Accident Analysis and Prevention 122 (2019) 226–238

Contents lists available at ScienceDirect

Accident Analysis and Prevention

journal homepage: www.elsevier.com/locate/aap

Crash injury severity analysis using a two-layer Stacking framework

Jinjun Tang[a], Jian Liang[a], Chunyang Han[a], Zhibin Li[b,*], Helai Huang[a]

[a] School of Traffic and Transportation Engineering, Smart Transport Key Laboratory of Hunan Province, Central South University, Changsha, 410075, China
[b] School of Transportation, Southeast University, Nanjing, 210096, China

ARTICLE INFO

Keywords:
Crash injury severity
Severity classification
Stacking model
Random Forests
Adaptive Boosting
Gradient Boosting Decision Tree

ABSTRACT

Crash injury severity analysis is useful for traffic management agency to further understand severity of crashes. A two-layer Stacking framework is proposed in this study to predict the crash injury severity: The fist layer integrates advantages of three base classification methods: RF (Random Forests), AdaBoost (Adaptive Boosting), and GBDT (Gradient Boosting Decision Tree); the second layer completes classification of crash injury severity based on a Logistic Regression model. A total of 5538 crashes were recorded at 326 freeway diverge areas. In the model calibration, several parameters including the number of trees in three base classification methods, learning rate, and regularization coefficient are optimized via a systematic grid search approach. In the model validation, the performance of the Stacking model is compared with several traditional models including the Support Vector Machine (SVM), Multi-Layer Perceptron (MLP) and Random Forests (RF) in the multi classification experiments. The prediction results show that Stacking model achieves superior performance evaluated by two indicators: accuracy and recall. Furthermore, all the factors used in severity prediction are classified into different categories according to their influence on the results, and sensitivity analysis of several significant factors is finally implemented to explore the impact of their value variation on the prediction accuracy.

1. Introduction

Crashes vary in the level of injury or property damage. Crash severity is defined as level of injury or property damage due to a crash (AASHTO, 2010). Crash injure severity is an important aspect in assessing safety performance. Over the past few decades, there has been a quantity of research on investigating the relationship between crash severity outcomes and their related risk factors such as the traffic volume, geometrical and environmental characteristic of traffic sites, etc. Methodologically, previous methods for prediction of crash frequency by severity can be classified into two categories. First group is to use separate models. Such method separately models crash frequencies at each severity level by using the univariate models and their variants. However, due to the fact that latent factors are likely to exist across crash rates at different levels of severity, significant correlations have been found between crash and injury counts (Bijleveld, 2005). Such method neglected the interdependence between severity levels. Another group is to use the joint models. This method is able to account for the correlations among crash frequencies at various severity levels. Ma and Kockelman, (2006) proposes a multivariate Poisson regression model, which adds a common error term into the Poisson distributions to account for their correlations. Multivariate Poisson-lognormal regression model (Ma et al., 2008) which is comparable with multivariate Poisson regression model by allowing for the common observed over-dispersion has been commonly used in analyses. Afterward, error terms with Gaussian conditional auto-regressive distribution have been introduced into the multivariate Poisson-lognormal model, in order to account for the spatial correlations (Barua et al., 2014, 2016). Besides, there are a series of other methods have been investigated such as simultaneous equations (Ye et al., 2009, 2013), a joint-probability approach (Pei et al., 2011) and two-stage bivariate/multivariate models (Wang et al., 2011; Xu et al., 2014).

All models mentioned above are based on a generalized linear functional form and certain assumed distributions of crash data. They need a well-defined function form for describing the relationship between crash occurrence and explanatory variables. The specification of the functional from can significantly affect the goodness-of-fit of generalized linear function (GLMs). Most of these models assume that risk factors influence crash frequency in a linear manner. However, researchers have found non-linear relationships exist between crashes and risk factors (Mahalel, 1986; Hauer et al., 1998; Hauer, 1997). Thus, once the assumptions of GLM are violated, biased inferences on effects of the related factors may be introduced.

Compared with the statistical models, which have good theoretical

* Corresponding author.
E-mail address: lizhibin@seu.edu.cn (Z. Li).

https://doi.org/10.1016/j.aap.2018.10.016
Received 20 May 2018; Received in revised form 18 October 2018; Accepted 22 October 2018
Available online 01 November 2018
0001-4575/

interpretability and clear calculation construction, machine learning models (Tang et al., 2018) are more flexible with no or little prior assumptions for input variables and have higher adaptability to processing outliers, missing and noisy data. For example, Artificial Neural Networks (ANN) (Abdelwahab and Abdel-aty, 2001; Delen et al., 2006; Zeng and Huang, 2014), Decision Trees (DT) (Abellán et al., 2013; Oña et al., 2013), Support Vector Machines (SVM) (Dong et al., 2015; Li et al., 2012; Iranitalab and Khattak, 2017), Random Forests (RF) (Das et al., 2009; Harb et al., 2009), and K-means Clustering (KC) (Anderson, 2009; Mauro et al., 2013) are widely applied in crash severity modeling, analysis, prediction and some traffic safety-related researches. Although various machine learning models had been proposed to analyze crash injury severity in the previous studies, each individual model has its own advantages and specific range of application. As a result, it could be helpful to combine several single models to enhance the analyzing performance for the crash injury severity. Stacking, a widely used ensemble method (Menahem et al., 2009; Schaffer, 1994; Deng et al., 2012; Bifet et al., 2010), combines several base classifiers into one meta-classifier. Such integration structure expresses the following advantages: simple structure, gaining higher performance and combining capability of different classifiers. Early studies focusing on the analysis of crash injury severity are mainly based on statistical methods. The advantages of the statistical models are that they express good theoretical interpretability with concrete calculation construction, so that they can exhibit direct and clear explanation to the relationship between accident severity and related variables than machine learning models. However, statistical models have the drawback that most of these models assume that risk factors influence crash frequency in a linear manner which may not be the truth according to several studies (Mahalel, 1986; Hauer et al., 1988; Hauer, 1997). Thus, once the assumptions of statistical models are violated, biased inferences on effects of the related factors may be introduced. The advantage of machine learning methods is that they are more flexible with no or little prior assumptions for input variables which leads to improved predicting performances. In addition, these approaches are more capable of processing outliers, missing and noisy data. The major disadvantage of machine learning methods is that they perform like a 'black box' approach in the analysis and prediction of severity classification and often lack a direct and clear interpretation between accident severity and related variables.

In this study, we design a Stacking framework with two layers to analyze crash injury severity. In the first layer, three base classifier including RF, AdaBoost, and GBDT are integrated to extract advantages of each individual model. In the second layer, a Logistic Regression is trained based on outputs of first layer, and calculate the final classifying results of crash injury severity

The rest of this study is organized as follows. Section 2 introduces the data samples and detailed collection process. Methodology structure and calculation procedure are described in Section 3. Model application and experimental results are provided in Section 4. Section 5 includes conclusions and discussions.

2. Data description

Crash injury severity data were collected in the segments of 326 freeway in the State of Florida, United States. The segment researched in this study includes a deceleration lane and an exit ramp. Furthermore, we define two influencing areas for the freeway segment, which include (1) upstream area, located within 457 m (1500 ft) upstream of the painted nose; (2) downstream area, located within 305 m (1000 ft) downstream of the pained nose. So, the total length of the freeway segment focused in this study is 762 m (2500 ft).

The data collecting time periods lasted three years from 2004 to 2006, and total 5538 crash records were used for injury severity classification and prediction in the selected freeway segments. The crash injury severity was initially categorized into five ordered levels, shown in Table 1. The level 1 indicates no-injury crashes, and corresponding number of samples take up more than half (52.5%) of all crashes records. The level 2 denotes possible or invisible injury crashes which accounts for 26.4% of total crashes. The level 3 represents no-capacitating injury crashes, and the level 4 belongs to incapacitating injury crashes part, which respectively account for 15.1% and 5.1% of all crashes. Finally, the level 5 denotes fatal crashes accident, which has the lowest percentage of 0.9%. As the sample size of the Level 5 is too small, we aggregate the Level 4 and 5 into one level and name as: Highest Injury Severity. In the latter injury severity analysis in Section 4, the experiment of multi-classification with four levels is applied to validate classification accuracy.

Table 1
Division of injury severity levels for the original crash data.

