IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 28 July 2023, accepted 13 August 2023, date of publication 21 August 2023, date of current version 30 August 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3307184

RESEARCH ARTICLE

An EWT-EnsemLSTM-LSSA Model for Metro Passengers Volume Prediction

KAILI LIAO AND WUNENG ZHOU
College of Information Sciences and Technology, Donghua University, Shanghai 201620, China
Corresponding author: Wuneng Zhou (zhouwuneng@163.com)

This work was supported by the Natural Science Foundation of Shanghai under Grant 20ZR1402800.

ABSTRACT Metro passenger volume prediction is an incredibly significant issue when it comes to traffic flow prediction problems. The accurate prediction of passenger flow is not only beneficial but necessary to adjust public transportation systems. As such, metro passenger volume prediction has become a crucial issue in the realm of Intelligent Transportation Systems (ITS). In this paper, a novel model called EWT-EnsemLSTM-LSSA has been proposed to deal with the complex issue of passenger flow prediction. This model assembles empirical wavelet transform (EWT), long short-term memory (LSTM), support vector regression (SVR), and logistic mapping sparrow search algorithm (LSSA) to create a comprehensive and robust solution. To start with, EWT is implemented to decompose the original dataset into five wavelet time-sequence data series for further prediction. A cluster of LSTMs with varying hidden layers and neuron counts is then deployed to scrutinize and exploit the implicit information within the EWT-decomposed signals. Subsequently, the output of LSTMs is integrated into a non-linear regression method SVR. Finally, LSSA is engaged to optimize the SVR automatically. The EWT-EnsemLSTM-LSSA model is put to the test in three case studies, employing data collected from the metro of Minneapolis, America, and Hangzhou, China. The results of these experiments are truly remarkable, as they indicate that the proposed model outperforms its conventional counterparts by reducing the mean average error to 189.27 and the root mean square error to 260.36 in Minneapolis data, and the mean average error to 24.97 and the root mean square error to 41.75 in Hangzhou data.

INDEX TERMS Traffic volume prediction, time series analysis, empirical wavelet transform (EWT), long short term memory (LSTM), support vector regression (SVR), logistic mapping sparrow search algorithm (LSSA).

I. INTRODUCTION

As the economy advances, there has been a steady rise in the number of vehicles, which in turn leads to an increase in traffic congestion [1]. This has created a demand for more intelligent transportation modes to tackle the issue, and the Internet of Vehicles (IoV) is one such promising solution. It is no surprise that this has become a key focus for the development of Intelligent Transportation Systems (ITS) [2]. Vehicle volume prediction aids transport agencies in setting self-regulating traffic lights to manage vehicles passing through crossroads and advising drivers on better routes. On the other hand, passenger volume prediction is instrumental in public transportation adjustments, such as buses and metros, to mitigate congestion on public transport. To solve this complex problem, traffic volume prediction can be modeled as a sequence-to-sequence problem, which is the foundation of current research. In other words, researchers analyze and forecast future traffic volumes by using observed historical data at various time steps.

In the realm of traffic flow prediction, parametric techniques have been widely employed in early research. One of the earliest algorithms, the autoregressive integrated moving average (ARIMA), was used to predict traffic conditions as a static process. This paved the way for several variants of ARIMA that were later proposed to improve traffic volume

The associate editor coordinating the review of this manuscript and approving it for publication was Sajid Ali.

prediction accuracy. For instance, Moshe [3] and Hamed [4] utilized ARIMA with autoregressive, integrated, and moving average polynomial orders of 0, 1, and 1, respectively, to forecast short-term traffic volume in expressway areas and urban arterial roads. To further enhance prediction accuracy, other models such as Seasonal ARIMA (SARIMA), Kohenen ARIMA (KARIMA) [5], and Vector ARIMA (VARIMA) [6] were introduced with additional seasonal parameters or residual modifications. Additionally, some researchers proposed the historical average (HA) [7] and [8], using the average of historical traffic data for future traffic volume prediction. In further attempts to increase prediction accuracy, Kalman filtering (KF) was employed in traffic prediction by Okutani [9], and a modified version of KF that utilized only the last two days' data for inputs was proposed by Kumar [10]. Emami et al. [11] proposed a method based on adaptive Kalman filters, whose performance had a significant margin with close to 11%.

As the research progressed, machine learning methods, such as K-Nearest Neighbor (KNN) and Support Vector Regression (SVR), were popular in non-parametric techniques research. Interestingly, KNN was used for the first time in traffic flow prediction by David [12]. Then, Unsok et al. [13] used the mutual information algorithm (MI) to transfer historical traffic time series and the spatial-temporal correlations into a traffic state vector, which was then applied to a KNN to handle the traffic state vector. On the other hand, Tang et al. [14] proposed a hybrid model combining the denoising method and support vector regression (SVR), to improve traffic volume prediction accuracy. Tempelmeier et al. [15] used KNN, SVR, and Ridge regression algorithms to predict the spatial dimension of the effect of events. These methods were a breakthrough in the field of traffic flow prediction, and the researchers put great effort into exploring these models.

With the advent of big data techniques, data-driven deep learning (DL) has been gaining traction in the realm of traffic volume forecasting. DL techniques are broadly classified into deep neural networks (DNN), convolutional neural networks (CNN), recurrent neural networks (RNN), autoencoder (AE), and their various permutations, which have been shown to outperform traditional methods. The efficacy of CNN methods has been demonstrated in extracting the spatial features tasks by Wu et al., [16]. Fouladgar et al. [17] proposed a decentralized method based on a CNN to predict the congestion state by the current state of neighboring stations. Zhang et al. [18] considered the weather and event data as the extra information based on a super deep structure CNN with 100 layers even up to 1000+ layers.

Lu, et al. [19] combined a multicast convolutional block with a stacked LSTM block to address the spatial dependencies of traffic data with high-dimensional temporal characteristics. Wang et al. [20] proposed a bi-directional memory LSTM scheme to capture the deep characteristics of the traffic flow, make the most of the traffic flow data over time and introduce climatological data. Majumdar et al. [21] turned to the LSTM architecture to forecast the spread of congestion on a road network by collecting speed data from sensing devices in two zones and predicts the spread of congestion every five minutes.

As the demand for accurate traffic forecasting has increased, researchers have been exploring combinations of forecasting algorithms to enhance the accuracy of traffic volume prediction. Wang et al. [22] proposed a deep polynomial neural network (DPNN) combined with SARIMA, which showed excellent performance in enhancing clarity on the spatial-temporal relationship in its deep architecture. Additionally, Peng et al. [23] came up with a dynamic graph recurrent convolutional neural network (Dynamic-GRCNN) that could deeply capture the spatial-temporal traffic flow features on urban passenger traffic flows. Incorporating fuzzy logic, Hou et al. [24] combined WNN and ARIMA to predict short-term traffic flow, while Cui et al. [25] utilized a new capsule network (CapsNet) and a nested LSTM (NLSTM) to forecast transportation network speed in Beijing by capturing the spatial features of traffic networks and the hierarchical temporal dependencies. To deal with high-impact traffic flow values of remarkably long sequence time steps, Yang et al. [26] added an attention mechanism to LSTM. Furthermore, Lin et al. [27] and Wei et al. [28] applied AE to obtain the internal relationship of traffic flow from upstream and downstream traffic flow data, and then utilized LSTM to analyze acquired characteristic data and the historical data to predict future traffic flow data. Kong et al. [29] constructed a novel Restricted Boltzmann Machine (RBM) to improve the performance of DBN for short-term traffic flow data. Finally, Zhou et al. [30] proposed δ-agree AdaBoost stacked autoencoder to improve the accuracy of short-term traffic flow forecasting.

Nevertheless, the previously mentioned techniques can be susceptible to the selection of hyperparameters, leading to suboptimal results. Therefore, this study presents a novel approach, EWT-EnsemLSTM-LSSA, which combines ensemble LSTM with a logistic mapping sparrow search algorithm to overcome this issue and enhance accuracy and efficiency. The main contributions of this paper are manifold:

1) To expose the latent features of the original data, we utilize the empirical wavelet transform method (EWT) to decompose the data.
2) We introduce an ensemble learning model, EnsemLSTM, which integrates LSTM and SVR to leverage time-series features and improve performance.
3) The Logistic mapping Sparrow search algorithm (LSSA) is employed to eliminate the impact of hyperparameters on the forecasting model.
4) Three case studies, comprising model experiments, ablation experiments, and parameter experiments, are performed to verify the effectiveness of the proposed approach. The results of the statistical analyses show the superiority of the proposed model.

