ANNALS OF APPLIED SCIENCES

MIDITERRANEAN JOURNALS

Original Research

DOI: 10.55085/aas.2024.720

A Novel Hybrid Model Based on Rule Learning and Dilated Convolution for Vehicle Collision Prediction

Nan Wu[1], Xue Liu[2], Zhiqiang Lv[2*]

[1] Beijing Institute of Computer Technology and Applications, Beijing, 100854, China
[2] College of Computer Science & Technology, Qingdao University, Qingdao, 266071, China

Check for updates

Received: 08 Oct 2023
Revised: 08 Jan 2024
Accepted: 20 Fab 2024
Published: 18 April 2024

Correspondence:
Dr. Zhiqiang Lv. College of Computer Science & Technology, Qingdao University, Qingdao, 266071, China
Email: 2019025929@qdu.edu.cn

Cite this article as: Wu N, Liu X, Lv Z. A Novel Hybrid Model Based on Rule Learning and Dilated Convolution for Vehicle Collision Prediction. Ann Applied Sciences. 2024:720. [**https://doi.org/10.55085/aas.2024.720**]

Authors' contributions
The participation of each author corresponds to the criteria of authorship and contributorship emphasized in the Recommendations for the Conduct, Reporting, Editing, and Publication of Scholarly work in Medical Journals of the International Committee of Medical Journal Editors. Indeed, all the authors have actively participated in the redaction, the revision of the manuscript, and provided approval for this final revised version.

Acknowledgments
None.

Funding
None

Conflict of interest
The authors declare that there is no conflict of interest regarding the publication of this article.

Ethical approval
Not applicable.

ABSTRACT

Vehicle collisions are a significant concern in road accidents, particularly with the rise of autonomous driving technology. However, existing studies often struggle to accurately predict collisions due to inconsistent correlations between collected data and collision labels. Therefore, this work quantitatively analyzes traffic accident data and constructs new features with strong correlations to the labels. In this study, a rule classification-dilated convolution network (R-DCN) model, which combines rule learning with dilated convolutional networks, is proposed. The rule learning model predicts partially collided vehicles using predefined rules, resulting in interpretability, high prediction efficiency, and quick computation. The remaining vehicle collisions are estimated using dilated convolutional layers, addressing the issue of missing important features in conventional convolution models. To distinguish between intense collisions (predicted by rule learning) and nonintense collisions (predicted by the dilated convolutional model), the data for training the network are those that remove the intense collision predicted by the rule learning model. The proposed model exhibits enhanced sensitivity to nonintense collision data. Compared to existing models, the approach presented in this work demonstrates superior evaluation metrics and training speed.

KEYWORDS: Autonomous driving, collision prediction, dilated convolution, rule learning, feature construction

1. INTRODUCTION

1.1 Research Status

Vehicle collision prediction is an important research field in autonomous driving. It helps ensure that the vehicle reaches its destination efficiently and safely, thereby reducing casualties, traffic jams, and environmental pollution caused by vehicle collisions. Recently, with the availability of numerous datasets, machine learning has surpassed traditional statistical methods in prediction. Researchers use machine learning algorithms such as decision trees, random forests, support vector machines (SVMs), and neural networks to predict vehicle collisions. They also use historical collision data and other relevant information as inputs to train models and learn traffic behavior patterns and collision probabilities. These methods have made some progress in accuracy and real-time prediction, but they still face challenges such as data inconsistency and computational resource requirements. Furthermore, deep learning methods like convolutional neural networks (CNNs) and recurrent neural networks (RNNs) are widely applied in vehicle collision prediction [1]. These approaches can learn feature representations from raw sensor data and possess strong expressive capabilities. Researchers have achieved favorable prediction results by leveraging large-scale labeled datasets and enhancing network structures. However, deep learning methods require high computational resources and lack interpretability.

Intelligent transportation systems have become increasingly important and are believed to contribute to the sustainable development of transportation. Prediction of traffic accidents is a crucial and challenging problem in the field of intelligent transportation. By analyzing the current driving state of the vehicle, accurate predictions of future traffic accident trends can be made, allowing drivers to adjust their driving state promptly and avoid accidents.

However, predicting vehicle collisions is difficult due to numerous influencing factors. Most existing research is based on data collected during vehicle driving, and these data are not strongly correlated with labels. Using this data directly has a limited impact on improving collision prediction accuracy and increases data collection difficulty. Additionally, vehicle collision prediction is a typical binary classification problem, classifying events as collision or noncollision. This poses another problem of the rarity of car crashes, as noncollision events are more common than collision events. As a result, the dataset may suffer from class imbalance issues.

Currently, several common methods for predicting vehicle collisions exist, including machine learning-based and physical model-based methods.

(1) Machine learning-based methods:

Supervised learning method: Using labeled training data to train models, common algorithms include decision trees, random forests, SVMs, and neural networks. However, the performance of these methods highly depends on the quality and quantity of training data, and there may be overfitting or underfitting issues.

Deep learning methods include CNNs and RNNs. These methods can automatically extract features from raw data, but they require a large amount of labeled data and computational resources and lack interpretability.

(2) Physical model-based methods:

Based on dynamic models: Use the vehicle's kinematic and dynamic models to predict collisions. These methods typically require accurate vehicle parameters and sensor measurement data and may pose challenges for complex traffic situations.

Rule-based approach: Use predefined rules and logic to determine potential collision situations. The performance of these methods highly depends on the accuracy and completeness of rules and cannot capture complex nonlinear relationships.

These methods have some common drawbacks when predicting vehicle collisions:

- Data inconsistency: The correlation between directly collected data and labels may be inconsistent, leading to a decrease in the accuracy of predictions.
- Incomplete data: It is often difficult to obtain complete and accurate vehicle behavior data and environmental information, which may affect the performance of the model.
- Model interpretability: Some methods, such as deep learning models, lack interpretability, making it difficult to understand how the model makes predictive decisions.
- Computing resource requirements: Some methods require many computing resources and extensive training time, which may be challenging for real-time applications and large-scale deployment.

To address the limitations of previous studies, this work applies data preprocessing techniques that align with the characteristics of the collected data. By analyzing the data features, this work constructs new relevant features verified to be highly correlated with labels. Additionally, to tackle the class imbalance issue, this work incorporates a synthetic minority oversampling technique (SMOTE). While previous studies have utilized different resampling techniques, the construction of test data is often done manually, which can impact the real-world application performance of the computational model and may result in the loss of valuable information regarding noncollision events. By exclusively applying SMOTE to the training set, this approach ensures that the test dataset reflects real-world information while retaining valuable information about noncollision events during the training process. Furthermore, this work introduces a novel prediction model called the R-DCN model for vehicle collision prediction. Leveraging prior knowledge and analysis of noncollision cases, this work derives rules for a rule learning model, where critical acceleration can indicate collision [2]. When critical acceleration features, such as instantaneous acceleration, fall below a specified value, this work identifies it as a collision. Given the consistent behavior of this type of data, it is collectively referred to as "intense collision." Conversely, the R-DCN model employs dilated convolutional networks to predict nonintense collisions. Compared to ordinary CNNs, dilated convolutional networks expand the receptive field, addressing potential information loss and considering the correlation between the whole and the parts. This makes them more accurate in predicting nonintense collisions, effectively capturing most collisions and displaying sensitivity to nonintense collision data.

1.2 Related Work

Traffic safety has always been a significant concern, and numerous literary works have discussed ways to enhance it. The existing safety system primarily comprises collision prediction and automatic braking, with collision prediction being of utmost importance. Therefore, first, we will review the previously proposed collision prediction algorithms.

