Article
XGBoost-DNN Mixed Model for Predicting Driver’s Estimation on the Relative Motion States during Lane-Changing Decisions: A Real Driving Study on the Highway

Chen Zhao 1, Xia Zhao 1,*, Zhao Li 1 and Qiong Zhang 2

1 School of Automobile, Chang’an University, Xi’an 710064, China; 2016022015@chd.edu.cn (C.Z.); 2019322001@chd.edu.cn (Z.L.)
2 China Communications Press Co., Ltd., Beijing 100101, China; zhangqiong@ccpress.com.cn
* Correspondence: zhaoxia26@chd.edu.cn; Tel.: +86-029-82334722

Abstract: This study is conducted on a real live highway to investigate the driver’s performance in estimating the speed and distance of vehicles behind the target lane during lane changes. Data on the participants’ estimated and actual data on the rear car were collected in the experiment. Ridge regression is used to analyze the effects of both the driver’s features, as well as the relative and absolute motion characteristics between the target vehicle and the subject vehicle, on the driver’s estimation outcomes. Finally, a mixed algorithm of extreme gradient boosting (XGBoost) and deep neural network (DNN) was proposed in this paper for establishing driver’s speed estimation and distance prediction models. Compared with other machine learning models, the XGBoost-DNN prediction model performs more accurate prediction performance in both classification scenarios. It is worth mentioning that the XGBoost-DNN mixed model exhibits a prediction accuracy approximately two percentage points higher than that of the XGBoost model. In the two-classification scenarios, the accuracy estimations of XGBoost-DNN speed and distance prediction models are 91.03% and 92.46%, respectively. In the three-classification scenarios, the accuracy estimations of XGBoost-DNN speed and distance prediction models are 87.18% and 87.59%, respectively. This study can provide a theoretical basis for the development of warning rules for lane-change warning systems as well as insights for understanding lane-change decision failures.

Keywords: lane-changing decision; XGBoost-DNN algorithm; prediction model; speed estimation; distance estimation

1. Introduction

Vehicle lane change is an essential maneuver for drivers while commuting. It is also a behavioral reflex closely related to road safety and traffic efficiency [1,2]. Studies have shown that lane changing causes about 4% of all traffic crashes and about 0.5% of all traffic fatalities in the United States [3]. Moreover, driver decision errors account for 75% of all lane-change-related crashes [4]. Eberhard et al. [5] mentioned that the main cause of lane-changing accidents is the driver’s misjudgment of the current traffic environment. Subsequently, the driver behavior model developed by Shawky [6] shows that drivers who suddenly change lanes are 2.53 times more likely to be involved in a traffic accident than others.

In order to reduce drivers’ psychological load during lane change and subsequently reduce the rate of related accidents, most current studies focus on vehicle warning systems that notify drivers when they change lanes [7–9]. However, the warning rules of current warning systems do not always correlate with the cognitive perception of drivers. Many users reported that the warning from the system is triggered too early, which is a nuisance, leading them to abandon the use of the lane change warning system altogether. In addition, lane change warning systems are not yet fully available in Chinese vehicles [10]. Therefore,
drivers more often make lane change decisions based on their own subjective assessment of risk. It is valuable to investigate the driver’s real-time decision-making paradigm in such a scenario.

The driver’s lane-changing decision-making process is a balance between the driver’s willingness to change lanes and his or her assessment of the related risk [11,12]. The driver’s assessment of the lane changing risk is mainly derived from the estimation of the distance and speed between the subject vehicle and the target vehicle. Furthermore, the accuracy of the driver’s assessment of the distance and speed between the target vehicle and the subject vehicle directly impacts the driver’s lane-changing decision and even the associated risk. If the driver tends to overestimate the speed of the target vehicle while underestimating the distance between the subject vehicle and the target vehicle, the driver will abandon the lane change operation and, thus, miss the time to change lanes. Conversely, if the driver underestimates the speed of the target vehicle and overestimates the distance between the two vehicles, the driver may perform the lane change, which will ultimately result in a significant lane change accident risk. Shawky [6] noted that drivers who look in the side mirror and out the window before changing lanes are, respectively, 4.61 and 3.85 times less likely to be involved in a crash than their counterparts who do not. Therefore, the key parameter for drivers to accurately assess the risk of lane change is to correctly perceive the relative motion state of the target vehicle and their own vehicle.

The relative motion state of the primary vehicle and the surrounding vehicles is the key input factor for researchers to build lane-change decision models. Qiu [13] chose the longitudinal distance and relative speed of the subject vehicle and the surrounding vehicles as the input variables of the lane change decision model and built a lane change decision model using traditional machine learning algorithms and gradient boosting decision trees (GBDTs), respectively, with the result showing the best overall prediction performance of GBDT. Xu et al. [14] also modeled the driver’s lane change decision based on the GBDT algorithm using the safe collision time between the subject car and the car behind the target lane and the traffic state around the subject car, and the model achieved good prediction performance. Although all of the above studies have good performance in predicting the lane change decision, the driver’s factor was not considered in these studies. In addition, the driver’s lane change process is generally divided into three stages: information perception, lane change decision, and action implementation. Most studies in the current research area focused on the study of lane change decisions and ignored the study of driver’s information perception.

Speed perception emerged in the field of traffic psychology. Volunteers undergo speed perception tests in order to evaluate their ability to accurately assess the speed of a moving object and rate their impatience and their tendency to drive precariously. Speed perception is a valid parameter for the assessment of driving safety since less experienced drivers have higher speed perception than average professional drivers [15], while drivers with poor driving behavior (with violation records) have lower speed perception than their average counterparts [16]. From the driver’s perspective, many parameters affect speed and distance perception. Drivers tend to underestimate the speed of a target object if it is large and in poor visibility conditions [17–19]. Driving speed also affects the driver’s ability to estimate speed, with drivers perceiving speed more accurately at moderate speeds (40–64 km/h) than at low speeds (5–32 km/h) and high speeds (72–97 km/h). The most accurate perception of speed is in the range of 40–56 km/h, but speeds below this range tend to be overestimated and speeds above this range are more likely to be underestimated [20,21]. In addition, factors such as continuous driving time, age, and driving seniority can affect the driver’s speed and distance perception [22]. The researchers also suggested that even veteran drivers tend to underestimate their driving speed while commuting [23,24].

