days. r represents the Pearson correlation coefficient to evaluate the relevance between observed and predicted values. The two figures indicate the prediction results in the task of speed prediction and volume prediction, respectively. From the observation of two figures, we can gain several conclusions as follows: the ELM-IBF model produces better prediction results with r compared with other models. In addition, the deep learning models outperform the statistical model and traditional nonparametric methods, because DBN and LSTM have complex structures and strong learning abilities. Furthermore, the ELM-IBF model works best among these models, because it combines the advantages of deep learning methods and ensemble learning theory.

5.2. Performance of the ELM-IBF Model for Multistep-Ahead Prediction. Table 4 lists the prediction performance of the ELM-IBF model for multistep-ahead prediction tasks. It can be observed that the MAEs, MAPEs, RMSEs, and TICs of the proposed method do not rise obviously as the look-ahead
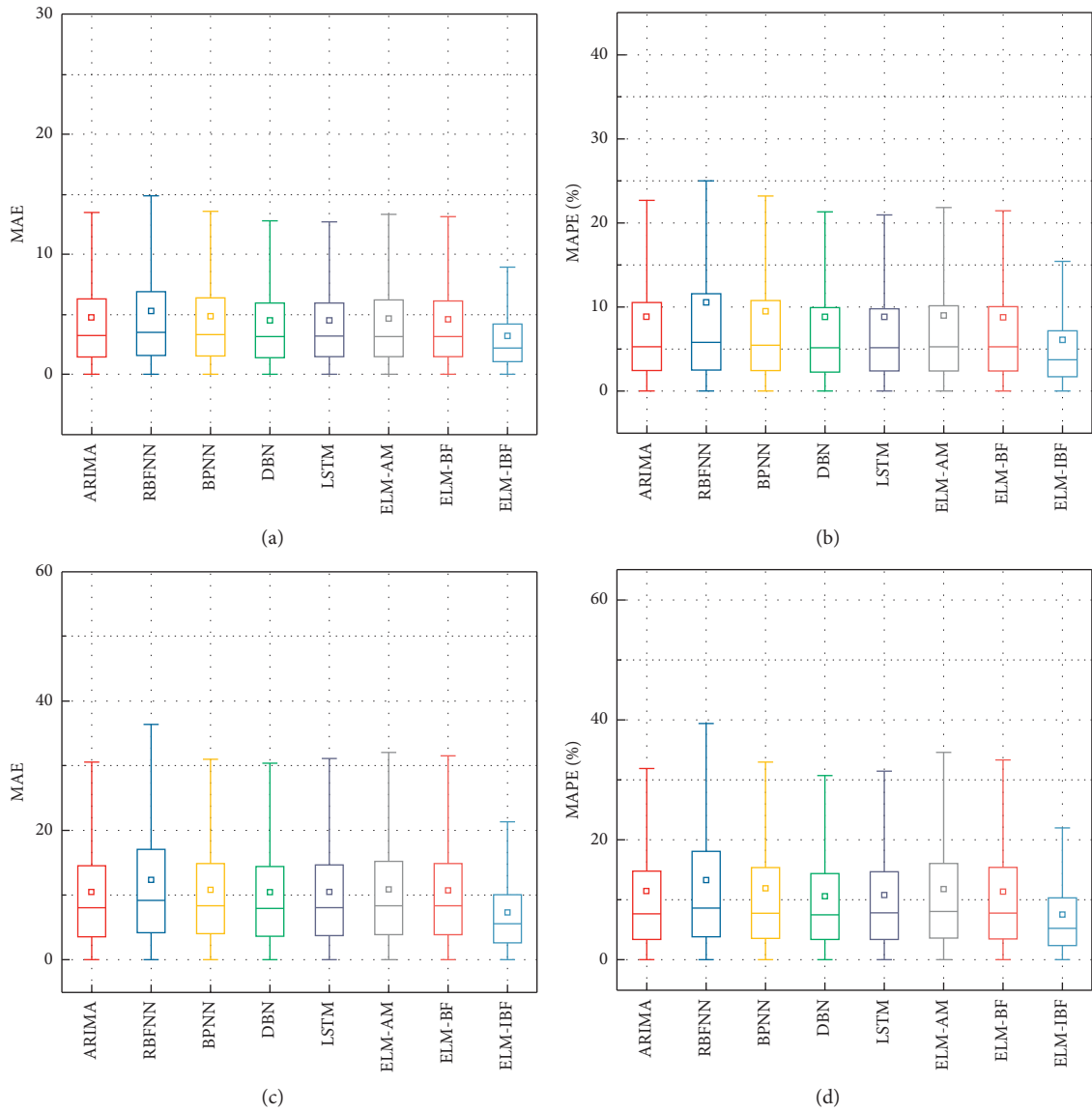
FIGURE 4: Comparison of the prediction errors for each method at the road section P1. (a) Speed prediction (MAE). (b) Speed prediction (MAPE). (c) Volume prediction (MAE). (d) Volume prediction (MAPE).
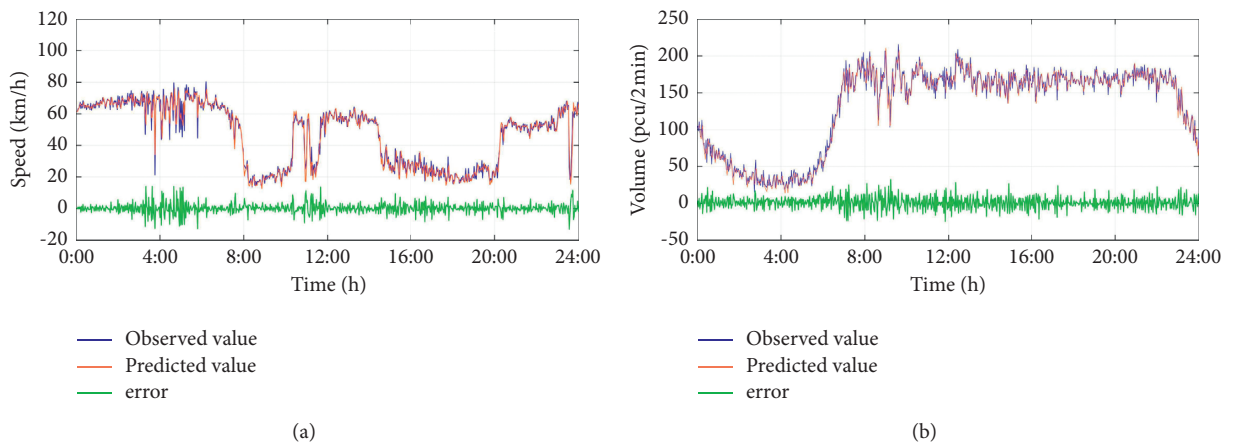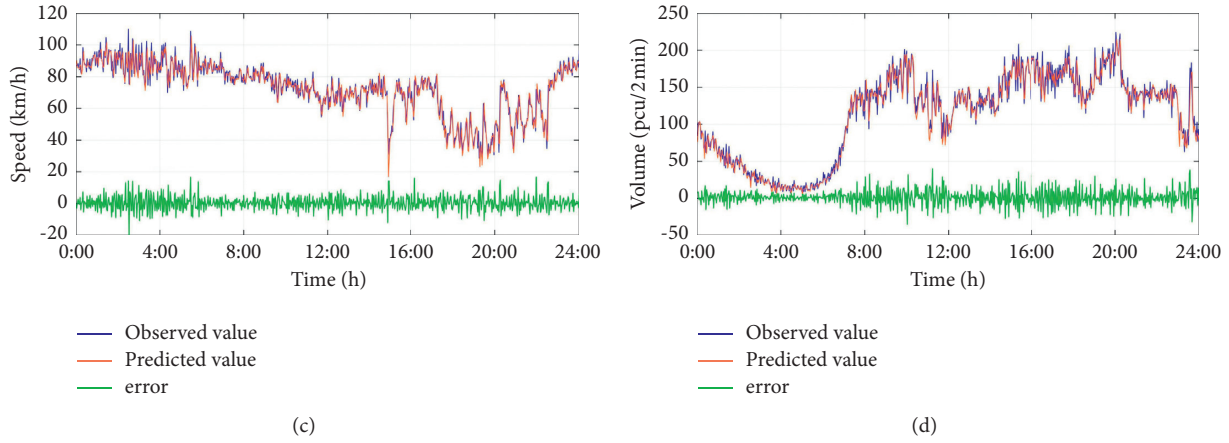

