

## Research Article

# Driver Lane-Changing Behavior Prediction Based on Deep Learning

Cheng Wei <sup>1</sup>, Fei Hui <sup>1</sup> and Asad J. Khattak<sup>1,2</sup>

<sup>1</sup>School of Information and Engineering, Chang'an University, Xi'an, ShaanXi 710064, China

<sup>2</sup>Civil and Environmental Engineering Department, University of Tennessee, Knoxville, TN 37996, USA

Correspondence should be addressed to Fei Hui; feihui@chd.edu.cn

Received 9 November 2020; Revised 12 March 2021; Accepted 9 April 2021; Published 28 April 2021

Academic Editor: Young-Jae Lee

Copyright © 2021 Cheng Wei et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A correct lane-changing plays a crucial role in traffic safety. Predicting the lane-changing behavior of a driver can improve the driving safety significantly. In this paper, a hybrid neural network prediction model based on recurrent neural network (RNN) and fully connected neural network (FC) is proposed to predict lane-changing behavior accurately and improve the prospective time of prediction. The dynamic time window is proposed to extract the lane-changing features which include driver physiological data, vehicle kinematics data, and driver kinematics data. The effectiveness of the proposed model is validated through the experiments in real traffic scenarios. Besides, the proposed model is compared with five prediction models, and the results show that the proposed prediction model can effectively predict the lane-changing behavior more accurate and earlier than the other models. The proposed model achieves the prediction accuracy of 93.5% and improves the prospective time of prediction by about 2.1 s on average.

## 1. Introduction

Driver lane-changing behavior is a key factor in driving safety. An improper lane-changing behavior may cause a vehicle collision [1, 2] or even a traffic accident [3–5]. In [6], it was indicated that nearly 18% of the total number of traffic accidents were caused by improper lane changing. Using a prediction model in the Advanced Driver Assistance Systems (ADASs) [7–9] could reduce the risk of accidents. Therefore, a model for accurate prediction of a driver lane-changing behavior using multiple data fusion is needed. Substantial research regarding the lane-changing prediction has been conducted. At present, there are two mainstream groups of methods for prediction models of lane-changing, namely, mathematical methods and artificial intelligence approaches. One of the lane-changing prediction models based on a mathematical method was introduced in [10, 11]. Also, in [12], the logistic regression method was used in a lane-changing prediction model, wherein the distances to the front and adjacent rear vehicles, forward time-to-collision (TTC), and turn signal were taken into account, and the results showed that this model performed well in certain circumstances.

Baumann et al. [13] improved a cognitive model to characterize driver behavior in an automotive environment and proved the correlation between drivers' cognitive processes and their driving movements. Salvucci [14] introduced an adaptive control of rational cognitive structures to monitor the lane-changing process in a multilane highway environment, and their model demonstrated how cognitive architectures could facilitate understanding of driver behavior. However, some of the mentioned studies were conducted in specific traffic scenarios such as highway entrance and ramp; besides, a part of the traffic scenarios was simulated instead of a real one. Therefore, those studies may not consider the lane-changing behavior in real traffic scenarios. Moreover, the above studies achieved the prediction accuracy ranges from 80% to 85%, so it has considerable space for improvement.

With the aim to build a more intelligent lane-changing prediction model, researchers have adopted machine learning. A lane-changing behavior prediction model based on the support vector machine (SVM) classifier and Bayesian filtering (BF) was proposed in [15], and it was shown that this model could predict driver lane-changing

behavior 1.3 s in advance. In addition, a lane-changing behavior and trajectory prediction model based on the Hidden Markov Model (HMM) was presented in [16], and the results showed that this model predicted the lane-changing trajectories very well, which makes it suitable for prediction of human-like lane-changing maneuvers. Hou et al. [17] proposed a lane-changing prediction model based on fuzzy logic, which was developed for the case of a forced lane-changing under the lane-descending conditions. The prediction accuracy of this model for nonmerging behavior and merging behavior was 86.3% and 87.5%, respectively. Furthermore, a novel lane-changing intention recognition algorithm which combines the HMM and BF models was proposed in [18], where the model input consisted of three signals from the CAN bus (steering angle, lateral acceleration, and yaw rate), and the output was behavior classification. The results revealed that the HMM-BF could achieve an average recognition accuracy of 91.9%. Zheng et al. [19] proposed a machine learning-based segmentation and classification algorithm consisting of three stages. The first stage includes data preprocessing and prefiltering, and its function is to reduce noise and remove clear left and right turning events. The second stage employs a spectral time frequency analysis segmentation approach to generalize all potential time-variant lane-changing and lane-keeping candidates. The third stage includes two possible classifications: lane-changing and lane-keeping. The results showed that the average accuracy of this three-stage algorithm exceeded 83.22%. Furthermore, in [20], a dynamic Bayesian network (DBN) was used to predict the lane-changing maneuvers, and the test using the real data showed that the lane changing was detected 1 s in advance. However, the machine learning methods are not suitable for multisource data fusion, and even a single-input data may result in lower accuracy.

In recent years, the deep learning algorithms have been widely used in lane-changing prediction because of its powerful high-dimensional data processing and autonomous learning capabilities, which is in sharp contrast with conventional mathematical methods. Xie et al. [21] proposed a deep learning-based method to predict the future trajectory of vehicles and achieved good results. It is demonstrated that the deep learning model can mine potential features from high-dimensional data and also indicated the feasibility of deep learning in lane-changing research. In [22], a backpropagation (BP) neural network was used as a controller of an automatic vehicle system, and a camera image was used as a neural network input to construct a lane-changing model. The results showed that one hidden layer was enough to provide good performance of a time-varying nonlinear dynamic system. Tomar et al. [23] proposed a method based on a multilayer perceptron to predict a lane-changing trajectory accurately. The prediction results showed that this model was able to predict the future path accurately only for discrete patches of a trajectory, but not for the complete trajectory. Ding et al. [24] developed a BP neural network-based model to predict the lane-changing trajectories, and they compared prediction results of the BP

neural network with that of the Elman network. It was found that the BP neural network-based model achieved better prediction performance under different sections and generated more reliable simulation results than the Elman network-based model. In [25], a fully connected neural network was applied to predict the lane-changing behavior of drivers; especially, the network model input consisted of multivehicle data, and the prediction accuracy of more than 90% was achieved. Moreover, a multifeature fusion neural network [26] that takes into account the physiological factors such as driver's head rotation was proposed to predict driver lane change behavior and the prediction accuracy exceeded 85%, while the prospective time was 1.5 s. Dou et al. [27] introduced a prediction model based on the SVM and BP neural network, which combined the results of SVM and BP neural network to improve the prediction accuracy, and the results showed that the average combined accuracy exceeded 92%. Furthermore, an MTSDepNet using a convolution kernel to process the multivariate time series data and a fully connected neural network to classify the lane-changing behavior were designed to predict the lane changing in [28], and the accuracy of this model exceeded 91.0%. Considering the driver's driving style, Li et al. [29] proposed a lane-changing intention estimation model based on Bayesian network and Gaussian mixture model, which achieved a good prospective time, but its accuracy is low. However, the lane-changing process is determined by both the driver and traffic environment, but the aforementioned research studies did not consider all the factors affecting the driver lane-changing behavior. Also, using a fully connected network for lane-changing prediction may cause data loss which further reduces model performance.

In summary, the existing studies have the limitations of low prediction accuracy and short prospective time. Two reasons for these limitations are "data problem" and "prediction model structure problem." Aiming at these two problems, a hybrid neural network driven by multiple types of data is proposed. The first level of the hybrid network is composed of Seq2Seq, a variant of RNN [30, 31], which is mainly used for time series data processing to reduce invisible data loss. The second level consists of a fully connected neural network for data fusion and lane-changing classification. There are three contributions of this study. (1) Three different types of data including vehicle kinematics data, driver kinematics data, and driver physiological data are collected and used. (2) A hybrid network with a two-level training model is proposed to deepen the number of network layers while avoiding the problem of gradient dispersion. (3) A dynamic time window algorithm is proposed to ensure the consistency and homogeneity of the model input data and extend the prediction prospective time.

The remainder of the paper is organized as follows. Section 2 describes the data source and introduces the data processing method. Section 3 elaborates the working principle and structure of the proposed Seq2Seq-FC neural network, as well as its mathematical relationships. Section 4 validates the model generalization ability. Lastly, Section 5 concludes the paper.

## 2. Data Collection and Processing

Since the data used in this study was derived from real traffic scenarios and different sensing equipment, the original data was first filtered and segmented, and then, the timestamp alignment was conducted, and the data with the same label was extracted by a time window.