Injury severity level	Description	Frequency	Percent(%)
Level 1	No injury	2902	52.5
Level 2	Possible/invisible injury	1463	26.4
Level 3	No-capacitating injury	837	15.1
Level 4	Incapacitating injury	285	5.1
Level 5	Fatal injury	51	0.9
Overall	–	5538	100.0

Considering the different number and arrangement of exit ramp lanes, the freeway exits were classified into four types which are denoted as type 1, type 2, type 3, and type 4, shown In Fig. 1. Type 1 is defined the exit ramp with a single lane and a taper section. Type 2 shows the exit ramp with a single lane and deceleration section. Type 3 is defined as the exit ramp with two lanes and deceleration section. Type 4 is denoted as the exit ramp with two lanes and a taper section.

In the injury database, various explanatory variables corresponding to each crash accident are also recorded to help analyzing causes. These variables or factors include roadway geometric design characteristics, environment/traffic conditions and the information of crashes. Detailed description and statistical information are expressed in Table 2. As this study mainly focuses on the injury severity classification with explanatory variables, the data collection procedure will not be discussed in depth here. For more details about data sources, please refer to the following studies (Chen et al., 2009; Li et al., 2012).

Note that collision type is represented by four categories where "others" category comprises of a high percentage of crashes. Other collision types in a descending order are hit concrete barrier (5.05%), hit guardrail (2.93%), overturned (2.11%), collision with moveable object on road (1.49%), run into ditch/culvert (0.92%), and other types listed in the report table (11.96%). Note that there are 11.34% of crash records whose types are not listed in the report table, and those data are also classified into the "OTHRES" crash type category in Table 2. In our modeling process, we only consider three major collision types (i.e. rear-end, sideswipe, and angle) as separate categories, because their sample sizes are relatively large. All other collision types are categorized into the "OTHERS" in Table 2 for simplicity.

3. Methodology

A Stacking framework is applied to integrate three methods, Random Forests classifiers, AdaBoost, and GBDT, in the crash injury severity prediction. This section will discuss the Stacking methods and calculation procedure.

3.1. Stacking

Stacking is an ensemble learning technique to combine multiple classification models via a meta-classifier (Wolpert, 1992). Generally, it contains two layers. In the first layer, the individual classification models are trained based on the complete training set. In the second layer, the meta-classifier is fitted based on the outputs of the individual

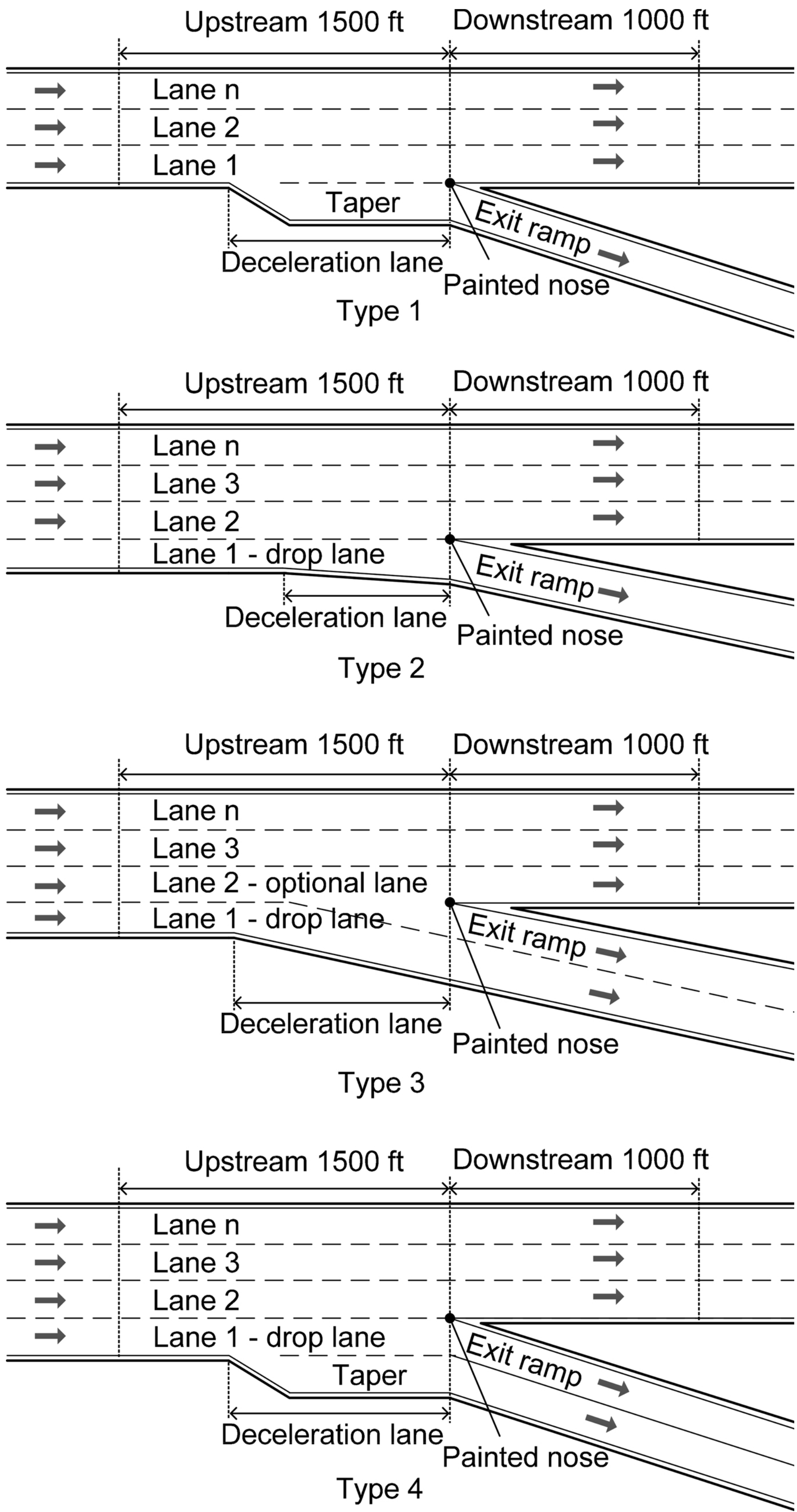


Fig. 1. Four types of freeways segments with different exit ramps.

classification models in the first layer. The output of the meta-classifier is the final result. Fig. 2 shows the structure of the proposed Stacking procedure. In the framework, the advantages of several methods can be integrated to improve classifying performance. Firstly, we divide raw crash data into training and testing dataset, the data are represented as $\{\mathbf{x}=[x_{i1}, x_{i2}, \ldots, x_{in}], y_i\}$, i = 1,2,3…, m and m indicates the number of

Table 2
Descriptions of selected variables for analysis.

Variable	Description	Frequency	Percent
Ramp type	Type 1 Exit Ramp	2387	43.1
	Type 2 Exit Ramp	1848	33.4
	Type 3 Exit Ramp	948	17.1
	Type 4 Exit Ramp	355	6.4
Main lanes	2 Lanes on mainline	535	9.7
	3 Lanes on mainline	1408	25.4
	4 Lanes on mainline	1298	23.4
	5 Lanes on mainline	1533	27.7
	6 Lanes on mainline	765	13.8
Ramp lanes	1 Lanes on exit ramp	4180	75.5
	2 Lanes on exit ramp	1308	23.6
	3 Lanes on exit ramp	51	0.9
DeLength	Length of deceleration lanes(mile)	Continuous	
RaLength	Length of entire ramps(mile)	Continuous	
SurfacType	1 Blacktop surface	4390	79.3
	0 others	1148	20.7
ShoulderType	1 Paved shoulder	3840	69.3
	0 no Paved shoulder	1698	30.7
ShoulderWidth	Right shoulder width(ft)	Continuous	
MainSpeed	Post speed limit on mainline(mph)	Continuous	
SpeedDiff	Difference of speed limit between mainline and exit ramps(mph)	Continuous	
Light	1 Daylight (including dusk and dawn)	3785	68.3
	0 No daylight	1753	31.7
Weather	1 Clear weather condition	3647	65.8
	0 Others	1891	34.2
Surface	1 Wet surface condition	1148	20.7
	0 Dry surface condition	4390	79.3
LandType	1 Business surroundings	3185	57.5
	0 Residential surroundings	2353	42.5
MainADT	Mainline ADT per year in thousand	Continuous	
RampADT	Exit ramp ADT per year in thousand	Continuous	
AlcDrug	1 Alcohol/drug involved	223	0.4
	0 No alcohol/drug involved	5315	99.6
Crash Type	Rear end crash	2347	42.4
	Sideswipe crash	760	13.7
	Angle crash	450	8.1
	Others	1981	35.8

data samples, n is the number of features or variables adopted in the study, and the values of y (y=L1,L2,L3 or L4) correspond to different injury severity levels. Secondly, the training samples are set as input to the tree base classifying methods in the first layer, and three parallel classifying results, probability of a single sample belongs to different severity levels, are calculated, which are represented as p_{i1}, p_{i2} and p_{i3} respectively, and detailed calculation process of these three base methods will introduced from Section 3.2.1 to 3.2.3. Thirdly, a Logistic Regression model is designed to fuse the classifying results from the first layer through establishing a sigmoid function to minimize the loss function by using the gradient decent algorithm, and the detailed information of Logistic Regression will be introduced in Section 3.2.4. Finally, the final classifying results can be calculated by using testing dataset base on the optimal classifying models, and the evaluation indicators are established to estimate the prediction performance of the proposed method.