FIGURE 5: Continued.

(c)

(d)

FIGURE 5: Prediction results of the ELM-IBF model at the road section P1. (a) Speed prediction on a weekday (2014.1.16). (b) Volume prediction on a weekday (2014.1.16). (c) Speed prediction on a weekend (2014.1.18). (d) Volume prediction on a weekend (2014.1.18).
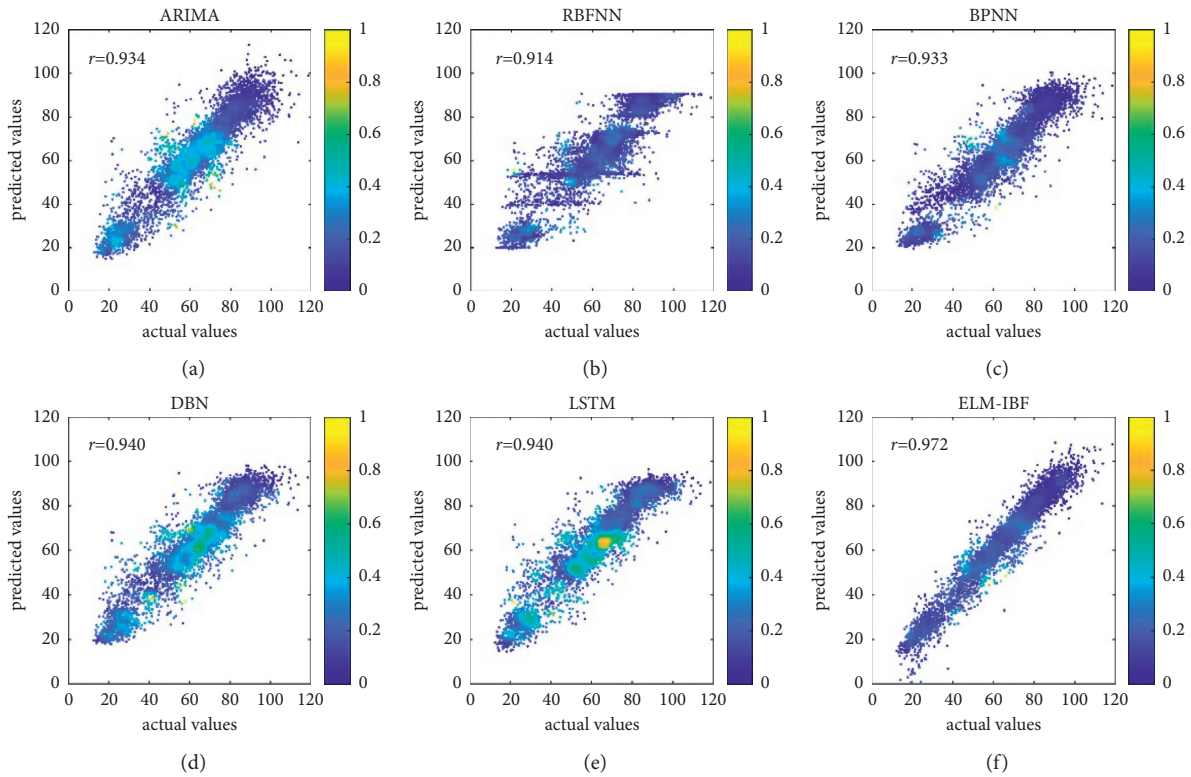