The remainder of this article is arranged as follows. The correlated methodologies used in this paper are introduced

in' Methodology'. 'The proposed forecasting model' presents the proposed EWT-EnsemLSTM-LSSA model in detail. 'Case study' records the processes and results of our experiments. Finally, in 'conclusion and future work', conclusions and future works are summarized.

II. METHODOLOGY

A. DATA CLEANING

Data cleaning is a critical aspect of big data analysis, without which the dataset may contain missing or incorrect values. To maintain dataset consistency, missing values must be filled in. In this study, to avoid the distortion caused by missing data, we use the conditional mean completer method (CMC) to fill in the gaps in traffic flow data. The CMC method leverages the feature of traffic flows and uses the mean values of the same time across different weeks to fill in the missing values. This method ensures that the missing data are replaced with meaningful values that maintain the overall consistency of the dataset.

B. EMPIRICAL WAVELET TRANSFORM

In the realm of signal processing, one technique that has been gaining increasing attention and admiration is the Empirical Wavelet Transform (EWT) [31]. As a relatively new method, EWT was first proposed in 2013, and since then, has continued to grow in popularity due to its impressive ability to adaptively analyze time-frequency components. EWT's power lies in its ability to combine and inherit the best attributes of both Wavelet Transform (WT) and Empirical Mode Decomposition (EMD), ultimately making it a formidable tool for signal analysis. EWT breaks down the Fourier spectrum into several consecutive intervals, then employes a group of constructed wavelet filters to filter each interval. After undergoing this process, the processed signals are then reconstructed as a set of components. This, in turn, allows for a more in-depth and comprehensive analysis of the signal components. As for the intricate details of the EWT algorithm, it is a complex process that requires a deep understanding of signal processing. Nonetheless, it is an incredibly efficient and accurate technique that is suitable for a wide range of applications, including audio, image, and speech processing, among others. With its promising potential and undeniable versatility, EWT is an impressive and exciting addition to the world of signal processing.

Firstly, for a given signal $f(t)$, the corresponding Fourier support interval $[0, \pi]$ is divided into N continuous sub-intervals by N components, while the boundaries are presented as $\omega_0 = 0$ and $\omega_N = \pi$. The filter in each sub-interval can be defined as $\Lambda_n = [\omega_{n-1}, \omega_n]$, where $\bigcup_{n=1}^{N} \Lambda_n = [0, \pi]$, $\forall n > 0$. The empirical wavelets function $\widehat{\psi}_n(\omega)$ and the empirical scaling function $\widehat{\phi}_n(\omega)$ are defined by Eqs. (1) and (2), as shown at the bottom of the next page, respectively, where the proportional relationship between τ_n and ω_n is $\tau_n = \gamma\omega_n$, and the proportionality coefficient γ satisfies $0 < \gamma < 1$. The function $\beta(\cdot)$ is an arbitrary function which is satisfied $\beta(x) + \beta(1 - x) = 1, x \in (0, 1)$, with $\beta(x) = 0, x \leq 0$, and $\beta(x) = 1, x \geq 1$. The most used in the literature is $\beta(x) = x^4(35 - 84x + 70x^2 - 34x^3)$.

Using classical WT for reference, the corresponding mathematical expression in EWT is defined as Eqs.(3) and (4), as shown at the bottom of the next page, where $W_f^{\varepsilon}(n, t)$ is the detail coefficient of the empirical wavelet, $W_f^{\varepsilon}(0, t)$ is the approximation coefficient, and $F^{-1}[\cdot]$ is the inverse Fourier transformation.

Finally, the original signal can be reconstructed as Eq.(5), as shown at the bottom of the next page, where $\hat{W}_f^{\varepsilon}(0, \omega)$ and $\hat{W}_f^{\varepsilon}(n, \omega)$ are the Fourier transformations of $W_f^{\varepsilon}(0, t)$ and $W_f^{\varepsilon}(n, t)$, respectively; $*$ is convolution symbol. Thus, the empirical mode is given by Eqs. (6) and (7).

$$f_0(t) = \omega_f^{\varepsilon}(0, t)^* \varphi_1(t), \tag{6}$$

$$f_k(t) = \omega_f^{\varepsilon}(k, t)^* \varphi_k(t). \tag{7}$$

After EWT decomposition, a given signal $f(t)$ can be expressed as an intrinsic mode function:

$$f(t) = \sum_{k=0}^{N} f_k(t). \tag{8}$$

C. LONG SHORT-TERM MEMORY

In the field of artificial intelligence (AI), Long Short-Term Memory (LSTM) [32] is a widely used model to tackle the notorious vanishing and exploding gradient problem in recurrent neural networks (RNN). LSTM achieves this by introducing a memory cell in its cell structure, which is responsible for preserving information over a prolonged period of time. The cell structure of LSTM includes a trio of gates, including input, forget, and output gates, which facilitate the flow of information within the neural network. By utilizing this innovative architecture, LSTM can efficiently store, forget, and retrieve information from its memory cell. This allows the network to accurately determine whether to remember or forget the previous state, based on the context of the current input. The structure of the LSTM processor unit is shown in Fig. 1.

The computing step of forecasting is shown below:

$$i_t = \sigma(W_{xi} \cdot x_t + W_{hi} \cdot h_{t-1} + W_{ci} \cdot c_{t-1} + b_i), \tag{9}$$

$$f_t = \sigma(W_{xf} \cdot x_t + W_{hf} \cdot h_{t-1} + W_{cf} \cdot c_{t-1} + b_f), \tag{10}$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot tanh(W_{xc} \cdot xt + W_{hc} \cdot h_{t-1} + b_c), \tag{11}$$

$$o_t = \sigma(W_{xo} \cdot x_t + W_{ho} \cdot h_{t-1} + W_{co} \cdot c_{t-1} + b_o), \tag{12}$$

$$h_t = o_t \cdot tanh(c_t), \tag{13}$$

where i_t, f_t, c_t and o_t are the vectors for input gate, forget gate, cell state, and output gate, respectively; h_t is the output variable; W_{xi}, W_{hi}, W_{ci} are respectively the corresponding weight coefficients in input gate; W_{xf}, W_{hf}, W_{cf} are respectively the corresponding weight coefficients in forget gate; W_{xc}, W_{hc} are respectively the corresponding weight coefficients in cell state; W_{xo}, W_{ho}, W_{co} are respectively the corresponding weight coefficients in output gate; b_i, b_f, b_c and b_o are

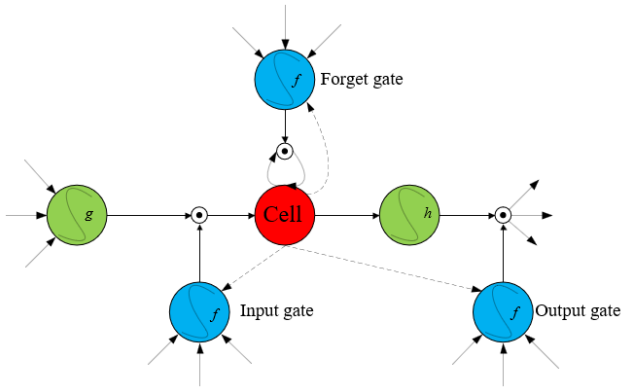


FIGURE 1. The units of LSTM.

respectively the corresponding bias vectors; σ is the sigmoid function used for the activation function between neurons.