**2.1. Data Collection.** In the data collection process, the speed and acceleration of a vehicle were collected by the Cohda-wireless dedicated short-range communication (DSRC) installed on the vehicle. The steering angle and angular velocity of the steering wheel were obtained by a corner tester mounted at the steering wheel. The electroencephalogram (EEG) and driver's head movement data were obtained by the brain wave analyzer. Besides, the electrocardiogram (ECG) stemmed from a heart rate tester. The number of driver's head rotations in the horizontal direction was determined using the driving video obtained by a driving recorder. The equipment used for data acquisition is shown in Figure 1. Also, a different data was used for model training and validating. The data was collected in the same way but using different traffic routes. The route used for the collection of training data is presented in Figure 2, and the route started at the Chang'an University going through the main roads in Xi'an and ended at Xi'an Cheng'nan Passenger Transport Center. During the process of data collection, the procedure was repeated for several times. Since all data collection equipment are connected to the same PC, a timestamp is added to the header of each data based on Beijing time to synchronize the data collected by different equipment. In addition, the purchased brainwave analyzer and DSRC equipment have supporting data receiving software, which can denoise and filter the data and other self-designed equipment receive data using the serial communication protocol, and the data are denoised using the period-average padding method and PauTa criterion built into hardware. Therefore, the data output by a different equipment has been denoised and can be used directly.

It should be noted that, during the research, we found the impact of other factors on the subject vehicle (such as surrounding environmental factors, traffic environmental factors, and driving purpose) will ultimately be reflected in the driver's control of the vehicle. For example, if the number of vehicles around the subject vehicle increases, the speed of the subject vehicle will decrease. If the subject vehicle is driving on a slippery road, its overall speed will be significantly lower than the speed on a normal road. Therefore, only the subject vehicle's own data and the driver's own data were collected.

**2.2. Features Extraction.** The time window is the basis for data processing, and the data in time window should contain the data referring to the entire lane-changing process, not only a part of it. As shown in Figure 3, during the lane-changing process, the  $\beta$  bands of brain wave signal change greatly at the beginning of the lane-changing, and vehicle's steering angle tends to become stable at the end of the lane-



FIGURE 1: Equipment used for data collection.



FIGURE 2: Data collection route.

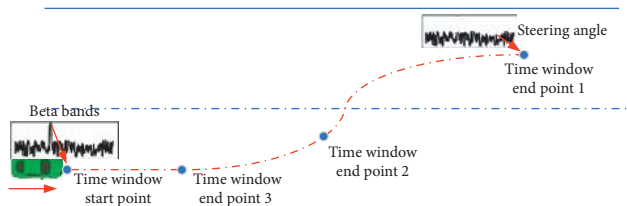


FIGURE 3: Lane-changing process.

changing process. Therefore, the point where the  $\beta$  bands change drastically is recorded as the starting point of the time window, and the stable point of the steering angle after the lane changing is completed and is recorded as the end point of the time window. Because the time taken for each lane changings is different, each lane-changing behavior has its corresponding time window. Considering the prospective time of the prediction model, the method of time window shrinking is adopted in this work. As shown in Figure 3, without considering the prediction accuracy, the endpoint of time window 2 has longer prospective time than that of the endpoint 1. Therefore, in the process of model training, the model is verified by shrinking the time window while comparing the accuracy. During data processing, the data

length of a current time window is important because the RNN-Seq2Seq used in this work requires input data of a fixed dimension. However, since a dynamic time window is used, a data extraction method is needed. The data extraction method not only ensures the consistency of input data but also minimizes data loss. Besides, the maximum speed of a driver's head in a three-dimensional space and the number of driver's head rotations in the horizontal direction within a time window are directly fed to the input of the fully connected neural network without being processed by the RNN-Seq2Seq.

**2.3. Data Processing.** Vehicle kinematics data processing. The length of the time window was dynamically adjusted, to ensure that the dimension of data extracted from each time window was equal, during the vehicle kinematics data processing, for each time series data in a time window, the maximum, minimum, average, and variance values were used. Table 1 shows the characteristic of data within a time window which were used for further data processing. When the time window was gradually shrunk, the label data of the Seq2Seq was a caudal data of the original time window, but the processing method stayed the same. As can be seen in Table 1, 15 data features were extracted and used. There are two reasons for selecting these 15 features. The first is that such feature extraction method can ensure the same dimension and consistency of model input. The second is that the selected 15 features can reflect the vehicle movement situation in a time window no matter how long the time window is, which is also the meaning of those 15 features.

Driver's physiological and kinematics data processing. The EEG data were filtered by a bandpass filter which was a Chebyshev type II filter with a lower cut-off frequency of 4 Hz and an upper cut-off frequency of 30 Hz. The  $\delta$  bands of 1–3 Hz in the EEG with an amplitude of 20–200  $\mu$ V were removed because the  $\delta$  wave appears in the human infant stage or immature mental development period or when an adult is under extreme fatigue and lethargy or anesthesia. Therefore, these bands do not change significantly when the driver considers the lane-changing. In addition, the  $\theta$  bands with a frequency of 4–7 Hz and an amplitude of 5–20  $\mu$ V, the  $\alpha$  bands with a frequency of 8–13 Hz and an amplitude of 20–100  $\mu$ V, and  $\beta$  bands with a frequency of 14–30 Hz and an amplitude of 100–150  $\mu$ V were needed. Namely, the waveforms corresponding to these three bands types changed significantly when people are in the depression state, normal state, and excited state, so they should be used as input to the prediction model because they are changeable and important pointers to the people emotions and states. Similarly, the average, maximum, and minimum values and the variance of  $\theta$ ,  $\alpha$ , and  $\beta$  bands of the EEG data within a time window were included. Six frequency domain features are extracted from each of the three EEG channels:  $\theta/(\theta + \alpha)$ ,  $\alpha/(\theta + \alpha)$ ,  $(\theta + \alpha)/\beta$ ,  $\alpha/\beta$ ,  $(\theta + \alpha)/(\theta + \beta)$ , and  $\theta/\beta$ . The reason for selecting these six features is that Martensson's research

TABLE 1: Vehicle kinematics data processing method.

Feature	Process method
Speed	Maximum, minimum, average, and variance
Acceleration	Maximum, minimum, and average
Steering angle	Maximum, minimum, average, and variance
Steering angular velocity	Maximum, minimum, average, and variance

[32] shows that these six features can reflect the driver's brain activity at a certain time and can also reflect the driver's thinking while driving. Like the method of vehicle kinematics data extraction, the selected 18 brainwave data related features can reflect the driver's brain thinking when performing lane changing.

The heart rate signal was processed similarly as the other data, and the maximum, minimum, and average values and variance of the heart rate in a time window were taken as the input data regarding the motion data of a driver's head in a time window, and we used the speed maximum value in a three-dimensional space and the number of rotations of a driver's head in the horizontal direction. These four features were not processed by the RNN, instead they were fed directly to the input of the fully connected network because, to a certain extent, the maximum speed of head movement can reflect the urgency of the driver to change lanes. For example, if the driver frequently turns his head during driving and the speed is high, the driver has a higher probability of performing lane changing. In summary, 26 data features were extracted and used as input parameters for data processing. The extracted features and their labels are shown in Table 2. And, the 26 data features in Table 2 can reflect the driving situation of the vehicle in two aspects (vehicle and driver) when input as a model. It also ensures that the dimensions of the input data are not affected by the length of the time window.

After the original data was processed as described above, the dimension of the input data will be the same in every time window regardless of its length. In this way, data consistency during model training and testing was ensured. In Figures 4 and 5, the vehicle kinematic data and the brain wave during the lane-changing process are presented, respectively. The reason why the steering angle shown in Figure 4 does not have a negative value is, when designing the steering wheel angle measuring equipment, we set its equilibrium state (middle value) to 0 and design it as a positive value regardless of turning to the right or turning to the left.

### 3. Proposed Model

As already mentioned, the proposed model is based on an RNN-Seq2Seq network and a fully connected neural network. The RNN-Seq2Seq network is used to extract the features of the original data and predict the value of the



TABLE 2: The extracted features and labels.

Parameters	Definition
$\theta_{\max} \theta_{\min} \theta_{ave} \theta_{var}$	Maximum, minimum, average and variance of theta bands
$\alpha_{\max} \alpha_{\min} \alpha_{ave} \alpha_{var}$	Maximum, minimum, average, and variance of alpha bands
$\beta_{\max} \beta_{\min} \beta_{ave} \beta_{var}$	Maximum, minimum, average, and variance of beta bands
$P_{\max} P_{\min} P_{ave} P_{var}$ $(\theta/(\theta + \alpha))(\alpha/(\theta + \alpha))$ $((\theta + \alpha)/\beta\alpha)/\beta$ $(\theta + \alpha)/((\theta + \beta)\theta/\beta)$	Maximum, minimum, average, and variance of heart rate  Six absolute power of theta, alpha, and beta bands
$\omega_{\max} \omega_{\min} \omega_{ave} \omega_{var}$	Maximum, minimum, average, and variance of steering angle
$\omega_a \omega_{a\max} \omega_{a\min} \omega_{aave} \omega_{avar}$	Maximum, minimum, average, and variance of angular velocity
$v_{\max} v_{\min} v_{ave} v_{var}$	Maximum, minimum, average, and variance of vehicle speed
$a_{\max} a_{\min} a_{ave}$	Maximum, minimum, average, and variance of vehicle acc
$H_x \max H_y \max H_z \max$	Maximum speed of the head in three dimensions
$H_c$	Number of rotations of the head within the time window

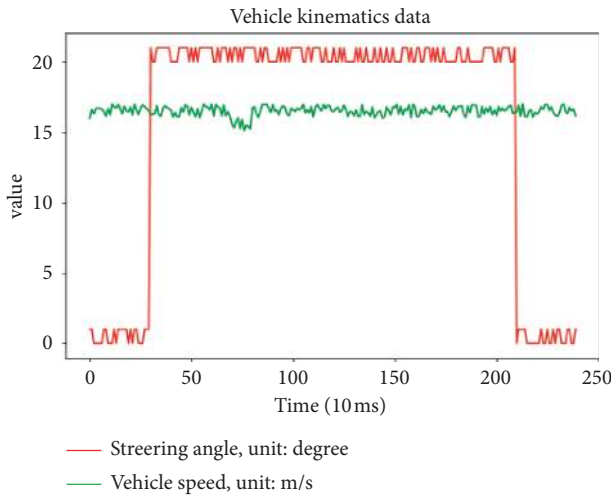


FIGURE 4: Vehicle kinematics data during the lane-changing process.