3.2. Selection of base classifiers in Stacking

Stacking is a representation learning method. The raw data, with messy and irregular features, will be processed through multiple classification models in the Stacking, and valid features can be extracted. The learning ability of Stacking comes mainly from the representation of the features, which is consistent with the structure of neural networks. The first layer in Stacking can be equivalent to the first n-1 layer in neural network, while the second layer in Stacking can be analogized to the last output layer in a neural network.

The first layer in Stacking can be considered as a highly complex non-linear feature converter. Different classifiers in Stacking represent heterogeneity for different features. In order to effectively extract features from the raw data, the base classifiers in the first layer need meet two requirements: (1) High diversity and (2) High accuracy. In this study, Random Forest, AdaBoost and GBDT are selected as the base classifiers in the first layer. The three models all accomplish learning tasks by combining multiple learners, but their modeling ideas are completely different. Because of their similarities and differences, and they all achieved high performance in cross-validation, we finally chose these three models combined in the first layer of the Stacking model.

For the second layer in the Stacking model, as features are extracted based on complex non-linear transformations, it is unnecessary to choose complex classifiers in the output layer. Logistic regression is a good candidate with simple structure and further additional advantages: with L_2 regularization Logistic regression can further prevent over-fitting.

3.2.1. Random forests

Random Forests (RF), proposed by Breiman (2001), used bootstrap sampling method to change the training set in order to build an ensemble of decision trees. In this study, the input samples for RF are represented as $\{\mathbf{x}=[x_{i1}, x_{i2}, \ldots, x_{in}], y_i\}$, i = 1,2,3…, m and m indicates the number of data samples, n is the number of features or variables

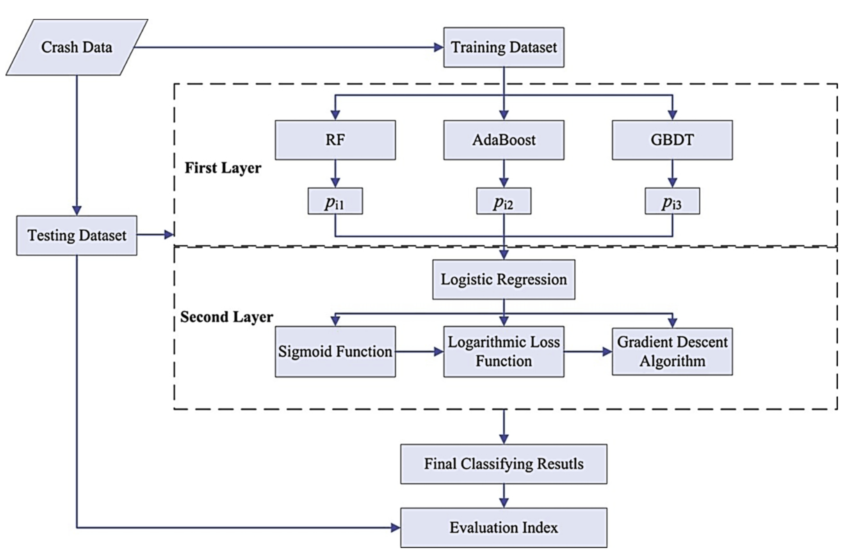


Fig. 2. Calculation structure of Stacking framework.

adopted in the study, the values of y (y=L1,L2,L3 or L4) correspond to different injury severity levels, and the output is the probability of a single sample belongs to different severity levels. It includes three basic calculation processes: sample set selection, decision tree generation and decision tree combination. The classification result of the new data depends on how many votes the decision tree makes. The main process of RF algorithm includes following several parts:

(1) Select the sample set

RF creates new training sets by re-sampling approach from the original data set n times, and the n indicates the number of samples in the original training set, randomly with replacement. This means the sample just being chosen will not be removed from the data set in the next extraction. Hence, some of the training samples will be chosen more than once while some others will not be chosen at all in a new set. After a total of K rounds of extraction, K new sample sets are obtained: $D_1, D_2, \ldots, D_K$. The K sample sets that are independently sampled and have the same distribution will be used to generate K decision trees.

(2) Generate decision tree

Assuming there are M features in the feature space, during each round of generating the decision tree, m features are ($m < M$) randomly selected from the feature space to form a new feature set and generate a decision tree. After K rounds calculation, K decision trees are generated. Since these trees are randomly chosen from training set and feature space, they are independent from each other.

(3) Combination of decision tree models

As the decision trees generated in this study are independent, the significance of each decision tree can be considered equal to the final results. Accordingly, the same weight of each decision tree is considered in the combination of RF. Furthermore, the final classification results are determined by the votes from all the decision trees. Fig. 3 expresses detailed calculation procedure in the RF algorithm.

3.2.2. AdaBoost

AdaBoost was proposed by Freund and Schapire (1996). AdaBoost is an iterative algorithm whose core idea is to train different weak learners (h_t) for the same training set and then group these weak learners to construct a stronger learner (H). AdaBoost has the following features: (1) Each iteration changes the distribution of the sample set, not repeat sampling; (2) The change in the sample set distribution depends on whether the sample is correctly classified. Each sample in training set D has a weight, represented by the vector $\boldsymbol{\beta}$. Samples that are often classified correctly have a low weight, and samples that are often misclassified have a high weight; (3) Each weak learner has a weight, represented by the vector. The input samples are same as RF represented as $\{\mathbf{x} = [x_{i1}, x_{i2}, \ldots, x_{in}], y_i\}$, i = 1,2,3…, m, and the output is the probability of a single sample belongs to different severity levels.The detailed calculation process of Adaboost algorithm contains following several steps described in Algorithm 1.

Algorithm 1: Adaboost algorithm

Inputs: Data set D = $\{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m),\}$;

Base learning algorithm £;

Weak learner (h_t);

Number of learning rounds K.

Process:

1. $\boldsymbol{\beta}_1(i) = \frac{1}{m}$. % Initialize the weight distribution
2. **for** k = 1, …, K:
3. $h_k = £(\mathrm{D}, D_k)$; %Train a learner h_k $from\ D\ using\ distribution\ D_k$
4. $\varepsilon_k = \frac{Number\ of\ samples\ incorrectly\ classified\ by\ h_k}{The\ number\ of\ all\ samples}$; %Measure the error
5. $\boldsymbol{\gamma}_k = \frac{1}{2}\ln(\frac{1-\varepsilon_k}{\varepsilon_k})$; $\%Determine\ the\ weight\ of\ h_k$
6. $\boldsymbol{\beta}_{k+1}(i) = \frac{D_t(i)}{Z_k} \times \begin{cases} \exp(-\alpha_k)\ if\ h_k(x_i) = y_i \\ \exp(\alpha_k)\ if\ h_k(x_i) \neq y_i \end{cases}$

$= \frac{D_k(\mathrm{i})\exp(-\alpha_k y_i h_k(x_i))}{Z_k}$ $\%Update\ the\ distribution$

$\%Z_k\ is\ a\ normalization\ \ factor\ which$ enable $D_{k+1}\ to\ be\ a\ distribution$

7. **end**

Output: $\mathrm{H}(x) = \mathrm{sign}(\sum_{k=1}^{K} \boldsymbol{\gamma}_k h_k(x))$ $\%sign(x) = \begin{cases} 1, x > 0 \\ 0, x = 0 \\ -1, x < 0 \end{cases}$

3.2.3. GBDT

GBDT was proposed by Friedman (2001). Like AdaBoost, GBDT repeats the selection of an ordinary model and adjusts it based on the performance of the previous model. The difference is that AdaBoost improves the model by increasing the weight of the misclassified samples and GBDT uses the gradient boosting to improve model performance. Gradient Boosting is a widely used method to optimize parameter and improve model performance, and its main idea is to update parameters in the gradient descent direction of the loss function of the model built in the previous time. Loss function is used to evaluate the performance of a model. Generally, the smaller the loss function, the better the performance. While the loss function continues to decline, the model can be continuously modified to improve performance. The input samples are represented as $\{\mathbf{x}=[x_{i1}, x_{i2}, \ldots, x_{in}], y_i\}$, i = 1,2,3…, m, and same as RF and AdaBoost, the output is the probability of a single sample belongs to different severity levels.