D. SUPPORT VECTOR REGRESSION

The support vector regression (SVR) algorithm [33] is a popular method for nonlinear regression tasks. Given a training sample $D = (x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)$, where each input sample x_i belongs to the m-dimensional space R^m and each output sample y_i is a scalar, the goal of SVR is to find a regression model $f(x)$ that approximates the true relationship between x and y as closely as possible. Specifically, SVR aims to minimize the total loss $Loss[f]$ between the predicted values $f(x_i)$ and the corresponding true values y_i, subject to the constraint that the difference between the predicted and true values should be less than or equal to a given value ϵ, which defines the radius of an interval zone centered around $f(x)$ in the high-dimensional space. The process of SVR involves finding the optimal value of ϵ that maximizes the interval radius, while simultaneously minimizing the total loss. The detailed description of this process is presented below.

$$f(x) = W^T x + b, \tag{14}$$

$$Loss[f] = \frac{1}{2}\|W\|^2 + C\sum_{i=1}^{m} L(x_i, y_i, f(x_i)), \tag{15}$$

where W, b are the regression coefficients vector and bias term in regression model $f(x)$; C is the punishment coefficients in total loss $Loss[f]$, $L(x_i, y_i, f(x_i))$ is the ϵ-insensitive loss function.

In order to release the limitation of the function, an ingenious ploy is employed by introducing the flexible and adaptable slack variables ξ in lieu of the rigid and inelastic ϵ in an actual mission. Thus, the task can be transformed and reshaped into a more malleable and amenable form, as follows.

$$\begin{aligned} \min_{w,b,\xi_i,\hat{\xi}_i} \ & \frac{1}{2}\|W\|^2 + C\sum_{i=1}^{m}(\xi_i, \hat{\xi}_i), \\ s.t. \ & f(x_i) - y_i \le \epsilon + \xi_i, \\ & y_i - f(x_i) \le \epsilon + \hat{\xi}_i, \\ & \xi_i \ge 0, \hat{\xi}_i \ge 0, \quad i = 1, 2, \ldots, m. \end{aligned} \tag{16}$$

where ξ and $\hat{\xi}_i$ are the slack variables, W and C are the regression coefficients vector and the punishment coefficients.

It is not possible to directly solve the eq.(16) for the reason that W is linear and inseparable. A common resolution is to map the W into high-dimension space where W is transferred into linearly separable, which is the thought of kernel methods. By leveraging the power of kernel tricks that have been proven to be effective in the context of support vector machine (SVM), we can introduce a similar approach in support vector regression (SVR) to enhance its performance.

$$\widehat{\phi}_n(\omega) = \begin{cases} 1 & \text{if } |\omega| < \omega_n - \tau_n \\ \cos\left[\frac{\pi}{2}\beta\left(\frac{1}{2\tau_n}(|\omega| - \omega_n + \tau_n)\right)\right] & \text{if } \omega_n - \tau_n \leqslant |\omega| < \omega_n + \tau_n \\ 0 & \text{otherwise}, \end{cases} \tag{1}$$

$$\widehat{\psi}_n(\omega) = \begin{cases} 1 & \text{if } \omega_n + \tau_n \leqslant |\omega| < \omega_{n+1} - \tau_{n+1} \\ \cos\left[\frac{\pi}{2}\beta\left(\frac{1}{2\tau_n}(|\omega| - \omega_{n+1} + \tau_{n+1})\right)\right] & \text{if } \omega_{n+1} - \tau_{n+1} \leqslant |\omega| < \omega_{n+1} + \tau_{n+1} \\ \sin\left[\frac{\pi}{2}\beta\left(\frac{1}{2\tau_n}(|\omega| - \omega_n + \tau_n)\right)\right] & \text{if } \omega_n - \tau_n \leqslant |\omega| < \omega_n + \tau_n \\ 0 & \text{otherwise}, \end{cases} \tag{2}$$

$$W_f^{\varepsilon}(n,t) = \langle f, \psi_n \rangle = \int f(\tau)\overline{\psi_n(\tau - t)}d\tau = F^{-1}\left[f(\omega)\overline{\psi_n(\omega)}\right], \tag{3}$$

$$W_f^{\varepsilon}(0,t) = \langle f, \varphi_1 \rangle = \int f(\tau)\overline{\varphi_1(\tau - t)}d\tau = F^{-1}\left[f(\omega)\overline{\varphi_1(\omega)}\right], \tag{4}$$

$$f(t) = W_f^{\varepsilon}(0,t)^*\phi_1(t) + \sum_{n=1}^{N} W_f^{\varepsilon}(n,t)^*\psi_n(t) = F^{-1}\left(\hat{W}_f^{\varepsilon}(0,\omega)^*\widehat{\phi}_1(\omega) + \sum_{n=1}^{N}\hat{W}_f^{\varepsilon}(n,\omega)^*\widehat{\psi}_n(\omega)\right). \tag{5}$$

The revised regression model takes a new form, which is achieved through the use of the kernel function to calculate the dot product between the input data points. This enables the SVR algorithm to capture non-linear relationships that might exist between the input and output variables, thereby improving the accuracy of the regression model.

$$K(x_i, x_j) = exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right), \tag{17}$$

$$f(x) = \sum_{i=1}^{m}(\alpha_i - \alpha_i^*) \cdot K(x_i, x_j) + b,$$

$$s.t.\ \alpha_i \geq 0, \quad \alpha_i^* \geq 0, \quad \sum_{i=1}^{m}(\alpha_i - \alpha_i^*) = 0, \tag{18}$$

where α_i and α_i^* are the Lagrange multipliers; $K(x_i, x_j)$ is the radial basis function (RBF), which is used for the kernel function.

E. SPARROW SEARCH ALGORITHM

The Sparrow search algorithm (SSA) [34] is a novel swarm intelligence algorithm employed in parameter optimization, which mimics the foraging behavior of sparrows. In this algorithm, the sparrow population is marked as X with a particular form as shown in Eq. (19).

$$X = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,d} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,d} \\ \cdots & \cdots & & \cdots \\ x_{n,1} & x_{n,2} & \cdots & x_{n,d} \end{bmatrix}, \tag{19}$$

where n is the number of the sparrow population, and d is the dimension of searching space, which means $X \in \mathbb{R}^{n\times d}$.

Fitness function F_X, shown in Eq.(20), is utilized to measure the foraging performance and point out the direction of new positions' updating.

$$F_X = \begin{bmatrix} f_1 \\ f_2 \\ \cdots \\ f_n \end{bmatrix} = \begin{bmatrix} f([x_{1,1}, x_{1,2}, \ldots, x_{1,d}]) \\ f([x_{2,1}, x_{2,2}, \ldots, x_{2,d}]) \\ \cdots \\ f([x_{n,1}, x_{n,2}, \ldots, x_{n,d}]) \end{bmatrix}, \tag{20}$$

where the function f takes all dimensions of an individual sparrow into the calculation, and $F_X \in \mathbb{R}^{n\times 1}$.

In each iteration of the SSA, the fitness value is sorted to identify the top-performing sparrows, who are then referred to as the producers. These feathered friends occupy the upper echelons of the sorted fitness value, indicating that they have successfully foraged for a more nutritious food source. The updating strategy of producers is shown as Eq.(21).

$$x_{i,d}^{t+1} = \begin{cases} x_{i,d}^t \cdot exp(\frac{i}{\alpha \cdot T}), & if\ R_2 < ST \\ x_{i,d}^t \cdot Q \cdot L, & if\ R_2 \geqslant ST, \end{cases} \tag{21}$$

where i and d represent the indexes of sparrows and the dimensions, respectively; t is the index of iteration time and T is the maximum number of t; $\alpha \in (0, 1]$ is a uniform random number, and Q is a random number obeying the normal distribution; L is a matrix of all ones with dimension d, which is used to broadcast Q into d dimension; $R_2 \in [0, 1]$ and $S_T \in [0.5, 1]$ are the warming value and the danger threshold, respectively.

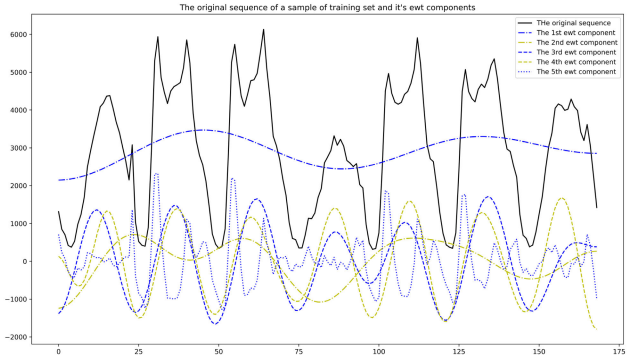


FIGURE 2. The original data and the EWT-processed data.