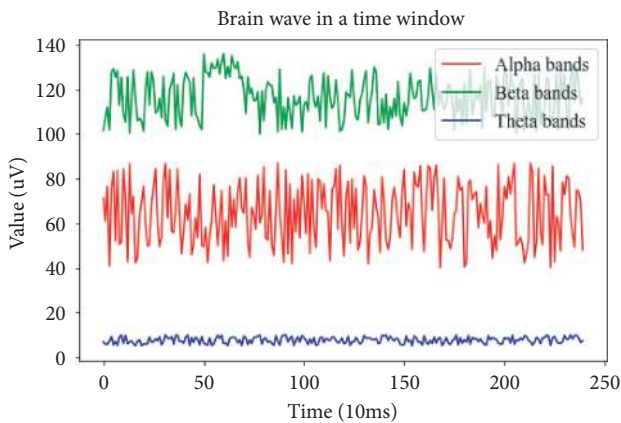


FIGURE 5: Brain wave during the lane-changing process.

next point which will be used as an input for the following fully connected neural network. In this work, 41 features are extracted and used for data processing. Except the data related to the driver's head rotation, the others features

are subjected to the RNN-Seq2Seq processing. Therefore, the Seq2Seq has 37 inputs and 37 outputs. In addition, the fully connected neural network has 41 inputs and 2 outputs. The data processing procedure is shown in Figure 6. It should be noted that when the entire prediction model is trained, the accuracy and prospective time of the model are compared by shrinking the time window.

**3.1. Seq2Seq Layer.** RNN is a kind of neural network that can model data sequences. The main advantage of this neural network type is that it can process time series data well. Since the lane-changing behavior often lasts for a long period of time, if the time series data is directly processed by a fully connected neural network, there will be inevitable data loss, which will decrease model accuracy. Additionally, when processing the time series data, an RNN considers the correlation between data in the data sequence, so the use of a data in a time window can be maximized.

Since our goal is to develop the prediction model whose input is consisted of time sequence data, an RNN structural variant, and the Seq2Seq structure, containing the encoder and decoder, is chosen, which represents an enhanced version of a normal RNN and consists of an encoder and a decoder. The computational kernel of both encoder and decoder is an LSTM (long short-term memory unit) or a GRU (gated recurrent unit).

Typical Seq2Seq structure denotes the encoder-decoder framework shown in Figure 7. The working mechanism of a Seq2Seq structure uses the encoder to map the input data to the semantic space to get a decoding vector  $c$  which represents the semantics and then use the decoder to obtain the required output.

As shown in Figure 7, the encoder-decoder framework has two inputs: one is  $x = \{x_1, x_2, \dots, x_n\}$  which represents the encoder input and the other is  $y = \{y_1, y_2, \dots, y_n\}$  which represents the decoder input. The inputs  $x$  and  $y$  are sequentially passed to the network in the respective order.

Assuming that the input sequence of the encoder is  $\{x_1, x_2, \dots, x_n\}$  and according to the RNN characteristics, the hidden state at time  $t$  in the input process is a function of the

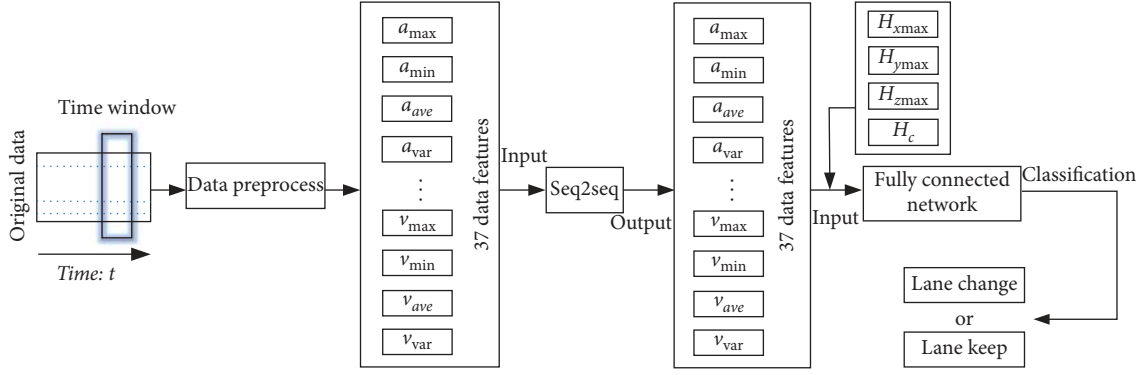


FIGURE 6: Data processing procedure.

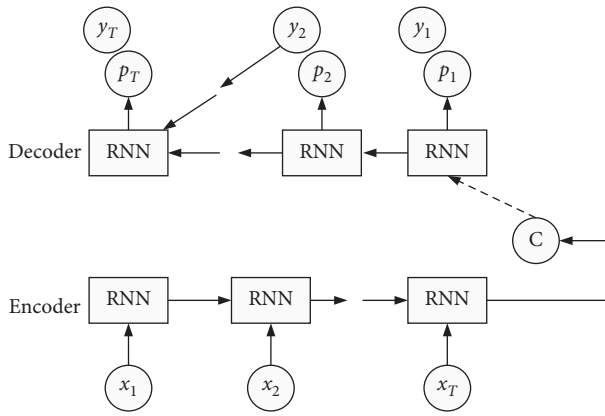


FIGURE 7: The Seq2Seq decoder-encoder based structure.

current input signal and the hidden layer output in the previous given as

$$h_t = f(h_{t-1}, x_t), \quad (1)$$

where  $f$  is a nonlinear activation function and the decoding vector  $c$  is the last state of  $h_t$ . Finally, a fixed-length decoding vector  $c$  is obtained. The decoder can be regarded as another RNN. When the decoding vector  $c$  is sequentially fed to the decoder, the decoder output at the time  $t$  can be expressed as

$$h_t = f(h_{t-1}, y_{t-1}, c). \quad (2)$$

Then, the conditional probability of  $y$  at time  $t$  is given by

$$p(y_t | y_{t-1}, y_{t-2}, \dots, y_1, c) = g(h_t, y_{t-1}, c). \quad (3)$$

where  $f$  and  $g$  are the given activation functions, and it must produce a valid probability. The two components of the encoder-decoder structure are jointly trained to maximize the conditional log-likelihood which is given by

$$\max_{\theta} \frac{1}{N} \sum_{n=1}^N \log p_{\theta}(y_n | x_n), \quad (4)$$

where  $\theta$  is the set of model parameters and  $\{x_n, y_n\}$  denotes the data pair in the training dataset.

Thus, in the Seq2Seq training,  $y$  participates in loss calculation and node operation, unlike general RNN, which

is used for loss supervision. Assuming the encoder input is  $\{x_n, y_n\}$ , the calculation process is as follows:

$$\begin{cases} h_t = zh_{t-1} + (1-z)h'_t, \\ h'_t = (\tanh[Wx_t] + [U(rh_{t-1})]), \\ z = \sigma([W_z x_t] + [U_z h_{t-1}]), \\ r = \sigma([W_r x_t] + [U_r h_{t-1}]). \end{cases} \quad (5)$$

In the above formulas,  $h'_t$ ,  $z$ , and  $r$  represent the intermediate variables,  $W$  and  $U$  are the training parameters, and  $\sigma$  is the activation function. When the hidden state of the encoding process is completed, the decoding vector is given by

$$c = \tanh(Vh^n), \quad (6)$$

where  $h^n$  is the final value of the encoder output after  $n$  epochs and  $V$  is the training parameter. After the decoding vector  $c$  is obtained, the decoder starts the decoding process initializing the initial hidden state  $h''_0$  which is given by

$$h''_0 = \tanh(Vhc). \quad (7)$$

The hidden state of the decoder at time  $t$  is given by

$$\begin{cases} h''_t = zh''_{t-1} + (1-z)\tilde{h}_t, \\ \tilde{h}_t = \tanh\left([W'y_{t-1}] + r'[U'h''_{t-1} + Cc]\right), \\ z' = \sigma([W'_z y_{t-1}] + [U'_z h''_{t-1}] + C_z c), \\ r' = \sigma([W'_r y_{t-1}] + [U'_r h''_{t-1}] + C_r c), \end{cases} \quad (8)$$

where  $W, U$ , and  $C$  and its deformations are variable training parameters. After obtaining the last hidden state, the condition probability is calculated by

$$p(y_t | y_{t-1}, \dots, y_1, X) = \frac{\exp(g s_t)}{\sum_1^k \exp(g s_t)}, \quad (9)$$

where  $k$  is the output dimension. Also, it holds that

$$\begin{aligned} s_t &= \max\{s_1 s_2, \dots, s_k\}, \\ s_i &= O_h h''_t + O_y y_{t-1} + O_c c, \quad (i = 1, 2, \dots, k), \end{aligned} \quad (10)$$

where  $O_h, O_y$ , and  $O_c$  are variable training parameters.