(a)

(b)

(c)

(d)

(e)

(f)

FIGURE 6: The correlation distribution of predicted results for the traffic speed. (a) ARIMA; (b) RBFNN; (c) BPNN; (d) DBN; (e) LSTM; (f) ELM-IBF.

time step increases. To be specific, the MAEs, MAPEs, and RMSEs fluctuate slightly with the look-ahead time step t_a increasing from 1 to 10, which corresponds to the prediction horizons ranging from 2 min to 20 min. It is interesting to find that the ELM-IBF model becomes more accurate when it is utilized to forecast traffic volume with a longer look-ahead time step. Meanwhile, the TICs rise slowly with t_a becoming larger, and it means that the fitting degree of the

ELM-IBF model drops a little, although its prediction accuracy is relatively stable.

Overall, the proposed method exhibits the advantages of making multistep-ahead traffic flow prediction. In the case of both traffic speed and volume prediction, the accuracy and stability of the results predicted by the ELM-IBF model with 10 look-ahead time steps are even higher than those predicted by its benchmark models (e.g., ARIMA, BPNN,
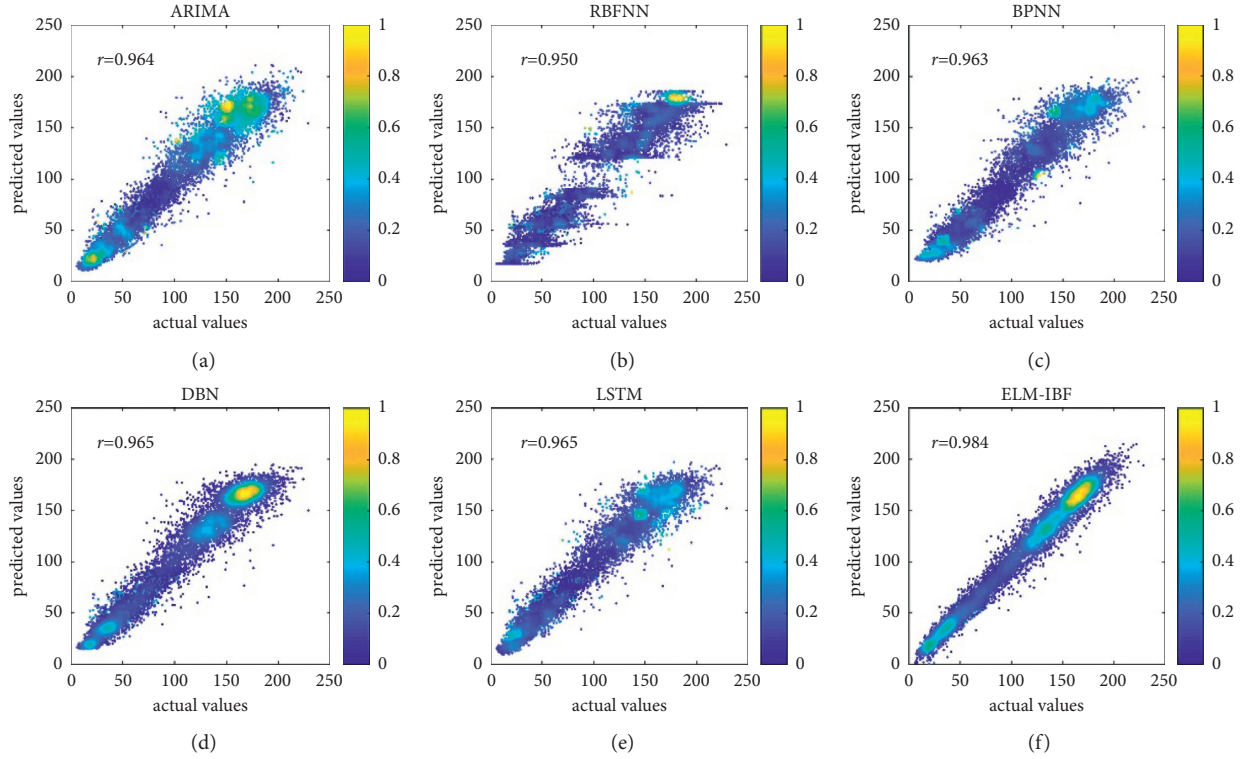
FIGURE 7: The correlation distribution of predicted results for the traffic volume. (a) ARIMA; (b) RBFNN; (c) BPNN; (d) DBN; (e) LSTM; (f) ELM-IBF.