The above studies all investigate the drivers’ ability to perceive their own vehicle, but other vehicles on the road are also a factor to be concerned about during vehicle operation, especially during lane-changing maneuvers. In the lane-changing study, the recognition
model than can identify the drivers’ lane change intention was established by Peng et al. (2015) using the speed of the self-car and the relative motion state between the self-car and the surrounding vehicles. In the Action Point Models proposed by Michaels [25], it is believed that the driver relies on the change in size of the preceding vehicle in the field of view to perceive the change in relative speed between the front and rear vehicles and judge whether he/she is approaching the front vehicle in the process of car following. Once the driver’s perception exceeds a certain threshold, he/she will perform deceleration until he/she can no longer perceive the change in relative speed. There is a significant difference between drivers’ spatial distance as well as speed discrimination in a dynamic traffic flow environment and in an idle state. Kang [26] analyzed drivers’ spatial distance discrimination in a dynamic ground environment and qualitatively analyzed the effect of color characteristics on drivers’ spatial distance discrimination. Wei et al. [27] conducted a distance perception test on drivers and analyzed the effect of obstacle color, ambient light characteristics, and vehicle speed on drivers’ distance discrimination characteristics.

The lane change behavior is similar to the car-following behavior in that the relative motion status between the vehicle and the other vehicles needs to be evaluated. However, during car-following behaviors, the driver can obtain the motion status of the vehicle in front of him/her by directly scanning it with his/her eyes. In lane-changing decisions, the driver needs to capture the motion information of the target vehicle, which means that the driver can acquire the relative motion status between the target vehicle and the driver’s own vehicle mainly through the rearview mirror instead of directly through the eyes [9]. There is a significant difference between the driver acquiring the relative motion state between the vehicle and other vehicles directly through the eyes and through the rearview mirror.

Few of the studies so far have examined the driver’s ability to estimate speed and distance through the rearview mirror, which is an important factor affecting lane change safety. In addition, experiments in related studies are rarely conducted in actual highway environments. Considering the actual road conditions in China, drivers have more demand for lane change on highways than on city roads, and drivers have different speed estimation and distance estimation abilities on highways than on ordinary roads, it is valuable to acquire data on drivers’ speed estimation and distance estimation through the rearview mirror on highways. In order to build a ground truth for the above scenarios, this study is conducted as a road experiment on the highway, during which the participants were asked to drive in a fixed lane at different speeds and to report the speed of the vehicle behind the left lane and the distance between it and their own vehicle through the rearview mirror at all times. The gathered information is used to investigate the speed estimation error and distance estimation error of the driver through the rearview mirror.

The research framework of this paper is shown in Figure 1, and it is organized as follows. Section 2 introduces the algorithms used to model the speed estimation and distance estimation. Section 3 presents the experiment and elaborates on the data sources for this study. The factors that affect the accuracy of driver speed estimation and distance estimation are analyzed in Section 4. In Section 5, the modeling process of XGBoost-DNN is shown, and the prediction performance of extreme gradient boosting (XGBoost) combined with deep neural network (DNN) algorithms proposed in this paper is compared with that of traditional machine learning algorithms. Section 6 discusses the results. Section 7 concludes the study and provides insights into future work.
2. Description of Algorithms

In this section, we first introduce the XGBoost ensemble learning algorithm and the DNN algorithm separately, and then we describe the mixed algorithm model proposed by combining the XGBoost algorithm and the DNN algorithm innovatively in this paper.

2.1. Description of XGBoost Algorithm

XGBoost, an ensemble learning algorithm, was proposed by Chen and Guestrin [28], and it is essentially an optimized implementation of GBDT [29], which sequentially generates and updates basic classifiers (weakly learned classifiers) to form integrated classifiers (strong classifiers) [30,31]. XGBoost has many advantages. First, XGBoost supports linear classifiers and enables both logistic regression and linear regression. Second, it performs a second-order Taylor expansion of the cost function, enabling the use of second-order derivatives when performing optimization. Third, XGBoost borrows from the RF algorithm and supports column sampling, which reduces computational effort while preventing overfitting [32].

XGBoost is commonly used in the fields of classification and regression, and in recent years, it has been widely noticed for its performance in efficiency and high prediction accuracy [33]. The model structure of XGBoost is shown in Figure 2 and the details of the model can be expressed as follows.

![Figure 1. Research framework.](image1)

**Figure 1.** Research framework.

2. Description of Algorithms

In this section, we first introduce the XGBoost ensemble learning algorithm and the DNN algorithm separately, and then we describe the mixed algorithm model proposed by combining the XGBoost algorithm and the DNN algorithm innovatively in this paper.

2.1. Description of XGBoost Algorithm

XGBoost, an ensemble learning algorithm, was proposed by Chen and Guestrin [28], and it is essentially an optimized implementation of GBDT [29], which sequentially generates and updates basic classifiers (weakly learned classifiers) to form integrated classifiers (strong classifiers) [30,31]. XGBoost has many advantages. First, XGBoost supports linear classifiers and enables both logistic regression and linear regression. Second, it performs a second-order Taylor expansion of the cost function, enabling the use of second-order derivatives when performing optimization. Third, XGBoost borrows from the RF algorithm and supports column sampling, which reduces computational effort while preventing overfitting [32].

XGBoost is commonly used in the fields of classification and regression, and in recent years, it has been widely noticed for its performance in efficiency and high prediction accuracy [33]. The model structure of XGBoost is shown in Figure 2 and the details of the model can be expressed as follows.

![Figure 2. XGBoost structure.](image2)

**Figure 2.** XGBoost structure.
\[ \hat{y}_i = \sum_{k=1}^{K} f_k(x_i), f_k \in F \]  

Here, \( \hat{y}_i \) is the prediction result of the sample \( i \), and \( f_k \) is a regression tree; \( F \) corresponds to the set of all regression trees; \( f_k(x_i) \) represents the calculated score of the \( k \)-th tree for the \( i \)-th sample in the data set.