Once the Seq2Seq finishes processing a given data sequence, its output is directly fed to the input of the fully connected neural network to perform lane-changing classification.

**3.2. Fully Connected Layer.** A fully connected network is the most basic and simple neural network, yet such a network performs well in the multiparameter fusion, so it is generally used in the complex nonlinear classification tasks. Since the lane-changing classification represents a nonlinear task including a large-amount input data, a fully connected network is used as a classification network.

In Figure 8,  $x(1), x(2), \dots, x(n-1), x(n)$  are the fully connected layer inputs that are from the Seq2Seq layer output and  $w_{ji}^l$  represents the weight of the  $i$ th synapse of the  $j$ th neuron in the  $l$ th layer. The induced local domain in the  $l$ th layer is

$$v_j^l(n) = \sum_i w_{ji}^l(n) y_i^{(l-1)}(n), \quad (11)$$

where  $y_i^{(l-1)}(n)$  is the output of the  $i$ th neuron in the previous layer (i.e., the  $(l-1)$ th layer) after  $n$  iterations. For  $i = 0$ , it holds that  $y_0^{(l-1)}(n) = 1$  and  $w_{j0}^{(l)}(n) = b_j^{(l)}(n)$ , where  $b_j^{(l)}$  denotes the bias of the  $j$ th neuron in the  $l$ th layer. Using

the SoftMax function as an activation function, the output of the  $j$ th neuron in the  $l$ th layer is given by

$$y_j^{(l)} = \varphi_j(v_j^{(l)}(n)). \quad (12)$$

If the neuron  $j$  is in the first hidden layer, then it holds that

$$y_j^{(0)} = x_j(n), \quad (13)$$

where  $x_j(n)$  is the  $j$ th element of the input vector  $x(n)$ .

Besides, if neuron  $j$  is in the output layer ( $l = L$ ,  $L$  is the depth of the network), then it holds that

$$y_j^{(L)} = o_j(n). \quad (14)$$

Therefore, the error is given by

$$e_j(n) = d_j(n) - o_j(n), \quad (15)$$

where  $d_j(n)$  is the  $j$ th element of the expected response vector  $d(n)$ .

After the forward propagation is completed, the backpropagation is performed to complete the weight optimization. The backpropagation is given by

$$\delta_j^{(l)}(n) = \begin{cases} e_j^{(L)}(n) \varphi_j'(v_j^{(L)}(n)), & \text{Output layer } L \text{ neurons } j, \\ \varphi_j'(v_j^{(L)}(n)) \sum_k \delta_k^{(l+1)}(n) w_{kj}^{(l+1)}(n), & \text{Hidden layer } L \text{ neurons } j. \end{cases} \quad (16)$$

where  $\delta_j(n)$  is the local gradient and  $\varphi_j'$  is the differentiation of an independent variable. After the local gradient is obtained, the weight updating process is performed. The weight updating process in the  $n$ th iteration is given by

$$w_{ji}^{(l)}(n+1) = w_{ji}^{(l)}(n) + \alpha [w_{ji}^{(l)}(n-1)] + \eta \delta_j^{(l)}(n) y_i^{(l-1)}(n), \quad (17)$$

where  $\alpha$  is the momentum constant and  $\eta$  is the learning rate.

**3.3. Model Training.** According to the survey statistic [33, 34], most drivers are more inclined to the car-following behavior than to the lane changing, so there are not a lot of lane-changing data. Therefore, we had to collect the data needed for the prediction model training and testing. During the data collection process, to avoid the situation where drivers deliberately change the lanes, drivers were not told the true purpose of their trip before the driving started. In addition, uncertain factors such as the driver's driving skills, driving style, and travel purpose will affect the predictive performance of the model. Therefore, in the process of data collection, the research team will provide the driver with some brief driving information (such as recreational driving or emergency driving) when inviting the driver and to make

the data more generalized. Different drivers were invited to participate in the data collection process in order to collect as many diverse data as possible under different conditions. After data collection, 7000 features of the lane-changing behavior were extracted from the collected dataset. Since the amount of data was not large, the model was trained and validated by the 10-fold cross-validation method whose pseudocode is shown in Algorithm 1.

We used Google's open source deep learning library TensorFlow to build the hybrid neural network described above. The computational kernel of Seq2Seq was GRU; during the backpropagation training, the Adam optimizer in TensorFlow was used, which automatically adjusts the learning rate parameters during training to avoid overfitting. The initial weights of the entire network were filled with several sets of data obeying the positive distribution, and the initial learning rate is 0.005.

The model training process and its discussion items are shown in Figure 9. As the time window shrinks, the model training process advances. The first step in training is to use the original time window; since the time window is not shrunk, there is no label data for the Seq2Seq. The fully connected neural network is added after the Seq2Seq, and since the lane-changing signs are the label of the entire network, only the accuracy is shown in this step. The second

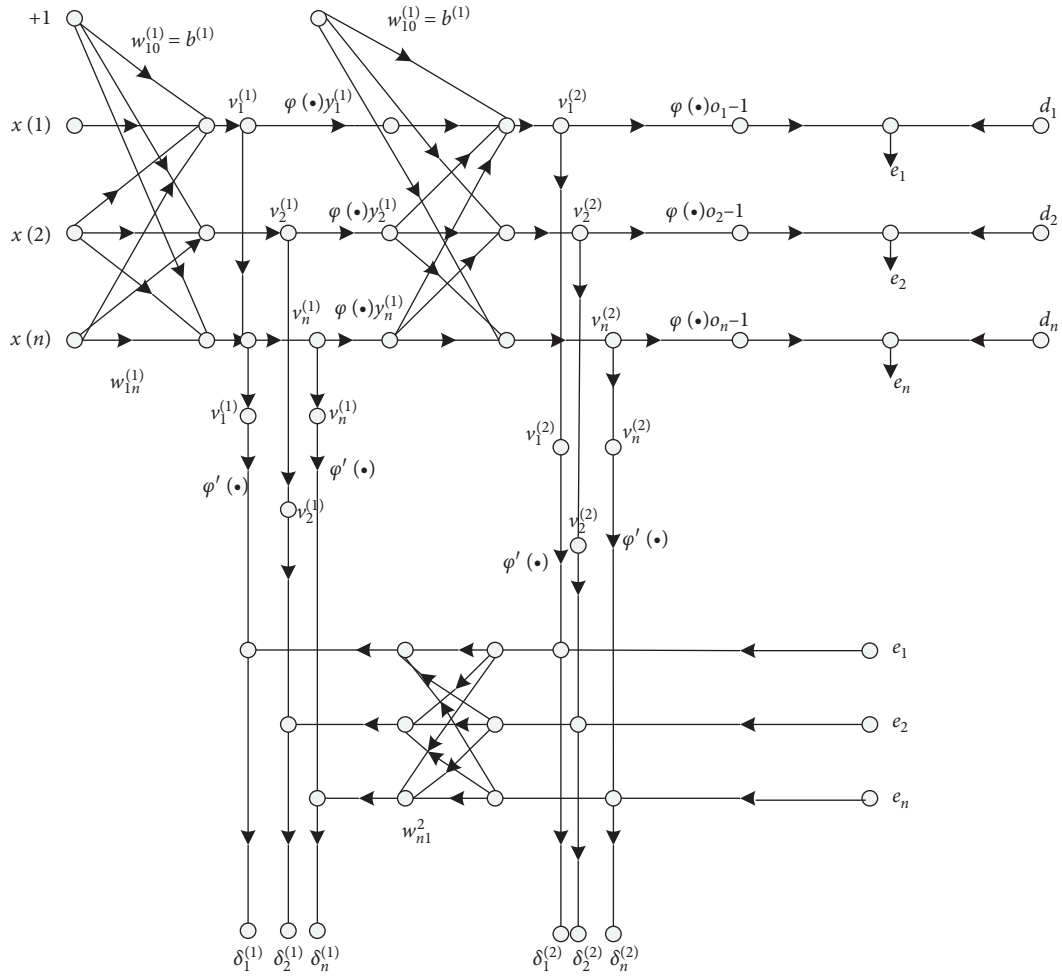


FIGURE 8: Fully connected neural network data flow diagram.