Since Gradient Boost is a framework that can contain many different algorithms, a method based on a boosting decision tree is called GBDT (Gradient Boosting Decision Tree). The boosting decision tree makes joint decisions by iterating multiple regression trees. Each regression tree learns the residuals of all previous trees and fits a current residual regression tree. The final result is the accumulation of all regression trees generated throughout the iteration. The GBDT uses the negative gradient of the loss function in the current model as an approximation of the residual in the boosting decision tree. The following steps describe GBDT algorithm in detail.

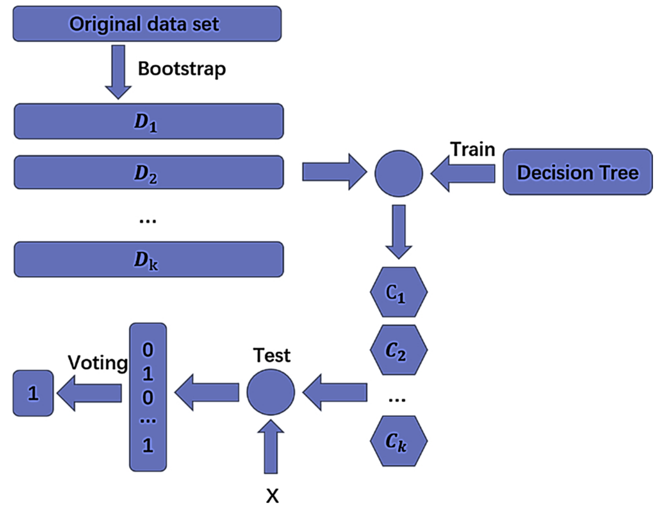


Fig. 3. Flowchart of RF in sample selection, decision tree generation and combination.

Algorithm 2: GBDT algorithm

Inputs: Data set D = $\{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m),\}$;

$Iterations\ rounds$: K;

Loss function : L(y, f) ($default$: $logistic\ funciton$);

Weak leanrer : h(x, θ) ($default$: $regression\ tree$);

Process:

1. Initialize $f_0(x) = argmin_\gamma \sum_{i=1}^{m} L(y_i, \gamma)$.

2. **for** $k = 1, \ldots, K$:

3. $Compute\ the\ negative\ gradient\ g_k(x)$;

4. Construct a new tree model $h(x, \theta_k)$;

5. Find the best gradient decent step p_k;

6. $p_k = \arg min_p \sum_{i=1}^{N} L[y_i, f_{t-1}(x_i) + ph(x_i, \theta_k)]$;

7. Update the function estimate:

$$f_k \leftarrow f_{k-1} + p_k h(x, \theta_k)$$

8. **end**

Output: $Y = \text{sign}(f_k)\quad \%sign(x) = \begin{cases} 1, x > 0 \\ 0, x = 0 \\ -1, x < 0 \end{cases}$

3.2.4. Logistic Regression(LR)

Logistic Regression is a linear model for classification, as shown in Fig. 2, it integrates the classifying results of three base methods in first layer through optimizing parameters in LR. The details of calculation process of LR in the second layer are summaries as follows:

Step.1 Calculate the probability of a single sample belongs to different severity levels, p_{ij}, $i = 1,2,3$, $j = 1,2,\ldots m$, m indicates the number of samples, and i is the index of three base classifying methods. Defining y as the different severity levels (y = L1, L2, L3, L4), for a single sample, the new dataset is defined as $\{(p_{1j}, p_{2j}, p_{3j}), yj\}$.

Step.2 In order to finish binary classification prediction in LR, different levels or classifications are firstly combined into 6 new binary classifiers, and y is updated as the combination of two different levels, y = {(L1,L2), (L1,L3), (L1,L4), (L2,L3), (L2,L4), (L3,L4)}.

Step.3 Define a sigmoid function to describe the possible outcomes of a single trial modeled:

$$h_\theta(p) = \frac{1}{1+e^{-z}} = \frac{1}{1+e^{-\theta^T p}} \tag{1}$$

where θ is the weight vector used to describe the contribution of base classifying method to the final result, and $z = \theta_0 + \theta_1 p_{1j} + \theta_2 p_{2j} + \theta_3 p_{3j}$ is defined as the input of sigmoid function.

Step.4 In the training process of LR, the logarithmic loss function is defined to measure the prediction of the model:

$$cost(h_\theta(p), y) = \frac{1}{m}\sum_{i=1}^{m}(y_i \log(h_\theta(p_i)) + (1 - y_i)\log(1 - h_\theta(p_i))) + \frac{1}{C}\sum_{i=1}^{m}\theta^T\theta \tag{2}$$

where m is the total number of samples, and C is a regularization coefficient to prevent overfitting. Furthermore, the gradient descent algorithm is applied to update the weights to minimize the loss function.

$$\theta := \theta - \alpha\frac{\partial cost(h_\theta(p), y)}{\partial\theta} \tag{3}$$

where α is the learning rate.

Step.5 Obtain the optimal values of θ, then calculate the value of sigmoid function $h_\theta(p)$ corresponding to 6 binary classifiers, and select the classifier with highest value of probability according to $\arg\max_{c} h_{\theta,c}(p)$ (c = 1,2,3…6).

Step.6 According to the probability of one selected binary classifier, if its value is larger than 0.5, then divide the samples into the first level in binary classification, otherwise, divide them into the second level.

3.3. Calculation procedure of stacking framework

As aforementioned, the first layer of Stacking is a feature extraction process. In order to extract features efficiently, we combine three base classifiers: RF, Adaboost and GBDT. Furthermore, to avoid over-fitting in model training, and a cross-validation method is adopted in this study. The raw data used in application is classified into training set and testing set. They were randomly separated with a ratio of 7:3, and a five-fold cross validation is applied to overcome over-fitting problem. Algorithm 3 shows pseudo codes of Stacking method proposed in this study.

Algorithm 3: Stacking framework

First layer:

Inputs: Training set $X1$, $Testing\ set\ X2$;
Three classifiers: $\{RF,\ AdaBoost,\ GBDT\}$;
CLF: Classifier in three classifiers $\{RF,\ AdaBoost,\ GBDT\}$;

Process:

1. **for** CLF in [RF, AdaBoost, GBDT]:
2. **for** $fold = 1, \dots, 5$:
3. 4/5 of X1 was used to train CLF;
4. Using the trained CLF to predict the remaining 1/5 of X1 to get P[fold];
5. Using the trained CLF to predict X2 to get T[fold];
6. **end**
7. $Stack\ P[1], \dots, P[5]\ to\ get\ P\_CLF(same\ shape\ with\ X1)$;
8. Average T[1], …, T[5] $according\ to\ the\ row\ to\ get\ T\_CLF(same\ shape\ with\ X2)$;
9. **end**

Second layer:

Inputs: $P = [P_{RF}, P_{AdaBoost}, P_{GBDT}]$, $T = [T_{RF}, T_{AdaBoost}, T_{GBDT}]$
Classifier：Logistic regression

Process: Using P as training set to train logistic regression

Output: Use trained logistic regression to predict T_{pred} and compare T_{pred} with T to evaluate the prediction.

4. Experimental results and discussion

The proposed Stacking model was mainly implemented in Python using scikit-learn (0.19.1) developed by Google. The crash injury severity includes four ordered levels ranking from Level 1 to Level 4. Level 1 represents no-injury crashes. Level 2–4 corresponds to the severity of the injury from low to high. In the original dataset, there are only a few samples belong to Level 4 and 5. This is reasonable because of the low probability of a major accident occurring in real life. So, we combine these two levels into one, and name as the highest injury severity level. In this phase, we evaluate prediction performance of multi-classification: four injury classes.

4.1. Performance evaluation

4.1.1. Accuracy

Accuracy is one of the most commonly used indexes to estimate prediction performance in classification tasks. For a specific sample set D, classification accuracy can be calculated using following equation.

$$Accuracy = \frac{the\ number\ of\ correct\ predictions}{N} \tag{4}$$

Table 3
Division of Recall in binary classification experiment.

True situation	Forecast results	
	Injury	Non-Injury
Injury	TP(true positive)	FN(false negative)
No-Injury	FP(false positive)	TN(true negative)

where N is the number of samples in D.