The objective function can be expressed as follows.

\[ \text{Obj} = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \]  

\[ \Omega(f_k) = \gamma T + \frac{1}{2} \lambda \sum_{j=1}^{T} w_j^2 \]  

The loss function of the model is \( l(y_i, \hat{y}_i) \), and \( \sum_{k=1}^{K} \Omega(f_k) \) is the regularization term. \( \gamma \) and \( \lambda \) denote the penalty coefficients on the model; \( T \) and \( w_j \) denote the number of leaves and the weights of the leaves of the \( k \)-th tree, respectively. By splitting the regularization direction into the first \( t-1 \) and the \( t \)-th iterations, the loss function becomes the following.

\[ \text{Obj}_t = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) + c \]  

The new function becomes \( f_t(x_i) \) and \( c \) is a constant. Subjecting the objective function to a Taylor second-order expansion, we obtain the following.

\[ g_i = \frac{\partial}{\partial y_i} l(y_i, \hat{y}_i^{(t-1)}) \]  

\[ h_i = \frac{\partial^2}{\partial y_i^2} l(y_i, \hat{y}_i^{(t-1)}) \]  

where \( g_i \) and \( h_i \) are, respectively, the first and second order derivatives of the loss function; thus, the objective function becomes the following.

\[ \text{Obj}_t = \sum_{i=1}^{n} \left( l(y_i, \hat{y}_i^{(t-1)}) + g_if_t(x_i) + \frac{1}{2} h_if_t^2(x_i) \right) + \Omega(f_t) + c \]  

By removing the constant term, this can be simplified into the following.

\[ \text{Obj}_t = \sum_{i=1}^{n} (g_if_t(x_i) + \frac{1}{2} h_if_t^2(x_i)) + \Omega(f_t) \]  

After transformation and merging the canonical terms, the objective function of the model is described as follows:

\[ \text{Obj}_t = \sum_{i=1}^{T} \left( \sum_{i \in l_j} g_i w_j + \frac{1}{2} \left( \sum_{i \in l_j} h_i + \lambda \right) w_j^2 \right) + \gamma T \]  

where \( l_j \) is the index number of the instance that falls on the \( j \)-th leaf node.

2.2. Description of DNN Model

DNN (Deep Neural Network) is a topic of interest in the field of machine learning. DNN model information is stored in the neurons in the network, which has strong robustness and fault tolerance [34]. In addition, DNN has a strong ability to synthesize
information and can coordinate well between multiple input information, which is suitable for fusing multiple features of inputs [35]. The internal neural network layers of DNN can be divided into three categories: input layer, hidden layer, and output layer, as shown in Figure 3. The first layer is the input layer, the last layer is the output layer, and the middle layers are all hidden layers. The layers are fully connected to each other, and any neuron in the first layer must be connected to any neuron in the second layer. The correspondence between the input and output values of neurons is as follows:

\[ y_i = f(b_j + \sum_{i=1}^{n} (x_i \times w_{ij})) \]  

(10)

where \( x_i \) is the input value, \( y_i \) is the output value, \( w_{ij} \) is the connection weight, and \( b_j \) is the bias value. The role of the activation function is to de-linearize and achieve a nonlinear classification task. In this paper, we introduce the DNN model using the Relu activation function, and the relation can be expressed as follows.

\[ f(\delta) = \begin{cases} \delta, (\delta \geq 0) \\ 0, (\delta < 0) \end{cases} \]  

(11)

**Figure 3.** DNN structure.

2.3. XGBoost-DNN Mixed Algorithm Model

In this section, we propose a mixed model structure of XGBoost model and DNN model in tandem (Figure 4), where the XGBoost model completes the feature transformation, and finally the DNN model is trained using the fused features, and the DNN model completes the final prediction. The general idea of the proposed model can be described as follows: First, the XGBoost model is trained with existing features; then, the tree learned by the XGBoost model is used to construct new features, and finally these new features are added to the original features to train the DNN model together. The new feature vector is constructed by taking values of 0 and 1, and each element of the vector corresponds to a leaf node of the tree in the XGBoost model. The length of the new feature vector is equal to the sum of the number of leaf nodes in all trees in the XGBoost model.
3. Study Implementation

3.1. Participants

A total of 14 participants (2 females and 12 males) volunteered in the experiment, with ages ranging from 27 to 48 years (mean = 34.7 years) and having 3 to 23 years (mean = 8.4 years) of driving experience. Seven experienced drivers (>5 years of driving experience) and seven non-experienced drivers (<5 years of driving experience) were included. The details of the participants’ information are described in Table 1. All participants had a valid driving license and had not experienced a major traffic accident in the past three years. Participants were prohibited from taking drugs and functional drinks the day before the trial. All were in good health on the day of the trial, and there were no traffic risks during the trial. Each participant willingly volunteered to engage in the experiment, which was approved by the Ethics Committee of Chang'an University. Each participant was given CNY 500 after the trial as compensation for their time.

Table 1. Description of participants’ details.

<table>
<thead>
<tr>
<th>Participant ID</th>
<th>Gender</th>
<th>Age</th>
<th>Driving Experience</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>male</td>
<td>29</td>
<td>4</td>
</tr>
<tr>
<td>02</td>
<td>male</td>
<td>29</td>
<td>4</td>
</tr>
<tr>
<td>03</td>
<td>male</td>
<td>28</td>
<td>3</td>
</tr>
<tr>
<td>04</td>
<td>male</td>
<td>27</td>
<td>3</td>
</tr>
<tr>
<td>05</td>
<td>male</td>
<td>28</td>
<td>4</td>
</tr>
<tr>
<td>06</td>
<td>male</td>
<td>48</td>
<td>21</td>
</tr>
<tr>
<td>07</td>
<td>male</td>
<td>34</td>
<td>7</td>
</tr>
<tr>
<td>08</td>
<td>male</td>
<td>36</td>
<td>10</td>
</tr>
<tr>
<td>09</td>
<td>female</td>
<td>42</td>
<td>15</td>
</tr>
<tr>
<td>10</td>
<td>male</td>
<td>38</td>
<td>8</td>
</tr>
<tr>
<td>11</td>
<td>male</td>
<td>36</td>
<td>7</td>
</tr>
<tr>
<td>12</td>
<td>male</td>
<td>48</td>
<td>23</td>
</tr>
<tr>
<td>13</td>
<td>female</td>
<td>31</td>
<td>4</td>
</tr>
<tr>
<td>14</td>
<td>male</td>
<td>32</td>
<td>5</td>
</tr>
</tbody>
</table>
3.2. Apparatus and Experimental Route

A multifunctional test platform combining a common vehicle and multiple data acquisition devices was deployed for this experiment, and the components of the multifunctional test platform are shown in Figure 5. The millimeter-wave radar was mounted on the center of the rear bumper of the subject vehicle to capture the relative position and relative vehicle speed data between the target vehicle and the subject vehicle. The video monitoring system was used to record the movement of the target vehicle; the GPS was used to record the geographic location of the subject vehicle; the CAN acquisition card obtained the speed information of the subject vehicle; and the wireless button recorded the moment while the participant was performing the task for subsequent data calibration. An industrial control computer was used to collect the data acquired by all components.