On the other hand, scroungers are subpar sparrows, as they have lower fitness values. A portion of the scroungers will embark on a journey to explore nearby producers to discover better food sources. In contrast, some scroungers will take the risk of venturing into the unknown to seek new food sources. The updating strategy of scroungers is shown in eq.(22).

$$x_{i,d}^{t+1} = \begin{cases} Q \cdot exp(\frac{x_{w(d)}^t - x_{i,d}^t}{A \cdot T}), & if\ i > \frac{n}{2} \\ x_{cb(d)}^{t+1} + |x_{i,d}^t - x_{cb(d)}^{t+1}| \cdot A^\dagger \cdot L, & otherwise, \end{cases} \tag{22}$$

where $x_{w(d)}^t$ is the worst position among populations at last iteration in d-dimension; $x_{cb(d)}^{t+1}$ is the best position at current iteration in d-dimension; $A \in \mathbb{R}^{1\times d}$ consists with elements 1 or -1 randomly, and $A^\dagger$ is the pseudo-inverse of A.

Some sparrows are being able to detect early warning in the whole population. Their updating strategy is shown in Eq.(23).

$$x_{i,d}^{t+1} = \begin{cases} x_{cb(d)}^t + \beta(x_{i,d}^t - x_{cb(d)}^t), & if\ f_i \neq f_g \\ x_{i,d}^t + K(\frac{x_{i,d}^t - x_{w(d)}^t}{|f_i - f_w| + \epsilon}), & if\ f_i = f_g, \end{cases} \tag{23}$$

where β is a step size control parameter obeying the standard normal distribution; K is a step size control parameter in the interval $[-1, 1]$; f_w and f_g are the fitness values at the worst and global best positions at the current interaction; ϵ is a minimal constant to prevent the denominator from being 0.

III. THE PROPOSED FORECASTING MODEL

The overall structure of the proposed model is shown in Fig.3(a). Firstly, CMC is utilized to fill missing value in the original data and the EWT is applied to transform time-varying data into these modal components to eliminate the effects of certain interfering signals. Fig.2 presents both original and EWT-proposed data. In fig.2, it can be observed that the original sequential data, which is plotted in black line, has an initial periodic regularity. However, there are irregular errors in the sequence. By utilizing EWT, the original

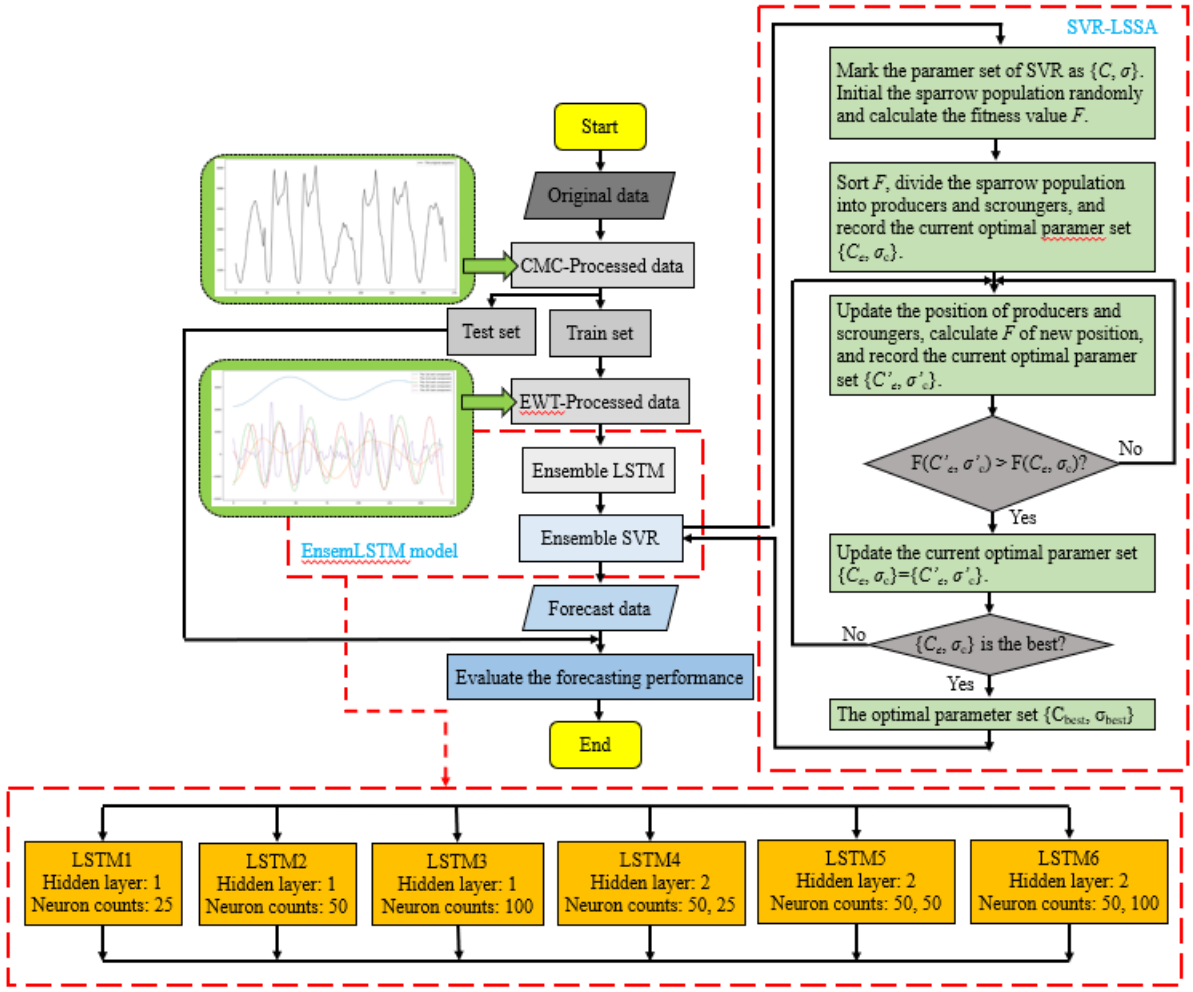


FIGURE 3. The structure of proposed model.

TABLE 1. The forecasting results on metro passenger data in minneapolis.

Model	MAE	RMSE
EWT-EnsemLSTM-LSSA	**189.27**	**260.36**
ARIMA	435.51	633.60
Wavelet-LSTM	211.60	302.90
BWA	222.58	320.07
SVR-LSTM	255.48	306.19

TABLE 2. The forecasting results on metro passenger data in Hangzhou.

Model	MAE	RMSE
EWT-EnsemLSTM-LSSA	**24.97**	**41.75**
ARIMA	50.51	101.30
Wavelet-LSTM	26.13	44.30
BWA	25.57	43.98
SVR-LSTM	31.02	53.57

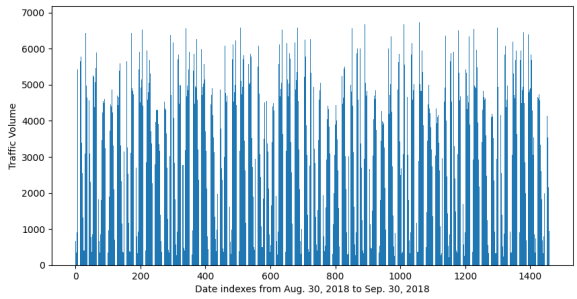
data is decomposed into five sub-sequential data. Four sub-sequential data are shown great regularity, and the range of the last one has been largely reduced after EWT.

Then, motivated by Ensemble learning which is a technique that combines multiple methods to overcome the shortages in a single model, a nonlinear ensemble time series prediction method, named EnsemLSTM, is proposed in this paper. The ensemble learning structure, combining the strengths of LSTM and SVR, is shown in the orange dotted frame of Fig.3(b). In the structure, SVR replaces the original output layer of LSTM, to enhance the non-linear forecasting capability.