4.1.2. Recall

For binary classification, the results can be classified into four categories based on a combination of real categories and forecasting categories: the true positive (TP) case, the false positive (FP) case, the true negative (TN) case and the false negative (FN) case, which indicates the number of corresponding samples, and the confusion matrix of binary classification results is shown in Table 3. The confusion matrix is also called as error matrix in machine learning field. It is a specific table used to evaluate accuracy of machine learning models. Generally, the row and column of the matrix respectively represent the prediction results of classification and actual classification. Using the confusion matrix, we can estimate the prediction performance based on following four cases: the true positive (TP), the false positive (FP), the true negative (TN) and the false negative (FN) in Table 3.

Accordingly, the Recall is defined by the following equation.

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

Accuracy is a comprehensive indicator to reflect how many of all samples are predicted correctly. However, in the crash injury severity analysis, what we really concerned is how many injury samples were predicted correctly. Comparing to the situation that No-Injury sample is classified into Injury part, if the Injury sample is classified into No-Injury category, it will lead to more serious consequences: ignoring the impact of serious accidents. It is expected in this study to design a Stacking classifying method to obtain higher prediction accuracy in injury sample set, and this is the reason why we chose Recall as a measure of performance.

Similarly, for the multi-classification experiment, the Recall is defined as:

$$Recall = \frac{\sum_{i=2}^{5} S_i}{N_s} \tag{6}$$

where S_i represents the number of samples predicted correctly in injury severity Level 2,3,4, N_s means the total number of injury samples.

4.2. Parameters optimization

Stacking is an integrated model with several important parameters needs to be optimized. In this process, a systematic grid search in sciktilearn is implemented. Although system grid search requires high computational costs, better results can be obtained by systematically tuning the parameter values.

4.2.1. Parameter optimization in the first layer

Three classifying algorithm including RF, AdaBoost and GBDT are integrated in the first layer of the Stacking model. As mentioned before,

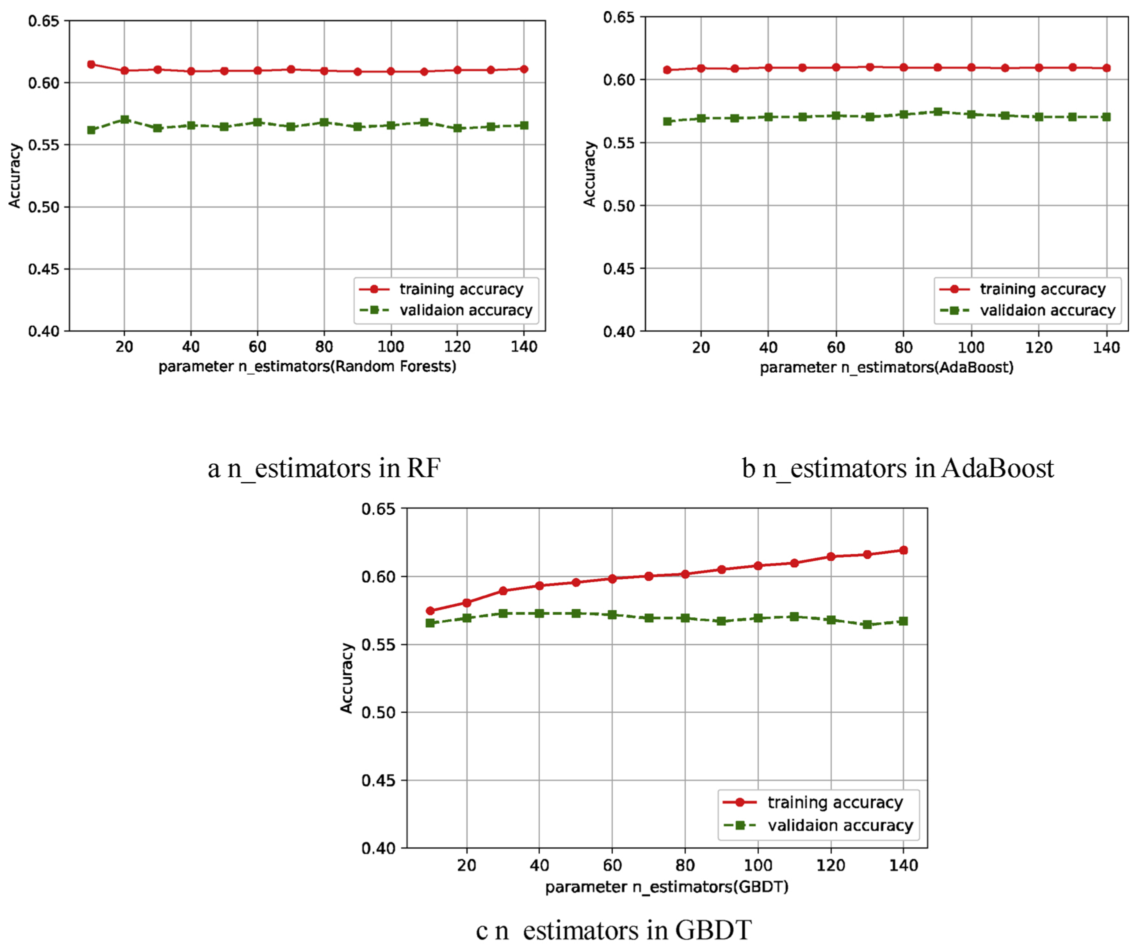


Fig. 4. The effect of n_estimators to prediction accuracy of three base classifying algorithms.

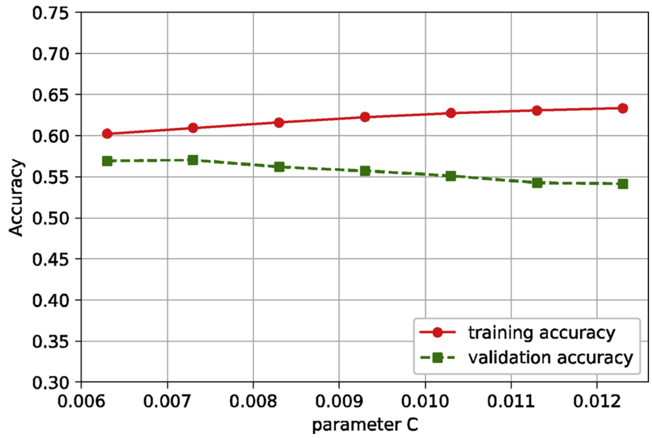


Fig. 5. Accuracy variations with different value of *C* in second layer.

these three models are ensemble learning methods that combine multiple weak learners to achieve significantly better generalization performance than a single learner. All three models have a common hyperparameter: the number of combined weak learners (n_estimators). The increase of n_estimators will affect the accuracy of three base models RF, Adaboost and GBDT. Before implementing the Stacking method, the parameter n_estimators needs to be optimized. Fig. 4a, b and c show the effect of the number of weak classifiers in the multi-classification on the prediction accuracy of three base models. From the observation of the figures, as the number of n_estimators increases, the training accuracy remains unchanged, and the validation accuracy decreases slightly, which shows that too large value of n_estimators may lead to over-fitting problem. The values of n_estimators are set as 20, 90 and 110 respectively according to the highest verification accuracy.

4.2.2. Parameter optimization in the second layer

The second level of the Stacking model is the logistic regression classifier, in which the regularization coefficient *C* is an important parameter to avoid over-fitting, and the smaller values contribute to stronger regularization. In Fig. 5, we express the effect of *C* on the prediction accuracy to the Stacking model with variation from 0.006 to 0.013. With the increase of *C*, the validation accuracy decreases smoothly. The optimal value of C is set as 0.0073 when the validation accuracy achieves the highest value.

4.2.3. Optimal parameters for the stacking model

According to the discussion before, the parameters of the model with the highest validation accuracy are identified as the optimal parameters. The Table 4 summaries all the optimal values of parameters in Stacking model for multi-classification.

Table 4
Optimal values of parameters in Stacking model.

Parameter	Values	Description
n_estimators	20	the number of trees in the RF.
	90	The number of weak learners in AdaBoost.
	110	The number of weak learners in GBDT.
Learning-rate	0.09	The learning rate in AdaBoost
	0.06	The learning rate in GBDT
max_depth	10	The maximum depth of the tree in RF.
	4	The maximum depth of the tree in AdaBoost.
	11	The maximum depth of the tree in GBDT.
class_weight	“Weighted”	Weights associated with classes in LR
C	0.0073	Regularization coefficient in LR

4.3. Comparison of prediction performance

In the application, the proposed Stacking model is compared with several traditional classification methods: SVM model with Radical Basis Function (RBF) kernel, the MLP model and the RF model. In order to ensure a fair comparison, all the models are trained based on the same training set and tested on the same validating set. In multi-classification, the raw crash samples were also randomly separated into a training set and a validation set with a ratio of 7:3. The prediction results of different models for the multi-classification are shown in Table 5. It is found that the performance of Stacking strategy on the validation set is superior to other models according to accuracy and recall. The overall classification accuracy and recall of proposed model based on the validation set are 57.69% and 20.35%, and these are the highest values comparing with other four traditional models.