![Multifunctional test platform](image)

**Figure 5.** Multifunctional test platform.

In this study, participants were required to look into the rearview mirror several times to estimate the speed and distance of the target vehicle, which may cause distractions while driving. To ensure the safety of the participants, the Xi’an roundabout highway was chosen as the experimental road, where the traffic volume is relatively light, with a two-way six-lane road and having a speed limit of 120 km/h. The total length of the experimental route is 135 km.

3.3. Experimental Design and Procedure

The ability of a driver to decide to change lane depends on an accurate estimation of the speed and distance of the target vehicle. To investigate such an ability, this study conducted realistic road tests at different vehicle speeds. Considering that the daily driving speed for commuters in China mainly ranges from 60 km/h to 90 km/h, we chose 60 km/h, 70 km/h, 80 km/h, and 90 km/h as test speeds. Considering that CCS (Cruise Control System) is becoming more and more common and is increasingly used on highways, in this study, it was simultaneously used to reduce the driver’s workload during driving and to control the longitudinal speed of the subject vehicle. The subject vehicle was required to drive in the middle lane during the test, while the target lane was the left lane (as shown in Figure 6). The target vehicle at the closest distance to the self-vehicle has the greatest influence on the lane change decision during the lane change process, and the driver obtains information about this vehicle through the rearview mirror in daily driving. Therefore, during the test, we requested the driver to estimate the speed and relative distance between the target vehicle and the subject vehicle by using the left rearview mirror.
Therefore, during the test, we requested the driver to estimate the speed of the target vehicle and the relative distance to the subject vehicle by using the rearview mirror as many times as possible, verbally report their estimations, and simultaneously press a wireless button mounted on the left side of the steering wheel under the premise of safe driving. In this experiment, an advisor was seated in the rear seat of the subject vehicle and was responsible for monitoring the target vehicle and alerting the participant to make corresponding speed and distance estimations, as well as recording the subject’s verbal report of the results. To ensure driving safety, a veteran driver was always seated in the co-driver’s seat of the subject vehicle to monitor the driving environment and alert the participant if there was a driving risk.

3.4. Variables Definition

In this study, the effects of the speed of the subject vehicle, the speed of the target vehicle, and the relative speed and relative distance between the subject vehicle and the target vehicle on the driver’s speed and distance estimation errors were investigated. Therefore, the analytical variables are the speed of the subject vehicle, the speed of the target vehicle, the relative speed, and the distance. Moreover, the variables are calculated as shown below.

The relative velocity between the subject vehicle and the target vehicle was expressed as follows:

\[
V_r = V_s - V_t 
\]

(12)

where \( V_s \) the speed of the subject vehicle \( S \), and \( V_t \) is the speed of the target vehicle \( T \). \( V_r < 0 \) indicates that the speed of the subject vehicle \( S \) is less than the speed of the target vehicle \( T \).

Driver’s speed estimation errors were expressed as follows:

\[
V_{er} = V_{es} - V_{re} 
\]

(13)

where \( V_{es} \) is the estimated speed of the target vehicle \( T \) from the subject vehicle \( S \), and \( V_{re} \) is the real speed of the target vehicle \( T \). \( V_{er} < 0 \) indicates that the driver underestimates the speed of the target vehicle \( T \), and \( V_{er} > 0 \) indicates that the driver overestimates the speed of the target vehicle \( T \).

Driver’s distance estimation errors were evaluated as follows:

\[
D_{er} = D_{es} - D_{re} 
\]

(14)

where \( D_{es} \) is the estimated distance between the target vehicle and the subject vehicle, and \( D_{re} \) is the real distance between them. \( D_{er} < 0 \) indicates that the driver underestimates the distance between the two vehicles; \( D_{er} > 0 \) indicates that the driver overestimates the distance between the two vehicles.
4. Statistical Analysis Results

Under most lane change circumstances, the speed of the target vehicle $T$ is greater than the speed of the subject vehicle $S$; therefore, the data for which the speed of vehicle $T$ is less than the speed of vehicle $S$ are excluded. In this study, 2116 sets of relative distance estimation data and 1542 sets of target vehicle speed estimation data were finally collected. Since there were strong linear correlations between the analytical variables, a ridge regression method was used.

Ridge regression, a modified least squares method, is a biased estimation method that is far more accurate than unbiased estimation in the face of the problem of multicollinearity where independent variables are highly correlated [36,37]. There are many methods to set or validate ridge regression parameter $k$, and the most often used is the ridge trace plot, which shows the coefficients as a function of $k$ [38]. The optimal ridge regression coefficient is the minimum $k$ value when the standardized regression coefficients of each independent variable are stabilized as determined by the ridge trace plot.

4.1. Analysis of Relative Distance Estimation Error

4.1.1. General Analysis

The distributions of the motion parameters and distance parameters of the subject vehicle and the target vehicle during the distance estimation are shown in Figure 7. The real distance between the target vehicle and the subject vehicle during the distance estimation fell in the interval [4.23, 186.18], of which about 90% of the distance was clustered between [15.11, 114.90] with a mean value of 56.56 m. The speed of the target vehicle ranged from 52.21 to 174.53 km/h, of which about 60% of the speed of the target vehicle ranged from 95.31 to 119.56 with a mean value of 105.59 km/h. The relative speed of the subject vehicle fell in the interval [4.23, 186.18], of which about 90% of the distance was clustered between [15.11, 114.90] with a mean value of 56.56 m. The speed of the target vehicle ranged from 95.31 to 119.56 with a mean value of 105.59 km/h. The relative distance estimation error fluctuated in the interval [−110.99, −0.04], of which about 75% was clustered in the interval [−49.29, −11.94] with a mean value of −31.66 km/h. The relative distance estimation error fluctuated in the interval [−79.92, 37.38 m], of which about 60% were concentrated in the interval [−12.92, 11.93] with a mean value of −1.15 m.

![Figure 7. Distribution of target vehicle motion and position parameters during relative distance estimation task.](image-url)

Among the 2116 valid speed estimation samples collected in the experiment, 1021 samples underestimated the target vehicle, accounting for 48.25% of the total number of samples, and 1095 samples overestimated the target vehicle, accounting for 51.75% of the total number of samples.
4.1.2. Ridge Regression Analysis

The position and speed parameters between the target vehicle and the subject vehicle with distance estimation errors were modeled and analyzed using ridge regression. The ridge trace plots of each parameter coefficient are shown in Figure 8. We found that the coefficients began to stabilize when $k = 0.32$; thus, 0.32 was chosen as the optimal ridge regression coefficient in this study.