Methods based on the Internet of Vehicles (IoV) can be categorized into traditional parametric methods and artificial intelligence-based methods. Parametric methods rely on mathematical formulas to identify the relationship between independent and dependent variables. Rear-end collision prediction extensively utilizes parametric methods, such as the Honda algorithm, which is a time-to-collision (TTC) algorithm. Sensitivity analysis resulted in a threshold value of 2.2 sec [3]. However, the thresholds in these parametric methods are fixed or derived from specific formulas, and their impact can vary in different environments. Moreover, even in similar environments, different drivers may have different perception thresholds. If the threshold is too low, drivers may not have sufficient time to react appropriately. Conversely, if the threshold is set too high, it could lead to numerous false warnings, potentially causing drivers to ignore or reduce the frequency of using warnings. Thus, it can be concluded that parametric methods do not apply effectively in real-world driving situations.

In a prior study [4], logistic regression (LR) was employed to examine the occurrence of traffic accidents in the United States. The results indicated that speed and seat belt usage are significant parameters for predicting the severity of traffic accidents. LR provides interpretability. However, LR is prone to overfitting, reducing the model's robustness. With the ease of data collection and advancements in computing, greater attention has been given to artificial intelligence-based methods. Researchers proposed several collision warning algorithms based on artificial intelligence, including prevalent machine learning algorithms such as decision trees and SVMs. These algorithms effectively address the overfitting issue and enhance prediction accuracy [5]. Furthermore, Huang et al. [6] introduced the application of a CNN with a dropout operation, whereas Ren et al. [7] suggested the use of a long short-term memory neural network (LSTM) for vehicle collision prediction. Such methods can extract high-dimensional information and capture hierarchical features in datasets. They overcome the challenge of low prediction accuracy encountered by logistic models and SVMs and demonstrate greater sensitivity. However, a previous study [8] comparing traffic incident prediction revealed that machine learning models lack variable explanation, and rule-based models require a substantial number of samples. This work combines an interpretable rule learning model with a dilated convolutional model to address these limitations.

Another study [9] argued that traffic variables are critical in predicting car crashes. Quantifying their impact has facilitated the development of countermeasures to enhance traffic safety. Since vehicle collisions primarily occur when vehicles are in motion, the influence of traffic variables on collisions is inevitable. Numerous studies have analyzed the relationship between potential collision probability and various traffic variables. For instance, in the work conducted by [10], the impact of specific variables on vehicles in hazardous conditions was observed, with traffic speed identified as the most crucial parameter. Based on this knowledge, this work constructs several related features based on the collected speed data to make corresponding predictions. Experimental results demonstrate substantial improvement in prediction accuracy.

1.3 Contributions

This work proposes a model R-DCN for predicting whether a vehicle has a collision. It uses the state information and motion information of the vehicle at a specific time to predict whether a collision could occur. The contributions of this paper are as follows:

(1) In this work, vehicle collisions are predicted from the data analysis perspective. Compared with the existing research, this work trains the model based on the features constructed from the collected vehicle information rather than directly using the collected data. It is of great significance for improving the accuracy of vehicle collision prediction and reducing the time used. It is also of great significance to improve the performance of the vehicle-assisted driving system.

(2) This work introduces a model R-DCN that fuses a rule learning model with a dilated convolutional network. The rule learning model is mainly based on the defined rules to predict vehicle collisions quickly. A dilated convolutional network makes predictions on the remaining vehicles. Thereby, prediction accuracy is improved and the time required for prediction is reduced.

(3) This work uses a real dataset for experimental evaluation. The results show that the model formed by combining rule learning with the dilated convolutional network is highly accurate for predicting vehicle collisions. Compared with existing models for predicting vehicle collisions, the model prediction accuracy is improved by at least 12%.

2. METHODOLOGY

2.1 Data Preprocessing

Sufficient analysis of the raw data is helpful for understanding and selecting appropriate data. Therefore, before conducting experiments, this work first implements data cleaning and solves the class imbalance problem.

2.1.1 Data Cleansing

Data cleansing refers to identifying and handling incomplete, incorrect, inaccurate, and irrelevant data in a dataset. Common methods used for processing include deletion, replacement, or modification. In this work, the focus is on the label set. Based on reading materials and general knowledge, the label set is expected to possess the following characteristics. Firstly, during a vehicle collision, the state of the main negative relay changes from connected to disconnected. This is due to the car's built-in short circuit prevention function. Specifically, when a collision occurs, the airbag controller sends a signal to the battery management system, which then disconnects the high voltage through the battery relay to prevent fires. Secondly, a significant change in vehicle speed is expected during a collision. Considering these factors, anomalies are identified in the raw label set and corrections are made to five labels. For example, for car number 5, there is a malfunction in the vehicle sensor, resulting in a constant speed of "0" before and after the collision. Hence, the collision label for that car is changed to "0."

Upon initial analysis, it was observed that some data in the dataset were missing. To address this, a novel data-cleaning approach is proposed. For data with a missingness rate exceeding 10%, deletion is directly applied. The quantity of deleted data is minimal and is not expected to impact the experimental results significantly. Furthermore, deduplication is performed on the dataset, keeping only the first data entry for vehicles with the same vehicle number and collection time. The data is then sorted based on collection time. Additionally, data where the "seat occupancy status of main driver" is recorded as "sensor failure" are also removed to ensure data authenticity and reliability.

2.1.2 Data Imbalance

After integrating the dataset with the label set, it was observed in this work that the number of collision data instances is significantly lower than that of noncollision data instances. Common methods used to address the class imbalance issue include under-sampling the noncollision data, oversampling the collision data, and other processing techniques. However, using under-sampling alone could result in the loss of a significant amount of noncollision data and important related information. Similarly, relying solely on oversampling could excessively emphasize the characteristics of collision data and lead to overfitting. To overcome these challenges, the SMOTE (synthetic minority oversampling technique) approach is employed in this experiment. This method is a traditional oversampling technique [11]. It involves creating a vector between the collision data and its k nearest neighbors and then multiplying this vector by a random constant between 0 and 1. The resulting vector is added to the dataset as new collision data. Using this approach, a collision data to noncollision data ratio of 1:1 is achieved. Previous studies, such as the one conducted by Basso et al., successfully utilized the SMOTE method to generate collision data for highway vehicles, demonstrating good alignment with real data [12]. Additionally, many existing collision prediction experiments have adopted algorithms based on SMOTE-balanced datasets, yielding promising predictive performance [13]. The accuracy of prediction can be improved using the SMOTE method.

2.2 Problem Definition and Data Design

2.2.1 Problem Definition

The vehicle collision prediction problem can be defined as predicting whether the vehicle will collide based on the current relevant attribute information of the vehicle ($\left[x_1, x_2, ..., x_{n-1}, x_n\right]$, where n represents the number of attributes). That is, the collected information of the vehicle in motion and the feature constructed on this basis are used as the input of the model. The output of the model is to predict whether a collision will occur. The formula is shown in (1). In this way, the task of predicting whether a vehicle collision occurs or not is realized.

$$y_i = M(x_1, x_2, \dots, x_{n-1}, x_n), \quad (1)$$

Among them, M represents the modeling method and n is the number of features.

2.2.2 Data Design

Usually, the description of an object could involve multiple attributes. These attributes include related features, irrelevant features, and redundant features. The dataset used in this study contains a total of 22 attributes. It is too complicated to use all the features to predict vehicle collisions, which could greatly increase the difficulty and time of training. Therefore, this experiment is based on this dataset to process the data.