Input: training set and validation set

Output: classification results

- (1) The entire dataset is randomly divided into 10 parts
- (2) Use one of 10 parts as a test set and the other 9 as a training set, and loop beginning at this step
- (3) Train the model with 9 training sets and record the relevant data
- (4) Test the model with the test set and record the relevant data
- (5) Return to 2, select new test set and training set, the sets must differ from the previous ones, and repeat this process until all 10 parts are used as a test set, and the loop ends
- (6) Compare the real value and prediction value and calculate the prediction accuracy of each training and validation step
- (7) Model training/verification complete

ALGORITHM 1: 10-fold cross-validation method.

step is shrinking the time window to 2/5 of the original one, and then, the model's prospective time and accuracy are determined. Similarly, the third step denotes the time window shrinking to 1/5 of the original size and then determines the new model's prospective time and accuracy. After shrinking the time window, the new prediction performance of the Seq2Seq is also displayed. This time window shrink mechanism enables the model to predict lane-changing behavior using only a small part of the header data of the entire lane-changing data, instead of using all lane-

changing data for lane-changing recognition. At the same time, when processing the data, we found that, within 1/5 of the data of the original window, lane-changing behavior did not occur from a macroperspective. Therefore, it is reasonable to use this part of data to predict whether the vehicle will change lanes in the future.

When there is no middle label in the model, only the accuracy is discussed. The proposed model structure is presented in Figure 10, and the loss and the accuracy after the iteration are displayed in Figure 11.



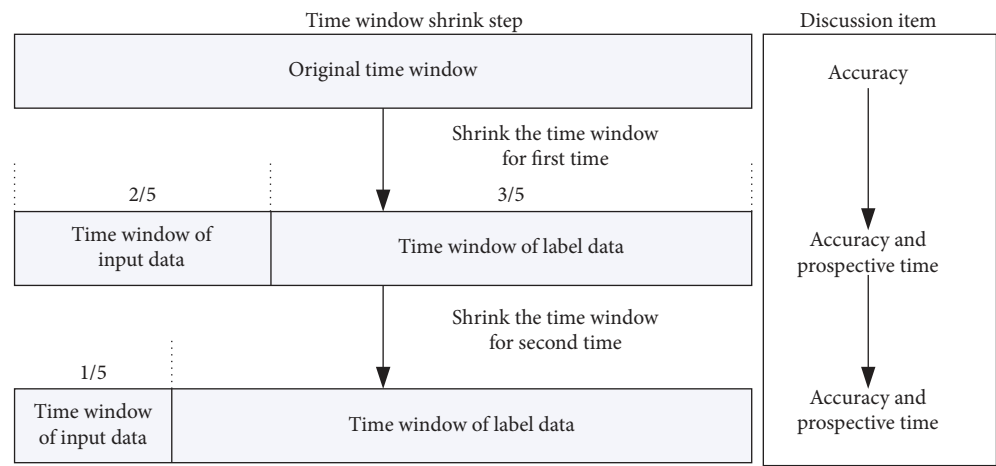


FIGURE 9: Model training procedure.

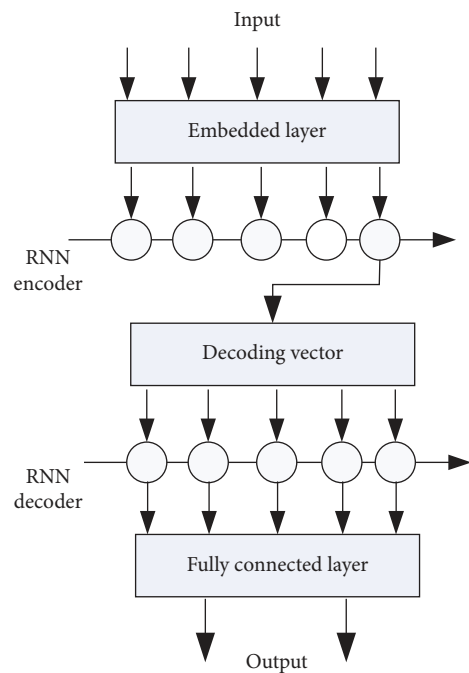


FIGURE 10: The proposed model structure.

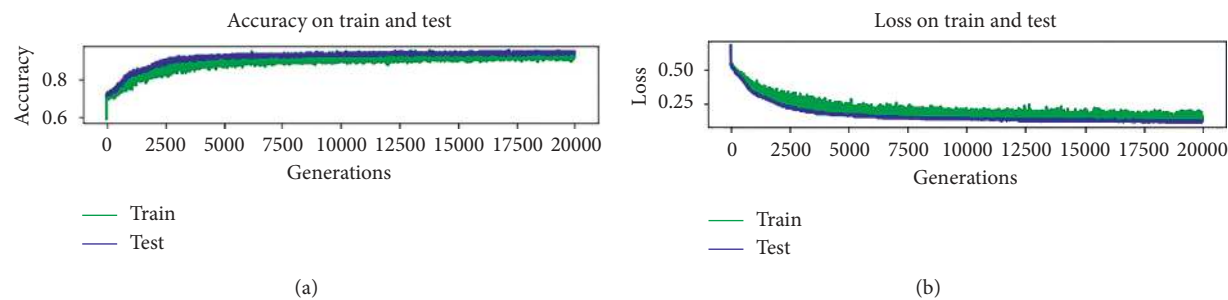


FIGURE 11: The accuracy (a) and the loss (b) after 20,000 iteration.

As presented in Figure 11, after iterating for 20,000 times, the model achieves the convergence state with the accuracy of 0.9858 and the loss of 0.11, which shows that the proposed model can recognize the driver's lane-changing behavior more accurately.

The training process of the entire model after shrinking the time window is given in Figure 9. Seq2Seq uses front data

of a time window to predict the caudal data. Because there are series of different input and output data as well the Seq2Seq layer which predicts the driver and vehicle status at the next time point, so the Euclidean distance is used to estimate the level of regression of the prediction result to the real result. The calculation formula is

$$\text{Average deviation} = \frac{\sqrt{(\text{actual value}_1 - \text{predictive value}_1)^2 + \dots + (\text{actual value}_n - \text{predictive value}_n)^2}}{n} \quad (18)$$

In Figure 12, the average Euclidean distance of the model training for once and twice shrunk time window is presented. It can be noticed that, after 200 iterations, the average deviation began to converge, and the deviation fluctuated within a small range, which shows that the Seq2Seq achieved a good prediction performance. Next, as already explained, the Seq2Seq prediction results were fed to the input of the fully connected network and subjected under the 10-fold cross-validation method. The obtained prediction accuracy and the loss for once shrunk time window are presented in Figure 13. When the number of iterations reached 20,000, the model converged with the accuracy of 0.935 and the loss of 0.12. The model accuracy was reduced due to the data loss caused by time window shrinking, but the prospective time was extended. The accuracy and the loss for twice shrunk time window are presented in Figure 14, where the accuracy increased to 0.938 and the loss was reduced to 0.11.

In Figures 15 and 16, the prospective time for once and twice shrunk time window is, respectively, presented.  $t_1$  and  $T_1$  denote the time needed that a vehicle changes a lane,  $t_2$  and  $T_2$  are the data extraction window for the Seq2Seq training and testing,  $t_3 + t_4$  and  $T_3 + T_4$  are the label extraction window for the Seq2Seq training and testing, and  $t_3$  and  $T_3$  are the time for predicting the lane changing, which is also the prospective time. When  $T_3 > t_3$ , the prediction method presented in Figure 16 had a longer prospective time than that presented in Figure 15.

In the model verification, the method presented in Figure 15 achieved the prediction accuracy of 0.935, and the model presented in Figure 16 achieved the prediction accuracy of 0.938. Thus, the latter method had better prediction performance. Besides, after statistical analysis of the data used in the paper, we came up with such a mathematical relationship:

$$\begin{aligned} T_2 = T_4 &= \frac{1}{5}T_1, \\ T_3 &= \frac{3}{5}T_1. \end{aligned} \quad (19)$$

The time cost and the corresponding number of lane changings are presented in Table 3, where the lane changing conducted in the interval 3-4s accounted for 47% of the total number of lane changings, which means the prospective time was 1.8-2.4s. Therefore, the average perspective time of

the model is 2.1s. This result indicates that the model could predict the lane changing well and achieve a high prediction accuracy. The comparison between the time cost of lane changing and the prediction time cost after sampling 50 times in all lane changing is presented in Figure 17, and the presented result shows that the prediction time was much shorter than the lane-changing time.

#### 4. Model Validation

After the model was trained, all the weight parameters in the prediction model were optimized. In order to test the generalization ability of the developed model, a different data was used for model training and testing. Also, different drivers were invited to drive vehicles to collect data on another route to obtain the test data. The routes used for validation data acquisition is presented in Figure 18. This route also started at the Chang'an University but ended at the Xi'an Chengbei Passenger Station, and the route direction was different compared with the one used for collection of training data presented in Figure 2. In addition, according to the data scale, the detailed structure and parameters of the cascade model, as shown in Table 4, were determined.