![Figure 8. Distance estimation task ridge trace plot.](image)

While observing the respective coefficients values (Figure 9), we noticed the following: The regression coefficient value of distance was $-0.106 (t = -20.193, p = 0.000 < 0.01)$, implying that distance has a significant negative incline on its corresponding estimation error; the regression coefficient value of relative vehicle speed was $0.286 (t = 10.775, p = 0.000 < 0.01)$, implying that relative vehicle speed has a significant positive effect on the distance estimation error; the regression coefficient value of subject vehicle speed km/h was $0.106 (t = 2.240, p = 0.025 < 0.05)$, implying that the subject vehicle speed has a significant positive bias on the distance estimation error; the regression coefficient value of the target vehicle speed was $-0.285 (t = -8.999, p = 0.000 < 0.01)$, implying that the target vehicle speed has a significant negative effect on the distance estimation error; finally, the regression coefficient value of the driving experience was $-0.061 (t = -0.298, p = 0.766 > 0.05)$, implying that the driving experience does not have a significant impact on the distance estimation error.

![Figure 9. The weights of the effects of each analytical variable on the error of distance estimation.](image)
In conclusion, it can be seen that the relative speed and subject vehicle speed would have a significant positive influence relationship on the distance estimation error. However, the distance and target vehicle speed would have a significant negative influence on the distance estimation error.

4.2. Analysis of Target Vehicle Speed Estimation Error

4.2.1. General Analysis

The distributions of the motion and distance parameters of the subject and target vehicles during the speed estimation task are shown in Figure 10. The actual distance between the target vehicle and the subject vehicle during the distance estimation fell in the interval [2.16, 196.95], of which about 85% of the distance values were clustered between [9.80, 101.71] with a mean value of 55.73 m. The speed of the target vehicle ranged from 54.17 to 155.29 km/h, of which about 60% of the speed of the target vehicle was between 95.18 and 119.56 with a mean value of 105.27 km/h. The relative speeds of the subject vehicles to the target vehicles varied between \(-85.35\) and \(-0.03\) km/h, of which about 75% fell in the interval \([-49.78, -10.94]\) with a mean value of 31.36 km/h. The speed estimation errors varied in the interval \([-41.00, 50.83]\), of which about 70% of them were concentrated in the interval \([-10.73, 16.65]\), with a mean value of \(-0.29\) km/h.

![Figure 10. Distribution of target vehicle motion and position parameters during relative speed estimation task.](image)

Among the 1542 valid speed estimation samples collected in the experiment, 804 samples underestimated the target vehicle, accounting for 52.14% of the total number of samples, and 738 samples overestimated the target vehicle, accounting for 47.86% of the total number of samples.

4.2.2. Ridge Regression Analysis

The distance, relative speed, subject vehicle speed, and target vehicle speed were used as independent variables, and speed estimation error was used as dependent variable to perform ridge regression analysis in the process of target vehicle speed estimation. According to the ridge trace plot shown in Figure 11, \(k = 0.07\) was the optimal choice.
The regression coefficients of each analytical variable on the error of speed estimation were shown in Figure 12. The regression coefficient value of distance was $-0.040 \ (t = -5.537, p = 0.000 < 0.01)$, implying that distance would have a significant negative impact on the speed judgment error. The regression coefficient value of relative speed was $0.185 \ (t = 20.888, p = 0.000 < 0.01)$, implying that the relative speed would have a significant positive influence on the speed judgment error. The regression coefficient value of the subject vehicle speed was $-0.179 \ (t = -10.599, p = 0.000 < 0.01)$, which means that the subject vehicle speed would have a significant negative influence on the speed judgment error. The regression coefficient value of the target vehicle speed was $-0.327 \ (t = -30.224, p = 0.000 < 0.01)$, implying that the target car speed would have a significant negative incline on the speed judgment error. The regression coefficient value of the driving experience was $0.016 \ (t = 0.035, p = 0.972 > 0.05)$, implying that the driving experience would not have a significant influence on the speed judgment error.

Figure 11. Speed estimation task ridge trace plot.

Figure 12. The weights of the effects of each analytical variable on the error of speed estimation.

Summarizing the analysis, it can be observed that the relative speed will have a significant positive bias on the speed judgment error. However, the distance, subject vehicle speed, and target vehicle speed would have a significant negative influence on the speed judgment error.
5. Modeling of Relative Distance and Velocity Estimation

Based on the factors affecting the driver speed estimation and distance estimation errors selected in the previous section, a prediction model for driver speed estimation and distance estimation based on the integrated learning eXtreme Gradient Boosting (XGBoost) algorithm is developed in this section. Firstly, a coarser two-class prediction model is established; that is, the driver’s estimation results are classified into two categories: overestimation and underestimation. Subsequently, the driver’s estimation results were further subdivided into three categories: overestimation, normal, and underestimation. A more detailed three-classification prediction model is derived.

A total of 1542 sets of speed estimation samples and 2116 sets of distance estimation samples were obtained from the actual road tests. In this paper, the data set was divided into 80% training set and 20% testing set. The input parameters of the model are divided into two categories. One is the driver’s experience; the other is the vehicle motion parameters: subject vehicle speed, target vehicle speed, relative vehicle speed, and relative distance.

5.1. Model Construction

5.1.1. Model Label Settings

To provide a comprehensive overview of the classification performance of the speed and distance estimation model, this paper sets the classification of the speed and distance estimation errors into two cases, namely, two and classifications, according to the specific values of the speed and distance estimation three errors and the distribution of the error values. Specifically, “two classification” is used to classify the driver’s estimation into “overestimation” and “underestimation”. Without a loss of generality, “three classification” is used to classify the driver’s estimation error into “overestimation”, “normal estimation”, and “underestimation”.

In the two-classification of the estimation error, the estimation error value 0 is used as a threshold, when the estimation error is greater than 0, the error value label is set to 1 (overestimation); when the estimation error is less than 0, the error label is set to 0 (underestimation), as shown in Equation (1). In the two-classification model, there are 1020 labels of class 0 and 1096 labels of class 1 in the distance estimation dataset; while in the speed estimation dataset, there are 795 labels of class 0 and 747 labels of class 1.