4.4. Estimation of significant features and sensitivity analysis

The first layer of the Stacking model is a hybrid structure, and it is difficult to directly identify the contribution of each factor in raw data to the prediction results of crash injury severity. In order to evaluate the contribution of different factors or features to the crash injury severity analysis, we extract the weights corresponding to each feature in three base classifying models in the first layer, see Fig. 6. Fig. 6a, b and c show the contribution of each factor in the multi-classification case, and Fig. 6d expresses the average weights of three base models. The weights in three base classifying models have the same effect on the final classification results.

In order to select the significant factors having important effect to final results, all the factors are classified into five categories from high to low according to the average weights, which is shown in Table 6. The Category 1 means the factors have the highest effect to final results, and from Category 2 to 5, the effects gradually decreases. Thus, the higher the weights of the variables are, the greater the effect of factors on the result. Through observation from table, it is found that DeLength, RaLength, MainADT, and RampADT in Category 1 take high contribution to final results. The factors including Ramp Type, Ramp Lanes, Land Type, and Shoulder Type in Category 5 only have weak effects to final classifying results. Furthermore, it can be found that different factors are divided into Category 2, 3 and 4.

Aiming to the four significant factors estimated in Table 6, we further analyze their changes to the sensitivity of the classifying results. We increase a selected factor by one unit at a time with other factors remain unchanged, and the unit here is defined as one-eighth of the average value of all samples belong to this factor, in order to obtain observations under a smaller changing scale. This sensitivity analysis is only executed on the validation sample set. The results of sensitivity analysis of four main factors in Category 1 are shown in Fig. 7. The x-axis represents the number of increasing units, and the y-axis indicates the change of validation accuracy according to different units. From the observation of figures, we can find several results. (1) For the factor RaLength, it has lowest average weights in multi classification case (see Fig. 6d), and the accuracy sensitivity expresses a variation pattern: increase to maximum and then decreases. The maximum of validation changes achieves at 7units in the multi-classification (see Fig. 7a). (2) For the factor DeLength, it has lower average weights (see Fig. 6d), and the accuracy variation expresses patterns as: firstly increase then decreases in the multi-classification (see Fig. 7b). (3) For the factor MainADT, it has the highest average weights in classification results (see Fig. 6d), and the accuracy shows a variation pattern as: increase to maximum and then decreases (see Fig. 7c). (4) For the factor RampADT, it has lower average weights similar with DeLength (see Fig. 6d), and the accuracy variation expresses patterns as: increase with the growth of units in the multi-classification (see Fig. 7d).

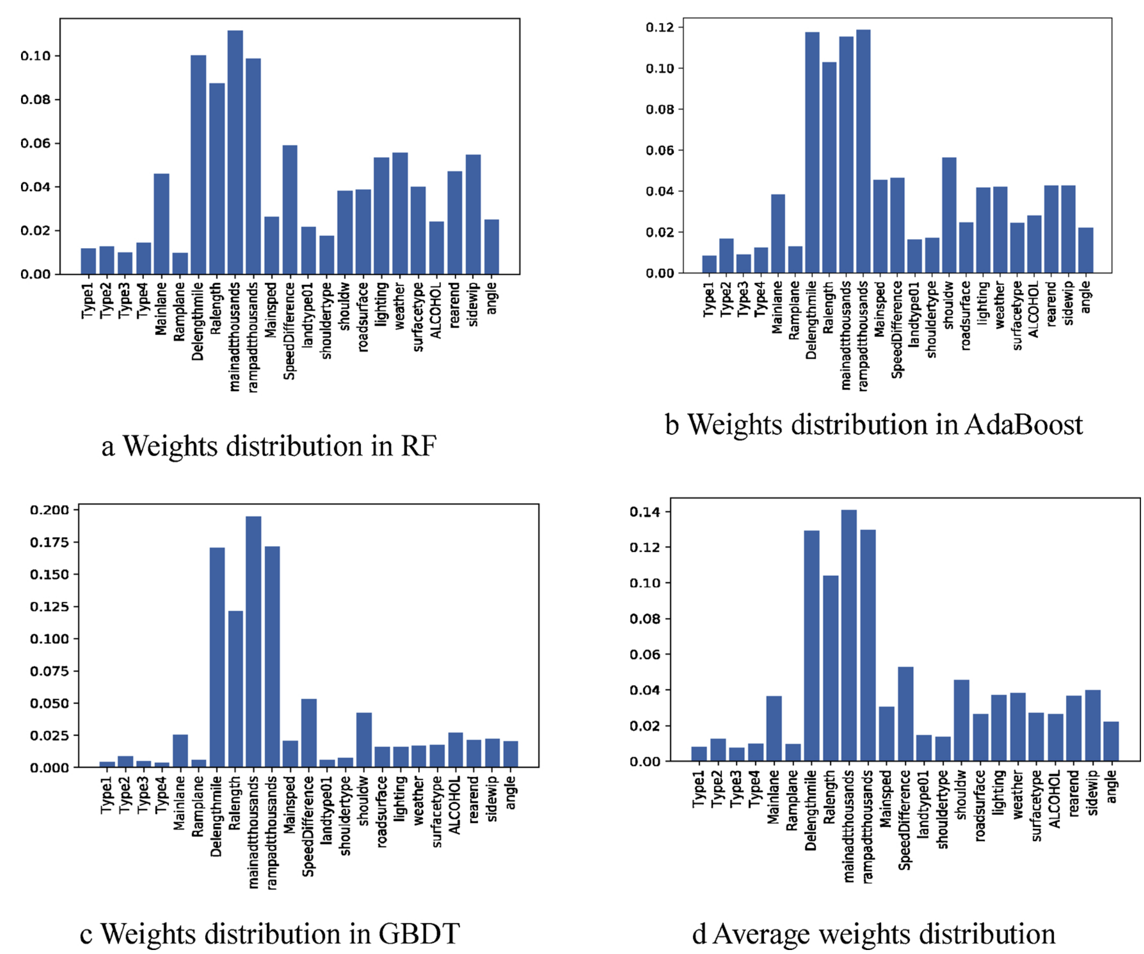
In recent years, a number of studies have applied the sensitivity analysis procedure to make an inference about the relationship between

Table 5
Comparison of prediction performance of different models.

	Injury severity	Training		Validation	
		Accuracy (%)	Recall (%)	Accuracy (%)	Recall (%)
SVM (rbf)	No Injury	94.55		90.18	
	Invisible injury	53.24		25.87	
	No-capacitating injury	30.46		6.33	
	Highest Injury Severity	17.56		3.87	
	Overall	68.71	31.17	55.01	15.69
MLP	No Injury	89.25		88.80	
	Invisible injury	18.89		18.24	
	No-capacitating injury	6.93		6.21	
	Highest Injury Severity	1.78		2.08	
	Overall	52.71	17.97	52.22	16.51
Random Forests	No Injury	95.35		91.80	
	Invisible injury	43.97		27.65	
	No-capacitating injury	18.28		1.67	
	Highest Injury Severity	8.03		0.89	
	Overall	64.35	30.79	55.42	17.23
Stacking Model	No Injury	95.66		88.73	
	Invisible injury	54.31		39.20	
	No-capacitating injury	31.18		6.57	
	Highest Injury Severity	18.45		4.46	
	Overall	69.72	41.06	57.69	20.35

the dependent and independent variables (see Zhang and Xie, 2007; Li et al., 2012; Chen et al., 2016). Sensitivity analyses are usually conducted through a data perturbation and before-after comparison, and the contribution of explanatory variables on the probabilities of injury severities are quantified. Through the sensitivity analysis, the quantitative impact of the input variables on crash severity can be identified.

To make it clearer, we gave an example illustrating how the prediction results by machine learning model can be used to improve safety at freeway diverge locations. The results are shown in Table 7. In



a Weights distribution in RF

b Weights distribution in AdaBoost

c Weights distribution in GBDT

d Average weights distribution

Fig. 6. Weights variation of different factors.

Table 6
Contribution of each factor in the binary classification and the multi-classification.