2.2.2.1 Data Transformation

Firstly, the features in the dataset are classified into two categories: state information and motion information. The state information mainly includes a description of the vehicle and the driver's state at the current time point. Conclusions can be drawn by calculating the correlation between state information and prediction results. For state information, the most important features are the start-stop state of the vehicle. And because the start-stop state of the vehicle could be inferred through the change of the vehicle's relay state, this work uses the relay state to construct the *if_off* feature to represent the start-stop state of vehicle. In the data preprocessing stage, this work encodes the state of the main negative relay of battery. Moreover, the difference between each data and the previous five times separately was calculated to get the state change of the battery five times. This work sums these five values to describe the state change of the battery over the previous time period. i.e., the value of *if_off*. As shown in Fig. 1, when the relay state changes from connected to disconnected, the feature value could change from continuous “1” to continuous “0.” On the *if_off* eigenvalue, it gradually changes from “-5” to “-1” and it is “0” at the rest of the time.

Battery pack main negative relay status	Battery status 0	Battery status 1	Battery status 2	Battery status 3	Battery status 4	if_off
1	0	0	0	0	0	0
1	0	0	0	0	0	0
1	0	0	0	0	0	0
1	0	0	0	0	0	0
1	0	0	0	0	0	0
0	-1	-1	-1	-1	-1	-5
0	0	-1	-1	-1	-1	-4
0	0	0	-1	-1	-1	-3
0	0	0	0	-1	-1	-2
0	0	0	0	0	-1	-1

Figure 1: if_off feature structure

Second, categorical feature encoding was performed; i.e., nonnumeric features in the raw dataset were encoded. Then, they were turned into statistical data types. The specific transformation is shown in Table 1.

Table 1: Features for categorical feature encoding.

Features	Nonnumeric value	Numeric value
Brake pedal status	Not stepped	0
	Step on	1
Driver departure reminder	No warning	0
	Warning (Remove the key)	1
Main driver's seat occupancy status	Vacant	0
	Occupancy	1
Driver's safety belt status	Tied	0
	Not tied	1
Handbrake status	Handbrake down	0
	Handbrake up	1
Vehicle key status	OFF	0
	ON	1
The current gear status of the vehicle	Neutral	0
	Forward	1

Finally, according to the conclusions drawn from previous studies, traffic speed has a greater impact on the occurrence of traffic accidents. This work has constructed the relevant feature. By analyzing the data in the dataset and the speed of the vehicle at different times, this work constructs several features about the vehicle's motion state, which are the vehicle's instantaneous acceleration ($v_{instant}^{i}$), local acceleration (v_{local}^{i}), speed difference ($v_{different}^{i}$), and acceleration statistics (a_{i_mean}, a_{i_min}, a_{i_max}). At the same time, the bucketing operation was performed on the vehicle speed difference feature. Features were defined as follows:

$$v_{instant}^{i} = \frac{v_i - v_{i-1}}{t_i - t_{i-1}} \tag{2}$$

$$v_{local}^{i} = -\frac{1}{3}\frac{v_{i-3} + v_{i-4} + v_{i-5}}{t_i - t_{i-4}} \tag{3}$$

$$v_{different}^{i} = v_i - v_{i-1} \tag{4}$$

$$a_{i\_mean} = mean(v\_diff1_{i-2} + v\_diff1_{i-1} + v\_diff1_{i}) \; a_{i\_min} = min(v\_diff1_{i-2} + v\_diff1_{i-1} + v\_c$$

$$a_{i\_max} = max(v\_diff1_{i-2} + v\_diff1_{i-1} + v\_diff1_{i}) \tag{5}$$

Among them, v_i represents the vehicle speed at the *i*-th time and t_i is the *i*-th time.

2.2.2.2 Correlation Analysis and Feature Selection

Selecting relevant features from these features is important for collision prediction since only relevant features can help improve the accuracy of model predictions. There are related strategies for feature engineering in different fields. Correlation can help identify features that are highly correlated with the target variable, thereby providing better predictive performance. This information can help eliminate redundant features, reduce data dimensionality, and improve the interpretability and generalization capability of the model. However, correlation as a feature selection method also has limitations, as it can only capture linear relationships and may not accurately describe nonlinear relationships. The principle of feature selection methods is based on measuring the degree of correlation between features and the target variable to identify the most relevant features to the target variable. Feature selection is of great importance in machine learning and data mining, as it can help improve the model's generalization capability and reduce the risk of overfitting [14]. This work mainly uses feature classification and computational correlation for feature selection.

Before using the model to predict vehicle collision, this work verifies the correlation between the given data to determine the necessity of using the model prediction. If the data in the dataset is highly correlated, then this work can directly use the linear model to make predictions. The training process of the linear model is much simpler than the model prediction. This work verifies the correlation of the features in the dataset except for the car number and collection time. The results are plotted as a Pearson correlation coefficient matrix. Its correlation coefficient is calculated according to formula (6). The result is shown in Fig. 2. The graph can describe the correlation between various features in detail and intuitively. Furthermore, according to this, you can determine whether you need to use a linear algorithm for prediction.

$$\rho(\mathrm{X}, \mathrm{Y}) = \frac{E\left[\left(X - \mu X\right)\left(Y - \mu Y\right)\right]}{\sigma X \sigma Y} = \frac{E\left[\left(X - \mu X\right)\left(Y - \mu Y\right)\right]}{\sqrt{\sum_{i=1}^{n}(X_i - \mu X)^2}\sqrt{\sum_{i=1}^{n}(Y_i - \mu Y)^2}}$$

The Pearson correlation coefficient map mainly represents the correlation between features according to the depth of the color [15]. The darker the color, the higher the correlation between the features. As shown in Fig. 2, the correlation between the given data is not high, indicating that simple linear models may not predict the occurrence of collisions well. Thus, we must solve this problem through a deep learning algorithm. Of course, the correlation between the two is still relatively high when it comes to the features of the vehicle's own state, such as the current and voltage of the current vehicle. This in itself is logically factual and speaks to the reliability of this dataset.

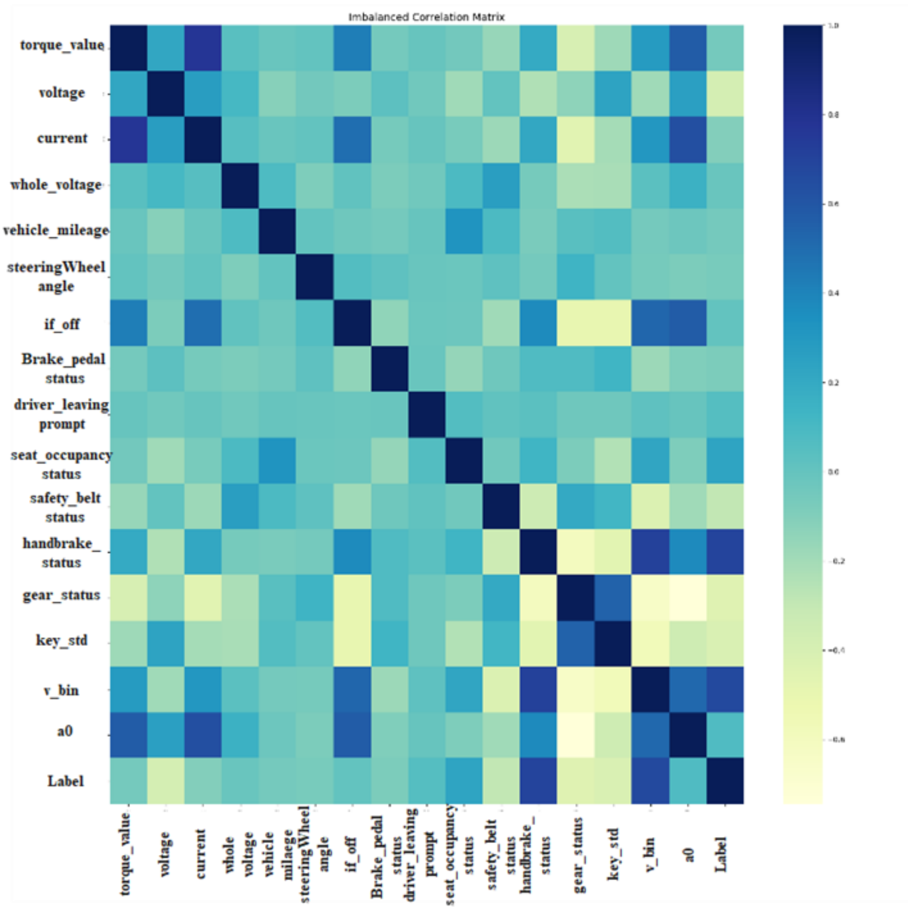


Figure 2: Correlation between features.