For tri-classification of estimation errors, the 50% median interval of estimation errors was used as label 2 (normal), the upper 25% interval was used as label 3 (overestimation), and the lower 25% interval was as label 1 (underestimation), as shown in Equations (2) and (3). For the three classifications, there are 519 labels of category 1, 1089 labels of category 2, and 508 labels of category 3 in the distance estimation dataset. In the velocity estimation dataset, there are 374 labels of category 1, 802 labels of category 2, and 366 labels of category 3.

\[
E_{\text{speed-estimate}}^{\text{2-class}} = \begin{cases} 
0, & (V_{er} < 0) \\
1, & (V_{er} \geq 0)
\end{cases} 
\]

\[
E_{\text{distance-estimate}}^{\text{2-class}} = \begin{cases} 
0, & (D_{er} < 0) \\
1, & (D_{er} \geq 0)
\end{cases} 
\]

\[
E_{\text{speed-estimate}}^{\text{3-class}} = \begin{cases} 
3, & (V_{er} > 9) \\
2, & (-9 \leq V_{er} \leq 9) \\
1, & (V_{er} < -9)
\end{cases} 
\]

\[
E_{\text{distance-estimate}}^{\text{3-class}} = \begin{cases} 
3, & (D_{er} > 10) \\
2, & (-10 \leq D_{er} \leq 10) \\
1, & (D_{er} < -10)
\end{cases} 
\]
5.1.2. Model Parameter Settings

In the process of training the prediction model based on XGBoost-DNN algorithm, the parameters were adjusted to achieve the best prediction performance. In the XGBoost model, the key parameters include the learning rate, the number of decision trees, and the maximum depth of the decision trees in which the learning rate can control the step length of updating weights in each iteration of training. The maximum depth of the tree can affect the overfitting phenomenon: The smaller the value, the slower the model training speed is and the easier the model is underfitted, while the larger the value, the easier the model is overfitted.

The main parameters of DNN models include the number of hidden layers and the number of neurons per layer, learning rate, activation function, and optimizer. The deeper the number of layers, the stronger the theoretical performance of the model, but it is prone to overfitting, and too many neurons will also result in overfitting. The learning rate affects the model loss function and changes the convergence rate of the model; the role of the activation function is used to incorporate nonlinear factors to solve problems that cannot be solved by a general linear model; and the role of the optimizer is to update and calculate the model network parameters to approximate or reach the optimal value.

In this paper, the model prediction accuracy is the main criterion to measure the parameter selection. In this paper, we start by determining the value range of each parameter, and then we use iterative calculation and control variables to obtain the value for each parameter that maximizes the model’s prediction accuracy. The specific parameter values are shown in Table 2.

In this paper, we use the Python-Sklearn package to train the model with an Intel i9-12900 K CPU and 64 GB of running memory. In the DNN module, the parameter tol is the condition for the model to stop training, and we set tol to 0.001, which means the model stops training when the loss value is less than or equal to 0.001. To avoid overfitting, we adjust the parameter’s subsample and colsample-bytree in the XGBoost module, which represent the ratio of the data and features used in training each tree to the total training set and features, respectively, with typical values of 0.5–1. By adjusting these two parameters, the overfitting of the model can be prevented.

Table 2. Hyperparameters tuning for XGBoost-DNN in this study.

<table>
<thead>
<tr>
<th>Class</th>
<th>Hyperparameter</th>
<th>Search Range</th>
<th>Model</th>
<th>Optimal Value</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Learning rate</td>
<td>[0.01, 0.5]</td>
<td>Speed</td>
<td>0.08</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance</td>
<td>0.07</td>
</tr>
<tr>
<td>XGBoost</td>
<td>Max tree depth</td>
<td>[2, 10]</td>
<td>Speed</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>Minimum loss reduction</td>
<td>[0, 5]</td>
<td>Speed</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>Number of estimators</td>
<td>[20, 150]</td>
<td>Speed</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance</td>
<td>64</td>
</tr>
<tr>
<td>2-class</td>
<td>Hidden layer sizes</td>
<td></td>
<td>Speed</td>
<td>(100,)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance</td>
<td>(100)</td>
</tr>
<tr>
<td>DNN</td>
<td>Learning rate</td>
<td>[0.001, 0.1]</td>
<td>Speed</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>Activation</td>
<td>Relu, tanh, Logistic</td>
<td>Speed</td>
<td>Relu</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance</td>
<td>SGD</td>
</tr>
<tr>
<td></td>
<td>Solver</td>
<td>SGD, Adam</td>
<td>Speed</td>
<td>200</td>
</tr>
<tr>
<td></td>
<td>Number of iterations</td>
<td></td>
<td>Distance</td>
<td>200</td>
</tr>
<tr>
<td></td>
<td>Batchsize</td>
<td></td>
<td>Speed</td>
<td>Auto</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance</td>
<td>Auto</td>
</tr>
</tbody>
</table>
Table 2. Cont.

<table>
<thead>
<tr>
<th>Class</th>
<th>Hyperparameter</th>
<th>Search Range</th>
<th>Model</th>
<th>Optimal Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>XGBoost</td>
<td>Learning rate</td>
<td>[0.01, 0.5]</td>
<td>Speed</td>
<td>0.05</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance</td>
<td>0.05</td>
</tr>
<tr>
<td></td>
<td>Max tree depth</td>
<td>[2, 10]</td>
<td>Speed</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Minimum loss reduction</td>
<td>[0, 5]</td>
<td>Speed</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>Number of estimator</td>
<td>[20, 150]</td>
<td>Speed</td>
<td>67</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance</td>
<td>73</td>
</tr>
<tr>
<td>3-class</td>
<td>Hidden layer sizes</td>
<td></td>
<td>Speed</td>
<td>(100)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance</td>
<td>(100)</td>
</tr>
<tr>
<td>DNN</td>
<td>Learning rate</td>
<td>[0.001, 0.1]</td>
<td>Speed</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance</td>
<td>0.001</td>
</tr>
<tr>
<td></td>
<td>Activation</td>
<td>Relu, tanh, Logistic</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Solver</td>
<td>SGD, Adam</td>
<td>Speed</td>
<td>Relu</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance</td>
<td>SGD</td>
</tr>
<tr>
<td></td>
<td>Number of iterations</td>
<td></td>
<td>Speed</td>
<td>200</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance</td>
<td>200</td>
</tr>
<tr>
<td></td>
<td>Batchsize</td>
<td></td>
<td>Speed</td>
<td>Auto</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Distance</td>
<td>Auto</td>
</tr>
</tbody>
</table>

5.2. Model Evaluation

5.2.1. Model Evaluation Methods

Confusion matrices with good visualization are often used to evaluate the performance of machine learning models [39–41]. In this paper, the confusion matrix is used to evaluate the performance of the binary prediction model and the triple classification prediction model. For \( k \)-element classification, the confusion matrix is represented as a matrix of size \( k \times k \). As an example, the binary confusion matrix is described in Table 2. True Positive (TP), False Negative (FN), True Negative (TN), and False Positive (FP) are the main indicators of the confusion matrix, and the substantial meaning of each indicator is shown in Table 3.