Category 1	Category 2	Category 3	Category 4	Category 5
DeLength	Speed Diff	Light	Crash Type	Ramp Type
RaLength	Shoulder Width	Main Lanes	AlcDrug	Ramp lanes
MainADT		Weather	Main Speed	Land Type
RampADT			Surface	Shoulder Type
			Surface Type	

the experiment, for main factors including RaLength (Length of entire ramps), DeLength (Length of deceleration lanes), MainADT (Mainline ADT per year) and RampADT (Exit ramp ADT per year) are considered to analyze their influence on the crash severity. Being consistent with the previous analysis of sensitivity, we also increase each selected factor by one unit at a time with other factors remain unchanged. To make a clear description about the impact of prediction results on crash severity, we combine three injury categories (Invisible injury, No-capacitating injury, Highest Injury Severity) into one class (Injury). The modeling results are briefly interpreted as follows:

For the factor RaLength, with the increase of its value, the percentage of class “No Injury” increases around from 83% to 90%, while the percentage of class “Injury” decreases from about 16% to 9%. This means that the severity of the accident declines, especially when the value of RaLength increases by 10 units, the prediction results become stable, see the bold part in the table. This result shows that setting a reasonable length of entire ramp can improve the adaptability of vehicles when merging into the traffic in main lanes, and finally enhance driving safety.

For the DeLength, with the increase of its value, the percentage of class “No Injury” increases approximately from 80% to 83%, while the percentage of class “Injury” decreases from about 19% to 16%. This indicates that setting a reasonable length of deceleration lane, when the DeLength increases by 6 units show in the table, can provide drivers with enough space to help them reduce speed, adapt to the driving environment on the exit ramp, and finally reduce the severity of the accident.

For the MainADT, with the increase of its value, the percentage of class “No Injury” increases approximately from 83% to 90%, while the percentage of class “Injury” decreases from about 16% to 9%. This indicates that reasonable control to traffic volume on the main road will be helpful to reduce the number of serious or injury accidents. The reason is that, with the increase of traffic volume, the speed of vehicles is reduced, hence results in lower severity of accidents.

For the RampADT, with the increase of its value, the distribution of percentage expresses different patterns compared with other three factors, the percentage of class “No Injury” decreases approximately from 80% to 69%, while the percentage of class “Injury” increases from about 19% to 30%. This shows that the increase of traffic volume on the exit ramp will seriously interfere with running state of vehicles in the main road lane. As the fast speed of vehicles in the main road, this interference will results in injury accidents. Therefore, through effectively guiding vehicles to leave mine road at previous or latter exit ramps, the traffic volume on specific exit ramp can be reduced, and finally results in the reduce of injury accident number.

The above sensitivity analysis suggests that the machine learning models can provide critical support for the implementation of safety improvement measurements on freeways as well as give quantitative evaluation of the effectiveness.

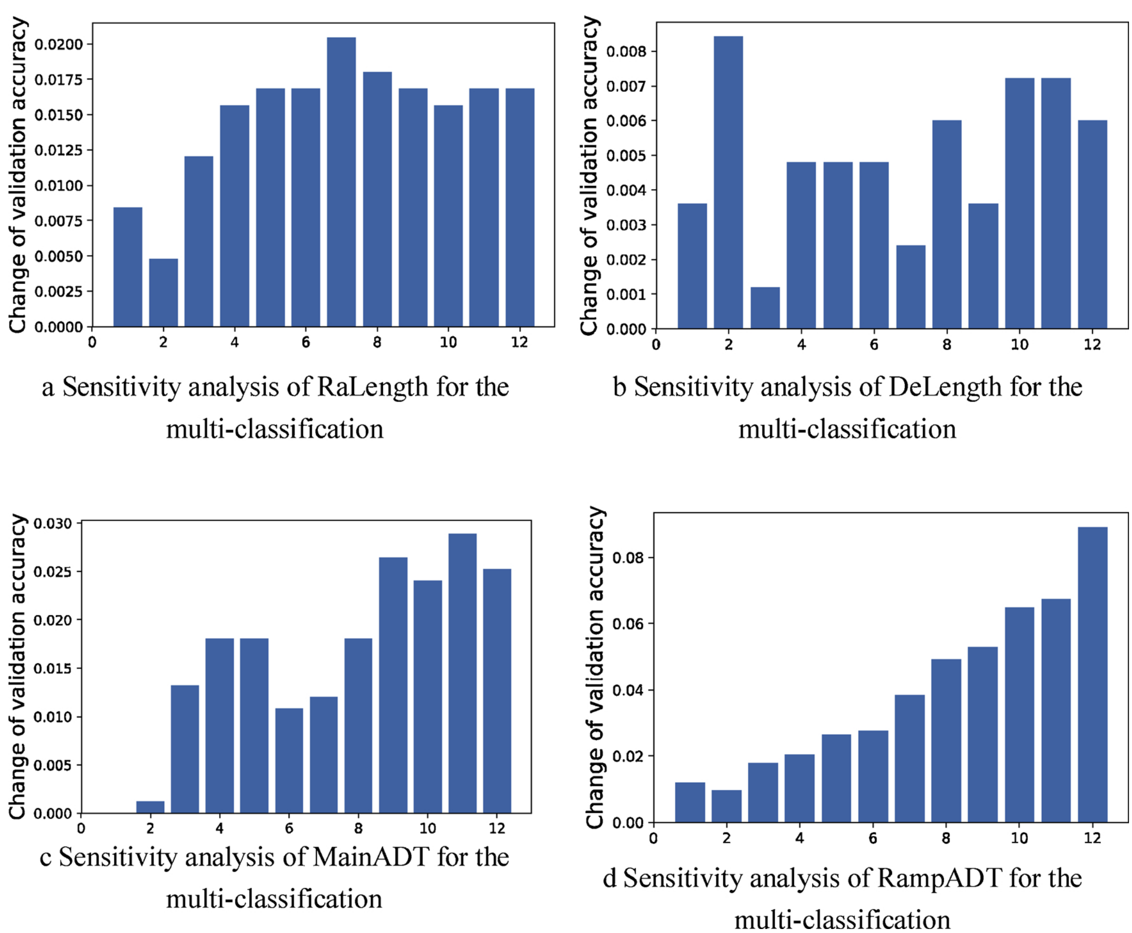


Fig. 7. Variation of validation accuracy with the changes of four main factors for the binary and multi classification.

Table 7
Percentage of predicted crash severity in sensitivity analysis for four factors.

Number of increasing unit	Factors							
	RaLength		DeLength		MainADT		RampADT	
	No Injury	Injury	No Injury	Injury	No Injury	Injury	No Injury	Injury
1	83.51	16.49	80.29	19.71	83.03	16.97	80.99	19.01
2	85.32	14.68	81.23	18.77	83.39	16.61	81.47	18.53
3	85.92	14.08	82.31	17.69	85.68	14.32	81.11	18.89
4	86.52	13.48	82.79	17.21	83.39	16.61	78.46	21.54
5	88.09	11.91	82.67	17.33	84.12	15.88	78.34	21.66
6	87.85	12.15	**83.03**	**16.97**	84.60	15.40	77.14	22.86
7	89.05	10.95	83.03	16.97	84.84	15.16	76.77	23.22
8	89.77	10.23	83.39	16.61	87.48	12.52	73.29	26.71
9	89.17	10.83	83.63	16.37	87.97	12.03	71.12	28.88
10	**90.13**	**9.87**	83.27	16.73	**90.25**	**9.75**	**69.31**	**30.68**
11	90.25	9.75	84.12	15.88	90.37	9.63	69.55	30.44
12	90.85	9.15	83.63	16.37	89.81	10.19	69.13	30.87

5. Conclusions

In this paper, a stacking framework combining RF, Adaboost, GBDT and Logistic Regression model was developed and employed to predict the crash injury severity based on crash data collected at 326 freeway diverge areas. The total data samples were classified into two parts: training and testing dataset. Considering the specialty of crash injury severity, we used accuracy and recall as evaluation indicators. A multi-classification experiment is applied in model validation and comparison. Compared with SVM, MLP and RF, the validation accuracy and recall for the Stacking model in the multi-classification case were 57.69% and 20.35%, which were higher than those in the SVM model (55.01% & 15.69%), the MLP model (52.22% & 16.51%) and the Random Forest model (55.42% & 17.23%). In addition, by examining the importance of factors in the three base classifiers, we analyzed the impact of factors on the crash injury severity.