The model was validated using the same data processing method as that used for model training, but the validation included only one input data, without any label, the data was used as the network input, and the focus was on the comparison between the lane changing predicted by the network and the real data. The cross-validation was performed using the data of 3000 lane changings. The accuracy of each validating batch and the average accuracy are presented in Figure 19, where the model accuracy decreased in the first few validating batches, but the overall performance was good, and the average accuracy of the test exceeded 93.5%. Moreover, the prospective time was the same as that of the training. The prospective time for randomly extracted 50 adjacent lane-changing data was 1/5 of the entire lane-changing time, as shown in Figure 20.

The proposed model was compared with the five most commonly used prediction models. The comparison results are presented in Figure 21 and Table 5. The performance parameters of MST-Net all come from the literature [27], so Table 5 does not include the calculation time of

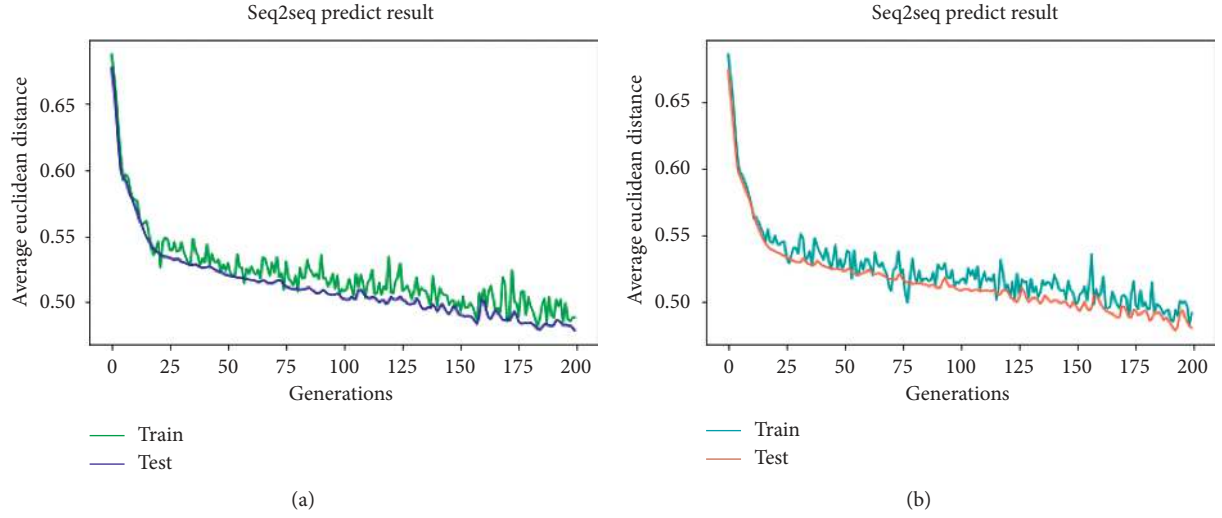


FIGURE 12: The average deviation for once (a) and twice (b) shrunk time window.

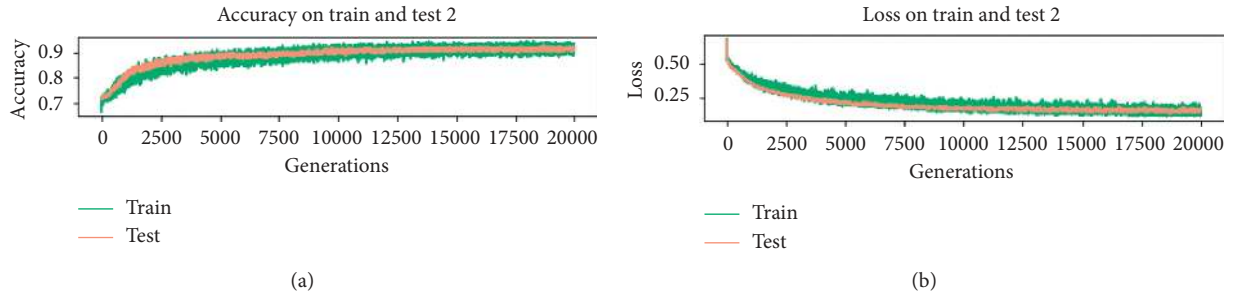


FIGURE 13: The accuracy (a) and the loss (b) for once shrunk time window.

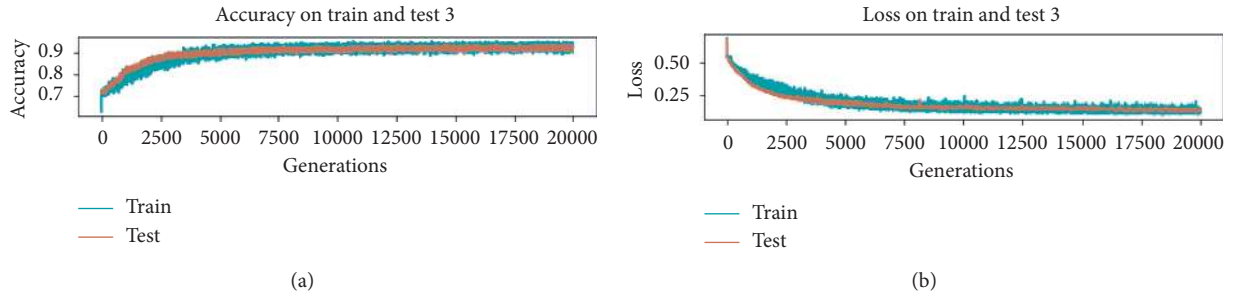


FIGURE 14: The accuracy (a) and the loss (b) for twice shrunk time window.

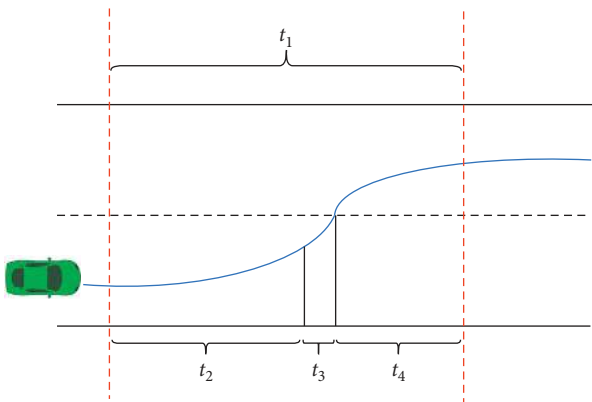


FIGURE 15: Prospective time for once shrunk time window.

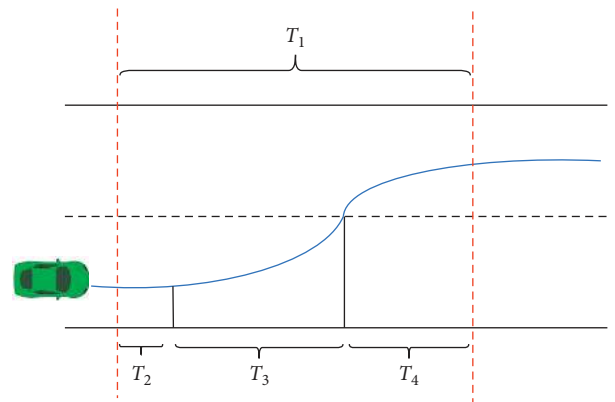


FIGURE 16: Prospective time for twice shrunk time window.

TABLE 3: Number of lane changes and time cost.

TC	0-1 s	1-2 s	2-3 s	3-4 s	4-5 s
NoLC	70	1372	1848	3290	420

TC: time cost; NoLC: number of lane changing.

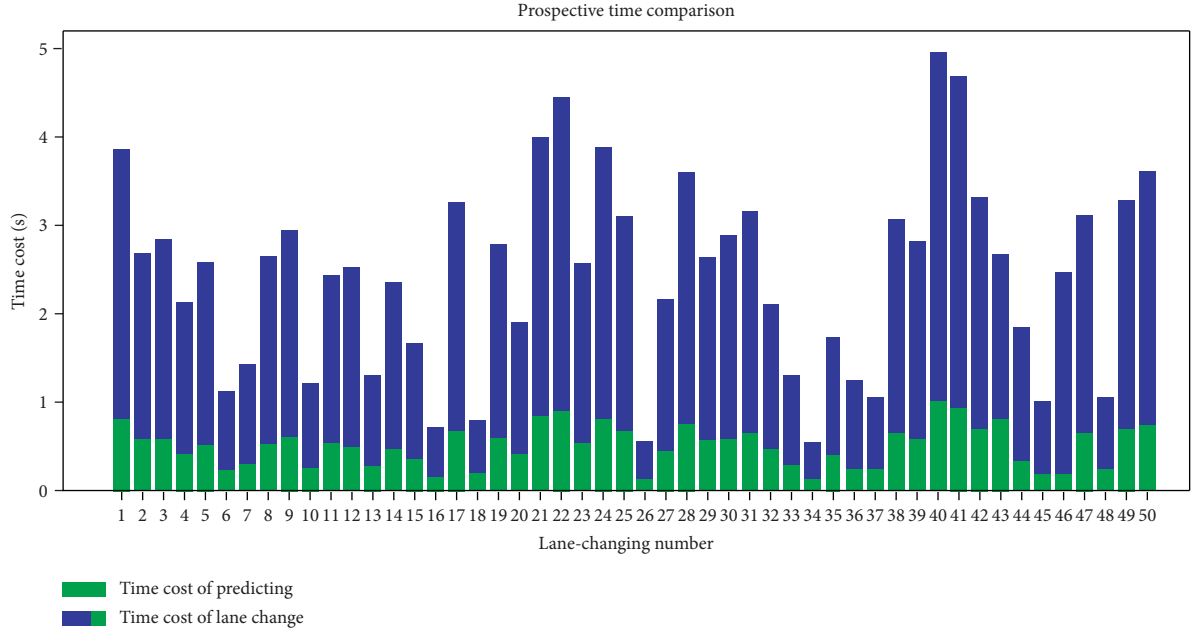


FIGURE 17: The time cost for different numbers of lane changes in testing.