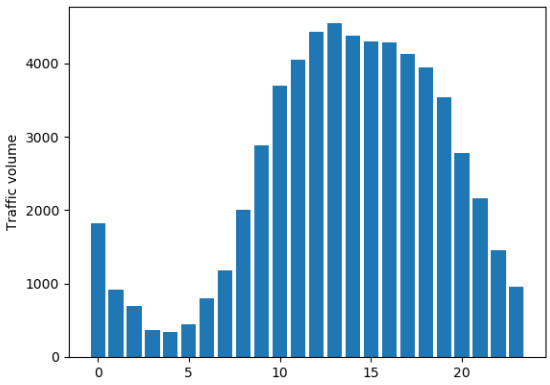
However, the forecasting performance of deep learning models can be affected by hyper-parameters such as the number of hidden layers and the counts of neurons in each hidden layer. With the art of setting hyper-parameters still being a mystery, the only way out of this dilemma is to perform trial-and-error experiments. To deal with the problem, swarm intelligence algorithms, such as SSA, are commonly used. Nevertheless, the performance of SSA is largely influenced by its initial distribution. To overcome the shortage, logistic mapping is introduced to improve SSA (LSSA).

Note the initial location of sparrow $x_{ij}(i = 1, 2, \ldots, N.j = 1, 2, \ldots, D)$, the logistic mapping location is given by eq.(24) - (26):

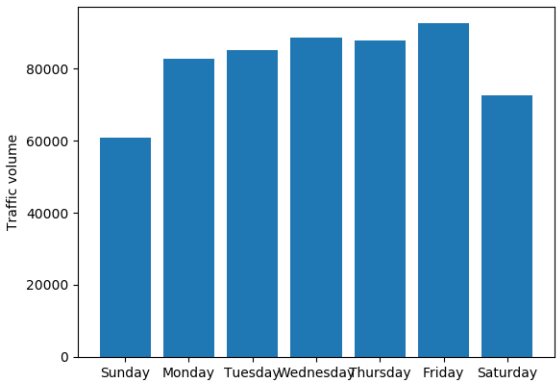
$$x_{ij}(0) = \frac{x_{ij} - lb}{ub - lb}, \tag{24}$$



(a) The traffic volume used in this paper.



(b) The traffic volume in one day



(c) The traffic volume in a week

FIGURE 4. The traffic volume in one day and a week.

$$x_{ij}(t) = \mu x_{ij}(t-1)(1 - x_{ij}(t-1)), \tag{25}$$

$$x'_{ij} = lb + x_{ij}(t) * (ub - lb), \tag{26}$$

where lb and ub are the lower bound and the upper bound of x_{ij}, t presents the times of logistic mapping, and μ is the control parameter.

Note that the dimension of the input sequence is d, and the length of the input sequence is N. The time complexity of EWT is $\mathcal{O}(Nd)$, and the time complexity of LSTM is $\mathcal{O}(Nd^2)$. Note that the updating time of the LSSA is M, thus the time complexity of EnsemLSTM-LSSA is $\mathcal{O}(MNd^2)$. Due to the combination of EWT and EnsemLSTM-LSSA is a linear stack, the time complexity of EWT-EnsemLSTM-LSSA is $\mathcal{O}(MNd^2)$. The EWT-EnsemLSTM-LSSA model can provide more accurate forecasting results which has been proofed in Sec.IV.

TABLE 3. The forecasting results of case study 2.

Model	MAE	RMSE
EWT-EnsemLSTM-LSSA	**189.27**	**260.36**
w/o LSSA (EWT-EnsemLSTM)	211.38	364.31
w/o EWT (EnsemLSTM-LSSA)	197.07	359.06
w/o EWT and LSSA (EnsemLSTM)	217.95	329.76
w/o SVR-LSSA (EWT-LSTM)	228.18	320.49
LSTM (baseline)	345.91	491.86
SVR (baseline)	548.89	779.56

IV. CASE STUDY

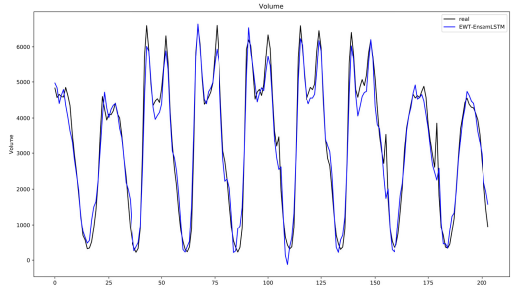
This section presents the forecasting performance of the proposed model. Especially to deserve to be mentioned, all the experiments were carried out in the python3.6 environment on a 1.6 GHz PC with process I5-8250U, and 20 GB RAM.

A. DATASET DESCRIPTION

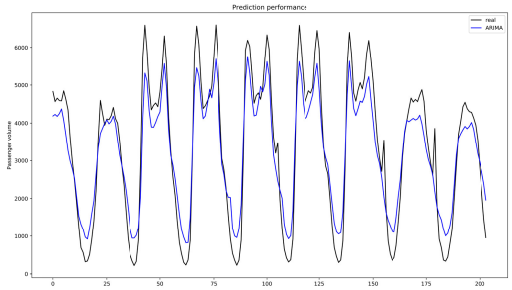
Two metro passenger volume datasets are employed to the proposed model. The first one is collected in Minneapolis, America, ranging from August 1, 2018, to September 30, 2018, was the key objective of this study. The data, which is initially collected, was processed in a one-hour sequence for the prediction model. Consequently, there are 1464 records after CMC ($(31 + 30) \times 24$), and the data distribution is illustrated by the histogram in Fig. 4(a), in which the max records is 6675 while the min is 274. The traffic value would display a certain regularity. To preliminarily demonstrate the underlying regular pattern, the metro passenger volumes in one day and one week were displayed in Fig.4. It is evident from Fig.4(b) and Fig.4(c) that the traffic volume is high during the day and low at night, as well as high on workdays and low on weekends. The second one is collected in Hangzhou, China, which ranges from Jun 1, 2019 to Jan 25, 2019. This dataset is recorded by a time interval of 15-minute, and there are 73 intervals in a day. The 1825(73×25) pieces of data are divided into 1650 groups in groups of 4 data, of which the number of training sets, validation sets and test sets is 1188, 132, and 330, respectively.

B. EVALUATION INDICES

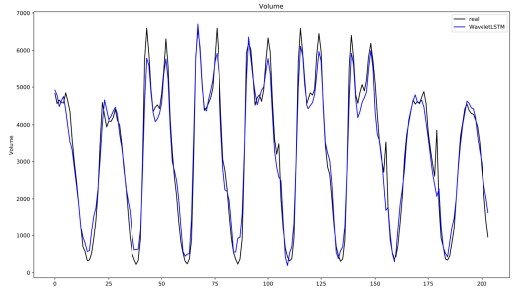
This article intends to measure the forecasting performance of the proposed model by utilizing two well-known statistical metrics, which have been widely adopted in previous research endeavors [35]. These two measures are of paramount importance in evaluating the performance of the forecasting model. Therefore, they are defined with great care and precision as follows.



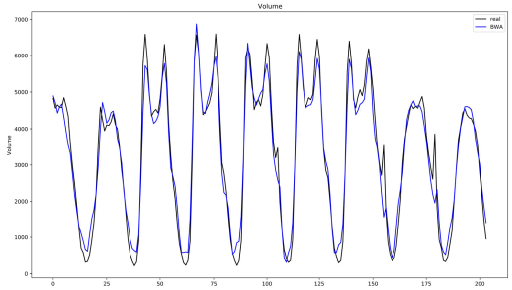
(a) The forecasting volume of EWT-EnsemLSTM.



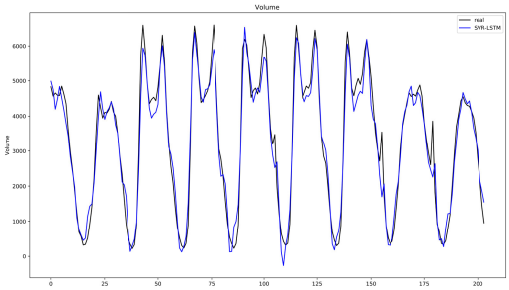
(b) The forecasting volume of ARIMA.



(c) The forecasting error of Wavelet-LSTM.



(d) The forecasting error of BWA.



(e) The forecasting volume of SVR-LSTM.