TABLE 4: Prediction results of the ELM-IBF model with longer prediction horizon.

Look-ahead time step	Speed prediction				Volume prediction			
	MAE	MAPE (%)	RMSE	TIC	MAE	MAPE (%)	RMSE	TIC
1	3.21	6.13	4.63	0.0354	6.91	7.27	9.21	0.0359
2	3.28	6.89	5.43	0.0424	6.95	7.47	9.30	0.0364
3	3.45	7.81	6.53	0.0515	7.00	7.58	9.50	0.0371
4	3.84	9.29	7.82	0.0616	7.03	7.66	9.63	0.0378
5	3.89	9.49	7.86	0.0621	7.13	7.75	9.70	0.0384
6	4.08	10.06	8.31	0.0653	7.22	7.93	9.88	0.0388
7	4.24	10.71	8.68	0.0682	7.36	8.22	10.18	0.0399
8	4.37	11.28	9.06	0.0712	7.43	8.32	10.28	0.0402
9	4.49	11.58	9.23	0.0724	7.77	8.79	10.85	0.0423
10	4.54	11.66	9.29	0.0729	7.97	9.02	10.90	0.0455

RBFNN, LSTM, DBN, ELM-AM, and ELM-BF) with only one look-ahead time step shown in Tables 2 and 3.

5.3. Sensitivity Analysis.

In this section, we conduct the sensitivity analysis and parameter tuning on the ELM-IBF model, where two kinds of critical parameters are investigated, including the length of the fusion step and the number of subpredictors.

The length of the fusion step λ in the IBF is an important parameter that affects the prediction performance of the proposed method. Figure 8 indicates the relationship between the prediction performance of the ELM-IBF model and the length of the fusion step under the condition that the look-ahead prediction time step t_a is chosen to be 1, and the number of DBN predictors is set as 3. Figure 8 reveals that the prediction accuracy and fitting degree decline gradually with the length of the fusion step λ increasing from 1 to 10. This may be caused by the phenomenon that the error compensation factor τ_t in equation (15) becomes closer to 1 when prediction errors during more previous time intervals are taken into the fusion process with the fusion step becoming larger. Therefore, the recommended length of the fusion step is $\lambda = 1$, which can ensure the accuracy and robustness of the ELM-IBF method without costing too much fusion time.