The contribution of this study and its significance for improving traffic safety can be summarized from following three aspects: (1) Use a Stacking framework combing three base classifying methods, RF, AdaBoost and GBDT, to achieve an improved prediction accuracy of crash injury severity. This result will be definitely helpful to understand the relationship between the crash injury severity and different factors. It is also beneficial to the traffic management department to predict the possibility or severity of accidents by integrating the road infrastructure information, traffic managing and controlling information, weather information, and other information from drivers to provide active traffic management strategy for safety. (2) Use weights distribution to analyze the influence of different explanatory variables or factors on the prediction of crash injury severity. Furthermore, all the variables are categorized into different influencing levels according to values of weights. This result can help to provide effective suggestion from aspects of traffic management, infrastructure construction, and driver's behavior for enhancing driving safety and reducing the possibility of accidents. (3) Analyze the sensitivity of several key factors, such as MainADT (Mainline ADT), RampADT (Exit ramp ADT), DeLength (Length of deceleration lanes) and RaLength (Length of entire ramps). This result can help to design managing and controlling measures to limit traffic volume in certain time periods for the freeway diverge area based on the conclusion that the traffic volume has a significant influence on the crash injury severity. Furthermore, optimizing the length of the deceleration lanes and entire ramps will also be useful to reduce the severity of accidents.

However, due to the complexity of traffic accident, our study has some limitations. First of all, because factor selection is a complete process, although there are some experiences reported in previous reference, we still need to make further study to select more reasonable set of factors. Second, since the Stacking method is an integrated framework, there are a large number of hyper parameters need to be calibrated, and the performance of the Stacking model is highly dependent on the choice of these parameters. In this research, a system grid search method is used to determine most of important parameters. However, several common parameters are set as default values. This may make the performance of the model not be optimal. Further research can focus on selecting more reasonable hyper parameters to improve the performance of the model. In addition, a future research direction could be how to consider the spatial-temporal correlations in the modeling structure between crash severity and explanatory factors.

Acknowledgements

This research was funded in part by the National Natural Science Foundation of China (No.71701215), Foundation of Central South University (No. 502045002), Science and Innovation Foundation of the Transportation Department in Hunan Province (No. 201725), Postdoctoral Science Foundation of China (No.140050005) and Scientific Research Project of Ministry of Housing and Urban-Rural Development (No. 2017-R2-032).

References

Abdelwahab, H.T., Abdel-Aty, M.A., 2001. Development of artificial neural network models to predict driver injury severity in traffic accidents at signalized intersections. Transp. Res. Rec. 1746, 6–13.
Abellán, J., López, G., Oña, J.D., 2013. Analysis of traffic accident severity using decision rules via decision trees. Expert Syst. Appl. 40 (15), 6047–6054.
American Association of state Highway and Transportation Officials, 2010. Highway Safety Manual. Washington, D.C.. .
Anderson, T.K., 2009. Kernel density estimation and k-means clustering to profile road accident hotspots. Accid. Anal. Prev. 41 (3), 359–364.
Barua, S., El-Basyouny, K., Islam, M.T., 2014. A full Bayesian multivariate count data model of collision severity with spatial correlation. Anal. Methods Accid. Res. 3, 28–43.
Barua, S., El-Basyouny, K., Islam, M.T., 2016. Multivariate random parameters collision count data models with spatial heterogeneity. Anal. Methods Accid. Res. 9, 1–15.
Bifet, A., Frank, E., Holmes, G., Pfahringer, B., 2010. Accurate ensembles for data streams: combining restricted hoeffding trees using stacking. J. Mach. Learn. Res. 13 (13), 225–240.
Bijleveld, F., 2005. The covariance between the number of accidents and the number of victims in multivariate analysis of accident related outcomes. Accid. Anal. Prev. 37 (4), 591–600.
Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32.
Chen, S.Y., Wang, W., Henk, J.Z., 2009. Construct support vector machine ensemble to detect traffic incident. Expert Syst. Appl. 36 (8), 10976–10986.
Chen, C., Zhang, G., Qian, Z., Tarefder, R., Tian, Z., 2016. Investigating driver injury severity patterns in rollover crashes using support vector machine models. Accid. Anal. Prev. 90, 128–139.
Das, A., Abdel-Aty, M., Pande, A., 2009. Using conditional inference forests to identify the factors affecting crash severity on arterial corridors. J. Saf. Res. 40 (4), 317–327.
Delen, D., Sharda, R., Bessonov, M., 2006. Identifying significant predictors of injury severity in traffic accidents using a series of artificial neural networks. Accid. Anal. Prev. 38 (3), 434–444.

Dong, N., Huang, H., Zheng, L., 2015. Support vector machine in crash prediction at the level of traffic analysis zones: assessing the spatial proximity effects. Accid. Anal. Prev. 82, 192–198.
Freund, Y., Schapire, R., 1996. Experiments with a new boosting algorithm. Proceedings of the 13th International Conference on Machine Learning 148–156.
Friedman, J.H., 2001. Greedy function approximation: a gradient boosting machine. Ann. Stat. 29 (5), 1189–1232.
Harb, R., Yan, X., Radwan, E., Su, X., 2009. Exploring precrash maneuvers using classification trees and random forests. Accid. Anal. Prev. 41 (1), 98–107.
Hauer, E., 1997. Observational Before–After Studies in Road Safety. Pergamon Press, Elsevier Science Ltd., Oxford, England.
Hauer, E., Ng, J.C.N., Lovell, J., 1988. Estimation of safety at signalized intersections. Transp. Res. Rec. 1185, 48–61.
Iranitalab, A., Khattak, A., 2017. Comparison of four statistical and machine learning methods for crash severity prediction. Accid. Anal. Prev. 108, 27–36.
Li, Z., Liu, P., Wang, W., Xu, C., 2012. Using support vector machine models for crash injury severity analysis. Accid. Anal. Prev. 45 (2), 478–486.
Ma, J., Kockelman, K.M., 2006. Bayesian multivariate Poisson regression for models of injury count, by severity. Transp. Res. Rec. 1950, 24–34.
Ma, J., Kockelman, K.M., Damien, P., 2008. A multivariate Poisson-lognormal regression model for prediction of crash counts by severity, using Bayesian methods. Accid. Anal. Prev. 40 (3), 964–975.
Mahalel, D., 1986. A note on accident risk. Transp. Res. Rec. 1068, 85–89.
Mauro, R., Luca, M., Acqua, G., 2013. Using a K-means clustering algorithm to examine patterns of vehicle crashes in before-after analysis. Mod. Appl. Sci. 7 (10), 11–19.
Menahem, E., Rokach, L., Elovici, Y., 2009. Troika – an improved stacking schema for classification tasks. Inf. Sci. (Ny) 179 (24), 4097–4122.
Oña, J.D., López, G., Abellán, J., 2013. Extracting decision rules from police accident reports through decision trees. Accid. Anal. Prev. 50 (2), 1151–1160.
Pei, X., Wong, S.C., Sze, N.N., 2011. A joint-probability approach to crash prediction models. Accid. Anal. Prev. 43 (3), 1160–1166.
Schaffer, C., 1994. Cross-validation, stacking and Bi-level stacking: meta-methods for classification learning. In: In: Cheeseman, P., Oldford, R.W. (Eds.), Selecting Models from Data. Lecture Notes in Statistics, vol. 89 Springer, New York.
Tang, J., Liu, F., Zhang, W., Ke, R., Zou, Y., 2018. Lane-changes prediction based on adaptive fuzzy neural network. Expert Syst. Appl. 91, 452–463.
Wang, C., Quddus, M.A., Ison, S.G., 2011. Predicting accident frequency at their severity levels and its application in site ranking using a two-stage mixed multivariate model. Accid. Anal. Prev. 43 (6), 1979–1990.
Wolpert, D., 1992. Stacked generalization. Neural Netw. 5 (2), 241–259.
Xu, X., Wong, S.C., Choi, K., 2014. A two-stage bivariate logistic-Tobit model for the safety analysis of signalized intersections. Anal. Methods Accid. Res. 3–4, 1–10.
Ye, X., Pendyala, R.M., Washington, S.P., Konduri, K., Oh, J., 2009. A simultaneous equations model of crash frequency by collision type for rural intersections. Saf. Sci. 47 (3), 443–452.
Ye, X., Pendyala, R.M., Shankar, V., Konduri, K.C., 2013. A simultaneous equations model of crash frequency by severity level for freeway sections. Accid. Anal. Prev. 57, 140–149.
Zeng, Q., Huang, H., 2014. A stable and optimized neural network model for crash injury severity prediction. Accid. Anal. Prev. 73, 351–358.
Zhang, Y., Xie, Y., 2007. Forecasting of short-term freeway volume with v-support vector machines. Transp. Res. Record 92–99.