FIGURE 5. The forecasting performance.

Mean absolute error (MAE) is

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|f(i) - h(i)|. \tag{27}$$

Root mean square error (RMSE) is

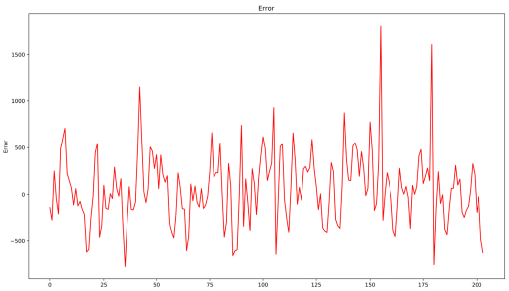
$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(f(i) - h(i))^2}. \tag{28}$$

where $f(i)$ is predicted value and $h(i)$ is actual value at time i. Furthermore, N is the total number of the dataset.

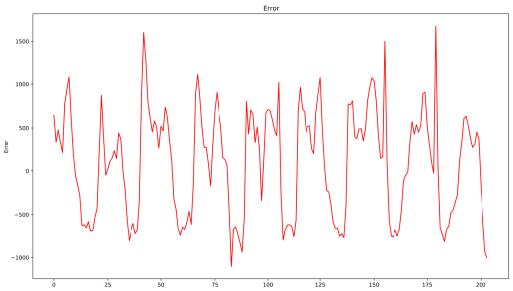
C. CASE STUDY 1: MODEL EXPERIMENT

In this case study, we aim to evaluate the effectiveness of our proposed model in comparison with various single forecasting methods. These methods include the Autoregressive Integrated Moving Average (ARIMA), Bi-LSTM method based on wavelet and attention (BWA) model [36], wavelet-LSTM [37], and fusion SVR-LSTM [33]. To process the whole dataset, we use sliding windows of 12 steps, which results in 1453 sequence samples. The training set comprises 70% of the samples, while the test set makes up the remaining 30%. The forecast performance and errors are shown in Fig.5 and Fig.6. Here are the details of these compared models.

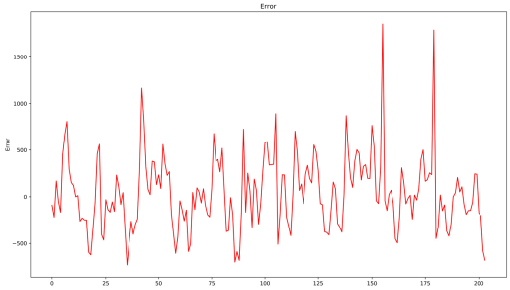
- ARIMA is a traditional and common used method in time series prediction. The parameters of ARIMA are chosen as follows: the number of autoregressive terms p, moving average terms q, and the differential order d are set as 2, 1, 1, respectively.
- Wavelet-LSTM can be divided into two parts. First, wavelet transformation is utilized to preprocess the time series data. Then, the traffic trends are learned by LSTM.



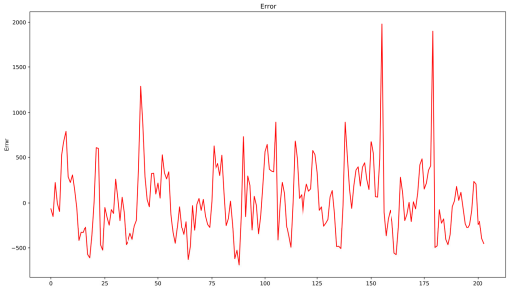
(a) The forecasting error of EWT-EnsemLSTM.



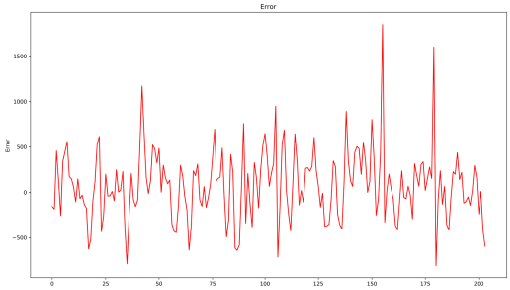
(b) The forecasting error of ARIMA.



(c) The forecasting volume of Wavelet-LSTM.



(d) The forecasting volume of BWA.



(e) The forecasting error of SVR-LSTM.

FIGURE 6. **The forecasting error.**

- BWA smooths the raw data by wavelet threshold denoising method and extracts features by adding an attention branch on Bi-direction LSTM.
- Fusion SVR-LSTM is used to compute the steady and temporal series.

In Minneapolis dataset, from Table 1, Fig. 5, and Fig. 6, the forecasting outcomes of different models are displayed, which suggest that the proposed hybrid model outperforms widely adopted prediction techniques with the least value of MAE as 189.27 and RMSE as 260.36. In contrast, Wavelet-LSTM performs the best among the compared prediction approaches with MAE as 221.60 and RMSE as 302.90. Meanwhile, ARIMA performs the worst with MAE as 435.51 and RMSE as 633.60.

In Hangzhou dataset, EWT-EnsemLSTM-LSSA is still the best performance model with MAE as 24.97 and RMSE as 41.75. However, different from the Minneapolis experiment, BWA model performs better than Wavelet-LSTM with MAE as 25.57 and RMSE 43.98. Nonetheless, it should be noted that the parameters utilized in these compared models may not be appropriate for achieving optimal forecasting performance. This prompts future research on the effect of different parameters in Case study 3.

D. CASE STUDY 2: ABLATION EXPERIMENT

A remarkable ablation study is conducted in this research to evaluate the effect of critical components on the improved performance of our proposed model using the same split ratio

TABLE 4. The forecasting results of case study 3.

Model	Parameter	Train : Test = 7:3		Train : Test = 8:2		Train : Test = 9:1	
		MAE	RMSE	MAE	RMSE	MAE	RMSE
EWT-EnsemLSTM-LSSA	———	**189.27**	**260.36**	**163.34**	**242.65**	**189.70**	**263.01**
ARIMA	2, 1, 1	435.51	633.60	431.57	632.13	399.01	595.34
	2, 0, 1	424.93	592.96	417.47	591.41	385.66	556.53
	2, 1, 2	426.67	593.40	418.66	591.86	386.67	557.05
	3, 1, 1	426.69	592.46	418.37	591.88	385.83	558.55
LSTM	10	305.86	405.82	257.67	341.02	285.75	398.55
	25	201.82	292.28	180.40	260.90	206.64	302.81
	50	191.64	281.25	177.48	260.25	200.88	288.67
	100	195.01	275.56	187.56	262.61	192.22	277.71
	50, 10	252.62	330.67	243.14	320.07	267.63	363.23
	50, 25	236.70	315.69	224.21	325.10	244.58	332.86
	50, 50	199.54	281.93	199.77	298.28	283.65	389.47
	50, 100	345.91	491.86	203.35	279.89	223.87	306.55
SVR	10, 0.05	548.09	779.56	320.96	413.70	333.51	441.96
	20, 0.05	348.32	447.43	318.75	409.52	334.95	441.67
	10, 0.25	275.37	359.20	253.81	337.61	262.70	392.51
	20, 0.25	227.32	301.86	201.96	298.22	239.73	326.19
	10, 0.65	536.24	640.65	495.15	588.55	540.42	616.21
	20, 0.65	536.24	640.65	495.15	588.55	540.42	616.21
BWA model	———	211.60	302.90	196.57	294.34	238.70	331.52
Wavelet-LSTM model	———	222.58	320.07	206.90	301.84	247.20	349.84
SVR-LSTM	———	255.48	306.19	235.77	279.30	283.64	331.90

dataset as in case study 1. Components are reduced gradually, and the variants of EWT-EnsemLSTM-LSSA are named as follows:

- **w/o LSSA**: LSSA is removed from EWT-EnsemLSTM, which is referred to EWT-EnsemLSTM. The C and σ in SVR are set as 10, 0.05 respectively.
- **w/o EWT**: EWT is removed from EWT-EnsemLSTM-SSA, which is referred to EnsemLSTM-LSSA, whose input is the original time sequence. The setting of hyper-parameters is as same as w/o LSSA.
- **w/o EWT and LSSA**: EWT and LSSA are removed from EWT-EnsemLSTM-LSSA, which is referred to EnsemLSTM. The setting of hyper-parameters is as same as w/o LSSA.
- **w/o SVR-SSA**: SVR-LSSA is removed from EWT-EnsemLSTM-LSSA, which is referred to EWT-LSTM.
- **LSTM**: the baseline.
- **SVR**: the baseline.