Figure 9 indicates the relationship between the prediction performance of the model and the number of DBN
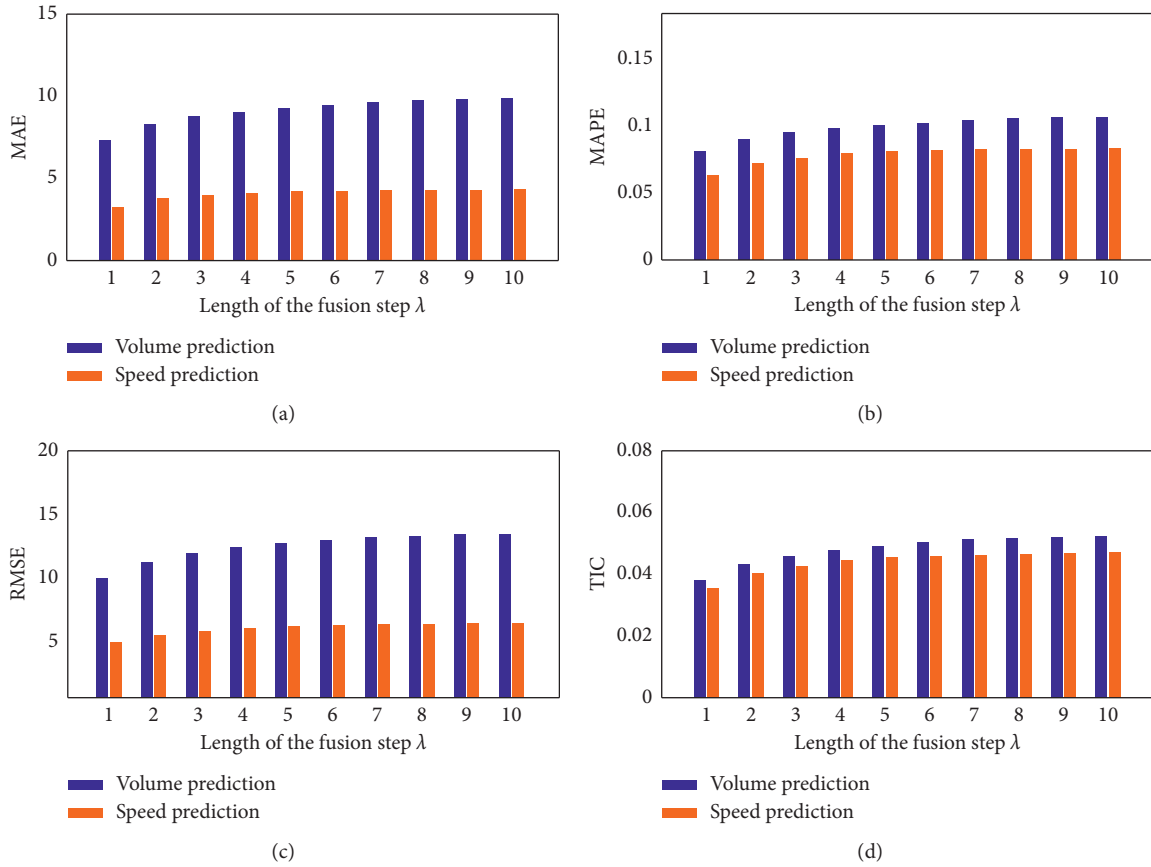
FIGURE 8: The influence of the length of the fusion steps on the ELM-IBF model. (a) MAE; (b) MAPE; (c) RMSE; (d) TIC.
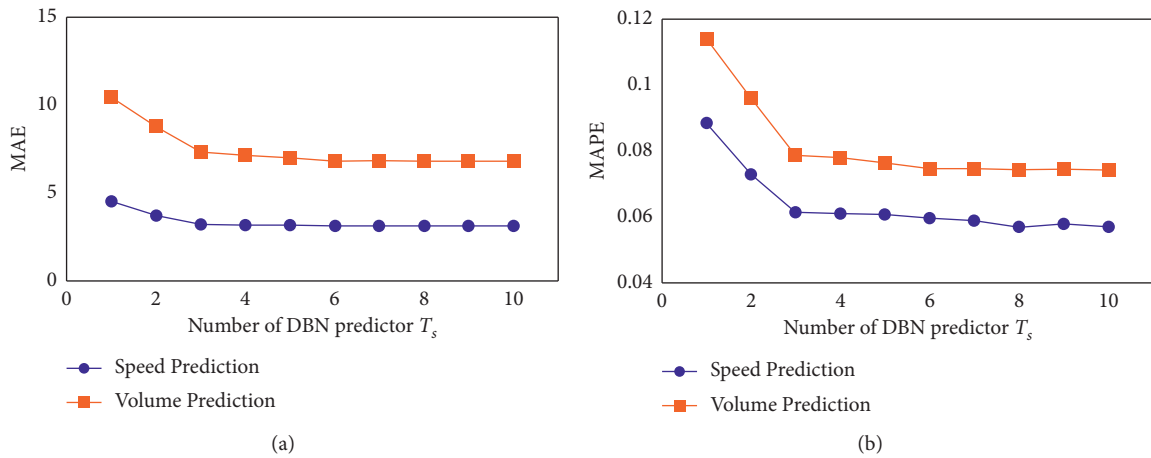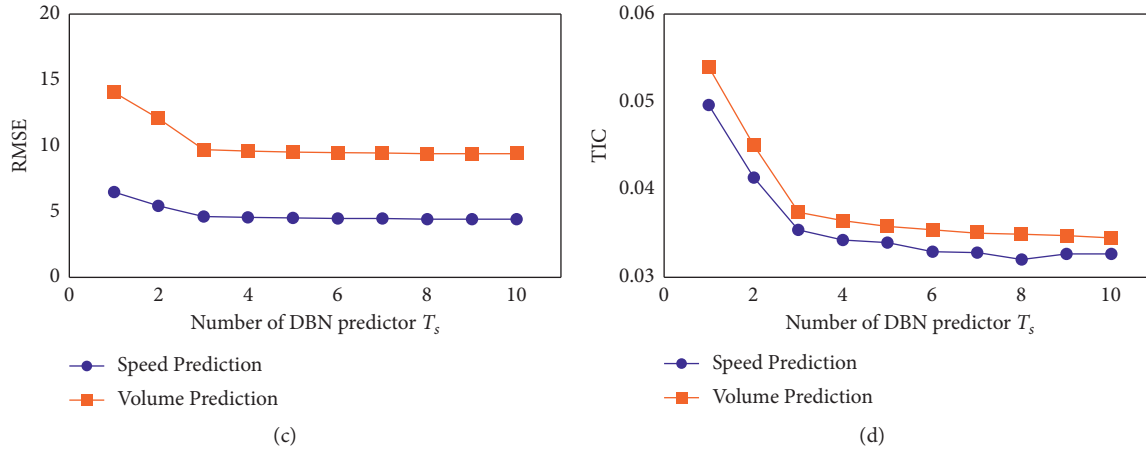

FIGURE 9: Continued.

(c)

(d)

Figure 9: The influence of the number of DBN predictors on the ELM-IBF model. (a) MAE; (b) MAPE; (c) RMSE; (d) TIC.

subpredictors when making a single-step-ahead prediction with the fusion time step set as 1. The number of DBN subpredictors is tuned from 1 to 10 with a step of 1. As illustrated in Figure 9, the MAEs, RMSEs, and MAPEs decrease more than 30% with the number of DBN subpredictors increasing from 1 to 5 in the task of forecasting both traffic speed and volume prediction, implying that the appropriate increase of the subpredictors can significantly improve the accuracy and fitting degree of the ensemble model. In addition, when the number of DBN subpredictors is larger than 5, the performance of the ELM-IBF model remains stable, and it demonstrates that the excessive increase of the number of subpredictors has little effect on the prediction performance probably due to the similarity of the homogeneous models. Therefore, the suitable number of the DBNs is around 3∼5 considering the prediction accuracy and the time consumption of training the deep learning models.