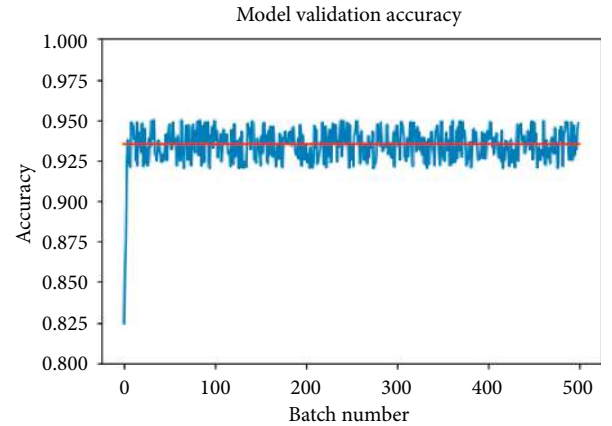


FIGURE 18: Validation data collection route.

TABLE 4: Cascade model structure during validation.

Level	Layer name	Number of nodes
1	Encoder-repeat-decoder	37-20-37
2	Input Hidden1 Hidden2 output	41-25-8-2

MTSDeepNet. The accuracy of the Dynamic Bayesian Network, the Decision Tree, and the SVM prediction was 75.0%, 84.0%, and 91.1%, respectively, and the accuracy of the BP neural network and the MTSDeepNet was 91.6% and 92.0%, respectively. The accuracy of the proposed Seq2Seq-FC structure was 93.5%, which was better than that of the



— Accuracy on every batch  
— Average accuracy

FIGURE 19: Validation accuracy.

other five algorithms. Furthermore, although the computation time of the proposed model is higher than several other methods, the microsecond-level increase does not affect applications such as vehicle anticollision which are built on it.

To better illustrate the superiority of the model proposed in the paper, it is necessary to illustrate the data and algorithms used in the six comparison models. As shown in



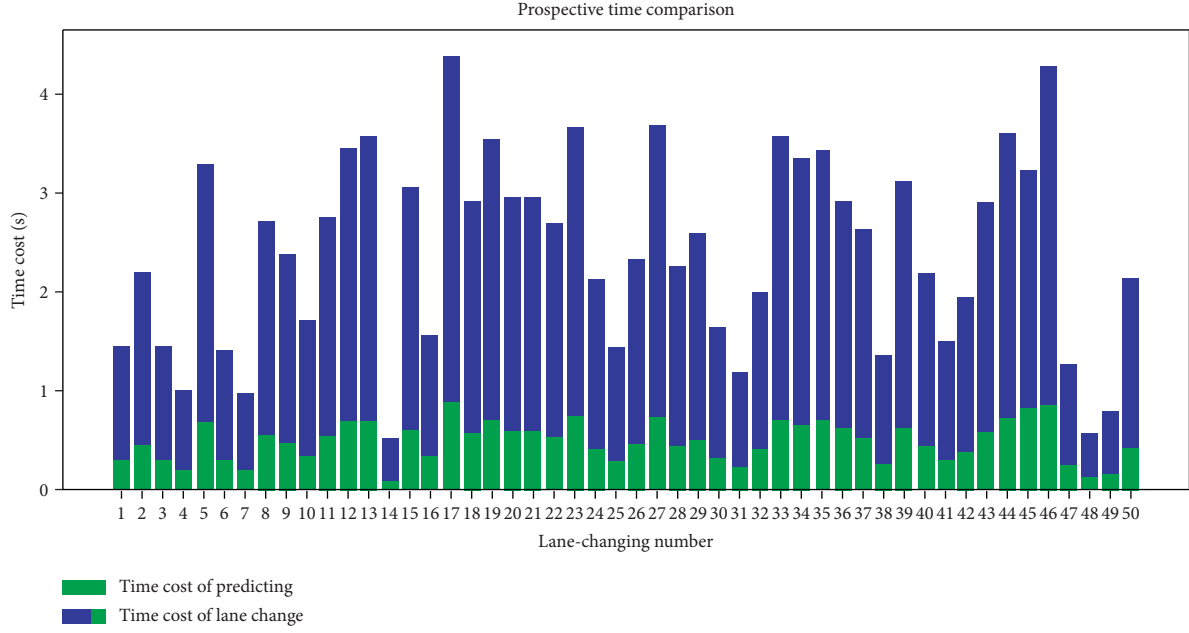


FIGURE 20: Time cost for different numbers of lane changes in validating.

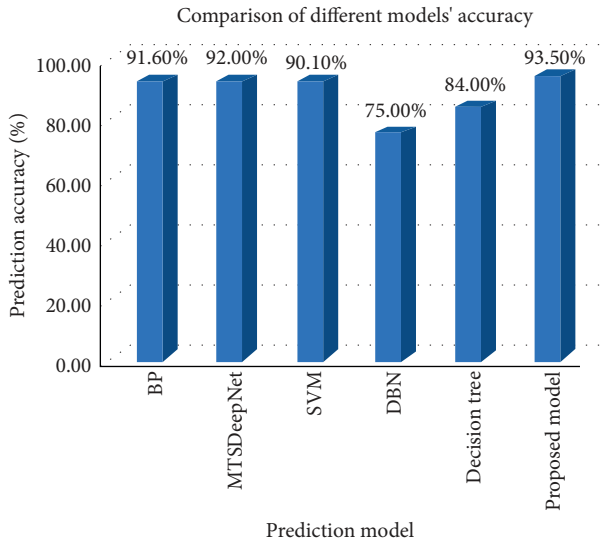


FIGURE 21: Comparison of the accuracy of different prediction models.

TABLE 5: Calculation time of different prediction models.

Prediction model	Signal data calculation time	Batch time (ms)
SVM	75us	22.8
DBN	64us	19.7
Decision tree	81us	24.7
Proposed model	160us	47.6

Table 6, it is the detailed information of the six models in the experimental process.

The word “other” means that the algorithms and data used in the comparison model are from other literatures; on

TABLE 6: Details of the six comparison models.

Prediction model	Data used in the model	Algorithm used in model
MSTDeepNet	Other [28]	Other [28]
BP	This paper	Other [22]
SVM	This paper	Other [15]
DBN	This paper	Other [20]
Decision tree	This paper	Other [35]
Proposed model	This paper	This paper

the contrary, “this paper” indicates that the algorithms and data used are all derived from this paper. The accuracy of MSTDeepNet comes from [28], and the BP, SVM, DBN, and decision tree use the data which is extracted in this paper and the algorithms used in other literatures. In the experiment, we input the 41 values which are from second shrinking of the time window into different classification algorithms to obtain the above accuracy. The comparison results show that this paper has certain advantages in terms of data or prediction model.

## 5. Conclusion

In this paper, a Seq2Seq-FC neural network for prediction of driver lane-changing behavior is introduced. The proposed model has two levels, where the first level denotes the Seq2Seq network whose function is to process the time series data and the second level denotes a fully connected neural network which works as a nonlinear classifier. In the proposed prediction model, the vehicle kinematics data (VKA), the drivers' kinematics data (DKA), and the drivers' physiology data (DPA) are used as input data for the fully connected network. In addition, a dynamic time window is proposed to extract the features of the lane-changing

process, and the method of time window shrinking is successively used to train the prediction model and improve the prospective time. Moreover, model testing was performed by data different from that used for model training to evaluate the model generalization ability. The test results showed that the proposed prediction model achieved a good performance.

In the data collection process, 35 drivers took part, and different routes were used for training and test data. The collected data consisted of 10,000 lane-changing samples, of which 7000 samples were used for model training, and the rest was used for model validating. The validating results proved the effectiveness and stability of the proposed model. Moreover, the proposed Seq2Seq-FC model was compared with five common prediction models: BP neural network, SVM, dynamic Bayesian network, decision tree, and MTSDeepNet. The comparison results showed that the Seq2Seq-FC network achieved higher prediction accuracy and longer prospective time than other models. The results presented in this study can be helpful to improve the practical effect of the ADAS and enhance lane-changing safety.