The MAE, RMSE, and MAPE on the test set are recorded in Table 3. From the result table, it can be observed that compared with the integral forecasting model, the performances of all variants have been diminished to varying degrees, which indicates that the effects of both the input sequence and hyper-parameter setting are fully considered in the proposed EWT-EnsemLSTM-LSSA model. The experiment of w/o LSSA suggests that the most suitable parameter setting provided by LSSA in SVR can decrease the impact of hyper-parameters. Additionally, EWT can preliminarily extract the features of input data for subsequent processing. The comparison between the variants of w/o LSSA and w/o SVR-LSSA indicates that SVR, which is set after the ensemble LSTM network, can strengthen the fitting ability of the forecasting model.

E. CASE STUDY 3: PARAMETER EXPERIMENT

In this case study, hyper-parameters' impact is investigated through parameter experiments on core hyper-parameters in models mentioned in Case study 1. The results of the parameter experiments are analyzed and discussed in Case study 3.

- For ARIMA, only one parameter is changed while fixing the other two parameters, with the initial hyper-parameters setting as $p, d, q = 2, 1, 1$. The settings in the remaining three sets of comparative experiments are $p, d, q = 2, 0, 1$, $p, d, q = 2, 1, 2$, and $p, d, q = 3, 1, 1$.
- For LSTM, the number of the hidden layer is chosen from 1, 2, and when the number of the hidden layer is 1, the neuron counts range from 10 to 100. When the number of the hidden layer is 2, the neuron counts in the second layer range from 10 to 100, with the neuron counts in the first layer fixing as 50.
- As for SVR, C ranges from 10, 20 while σ ranges from 0.05, 0.25, 0.65.
- Moreover, the split ratio of the training and test sets is $7:3$, $8:2$, and $9:1$, to evaluate the impact of dataset splitting.

Table.4 presents the results of the MAE and RMSE on the test set for various models. The findings show that the performance of the ARIMA model can be enhanced by increasing the values of p and q or decreasing d. For the LSTM model, increasing the number of hidden layers and the neuron counts can lead to a reduction in the prediction error. However, it is essential to maintain a reasonable number of hidden layers and neuron counts to avoid overfitting, which can result in a degraded model performance. The SVR experiments reveal that increasing the value of C can lead to better performance improvement than increasing σ. However,

it is crucial to note that changes in C may not necessarily enhance the prediction accuracy if σ is not at an appropriate value. Additionally, a more suitable split ratio in the dataset can contribute to more accurate forecasting performance by the models. Overall, these experiments highlight the critical role of hyper-parameters and data splitting in enhancing the performance of forecasting models.

V. CONCLUSION AND FUTURE WORK

In this article, we present a complex and novel approach to tackle the problem of traffic volume prediction, by utilizing a hybrid model consisting of EWT, ensemble learning of LSTM, SVR, and LSSA. Our proposed method aims to decompose the original traffic volume into specific model components, by utilizing EWT as a preprocessing tool, which is specifically designed to capture the characteristics of the traffic volume itself. Given that traffic volume prediction is a time series prediction problem, LSTM serves as the basic framework for our approach. LSTM is improved by ensemble learning method. Ensemble learning utilizes a set of LSTM layers with different number of hidden layers and neuron counts to extract hidden feature in the passenger volume sequence. Next, SVR is used to replace the original output layer in LSTM to enhance the non-linear forecasting capability. However, to overcome the potential issues caused by hyper-parameters, we adopt an ensemble learning approach in our model. This involves combining six LSTM networks with diverse hyper-parameter settings, and integrating an SVR structure into the top layer of LSTMs. To further improve the model's performance, we introduce LSSA by giving a more balanced initial location to accelerate the optimization speed to fine-tune the hyper-parameters in SVR. Overall, this hybrid model approach offers a unique and powerful solution to the complex problem of traffic volume prediction.

Based on the case studies, the contribution of this paper can be concluded as follows:

1) EWT can get a proper decomposition of the original dataset.
2) Ensemble learning module inherits the advantages of LSTM about extracting time-series features and SVR about nonlinear regression.
3) LSSA can find the optimal hyper-parameter setting to strengthen the forecasting accuracy.

In the experiments, EWT-EnsemLSTM-LSSA model has shown a remarkable potential in handling time series data with complex patterns, including trend and seasonal components. However, it is worth noting that this model's accuracy may be affected by other seasonal variations in different months of the year, such as changes in passenger volume due to weather or school schedules. Furthermore, it is essential to consider external factors like holidays, special events, and public health crises in selecting the time period for prediction, which may impact the model's performance.

Moreover, the problem of traffic prediction is not just a single structure issue but a complex topological problem. Thus, in the future, multi-factors and multi-node synergy prediction in the traffic area is an exciting topic for further research.

CONFLICT OF INTEREST

None of the authors have a conflict of interest to disclose.

REFERENCES

[1] T. Xu, G. Han, X. Qi, J. Du, C. Lin, and L. Shu, "A hybrid machine learning model for demand prediction of edge-computing-based bike-sharing system using Internet of Things," *IEEE Internet Things J.*, vol. 7, no. 8, pp. 7345–7356, Aug. 2020, doi: 10.1109/JIOT.2020.2983089.

[2] A. Boukerche and J. Wang, "Machine learning-based traffic prediction models for intelligent transportation systems," *Comput. Netw.*, vol. 181, Nov. 2020, Art. no. 107530, doi: 10.1016/j.comnet.2020.107530.

[3] M. Levin and Y. Tsao, "On forecasting freeway occupancies and volumes (abridgment)," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 773, no. 1, pp. 47–49, 1980.

[4] M. M. Hamed, H. R. Al-Masaeid, and Z. M. B. Said, "Short-term prediction of traffic volume in urban arterials," *J. Transp. Eng.*, vol. 121, no. 3, pp. 249–254, May 1995, doi: 10.1061/(ASCE)0733-947X(1995)121:3(249).

[5] M. Van Der Voort, M. Dougherty, and S. Watson, "Combining Kohonen maps with ARIMA time series models to forecast traffic flow," *Transp. Res. C, Emerg. Technol.*, vol. 4, no. 5, pp. 307–318, 1996, doi: 10.1016/s0968-090x(97)82903-8.

[6] J. L. Gallego, "The exact likelihood function of a vector autoregressive moving average process," *Statist. Probab. Lett.*, vol. 79, no. 6, pp. 711–714, Mar. 2009, doi: 10.1016/j.spl.2008.10.030.

[7] Y. J. Stephanedes, P. G. Michalopoulos, and R. A. Plum, "Improved estimation of traffic flow for real-time control," *Transp. Res. Rec.*, vol. 7, no. 9, pp. 28–39, 1981.

[8] A. G. Hobeika and C. K. Kim, "Traffic-flow-prediction systems based on upstream traffic," in *Proc. Vehicle Navigat. Inf. Syst. Conf.*, 1994, pp. 345–350, doi: 10.1109/VNIS.1994.396815.

[9] I. Okutani and Y. J. Stephanedes, "Dynamic prediction of traffic volume through Kalman filtering theory," *Transp. Res. B, Methodol.*, vol. 18, no. 1, pp. 1–11, 1984, doi: 10.1016/0191-2615(84)90002-X.

[10] S. V. Kumar, "Traffic flow prediction using Kalman filtering technique," *Proc. Eng.*, vol. 187, pp. 582–587, 2017, doi: 10.1016/j.proeng.2017.04.417.

[11] A. Emami, M. Sarvi, and S. A. Bagloee, "Short-term traffic flow prediction based on faded memory Kalman filter fusing data from connected vehicles and Bluetooth sensors," *Simul. Model. Pract. Theory*, vol. 102, Jul. 2020, Art. no. 102025, doi: 10.1016/j.simpat.2019.102025.