Note that though the ELM-IBF model exhibits its capability of predicting traffic flow parameters accurately and stably, the shortcoming of the ELM-IBF may lie in its efficiency since the ensemble learning framework needs to combine several DBNs, and the aggregate time of training several DBNs is much larger than one DBN model. However, with the rapid development of data processing, data storage, and parallel computing technology, the training time consumption of the deep learning model may be shortened dramatically, and the sensitivity analysis of the above two critical parameters is also beneficial to reduce the time consumption and keep the accuracy of the ELM-IBF model.

6. Conclusions

Short-term traffic flow prediction is a significant problem in ITS. This paper establishes a novel hybrid model (ELM-IBF) for short-term traffic flow forecasting based on the combination of ensemble learning theory and deep belief network. At first, the Bagging algorithm is employed to divide the training dataset into several subdatasets and determine the entire structure of the ELM-IBF model. Then, the deep belief networks are introduced as the subpredictors trained

by the divided datasets. Afterward, an improved Bayesian fusion approach is proposed to integrate the prediction results of DBNs more efficiently. Finally, the measured traffic flow data collected on the six road sections of expressways in Beijing are utilized to examine the accuracy and robustness of the proposed method.

The main conclusions of this study can be summarized as follows: (1) the overall prediction results demonstrate that the ELM-IBF model outperforms the single model-based benchmarks (e.g., ARIMA, BPNN, RBFNN, LSTM, and DBN) in terms of accuracy and fitting degree when making single-step-ahead traffic volume and speed prediction. (2) Compared with other ensemble learning methods (e.g., ELM-AM and ELM-BF) with the same subpredictors, the ELM-IBF has lower MAE, MAPE, RMSE, and TIC. It proves that the IBF combination methods can improve the performance of the bagging framework significantly and work better than AM and BF methods. (3) The ELM-IBF model illustrates a stable and accurate prediction performance in the task of forecasting multistep-ahead traffic flow. (4) Sensibility analysis confirms that the length of the fusion step and the number of the subpredictors both affect the predictive performance of the ELM-IBF model, and the recommended values of the two aforementioned parameters are 1 and 5, respectively.

In the future, we will concentrate on utilizing multi-source traffic flow data to enrich the input variables of the proposed method to further improve model performance. Larger amounts of data including more abnormal traffic conditions are necessary to be used for training and examining the proposed models. Furthermore, this paper only applies homogeneous component predictors in the ELM-IBF model, while heterogeneous component predictors could be added and investigated to enhance the generalization ability of the ensemble learning model.

Data Availability

The data used to support the findings of this study were provided by the Beijing Municipal Commission of Transport and were not made public.

Conflicts of Interest

Acknowledgments

References

[1] Y. Qi and S. Ishak, "A hidden Markov model for short term prediction of traffic conditions on freeways," *Transportation Research Part C: Emerging Technologies*, vol. 43, pp. 95–111, 2014.

[2] F. Zhan, X. Wan, Y. Cheng, and B. Ran, "Methods for multi-type sensor allocations along a freeway corridor," *IEEE Intelligent Transportation Systems Magazine*, vol. 10, no. 2, pp. 134–149, 2018.

[3] Z. Li and J. Huang, "How to mitigate traffic congestion based on improved ant colony algorithm: a case study of a congested old area of a metropolis," *Sustainability*, vol. 11, no. 4, p. 1140, 2019.

[4] X. Chen, Z. Li, Y. Yang, L. Qi, and R. Ke, "High-resolution vehicle trajectory extraction and denoising from aerial videos," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 5, pp. 3190–3202, 2021.

[5] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, "Short-term traffic forecasting: where we are and where we're going," *Transportation Research Part C: Emerging Technologies*, vol. 43, pp. 3–19, 2014.

[6] B. Coifman, "Freeway detector assessment," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1917, no. 1, pp. 149–163, 2005.

[7] J. R. Junger, J. P. Havlicek, R. D. Barnes, and M. P. Tull, "Prediction aggregation of remote traffic microwave sensors speed and volume data," in *Proceedings of the 2009 12th International IEEE Conference on Intelligent Transportation Systems*, pp. 1–6, St. Louis, MO, USA, October 2009.

[8] X. Yu and P. D. Prevedouros, "Performance and challenges in utilizing non-intrusive sensors for traffic data collection," *Advances in Remote Sensing*, vol. 2, no. 2, pp. 45–50, 2013.

[9] H.-C. Shin, H. R. Roth, M. Gao et al., "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.

[10] N. G. Polson and V. O. Sokolov, "Deep learning for short-term traffic flow prediction," *Transportation Research Part C: Emerging Technologies*, vol. 79, pp. 1–17, 2017.

[11] Y. Liu, Z. Liu, and R. Jia, "DeepPF: a deep learning based architecture for metro passenger flow prediction," *Transportation Research Part C: Emerging Technologies*, vol. 101, pp. 18–34, 2019.