The correlations between two types of features and labels, state information and motion information, are calculated separately. The results are shown in Fig. 3. This work can conclude that the motion information of the vehicle has a greater impact on the accuracy of collision prediction. Therefore, this work selects all the features in the vehicle motion information and the top two features in the state information, i.e., the vehicle instantaneous acceleration ($v^i_{instant}$), the local acceleration (v^i_{local}), the acceleration statistics (a_{i_mean}、a_{i_min}、a_{i_max}), and the key state (*key_std*) and *if_off* . Research on vehicle collision prediction was conducted using these seven classes of features.

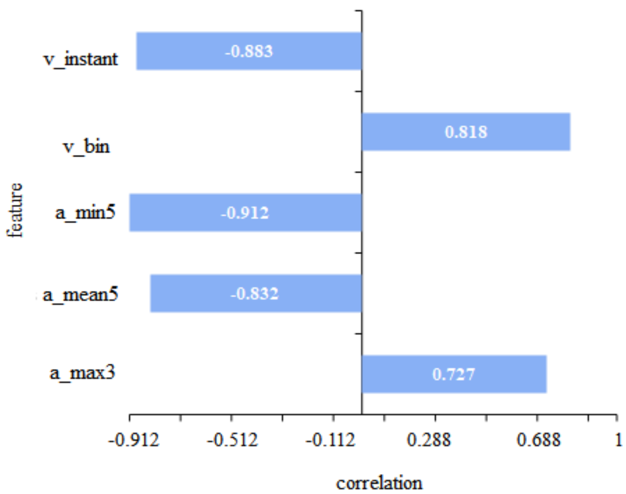


Figure 3a: Correlation between motion state and label.

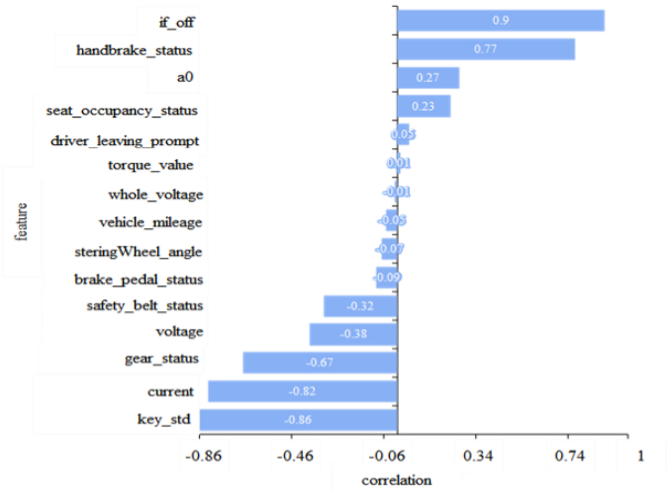


Figure 3b: Correlation between state information features and collision labels.

2.3. Model Design

The main structure of the model is introduced in this section in Fig. 4. Firstly, all the preprocessed data apply a rule learning model. The model predicts vehicle collisions through defined rules. Second, the relevant information of noncollision vehicles predicted by rule learning is used as the input data of the dilated convolutional network. This work selects seven features with strong label correlation as input in this process. Finally, the final features are normalized and linearized to the result of whether the vehicle has a collision. The dilated convolutional layer expends the receptive field, allowing each convolutional output to contain a wider range of information. Furthermore, this layer reduces the computational complexity of the model.

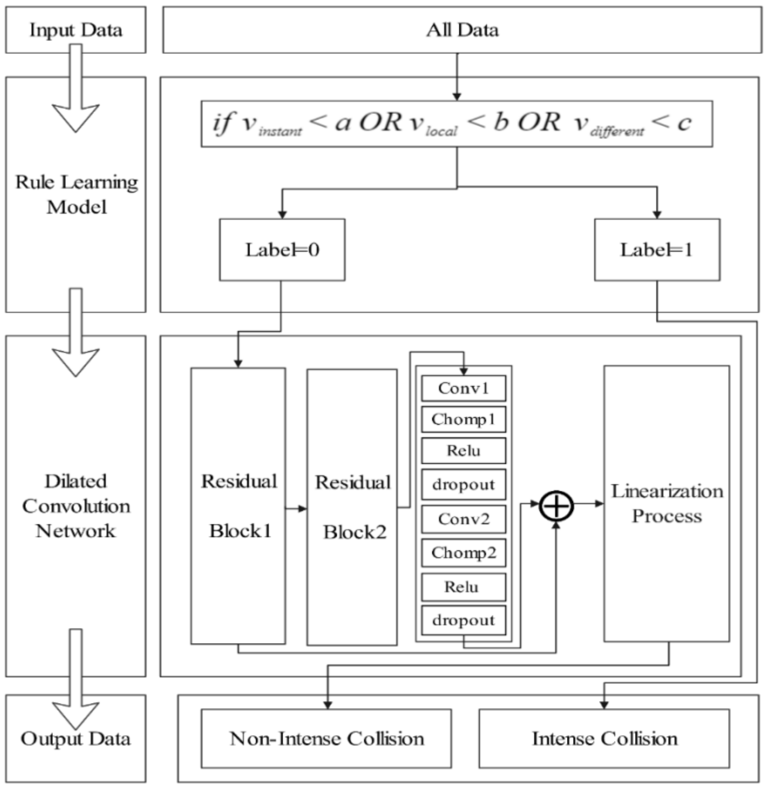


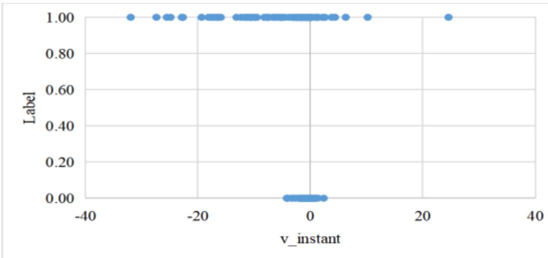
Figure 4: The main structure of the model

2.3.1 Rule Classification Layer

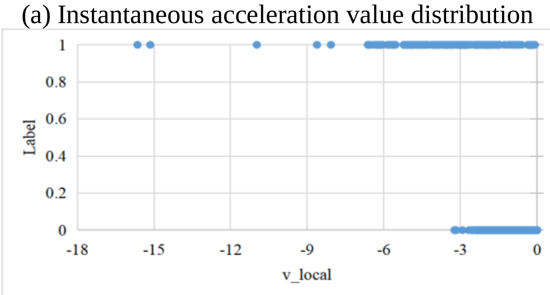
In this experiment, the rule learning model is used as the rule classification layer. The function of this layer is to quickly predict partial results according to the defined association rules [16]. This improves the speed of model prediction. Rule learning is to learn a set of rules for judging unseen examples from training data. It is also a manifestation of the characteristics of the predicted target data. Building a rule model itself is an advanced feature engineering. Rule learning has better explanations than "black-box models," such as neural networks and SVMs. It can give a more intuitive understanding of the classification process. It can be obtained from the previously analyzed correlation of features and collision labels. Several velocity-related features constructed have the highest correlation with the predicted results. Some previous work found that the value based on the critical deceleration is a good way to judge whether a collision occurs. Therefore, some studies use critical deceleration as a sign of collision, for example, 5.892 [17]. In this work, a rule model is established based on the relationship between the three characteristics of instantaneous acceleration, local acceleration, and velocity difference from the previous moment and the predicted target. The specific representation is as follows:

If $v_{instant}$<a OR v_{local} < b OR $v_{different}$ <c THEN Label=1 (7)

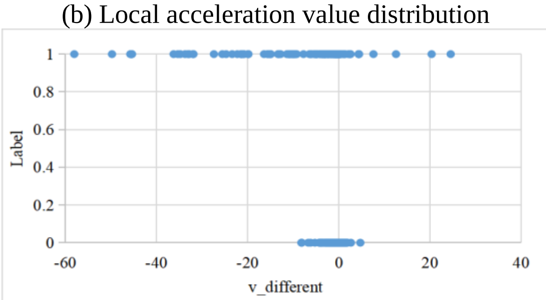
To better adapt to the actual environment, this work visualizes the distribution of the constructed features, as shown in Fig. 5. This work revealed that when the instantaneous acceleration ($v_{instant}^{i}$) is less than -4, the local acceleration (v_{local}^{i}) is less than -3, and the speed difference from the previous moment ($v_{different}^{i}$) is less than -20, the phenomenon of collision begins to appear. Considering the generalization ability of the algorithm, this work sets the separation threshold very large. More than twice the minimum value of the three features of the noncollision data is taken as the separation threshold to avoid overfitting. In this study, this work defines a collision as a rule when the instantaneous acceleration is less than -9 $\mathrm{m/s^2}$, the local acceleration is less than -6 $\mathrm{m/s^2}$, and the acceleration difference is less than -40 $\mathrm{m/s^2}$. The vehicle collision predicted by the rule learning model has high data consistency and a large absolute value, so this work calls it an intense collision. The data predicted by the rule learning model as noncollision are input into the dilated convolution model, and the predicted vehicle collision is a nonintense collision.



(a) Instantaneous acceleration value distribution



(b) Local acceleration value distribution



(c) Distribution of velocity difference from the previous moment

Figure 5: The value of features distribution.

2.3.2 Dilated Convolution Model

The convolutional network used in this experiment is composed of the dilated network as the standard convolutional layer. Furthermore, this work encapsulates the convolution layer and the identity map into a residual module. The deep network is then stacked by the residual module. Finally, set the fully connected layer.

The residual module structure is shown in Fig. 6. The introduction of the residual module solves the training problem of deep networks and the problem of gradient disappearance or gradient explosion [18]. The dilated convolution formula is shown in (8). $\{x_1, x_2, ..., x_t\}$ is the input sequence, $\{y_1, y_2, ..., y_t\}$ is the output sequence of the hidden layer, and $\{f_1, f_2, ..., f_t\}$ are the filters. The filter sequence represents the change of the convolution kernel in multiple hidden layers, which determines the change of the receptive field of the convolutional network in the hidden layer. *d* represents the dilation factor, which varies by an exponential of "2" depending on the depth of the network.

$$y_t = \sum_{i=0}^{K-1} f_i \cdot x_{t-i\cdot d} \tag{8}$$

Among them, the model's input is three-dimensional (N, C, L). N represents the batch size, C is the dimension of the input data, and L represents the length of the sequence, i.e., the number of features of the data. In the first residual block, the input data is 50×1×7 and the output data is 50×6×7. The input and output data are both 50×6×7 in the remaining two residual blocks.

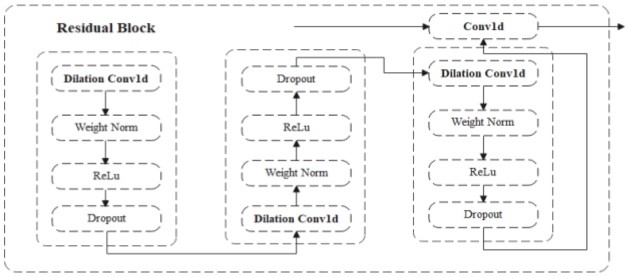


Figure 6: Residual block structure.

The function of the fully connected layer is to integrate the features and output them as a value. As shown in formula (9). The input of this layer is the feature extracted by the network from the input data, i.e., x. In this experiment, the dimension of the input data is 50×6×7. We can regard w_j as the weight of the feature under the j-th category, i.e., the importance of each dimension feature or the degree of influence on the final classification. Moreover, scores for each category are obtained by weighted summation of the features. The final output data is 50×2.

$$z_i = w_i \cdot x + b_i = w_{i1}x_{i1} + w_{i2}x_{i2} + ... + w_{in}x_{in} + b_i \tag{9}$$

This experiment predicts whether a vehicle collides, which is a binary classification problem. Therefore, this work applies a normalization function to the data output by the fully connected layer. The normalization function mainly plays the role of mapping the output of the fully connected layer to the probability of its category. The normalization function used in this experiment is Log SoftMax, which mainly solves the phenomenon of data overflow in the SoftMax function. Log SoftMax can also convert exponential calculations into addition calculations, thereby improving calculation efficiency and data stability [19]. The calculation formula is as follows:

$$\log[f(x_i)] = \log\left(\frac{e^{x_i}}{e^{x_1} + e^{x_2} + ... + e^{x_n}}\right) = \log\left(\frac{\frac{e^{x_i}}{M}}{\frac{e^{x_1}}{M} + \frac{e^{x_2}}{M} + ... + \frac{e^{x_n}}{M}}\right) = \log\left(\frac{e^{(x_i - M)}}{\sum_j^n e^{(x_j - M)}}\right)$$

$$= \log(e^{(x_i - M)}) - \log(\sum_j^n e^{(x_j - M)}) = (x_i - M) - \log(\sum_j^n e^{(x_j - M)})$$

Among them, $M = \max(x_i) \quad i = 1, 2, ..., n$. That is, M is the maximum value among all x_i.

3. EXPERIMENTS AND RESULTS

This section introduces the specific experimental process and the experimental results obtained according to our proposed method. The experimental process is shown in Fig. 7. To verify that this method predicts better, this work also implements several other models. Furthermore, using the same dataset to train the model, the prediction results show that our proposed model fused rule learning and dilated convolution (R-DCN) has high accuracy for predicting collisions.

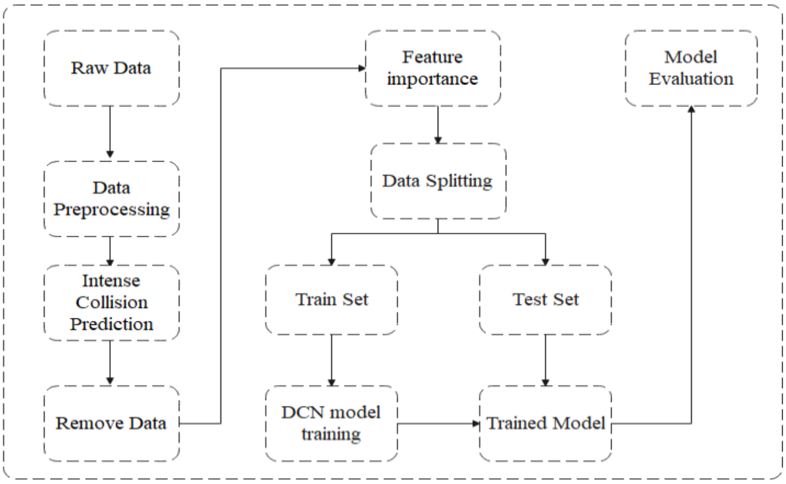


Figure 7: Experimental flowchart.