Table 3. Description of the actual meaning of confusion matrix.

<table>
<thead>
<tr>
<th>Confusion matrix</th>
<th>Predicted values (Model prediction outcome)</th>
<th>Overestimation (Positive)</th>
<th>Underestimation (Negative)</th>
</tr>
</thead>
<tbody>
<tr>
<td>True values</td>
<td>Overestimation (Positive)</td>
<td>True Positive (TP)</td>
<td>False Negative (FN)</td>
</tr>
<tr>
<td>(Driver’s estimation outcome)</td>
<td>Underestimation (Negative)</td>
<td>False Positive (FP)</td>
<td>True Negative (TN)</td>
</tr>
</tbody>
</table>

The metrics accuracy rate (\( Acc \)), precision rate (\( P \)), recall rate (\( R \)), and \( F_1 \)-score (\( F_1 \)), which are commonly used for classification model evaluation, can be calculated using these metrics [42]. The formula for each metric is shown below.

\[
Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{19}
\]

\[
P = \frac{TP}{TP + FP} \tag{20}
\]

\[
R = \frac{TP}{TP + FN} \tag{21}
\]
\[ F_1 = \frac{2PR}{P + R} \] (22)

The Receiver Operating Characteristic (ROC) curve is also used to measure the training performance of the model. The ROC curve shows the probability of TP rate and FP rate under different threshold settings, while the Area Under Curve (AUC), which is the region under the ROC curve, reveals better classification performance when its value ranges from 0.5 to 1. The ROC curve is more suitable for evaluating the binary prediction problem; therefore, when evaluating the latter, it is used in addition to the confusion matrix.

5.2.2. Model Evaluation Results

Without loss of generality, in order to verify the applicability of the prediction model based on the XGBoost-DNN mixed algorithm for predicting the driver’s lane change speed and distance estimation, this paper compared its performance with traditional machine algorithm models such as Logistic Regression (LR) [43], Random Forest (RF) [44], Support Vector Machine (SVM) [45,46], K-Nearest Neighbor (KNN) [47], Deep Neural Networks (DNN) [34,35], Gradient Boosting Decision Tree (GBDT) [48], and XGBoost models [49].

Table 4 reports the results of the two-classification model evaluation for speed estimation and distance estimation, and it was found that the model’s accuracy reaches 91.03% in speed estimation and 92.46% in distance estimation.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Speed Estimation</th>
<th>Distance Estimation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Acc (%)</td>
<td>P</td>
</tr>
<tr>
<td>RF</td>
<td>75.32</td>
<td>75.38</td>
</tr>
<tr>
<td>SVM</td>
<td>69.23</td>
<td>69.07</td>
</tr>
<tr>
<td>LR</td>
<td>73.08</td>
<td>72.96</td>
</tr>
<tr>
<td>KNN</td>
<td>76.92</td>
<td>76.89</td>
</tr>
<tr>
<td>DNN</td>
<td>75.96</td>
<td>76.85</td>
</tr>
<tr>
<td>GDBT</td>
<td>85.58</td>
<td>85.64</td>
</tr>
<tr>
<td>XGBoost</td>
<td>88.78</td>
<td>88.83</td>
</tr>
<tr>
<td>XGBoost-DNN</td>
<td>91.03</td>
<td>91.02</td>
</tr>
</tbody>
</table>

It can be observed from Table 4 that XGBoost-DNN has the highest recognition accuracy of 91.03% in speed estimation, which is about 22% greater than the accuracy of SVM model. Compared with RF, LR, KNN, and GBDT models, XGBoost-DNN showed considerable advantages. In addition, the recognition accuracy of XGBoost-DNN is about 2.25% higher than that of the XGBoost model. Similar results are found in distance estimation, and the recognition accuracy of XGBoost-DNN is also the highest, reaching 92.46%, which is about 24% higher than that of the SVM model. Moreover, the recognition accuracy of XGBoost-DNN is about 1.91% higher than that of XGBoost model.

Figure 13 depicts the values of model evaluation metrics for speed estimation and distance estimation for the two-classification case for different models. It is apparent from Figure 13 that the XGBoost-DNN proposed in this paper outperforms the other seven models. Moreover, the prediction performance of XGBoost is ranked second among the seven prediction models.
is indicates that the ROC estimation prediction model; (c) distance estimation prediction model.

Figure 14 depicts the AUC values of the XGBoost-DNN speed estimation model and distance estimation model are 0.957 and 0.963, respectively, which are higher than those obtained using traditional machine learning models. This indicates that the ROC curves also support the observation that the XGBoost-DNN model has better performance in the two-classification cases.

Table 5 reports the results of the three-classification model evaluation for speed estimation and distance estimation, and it was found that the model accuracy reaches 87.18% in speed estimation and 87.59% in distance estimation.
Table 5. Model evaluation results for different algorithms (three-classification).

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Speed Estimation</th>
<th>Distance Estimation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Acc (%)</td>
<td>P</td>
</tr>
<tr>
<td>RF</td>
<td>68.27</td>
<td>75.33</td>
</tr>
<tr>
<td>SVM</td>
<td>63.78</td>
<td>68.26</td>
</tr>
<tr>
<td>LR</td>
<td>63.46</td>
<td>62.31</td>
</tr>
<tr>
<td>KNN</td>
<td>63.46</td>
<td>62.51</td>
</tr>
<tr>
<td>DNN</td>
<td>65.06</td>
<td>67.40</td>
</tr>
<tr>
<td>GDBT</td>
<td>80.13</td>
<td>79.72</td>
</tr>
<tr>
<td>XGBoost</td>
<td>85.25</td>
<td>84.66</td>
</tr>
<tr>
<td>XGBoost-DNN</td>
<td>87.18</td>
<td>88.05</td>
</tr>
</tbody>
</table>

Figure 15 depicts the values of model evaluation metrics for speed estimation and distance estimation for the three-classification case for different models. It is apparent from Figure 15 that the XGBoost-DNN proposed in this paper outperforms the other seven models. Moreover, the prediction performance of XGBoost is ranked second among the seven prediction models.