[12] Y. Lv, Y. Duan, W. Kang, Z. Li, and F. Y. Wang, "Traffic flow prediction with big data: a deep learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 865–873, 2015.

[13] L. Li, L. Qin, X. Qu, J. Zhang, Y. Wang, and B. Ran, "Day-ahead traffic flow forecasting based on a deep belief network optimized by the multi-objective particle swarm algorithm," *Knowledge-Based Systems*, vol. 172, pp. 1–14, 2019.

[14] W. Huang, G. Song, H. Hong, and K. Xie, "Deep architecture for traffic flow prediction: deep belief networks with multitask learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2191–2201, 2014.

[15] W. Min and L. Wynter, "Real-time road traffic prediction with spatio-temporal correlations," *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 4, pp. 606–616, 2011.

[16] J. Guo, W. Huang, and B. M. Williams, "Adaptive Kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification," *Transportation Research Part C: Emerging Technologies*, vol. 43, pp. 50–64, 2014.

[17] M. S. Ahmed and A. R. Cook, "Analysis of freeway traffic time-series data by using box-jenkins techniques," *Transportation Research Record*, vol. 722, pp. 1–9, 1979.

[18] M. M. Hamed, H. R. Al-Masaeid, and Z. M. B. Said, "Short-term prediction of traffic volume in urban arterials," *Journal of Transportation Engineering*, vol. 121, no. 3, pp. 249–254, 1995.

[19] Y. Peng, M. Lei, J.-B. Li, and X.-Y. Peng, "A novel hybridization of echo state networks and multiplicative seasonal ARIMA model for mobile communication traffic series forecasting," *Neural Computing & Applications*, vol. 24, no. 3-4, pp. 883–890, 2014.

[20] G. E. P. Box and G. M. Jenkins, "Time series analysis, forecasting and control," *Journal of the American Statistical Association*, vol. 134, p. 1971, 2013.

[21] B. M. Williams and L. A. Hoel, "Modeling and forecasting vehicular traffic flow as a seasonal ARIMA process: theoretical basis and empirical results," *Journal of Transportation Engineering*, vol. 129, no. 6, pp. 664–672, 2003.

[22] X. Yang, Y. Zou, J. Tang, J. Liang, and M. Ijaz, "Evaluation of short-term freeway speed prediction based on periodic analysis using statistical models and machine learning models," *Journal of Advanced Transportation*, vol. 2020, Article ID 9628957, 16 pages, 2020.

[23] W.-C. Hong, Y. Dong, F. Zheng, and C.-Y. Lai, "Forecasting urban traffic flow by SVR with continuous ACO," *Applied Mathematical Modelling*, vol. 35, no. 3, pp. 1282–1291, 2011.

[24] Q. Bing, B. Gong, Z. Yang, Q. Shang, and X. Zhou, "Short-term traffic flow local prediction based on combined kernel function relevance vector machine model," *Mathematical Problems in Engineering*, vol. 2015, Article ID 154703, 9 pages, 2015.

[25] F. G. Habtemichael and M. Cetin, "Short-term traffic flow rate forecasting based on identifying similar traffic patterns," *Transportation Research Part C: Emerging Technologies*, vol. 66, pp. 61–78, 2016.

[26] Y. Xu, J. Zhang, X. Gong, K. Jiang, H. Zhou, and J. Yin, "A method of real-time traffic classification in secure access of the power enterprise based on improved random forest algorithm," *Power System Protection and Control*, vol. 44, pp. 82–89, 2016.

[27] W. Lu, Y. Rui, Z. Yi, B. Ran, and Y. Gu, "A hybrid model for lane-level traffic flow forecasting based on complete ensemble empirical mode decomposition and extreme gradient boosting," *IEEE Access*, vol. 8, pp. 42042–42054, 2020.

[28] M. G. Karlaftis and E. I. Vlahogianni, "Statistical methods versus neural networks in transportation research: differences, similarities and some insights," *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 3, pp. 387–399, 2011.

[29] J. Hua and A. Faghri, "Applications of artificial neural networks to intelligent vehicle-highway systems," *Transportation Research Record*, vol. 1453, pp. 83–90, 1994.

[30] D. Park and L. R. Rilett, "Forecasting freeway link travel times with a multilayer feedforward neural network," *Computer-Aided Civil and Infrastructure Engineering*, vol. 14, no. 5, pp. 357–367, 2010.

[31] J. Tang, X. Zhang, W. Yin, Y. Zou, and Y. Wang, "Missing data imputation for traffic flow based on combination of fuzzy neural network and rough set theory," *Journal of Intelligent Transportation Systems*, vol. 25, pp. 1–16, 2020.

[32] Z. Yang, Q. Bing, C. Lin, N. Yang, and D. Mei, "Research on short-term traffic flow prediction method based on similarity search of time series," *Mathematical Problems in Engineering*, vol. 2014, Article ID 184632, 8 pages, 2014.

[33] A. W. Kuang and Z. X. Huang, "Short-term traffic flow prediction based on RBF neural network," *Systems Engineering*, vol. 22, no. 2, pp. 63–65, 2004.

[34] P. Lingras, S. Sharma, and M. Zhong, "Prediction of recreational travel using genetically designed regression and time-delay neural network models," *Transportation Research Record*, vol. 1805, pp. 16–24, 2002.