3.1 Dataset Introduction

This research and analysis are based on the IoV, and various information data are obtained through various sensors. The IoV information primarily includes vehicle control information, battery information, and motor information. The data comes from real datasets. There are 19 features of vehicle information data, such as vehicle number, accelerator pedal position, battery pack main negative relay status, battery pack main positive relay, brake pedal status, a driver leaving prompt, main driver seat occupancy status, driver seat belt status, driver demand torque value, handbrake status, vehicle key status, low voltage battery voltage, vehicle current gear status, vehicle current, vehicle current voltage, vehicle mileage, vehicle speed, and steering wheel angle and other characteristics. Table 2 shows the structure of this dataset.

Table 2: Dataset structure

Num	Collection time	Accelerator pedal position	Battery pack main negative relay status	Battery pack main positive relay status	Brake pedal status
1	2020/8/30 9:30	27	Link	Link	Not stepped
Driver departure reminder	Main driver's seat occupancy status	Driver seat belt status	Handbrake status	vehicle key status	Low voltage battery voltage
No Warning	Someone	Already tied	Put down the handbrake	ON	13.89
The current gear status of the vehicle	The current total current of the vehicle	The current total voltage of the vehicle	Vehicle miles	Speed	Steering wheel angle
forward	53.6	121.9	7667	24.2	9.5
if_off	Instantaneous acceleration	Local acceleration	Speed difference	Acceleration statistic maximum	Statistical minimum of acceleration
-5	-19.7	-20.9	-10.7	2.06	-12.81
Statistical average of acceleration	Label				
-8.06	1				

3.2 Baseline Introduction

To prove the performance of the model, this work selects several representative baselines for comparison. The selected baselines include a traditional statistical model (Honda algorithm), a machine learning model (LR model), two traditional recurrent neural networks (RNN, GRU), and three new models (RFCNN, Multi-Task DNN, CNN-GRU). The details are as follows:

1) Honda algorithm: The Honda algorithm is an understanding warning algorithm based on the TTC algorithm. According to sensitivity analysis, the threshold is set to 2.2 seconds [3]. Its formula is (11).

$$R_{warning} = 2.2V_{rel} + 6.2 \tag{11}$$

2) Logistic Regression: The LR model is widely used in traffic safety [13]. The essence of this model is a classification model, and the most common one is to implement binary classification. The idea of this algorithm is to map the result of a linear function onto a sigmoid function. Usually, the dependent variable is a binary indicator of collision occurrence. The probability of collision is P(y=1), and the probability of noncollision is P(y=0).

Related parameter settings:

C: The reciprocal of regularization intensity, floating-point number, is set to 1.0 in this experiment. Regularization helps avoid overfitting, and a smaller C value indicates a stronger regularization term; Penalty: Regularization type, string {'l1 ','l2', 'elasticnet', 'none'}, set to 'l2' in this experiment. It determines the type of norm used for regularization, where 'l1' represents L1 regularization and 'l2' represents L2 regularization; Max_ Iter: Maximum number of iterations, integer, set to 100 in this experiment; Solver: Optimization algorithm, string {'newton cg ',' lbfgs', 'liblinear', 'sag', 'saga'}, set to 'lbfgs' in this experiment. The formula looks like this:

$$y \ Bernoull(p) \tag{12}$$

$$\log it(p) = \alpha + \beta x \tag{13}$$

3) Convolutional Neural Network (CNN): CNN is a deep neural network mainly used to process complex data [19]. CNN models have convolutional layers, pooling layers, activation layers, dropout layers, and flatten layers. The convolutional layer is used to extract the features of the data, the pooling layer reduces the size of the features, the dropout layer is used to reduce overfitting, and the flatten layer is used to convert the input data into an array. It is widely used in the field of predicting vehicle collisions. Especially when the input data is multidimensional, it can better capture the influence of features on the prediction results. Convolutional kernel size is 3 * 3, using maximum pooling

4) GRU: GRU model is a variant of LSTM model and RNN model. It is like LSTM, so GRU can also solve the problem of long-term dependency caused by gradient explosion or gradient disappearance in RNN. Concurrently, compared with LSTM, the structure is simpler and easier to calculate [20].

Related parameter settings:

Hidden size: 256; embedding dimension of words: 300; batch size: 64; learning rate: 0.001; number of iterations of training: 10.

5) RFCNN: RFCNN is obtained by combining tree-based machine learning models and deep learning models (RF and CNN). The final class is decided according to the maximum average probability mainly by soft voting the predicted probabilities made by the two classifiers. It is mainly used to predict traffic accidents.

Related parameter settings:

N_ Estimators: The number of trees in a random forest. Set to 200; Max_ Depth: The maximum depth of each tree. Control the growth depth of the tree and set it to 20; Min_ Samples_ Split: The minimum number of samples required for node splitting, set to 3; Min_ Samples_ Leaf: The minimum number of samples required for the leaf node, set to 2; Criterion: Select 'gini' as the standard used for splitting.

6) CNN-GRU: CNN-GRU is a model obtained by combining two deep learning methods. The primary consideration is that various models have different advantages and can be combined to get more accurate predictions. The CNN part of the model can mainly learn local representations in the data. The GRU can extract temporal dependencies from the local features. It also has applications in the field of traffic safety. Its parameter settings are the same as those of CNN and GRU.

7) MCWA: MCWA is a collision warning algorithm based on a multilayer perceptron neural network, which is mainly composed of an input layer, a hidden layer, and an output layer. It mainly achieves the ability to predict and detect possible severe decelerations in the ensuing seconds, thereby reducing the risk of collisions without being affected by the PRT of different humans. The feature extractor chooses to use a pretrained CNN, ResNet.

3.3 Quantitative Analysis

In the aspect of the evaluation model, this work adopts four indicators and an ROC curve based on the confusion matrix. In the confusion matrix, accuracy is an evaluation index for traditional classification problems, which represents the proportion of the total data set of all correctly judged result stations in the entire classification model [21]. However, the accuracy rate has a disadvantage. When the categories of the data are not uniformly distributed, the accuracy rate is no longer objective. Therefore, this work introduces Precision, Recall, and F1 Score as evaluation metrics [22]. The specific calculation is shown in Table 3.

Table 3: Calculation method of evaluation index.

Indicators	Calculation methods
Accuracy	$\frac{TP+TN}{TP+TN+FP+FN}$
Precision	$\frac{TP}{TP+FP}$
Recall	$\frac{TP}{TP+FN}$

The ROC curve is a graph showing the performance of the classification model under all classification thresholds. The vertical axis of this curve is true positive rate (TPR), and the horizontal axis is false positive rate (FPR). The calculation formula is as follows: the closer the classification target to the (0,1) point, the better the performance of the classifier [23]. The ROC curve of the above model is shown in Fig. 8. Compared with other models, the ROC curve of the R-DCN model is closer to the upper left corner, i.e., the classification performance of R-DCN is better than that of other models.

$$TPR = \frac{TP}{TP+FN} \tag{14}$$

$$FPR = \frac{FP}{FP+TN} \tag{15}$$

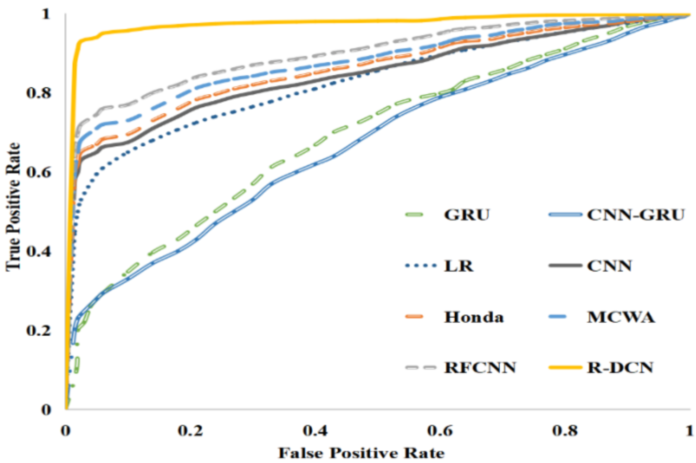


Figure 8: ROC curves of various models.