In our future research, we will improve the proposed model from several aspects. Firstly, many researchers have demonstrated that driver's decision to change a lane is also affected by vehicle type and driver's driving skills [36, 37]; for instance, a car has different lane-changing factor compared with a bus. However, in this study, we did not consider vehicle type because we used the vehicles of the same type. Therefore, in our future work, we will take different types of vehicles into account. Secondly, although many different road conditions were included in the traffic route, some of the road types such as rugged mountain road and country road were still not included, but they will be considered in our future research. Thirdly, since the neural network considers the fuzzy relationship between input and output, input data redundancy can be caused, and model calculation speed can be reduced. However, the sensitivity analysis can be used to eliminate some variables that have little effect on model classification, which may make the model more optimized and concise.

## Data Availability

The data used to support the findings of the study are available from the first author ChengWei (chengwei@chd.edu.cn) upon reasonable request.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

This work was supported by National Key R&D Program of China (Grant no. 2018YFB1600604).

## References

- [1] S. Gong and L. Du, "Optimal location of advance warning for mandatory lane change near a two-lane highway off-ramp," *Transportation Research Part B: Methodological*, vol. 84, pp. 1–30, 2016.
- [2] H. Yu, H. E. Tseng, and R. Langari, "A human-like game theory-based controller for automatic lane changing," *Transportation Research Part C: Emerging Technologies*, vol. 88, pp. 140–158, 2018.
- [3] M. Keyvan-Ekbatani, V. L. Knoop, and W. Daamen, "Categorization of the lane change decision process on freeways," *Transportation Research Part C: Emerging Technologies*, vol. 69, pp. 515–526, 2016.
- [4] A. Yasir, Z. Zuduo, and H. Mazharul, "Connectivity's impact on mandatory lanechanging behavior: evidences from a driving simulator study," *Transportation Research Part B: Methodological*, vol. 93, pp. 292–309, 2018.
- [5] J.-B. Sheu and S. G. Ritchie, "Stochastic modeling and real-time prediction of vehicular lane-changing behavior," *Transportation Research Part B: Methodological*, vol. 35, no. 7, pp. 695–716, 2001.
- [6] Y. Dou, F. Yan, and D. Feng, "Lane changing prediction at highway lane drops using support vector machine and artificial neural network classifiers," in *Proceedings of the IEEE International Conference on Advanced Intelligent Mechatronics IEEE*, Delft, The Netherlands, 2016.
- [7] H. Liu, H. Wei, T. Zuo, Z. Li, and Y. J. Yang, "Fine-tuning ADAS algorithm parameters for optimizing traffic safety and mobility in connected vehicle environment," *Transportation Research Part C: Emerging Technologies*, vol. 76, pp. 132–149, 2015.
- [8] A. Simic, O. Kocic, M. Z. Bjelica, and M. Milosevic, "Driver monitoring algorithm for advanced driver assistance systems," *TELFOR*, vol. 7, 2016.
- [9] Y. Saito, M. Itoh, and T. Inagaki, "Driver assistance system with a dual control scheme: effectiveness of identifying driver drowsiness and preventing lane departure accidents," *IEEE Transactions on Human-Machine Systems*, vol. 46, no. 5, pp. 660–671, 2016.
- [10] M. Errampalli, M. Okushima, and T. Akiyama, "Fuzzy logic based lane change model for microscopic traffic flow simulation," *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 12, no. 7, pp. 172–181, 2008.
- [11] W. Shi and Y. P. Zhang, "Decision analysis of lane change based on fuzzy logic," *Applied Mechanics and Materials*, vol. 419, pp. 790–794, 2013.
- [12] E. C. B. Olsen, *Modeling Slow Lead Vehicle Lane Changing*, Virginia Polytechnic Institute and State University, Blacksburg, VA, USA, 2013.
- [13] M. Baumann and J. F. Krems, "Situation awareness and driving: a cognitive model," *Modelling Driver Behaviour in Automotive Environments*, vol. 12, pp. 253–265, 2007.
- [14] D. D. Salvucci, "Modeling driver behavior in a cognitive architecture," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 48, no. 2, pp. 362–380, 2006.
- [15] P. Kumar, M. Perrollaz, S. Lefevre, and C. Laugier, "Learning-based approach for online lane change intention prediction," *Intelligent Vehicles*, vol. 4, pp. 797–802, 2013.

- [16] W. Yao, H. Zhao, P. Bonnifait, and H. Zha, "Lane change trajectory prediction by using recorded human driving data," *Intelligent Vehicles Symposium*, vol. 7, pp. 430–436, 2013.
- [17] Y. Hou, P. Edara, and C. Sun, "A genetic fuzzy system for modeling mandatory lane changing," *Intelligent Transportation Systems*, vol. 12, pp. 1044–1048, 2012.
- [18] K. Li, X. Wang, Y. Xu, and J. Wang, "Lane changing intention recognition based on speech recognition models," *Transportation Research Part C: Emerging Technologies*, vol. 69, pp. 497–514, 2016.
- [19] Y. Zheng and J. H. L. Hansen, "Lane-change detection from steering signal using spectral segmentation and learning-based classification," *IEEE Transactions on Intelligent Vehicles*, vol. 2, no. 1, pp. 14–24, 2017.
- [20] D. Kasper, G. Weidl, T. Dang et al., "Object-oriented bayesian networks for detection of lane change maneuvers," *IEEE Intelligent Transportation Systems Magazine*, vol. 4, no. 3, pp. 19–31, 2012.
- [21] D.-F. Xie, "A data-driven lane-changing model based on deep learning," *Transportation Research Part C Emerging Technologies*, vol. 106, pp. 41–60, 2019.
- [22] R. M. H. Cheng, J. W. Xiao, and S. Lequoc, "Neuromorphic controller for AGV steering," in *Proceedings of the 2010 IEEE International Conference on Robotics & Automation*, IEEE, Anchorage, AK, USA, 1992.
- [23] R. S. Tomar, S. Verma, and G. S. Tomar, "Prediction of lane change trajectories through neural network," in *Proceedings of the 2010 International Conference on Computational Intelligence and Communication Networks*, pp. 125–146, Bhopal, India, 2010.
- [24] C. Ding, W. Wang, X. Wang, and M. Baumann, "A neural network model for driver's lane-changing trajectory prediction in urban traffic flow," *Mathematical Problems in Engineering*, vol. 2013, Article ID 967358, 2013.
- [25] J. Zheng, K. Suzuki, and M. Fujita, "Predicting driver's lane-changing decisions using a neural network model," *Simulation Modelling Practice and Theory*, vol. 42, pp. 73–83, 2014.
- [26] J. Peng, Y. Guo, R. Fu, W. Yuan, and C. Wang, "Multi-parameter prediction of drivers' lane-changing behaviour with neural network model," *Applied Ergonomics*, vol. 50, pp. 207–217, 2016.
- [27] Y. Dou, F. Yan, and D. Feng, "Lane changing prediction at highway lane drops using support vector machine and artificial neural network classifiers," in *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp. 901–906, Delft, The Netherlands, 2016.
- [28] X. Wang, Y. L. Murphey, and D. S. Kochhar, "MTS-DeepNet for lane change prediction," in *Proceedings of the International Joint Conference on Neural Networks*, pp. 4571–4578, 2016.
- [29] X. Li, W. Wang, and M. Roetting, "Estimating driver's lane-change intent considering driving style and contextual traffic," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 9, pp. 3258–3271, 2019.
- [30] Y. Qiao, K. Hashimoto, A. Eriguchi, H. Wang, D. Wang, and Y. Tsuruoka, "Parallelizing and optimizing neural encoder-decoder models without padding on multi-core architecture," *Future Generation Computer Systems*, vol. 108, pp. 1206–1213, 2020.
- [31] L. Zheng zhong and S. G. D. O. David, "The impact of encoding-decoding schemes and weight normalization in spiking neural networks," *Neural Networks*, vol. 108, pp. 365–378, 2019.
- [32] H. Mårtensson, O. Keelan, and C. Ahlström, "Driver sleepiness classification based on physiological data and driving performance from real road driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 2, pp. 421–430, 2019.
- [33] C. Marina Martinez, M. Heucke, F. Y. Wang, B. Gao, and D. Cao, "Driving style recognition for intelligent vehicle control and advanced driver assistance: a survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 3, pp. 666–676, 2018.
- [34] Z. Zheng, "Recent developments and research needs in modeling lane changing," *Transportation Research Part B: Methodological*, vol. 60, no. 1, pp. 16–32, 2014.
- [35] Y. Li, M. Dong, and R. Kothari, "Classifiability-based omnivariate decision trees," *IEEE Transactions on Neural Networks*, vol. 16, no. 6, pp. 1547–1560, 2019.
- [36] E. Suzdaleva and I. Nagy, "An online estimation of driving style using data-dependent pointer model," *Research Part C: Emerging Technologies*, vol. 86, pp. 16–23, 2019.
- [37] P. Cai, Y. Wang, G. Lu, P. Chen, C. Ding, and J. Sun, "A spatiotemporal correlative k-nearest neighbor model for short-term traffic multistep forecasting," *Transportation Research Part C: Emerging Technologies*, vol. 62, pp. 21–34, 2019.