[35] Z. Zhao, W. Chen, X. Wu, P. C. Y. Chen, and J. Liu, "LSTM network: a deep learning approach for short-term traffic forecast," *IET Intelligent Transport Systems*, vol. 11, no. 2, pp. 68–75, 2017.

[36] Q. Liu, B. Wang, and Y. Zhu, "Short-term traffic speed forecasting based on attention convolutional neural network for arterials," *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 11, pp. 999–1016, 2018.

[37] L. Li, J. Zhang, Y. Wang, and B. Ran, "Missing value imputation for traffic-related time series data based on a multi-view learning method," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 8, pp. 2933–2943, 2019.

[38] E. Chen, Z. Ye, C. Wang, and M. Xu, "Subway passenger flow prediction for special events using smart card data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 3, pp. 1109–1120, 2020.

[39] W. Lu, Z. Yi, W. Liu, Y. Gu, Y. Rui, and B. Ran, "Efficient deep learning based method for multi-lane speed forecasting: a case study in Beijing," *IET Intelligent Transport Systems*, vol. 14, no. 14, pp. 2073–2082, 2020.

[40] H.-F. Yang, T. S. Dillon, and Y.-P. P. Chen, "Optimized structure of the traffic flow forecasting model with a deep learning approach," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2371–2381, 2017.

[41] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang, "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data," *Transportation Research Part C: Emerging Technologies*, vol. 54, pp. 187–197, 2015.

[42] J. Wang, R. Chen, and Z. He, "Traffic speed prediction for urban transportation network: a path based deep learning approach," *Transportation Research Part C: Emerging Technologies*, vol. 100, pp. 372–385, 2019.

[43] H. Peng, B. Du, M. Liu et al., "Dynamic graph convolutional network for long-term traffic flow prediction with reinforcement learning," *Information Sciences*, vol. 578, pp. 401–416, 2021.

[44] Y. Wu, H. Tan, L. Qin, B. Ran, and Z. Jiang, "A hybrid deep learning based traffic flow prediction method and its understanding," *Transportation Research Part C: Emerging Technologies*, vol. 90, no. 5, pp. 166–180, 2018.

[45] Y. Zhang, Y. Zhang, and A. Haghani, "A hybrid short-term traffic flow forecasting method based on spectral analysis and statistical volatility model," *Transportation Research Part C: Emerging Technologies*, vol. 43, pp. 65–78, 2014.

[46] J. Wang, W. Deng, and Y. Guo, "New Bayesian combination method for short-term traffic flow forecasting," *Transportation Research Part C: Emerging Technologies*, vol. 43, pp. 79–94, 2014.

[47] Y. Wei and M.-C. Chen, "Forecasting the short-term metro passenger flow with empirical mode decomposition and neural networks," *Transportation Research Part C: Emerging Technologies*, vol. 21, no. 1, pp. 148–162, 2012.

[48] M. Wang, W. Daamen, S. Hoogendoorn, and B. Arem, "Connected variable speed limits control and car-following control with vehicle-infrastructure communication to resolve stop-and-go waves," *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, vol. 20, 2016.

[49] Y. Gu, W. Lu, X. Xu, L. Qin, Z. Shao, and H. Zhang, "An improved bayesian combination model for short-term traffic prediction with deep learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 3, pp. 1332–1342, 2020.

[50] L. Li, B. Du, Y. Wang, L. Qin, and H. Tan, "Estimation of missing values in heterogeneous traffic data: application of multimodal deep learning model," *Knowledge-Based Systems*, vol. 194, Article ID 105592, 2020.

[51] Y. Gu, W. Lu, L. Qin, M. Li, and Z. Shao, "Short-term prediction of lane-level traffic speeds: a fusion deep learning model," *Transportation Research Part C: Emerging Technologies*, vol. 106, pp. 1–16, 2019.

[52] E. I. Vlahogianni, "Optimization of traffic forecasting: intelligent surrogate modeling," *Transportation Research Part C: Emerging Technologies*, vol. 55, pp. 14–23, 2015.

[53] H. Qiu, R. Li, and H. Liu, "Integrated model for traffic flow forecasting under rainy conditions," *Journal of Advanced Transportation*, vol. 50, no. 8, pp. 1754–1769, 2016.

[54] G. E. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[55] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski, "A learning algorithm for Boltzmann machines," *Cognitive Science*, vol. 9, no. 1, pp. 147–169, 1985.

[56] G. E. Hinton, "Learning multiple layers of representation," *Trends in Cognitive Sciences*, vol. 11, no. 10, pp. 428–434, 2007.

[57] W. Zheng, D.-H. Lee, and Q. Shi, "Short-term freeway traffic flow prediction: bayesian combined neural network approach," *Journal of Transportation Engineering*, vol. 132, no. 2, pp. 114–121, 2006.

[58] T. Hastie, R. Tibshirani, and J. Friedman, "Ensemble learning," *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Springer, Berlin, Germany, pp. 605–624, 2009.

[59] M. Collins, R. Schapire, and Y. Singer, "Logistic regression, AdaBoost and bregman distances," *Machine Learning*, vol. 48, no. 1–3, pp. 253–285, 2002.

[60] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.