Table 4: Quantitative analysis results.

	Accuracy	Precise	Recall	F1 Score
GRU	0.4022	0.4416	0.439	0.4403
CNN-GRU	0.500	0.5384	0.5698	0.5537
LR	0.745	0.858	0.586	0.696
CNN	0.712	0.674	0.770	0.722
Honda	0.59	0.78	0.7547	0.767
MCWA	0.836	0.800	0.8118	0.8058
RFCNN	0.812	0.842	0.864	0.853
R-DCN	0.9533	0.9675	0.9320	0.9477

The evaluation results of all models are shown in Table 4. R-DCN has significant performance improvement compared to other models. For the F1 Score indicator, the R-DCN model has an increase of 115.92% relative to GRU, 71.42% relative to CNN-GRU, 36.22% relative to LR, 31.12% relative to CNN, 23.48% relative to Honda, 17.57% relative to MCWA, and 11.12% relative to RFCNN. The results show that the proposed R-DCN is effective. The Honda algorithm is a traditional statistical method, and the experimental results vary significantly in different environments, so accuracy is low. The LR model is a simple machine learning model that solves the problem of different external environments that make the results different. However, due to the structure of LR being relatively simple, there could be underfitting during the experiment, resulting in poor performance; CNN and

GRU are traditional deep learning models, and the accuracy of the model has been improved. But they have a narrower field of view compared to dilated convolutions. So, they perform poorly under the same conditions; the CNN-GRU and RFCNN integrated model have complex structures, and their time-consuming increases in the case of improved accuracy in predicting vehicle collisions. In contrast, the combination of the R-DCN model improves the accuracy of the model. Concurrently, the rule classification algorithm uses prior knowledge to judge collisions, greatly reducing the time of model training and predicting.
The rule learning and dilated convolution models can reduce training and inference time. The evaluation results of all models are shown in Table 5. The results in Table 5 show that the R-DCN model can reduce training and inference time compared to other baseline models.

Table 5: Quantitative analysis results.

	Training time	Inference time
GRU	3.9 min	2 ms
CNN-GRU	39 min	16 ms
LR	17 s	2 ms
CNN	3.5 min	2 ms
Honda	4.9 min	3 ms
MCWA	16 min	4 ms
RFCNN	1.6 h	7 s
R-DCN	1.3 min	2 ms

3.4 Qualitative Analysis

As shown in Fig. 9, in this experiment, the data in the raw dataset, the data after feature construction, and the data after feature engineering are input into the model for experiments. The experimental results show that the data after feature engineering as the model's input has the highest prediction accuracy. This is mainly because, after feature engineering, this work selects seven features from data that are highly correlated with labels as input, which improves prediction accuracy and saves time in training the model. The data in the raw dataset is weakly correlated with the labels, making the model perform poorly. In the data after feature construction, there are features with a strong correlation with the label but also features with a weak correlation or irrelevance. Therefore, the model's performance is not good, and the training time of the model is increased.

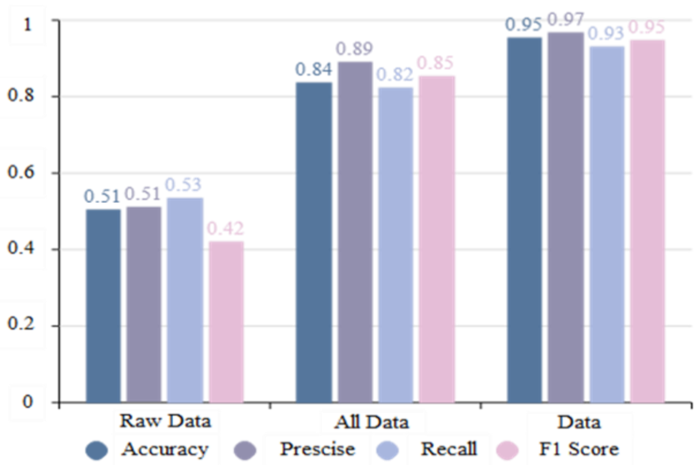
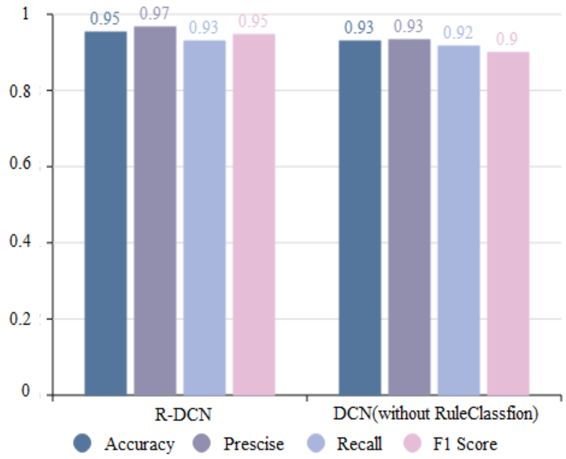


Figure 9: Prediction results of raw data, all data, and selected data as input, respectively

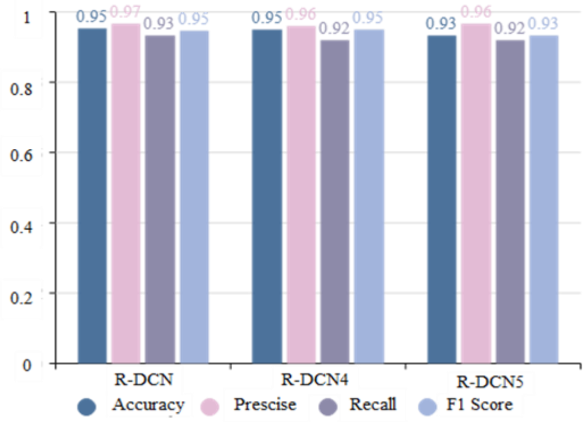
3.5 Ablation Experiments

In this section, to verify the significant superiority of the proposed model, this work sets up two related ablation experiments. Firstly, it is verified that the fusion of the rule learning model and the DCN model has significant advantages in traffic collision prediction. This work compares it with the result obtained by directly inputting the processed data into DCN without the rule learning part. Second, to verify that three-layer dilated convolution is the most suitable number of layers, we design an ablation experiment on the number of layers, using four-layer and five-layer dilated convolution for prediction, respectively. The computational progress of each algorithm is calculated and recorded by the same computer, and the results are shown in Fig. 10.

Fig. 10-a shows that the rule learning model [24] plays a vital role in the entire model, which helps to improve the prediction of nonintense collisions. The model without the rule classification layer has the worst prediction accuracy. At the same time, this result also shows the necessity of feature extraction for the raw data. This work finally selects seven features for prediction by calculating the correlation coefficient between the raw features and the label of collision. This not only improves the predictive performance of the model but also reduces the cost of data collection. Fig. 10-b shows that when the number of layers is set to 3 for the dilated convolutional model, its performance is the best. The accuracy begins to decline when the number of layers is greater than 3. Although the difference is not obvious, it also helps improve the prediction accuracy and, simultaneously, reduces the time to train the model.



(a) Comparison of experimental results of models with and without regular learning layers.



(b) Comparison of the prediction results of models with different layers.

Figure 10: Comparison of ablation experiments results.