[12] G. A. Davis and N. L. Nihan, "Nonparametric regression and short-term freeway traffic forecasting," *J. Transp. Eng.*, vol. 117, no. 2, pp. 178–188, Mar. 1991, doi: 10.1061/(ASCE)0733-947X(1991)117:2(178).

[13] U. Ryu, J. Wang, T. Kim, and S. Kwak, "Construction of traffic state vector using mutual information for short-term traffic flow prediction," *Transp. Res. C, Emerg. Technol.*, vol. 96, pp. 55–71, Nov. 2018, doi: 10.1016/j.trc.2018.09.015.

[14] J. Tang, X. Chen, Z. Hu, F. Zong, C. Han, and L. Li, "Traffic flow prediction based on combination of support vector machine and data denoising schemes," *Phys. A, Stat. Mech. Appl.*, vol. 534, Nov. 2019, Art. no. 120642, doi: 10.1016/j.physa.2019.03.007.

[15] N. Tempelmeier, S. Dietze, and E. Demidova, "Crosstown traffic–supervised prediction of impact of planned special events on urban traffic," *GeoInformatica*, vol. 24, no. 2, pp. 339–370, Apr. 2020, doi: 10.1007/s10707-019-00366-x.

[16] Y. Wu, H. Tan, L. Qin, B. Ran, and Z. Jiang, "A hybrid deep learning based traffic flow prediction method and its understanding," *Transp. Res. C, Emerg. Technol.*, vol. 90, pp. 166–180, May 2018, doi: 10.1016/j.trc.2018.03.001.

[17] M. Fouladgar, M. Parchami, R. Elmasri, and A. Ghaderi, "Scalable deep traffic flow neural networks for urban traffic congestion prediction," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, May 2017, pp. 2251–2258, doi: 10.1109/IJCNN.2017.7966128.

[18] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," in *Proc. AAAI Conf. Artif. Intell.*, vol. 31, no. 1, 2017, p. 10735, doi: 10.1609/aaai.v31i1.10735.

[19] H. Lu, D. Huang, Y. Song, D. Jiang, T. Zhou, and J. Qin, "ST-TrafficNet: A spatial–temporal deep learning network for traffic forecasting," *Electronics*, vol. 9, no. 9, p. 1474, Sep. 2020, doi: 10.3390/electronics9091474.
[20] J. Wang, X. Xu, F. Wang, C. Chen, and K. Ren, "A deep prediction architecture for traffic flow with precipitation information," in *Proc. 9th Int. Conf. (ICSI)*. Shanghai, China: Springer, Jun. 2018, pp. 329–338, doi: 10.1007/978-3-319-93818-9_31.
[21] S. Majumdar, M. M. Subhani, B. Roullier, A. Anjum, and R. Zhu, "Congestion prediction for smart sustainable cities using IoT and machine learning approaches," *Sustain. Cities Soc.*, vol. 64, Jan. 2021, Art. no. 102500, doi: 10.1016/j.scs.2020.102500.
[22] W. Wang, H. Zhang, T. Li, J. Guo, W. Huang, Y. Wei, and J. Cao, "An interpretable model for short term traffic flow prediction," *Math. Comput. Simul.*, vol. 171, pp. 264–278, May 2020, doi: 10.1016/j.matcom.2019.12.013.
[23] H. Peng, H. Wang, B. Du, M. Z. A. Bhuiyan, H. Ma, J. Liu, L. Wang, Z. Yang, L. Du, S. Wang, and P. S. Yu, "Spatial temporal incidence dynamic graph neural networks for traffic flow forecasting," *Inf. Sci.*, vol. 521, pp. 277–290, Jun. 2020, doi: 10.1016/j.ins.2020.01.043.
[24] Q. Hou, J. Leng, G. Ma, W. Liu, and Y. Cheng, "An adaptive hybrid model for short-term urban traffic flow prediction," *Phys. A, Stat. Mech. Appl.*, vol. 527, Aug. 2019, Art. no. 121065, doi: 10.1016/j.physa.2019.121065.
[25] Z. Cui, R. Ke, Z. Pu, X. Ma, and Y. Wang, "Learning traffic as a graph: A gated graph wavelet recurrent neural network for network-scale traffic prediction," *Transp. Res. C, Emerg. Technol.*, vol. 115, Jun. 2020, Art. no. 102620, doi: 10.1016/j.trc.2020.102620.
[26] B. Yang, S. Sun, J. Li, X. Lin, and Y. Tian, "Traffic flow prediction using LSTM with feature enhancement," *Neurocomputing*, vol. 332, pp. 320–327, Mar. 2019, doi: 10.1016/j.neucom.2018.12.016.
[27] F. Lin, Y. Xu, Y. Yang, and H. Ma, "A spatial–temporal hybrid model for short-term traffic prediction," *Math. Problems Eng.*, vol. 2019, pp. 1–12, Jan. 2019, doi: 10.1155/2019/4858546.
[28] W. Wei, H. Wu, and H. Ma, "An AutoEncoder and LSTM-based traffic flow prediction method," *Sensors*, vol. 19, no. 13, p. 2946, Jul. 2019, doi: 10.3390/s19132946.
[29] F. Kong, J. Li, B. Jiang, and H. Song, "Short-term traffic flow prediction in smart multimedia system for Internet of Vehicles based on deep belief network," *Future Gener. Comput. Syst.*, vol. 93, pp. 460–472, Apr. 2019, doi: 10.1016/j.future.2018.10.052.
[30] T. Zhou, G. Han, X. Xu, Z. Lin, C. Han, Y. Huang, and J. Qin, "δ-agree AdaBoost stacked autoencoder for short-term traffic flow forecasting," *Neurocomputing*, vol. 247, pp. 31–38, Jul. 2017, doi: 10.1016/j.neucom.2017.03.049.
[31] J. Gilles, "Empirical wavelet transform," *IEEE Trans. Signal Process.*, vol. 61, no. 16, pp. 3999–4010, Aug. 2013, doi: 10.1109/TSP.2013.2265222.
[32] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. 25th Int. Conf. Neural Inf. Process. Syst.*, vol. 1, 2012, pp. 1097–1105, doi: 10.1145/3065386.
[33] J. Guo, Z. Xie, Y. Qin, L. Jia, and Y. Wang, "Short-term abnormal passenger flow prediction based on the fusion of SVR and LSTM," *IEEE Access*, vol. 7, pp. 42946–42955, 2019, doi: 10.1109/ACCESS.2019.2907739.
[34] J. Xue and B. Shen, "A novel swarm intelligence optimization approach: Sparrow search algorithm," *Syst. Sci. Control Eng.*, vol. 8, no. 1, pp. 22–34, Jan. 2020, doi: 10.1080/21642583.2019.1708830.
[35] A. Miglani and N. Kumar, "Deep learning models for traffic flow prediction in autonomous vehicles: A review, solutions, and challenges," *Veh. Commun.*, vol. 20, Dec. 2019, Art. no. 100184, doi: 10.1016/j.vehcom.2019.100184.
[36] M. Zhao, X. Zhang, and Y. Jin, "Wavelet embedded attentive bi-LSTM for short-term passenger flow forecasting," in *Proc. IEEE 7th Int. Conf. Big Data Comput. Service Appl. (BigDataService)*, Aug. 2021, pp. 177–183, doi: 10.1109/BigDataService52369.2021.00028.
[37] H. Lu and F. Yang, "A network traffic prediction model based on wavelet transformation and LSTM network," in *Proc. IEEE 9th Int. Conf. Softw. Eng. Service Sci. (ICSESS)*, Nov. 2018, pp. 1–4, doi: 10.1109/ICSESS.2018.8663884.

KAILI LIAO received the B.S. degree in electrical engineering and its automation from Donghua University, Shanghai, China, in 2014, where he is currently pursuing the Ph.D. degree in control science and engineering with the Department of Information Science and Technology. His research interests include time series forecasting and swarm intelligence algorithm.

WUNENG ZHOU received the B.S. degree in mathematics from Huazhong Normal University, Hubei, China, in 1982, and the Ph.D. degree in control science and engineering from Zhejiang University, Zhejiang, China, in 2005. He is currently a Professor with Donghua University, Shanghai, China. His current research interests include stability, synchronization, control for neural networks, multiagent system stability analysis, and formation control.