In summary, for the two-classification problem, high performance is achieved by all three models where the accuracy of all three models is above 72%. However, the XGBoost-DNN model has the highest prediction accuracy, with, respectively, 91.03% and 92.46% for speed and distance estimations, which is much higher than traditional machine learning models, proving excellent prediction accuracy. When compared to the two-classification prediction models, the performances of the three-classification prediction models built using the traditional machine learning algorithms and the XGBoost-DNN mixed algorithm proposed in this paper all plummeted. However, the XBoost-DNN model showed the lowest drop in performance. For speed and distance estimations, the accuracy is 87.18% and 87.59%, respectively, and the prediction accuracy is still better compared to the other traditional machine learning models.

6. Discussion

The risk assessment in driver lane changing mainly comes from the assessment of relative speed and relative distance between the target vehicle and the subject’s own vehicle, and the results of driver assessments are directly related to traffic risk and traffic efficiency. However, in the current lane-change research area, few researchers have conducted studies...
on the speed and distance estimation of drivers by using the rearview mirror during the lane-changing decision process. For this reason, in this study, an actual live highway test was conducted, during which the participants were asked to turn on CCS and make sequential estimates of the absolute speed of the target vehicle and the distance from the subject vehicle at four speeds controlled by CCS: 60, 70, 80, and 90 km/h. Firstly, we explored the surrounding factors that affect driver’s speed estimation and distance estimation errors. Subsequently, two- and three-classification prediction models for driver speed estimation and distance estimation results were developed based on the XGBoost-DNN mixed algorithm.

The results of the ridge regression showed that, in the speed and distance estimations, the speed of the subject vehicle, the speed of the target vehicle, the relative speed, and the relative distance between the subject vehicle and the target vehicle all had significant impact on the estimation error. The larger the relative speed and target vehicle speed, the more the driver tends to underestimate the speed and distance of the target vehicle. The smaller the relative speed and target speed, the more the driver tends to overestimate the speed and distance of the target vehicle. In addition, it can be observed in speed estimation that the higher the speed of the subject vehicle, the more the driver tends to underestimate the speed of the target vehicle; the lower the speed of the subject vehicle, the more the driver tends to overestimate the speed of the target vehicle. However, in distance estimation, the lower the speed of the subject vehicle, the more the driver tended to underestimate the relative distance; the higher the speed of the subject vehicle, the more the driver tended to overestimate the relative distance. In this study, we did not find a conclusive effect of driving experience on speed estimation error and distance estimation error. In addition, it can be observed from the coefficients of the variables in the ridge regression model that both the target vehicle speed and the relative vehicle speed have a large effect on the errors in speed estimation and distance estimation. This study on the effect of test and target vehicle speeds on driver estimation error is consistent with the findings by Wang, et al. [50].

In this study, a driver speed estimation and distance estimation model based on XGBoost-DNN mixed algorithm was established. In the two-classification models, we classify drivers into two categories based on their estimation errors with respect to zero: underestimation and overestimation. The prediction accuracy of the XGBoost-DNN algorithm-based binary classification model for speed estimation and distance estimation reached 91.03% and 92.46%, respectively, which is about two percentage points higher than that of the XGBoost algorithm. Both the confusion matrix method and the ROC curve method proved that the prediction model based on XGBoost-DNN for the two-classification estimation outperformed other traditional machine learning models. In the three-classification model, drivers in the data sample were classified into three categories according to their estimation error distribution: underestimation, normal, and overestimation. The prediction accuracy of the three-classification model based on XGBoost-DNN algorithm reached 87.18% and 87.59% for speed estimation and distance estimation, respectively. The performance of the XGBoost-DNN algorithm-based dichotomous estimation prediction model was verified to be better than that of traditional machine learning models by the confusion matrix method.

Although the prediction accuracy of the XGBoost-DNN algorithm-based driver speed estimation and distance estimation three-classification prediction model is lower than that of the two-classification model, its classification is more specific and can predict the driver’s estimation error situation more accurately.

7. Conclusions and Future Work

In this study, we found that environmental factors affect drivers’ estimation results. The greater the relative speed and target vehicle speed, the more drivers tend to underestimate the speed and distance of the target vehicle. Therefore, drivers should be reminded to focus extra attention to lane-change safety in such cases. We suggest that driver training efforts can be carried out to train drivers to correctly estimate the target
vehicle. The XGBoost-DNN model has a better prediction of driver estimation, which is helpful for providing a theoretical basis for a smarter warning rule for driver lane change warning systems.

In order to further improve the intelligence of the lane change warning system, we will continue to extend this research work in the future. Firstly, we will conduct larger scale trials on real roads to obtain more participants’ environmental estimation data. Secondly, we will recruit more types of participants (e.g., more age and driving experience coverage) in future trials, and we will try to achieve gender balance in the participants. Finally, the driver’s lane change process can be divided into three stages: information perception, lane change decision, and action execution. In this study, we explore the process of information perception, and we will gradually explore the other two stages in our future research.

Author Contributions: Conceptualization, C.Z. and Q.Z.; methodology, X.Z. and Z.L.; software, X.Z. and Z.L.; validation, C.Z. and Q.Z.; formal analysis, X.Z. and Z.L.; investigation, C.Z.; resources, C.Z. and Q.Z.; data curation, Q.Z.; writing—original draft preparation, X.Z. and Z.L.; writing—review and editing, X.Z. and Z.L.; visualization, X.Z. and Z.L.; supervision, C.Z.; project administration, C.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Ministry of Science and Technology (National Key Research and Development Program of China, 2019YFB1600500); the National Natural Science Foundation of China (National Natural Science Foundation of China, 51908054); and Chang’an University (Fundamental Research Funds for the Central Universities, CHD 300102220202).

Institutional Review Board Statement: The research falls within the IRB exempt status.

Informed Consent Statement: Informed consent was obtained from all the participants involved in the study.

Data Availability Statement: Data sharing is not applicable for this article.

Conflicts of Interest: The authors declare that they have no competing interests, and there is no relevant potential conflict of interest with China Communications Press Co.

References
45. Yang, X.; Yu, Q.; He, L.; Guo, T. The one-against-all partition based binary tree support vector machine algorithms for multi-class classification. *Neurocomputing* 2013, 113, 1–7. [CrossRef]