4. SUMMARY

Traffic accidents are the fundamental cause of casualties and property damage, and they are also a key issue for public health and safety. Prediction of traffic accidents is essential to preventing and reducing the occurrence of traffic accidents [25]. In this study, this work first cleans up the data and constructs and selects features based on the original feature data to ensure a higher correlation between input data and results and more accurate prediction results. Second, a model R-DCN was designed that fused a rule learning model and a dilated convolutional model to predict the occurrence of vehicle collisions. The experimental results show that the prediction results of the model have improved accuracy compared with existing models. On the other hand, this work constructs new features based on the raw features. By calculating the correlation of the overall features, the salient features, i.e., related features, are identified. By experimenting with overall features and correlation features separately [26-29], we can see that the experimental results are greatly improved. Therefore, we can also conclude that using relevant features can help improve the model's predictive performance and reduce the data collection cost. Despite the superiority of the proposed model, it also increases the complexity compared to a single model, which is an issue that needs to be addressed in future work. Furthermore, the next step is to apply the model to other datasets to demonstrate its effectiveness and generalizability.

REFERENCES

[1] Xu Z, Lv Z, Chu B, Sheng Z, Li J. Progress and prospects of future urban health status prediction. Eng Appl Artif Intell. 2024; 129: 107573. DOI: 10.1016/j.engappai.2023.107573

[2] Lv Z, Wang X, Cheng Z, Li J, Li H, Xu Z. A new approach to COVID-19 data mining: A deep spatial–temporal prediction model based on tree structure for traffic revitalization index. Data Knowl Eng. 2023 Jul; 146:102193. DOI: 10.1016/j.datak.2023.102193

[3] Wang X, Liu J, Qiu T, Mu C, Chen C, Zhou P. A real-time collision prediction mechanism with deep learning for intelligent transportation system. IEEE transactions vehic tech. 2020; 69(9): 9497-9508. DOI: 10.1109/TVT.2020.3003933

[4] Fujita Y, Akuzawa K, Sato M. Radar brake system. J sae Review. 1995; 1(16): 113. DOI: 10.1016/0389-4304(95)94949-N

[5] Kononen DW, Flannagan CA, Wang SC. Identification and validation of a logistic regression model for predicting serious injuries associated with motor vehicle crashes. Accid Anal Prev. 2011 Jan;43(1):112-22. DOI: 10.1016/j.aap.2010.07.018

[6] Gerla M, Lee EK, Pau G, Lee U. Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds. IEEE world forum on internet of things (WF-IoT). 2014; 241-246. DOI: 10.1109/WF-IoT.2014.6803166

[7] Huang T, Wang S, Sharma A. Highway crash dectection and risk estimation using deep learning. Accid Anal Prevent. 2020; 135: 105392. DOI: 10.1016/j.aap.2019.105392

[8] Yuan J, Abdel-Aty M, Gong Y, Cai Q. Real-time crash risk prediction using long short-term memory recurrent neural network. Transport research record. 2019; 2673(4): 314-326. DOI: 10.1177%2F0361198119840611

[9] Qiu T, Li B, Qu W, Ahmed E, Wang X. TOSG: A topology optimization scheme with global small world for industrial heterogeneous Internet of Things. IEEE Transactions Industr Infor. 2018; 15(6): 3174-3184. DOI: 10.1109/TII.2018.2872579

[10] Wang X, Fan T, Chen M, Deng B, Wu B, Tremont P. Safety modeling of urban arterials in Shanghai, China. Accident Analysis Prevent. 2015: 83; 57-66. DOI: 10.1016/j.aap.2015.07.004

[11] Xu C, Wang W, Liu P. A genetic programming model for real-time crash prediction on freeways. IEEE Transactions on Intelligent Transportation Systems. 2012; 14(2): 574-586. DOI: 10.1109/TITS.2012.2226240

[12] Lee D, Yeo H. Real-time rear-end collision-warning system using a multilayer perceptron neural network. IEEE Transactions on Intelligent Transportation Systems. 2016; 17(11): 3087-3097. DOI: 10.1109/TITS.2016.2537878

[13] Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: synthetic minority over-sampling technique. J artificial intelligence research. 2002; 16: 321-357. DOI: 10.1613/jair.953

[14] Li J, Cheng K, Wang S, Morstatter F, Trevino RP, Tang J, et al. Feature selection: A data perspective. ACM computing surveys (CSUR). 2017; 50(6): 1-45. DOI: 10.1145/3136625

[15] Benesty J, Chen J, Huang Y. On the importance of the Pearson correlation coefficient in noise reduction. IEEE/ACM Trans. Audio Speech Lang. Process. 2008; 16(4): 757-765. DOI: 10.1109/TASL.2008.919072

[16] Stahl DO. Boundedly rational rule learning in a guessing game. Games Eco Behav. 1996; 16(2): 303-30. DOI: 10.1006/game.1996.0088

[17] Basso F, Basso LJ, Bravo F, Pezoa R. Real-time crash prediction in an urban expressway using disaggregated data. Transp Res Part C Emerg techn. 2018; 86: 202-19. DOI: 10.1016/j.trc.2017.11.014

[18] Cai Q, Abdel-Aty M, Yuan J, Lee J, Wu Y. Real-time crash prediction on expressways using deep generative models. Transp Res Part C Emerg techn. 2020 ; 117 : 102697. DOI: 10.1016/j.trc.2020.102697

[19] Wang L, Abdel-Aty M, Lee J, Shi Q. Analysis of real-time crash risk for expressway ramps using traffic, geometric, trip generation, and socio-demographic predictors. Accid Anal Prev. 2019 Jan;122:378-84. DOI: 10.1016/j.aap.2017.06.003

[20] O'Shea K, Nash R. An introduction to convolutional neural networks. arXiv. 2015; 1511 08458. DOI: 10.48550/arXiv.1511.08458

[21] Li J, Yuan G, Yang Z. Edge-assisted Object Segmentation Using Multimodal Feature Aggregation and Learning. ACM Transactions on Sensor Net. 2023; 20(1): 1-22. DOI: 10.1145/3612922

[22] Sheng Z, Lv Z, Li J, Xu Z, Sun H, Liu X, et al. Taxi travel time prediction based on fusion of traffic condition features. Computers Elect Engin. 2023; 105: 108530. DOI: 10.1016/j.compeleceng.2022.108530

[23] Gu S, Yang L, Li Y, Li H. Multi-label Learning by Exploiting Imbalanced Label Correlations. In Pacific Rim International Conference on Artificial Intelligence. 2021; 585-96. DOI: 10.1007/978-3-030-89363-7_44

[24] Xu Z, Lv Z, Chu B, Li J. Fast autoregressive tensor decomposition for online real-time traffic flow prediction. Knowledge-Based Systems. 2023; 282: 111125. DOI: 10.1016/j.knosys.2023.111125

[25] Lv Z, Li J, Dong C, Li H, Xu Z. Deep learning in the COVID-19 epidemic: A deep model for urban traffic revitalization index. Data Knowl Eng. 202:135:101912. DOI: 10.1016/j.datak.2021.101912

[26] Lv Z, Li J, Xu Z, Wang Y, Li H. (). Parallel computing of spatio-temporal model based on deep reinforcement learning. In International Conference on Wireless Algorithms, Systems, and Applications. Cham: Springer International Publishing. 2021 Jun; 391-403. DOI: 10.1007/978-3-030-85928-2_31

[27] Fenili A. Nonlinear velocity and shape control of a rotating smart flexible beam-like structure. Annals of Applied Sciences. 2022. DOI: 10.55085/aas.2022.669

[28] Li J, Lv Z, Ma Z, Wang X, Xu Z. Optimization of spatial-temporal graph: A taxi demand forecasting model based on spatial-temporal tree. Info Fusion. 2024; 104: 102178.
DOI: 10.1016/j.inffus.2023.102178

[29] Lv Z, Cheng Z, Li J, Xu Z, Yang Z. TreeCN: time series prediction with the tree convolutional network for traffic prediction. IEEE Transactions on Intelligent Transportation Systems. 2023.