applied sciences

MDPI

Article

Different Forecasting Model Comparison for Near Future Crash Prediction

Bowen Cai [1,2,*] and Qianli Di [3]

[1] College of Transportation Engineering, Tongji University, Shanghai 201804, China
[2] Department of Civil and Environmental Engineering, Imperial College, London SW7 2AZ, UK
[3] Department of Computer Science, The University of Sheffield, Sheffield S10 2TN, UK
* Correspondence: caibowen@tongji.edu.cn

Abstract: A traffic crash is becoming one of the major factors that leads to unexpected death in the world. Short window traffic crash prediction in the near future is becoming more pragmatic with the advancements in the fields of artificial intelligence and traffic sensor technology. Short window traffic prediction can monitor traffic in real time, identify unsafe traffic dynamics, and implement suitable interventions for traffic conflicts. Crash prediction being an important component of intelligent traffic systems, it plays a crucial role in the development of proactive road safety management systems. Some near future crash prediction models were put forward in recent years; further improvements need to be implemented for actual applications. This paper utilizes traffic accident data from the study Freeway in China to build a time series-based count data model for daily crash prediction. Lane traffic flow, weather information, vehicle speed, and truck to car ratio were extracted from the deployment of non-intrusive detection systems with support of the Bridge Management Administration study and were input into the model as independent variables. Different types of prediction models in machine learning and time series forecasting methods such as boosting, ARIMA, time-series count data model, etc. are compared within the paper. Results show that integrating time series with a count data model can capture traffic accident features and account for the temporal structure for variable serial correlation. A prediction error of 0.7 was achieved according to Root Mean Squared Deviation.

Keywords: crash prediction; time series count data model; ensemble learning; daily crash prediction; ARIMA and seasonal-ARIMA

check for updates

Citation: Cai, B.; Di, Q. Different Forecasting Model Comparison for Near Future Crash Prediction. *Appl. Sci.* **2023**, *13*, 759. https://doi.org/10.3390/app13020759

Academic Editor: Francesco Clementi

Received: 25 October 2022
Revised: 27 December 2022
Accepted: 31 December 2022
Published: 5 January 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

More than 1.35 million people die annually worldwide in road crashes, and over 50 million people are seriously injured [1,2]. A road crash is a complex phenomenon involving the interaction of road geometry, environment, traffic dynamics, vehicles, and humans [3–5]. Due to obstacles of data acquirement, conventional road crash prediction models rarely take traffic dynamics, such as speed, traffic flow, and vehicle types, into analysis. Safety performance functions (SPFs), sometimes referred to as crash estimation techniques, are mathematical methods that forecast the probability of collisions on certain highway infrastructure. They also look at how different factors affect the crash indicator. Real-time crash prediction aids in the detection and prevention of traffic accidents. Numerous real-time accident forecasting algorithms have been studied for decades in order to offer useful data for proactive traffic management. However, atypical changes in direction and speed that are indicated by traffic dynamics usually place other road users in jeopardy of collision [6]. Much evidence shows that traffic conflicts are a crash precursor, and traffic dynamics are, therefore, related to traffic crashes [7]. In recent years, a high-resolution detector has been able to provide abundant data regarding vehicle speed, lane traffic flow, vehicle types, and weather information, which makes real time crash prediction possible and more accurate.

Typical crash prediction models include statistical methods, machine learning methods, and the deep learning neural network. Bayesian dynamic logistic regression can tune model parameters by integrating a new instance with prior knowledge [8]. The main benefit is that, unlike traditional extrapolation, which usually only gets a prediction equation as well as a probability value, this Bayesian analysis also gets the whole spectrum of interpretive possibilities. The Bayesian hierarchical structure model accounts for non-stationarity and unobserved heterogeneity in conflict extremes to improve prediction accuracy [9]. The applied Bayesian Belief Net is used with involvement of ramp flow and speed of the ramp location as variables. Ref. [5] included vehicle trajectory into the analysis and investigated influential factors for crash happening. Refs. [10,11], implemented an autoregressive integrated moving average with explanatory variables (ARIMAX) model to analyze road crashes in Nigeria. The traditional ARIMA framework enables projections based solely on the forecasted variable's historical performance. The paradigm makes the assumption that a variable places considerable correlate proportionally on both its previous values and the magnitudes of any prior (probability) disruptions. ARIMAX was found to effectively capture short-term data features and can predict near future movements. According to ARIMA, previous time series analyses are utilized to anticipate consumption. Although ARIMA is a univariate approach, ARIMAX forecasts work by combining historical consumption information from multiple parameters that include datasets (such as climatic conditions) [12]. Ref. [13] used multivariate probit models to build models between crash types and prevailing traffic conditions. Ref. [14] used a multivariate negative binomial (MVNB) model by embedding it into the supervised fine-tuning module as a regression layer to address the unobserved heterogeneity issues in the traffic crash prediction. Ref. [2] proposed a long short-term memory (LSTM)—gradient boosting regression trees (GBRT) model to predict traffic crashes by strengthening the LSTM classifier by boosting regression trees. Gradient boosting constructs gradually weaker (easier) computational methods, every one of which attempts to anticipate the uncertainty left behind by the preceding algorithm. As a result, the system has a propensity to overload quite quickly. Ref. [14] proposed a grey neural network model which compensated for the small sample's data distortion. Ref. [15] used a back propagation neural network to tackle nonlinear mapping and complex internal mechanisms.

Road traffic crashes are non-negative, integer, and random event counts [16]. Count data models, such as negative binomial (NB) and Poisson models, have been widely used to analyze traffic accident data for the satisfaction of underlying distribution assumptions of traffic accidents [16]. A cross-sectional or panel count data model cannot take into account the temporal structure which existed in the time series data. Serial correlation will largely distort modeling results if variables are highly correlated to each other. A quantitative research method, called an autoregressive integrated moving average, or ARIMA, utilizes series data for the period, perhaps to properly comprehend the set of numbers or forecast future patterns. If a mathematical analysis forecasts possible trends using data from the past, it is said to be autoregressive. The modification of ARIMA, known as Seasonal Autoregressive Integrating Movement Estimate, or Seasonal ARIMA, specifically enables unitary series information for the period with a constant term. The autoregressive integrated moving average (ARIMA) model is commonly used in time series temporal analysis with serial correlation. ARIMA and Seasonal-ARIMA have been used to model time series count data and enjoyed many applications for the past years [16]. In Box-Jenkins equations, partial autocorrelation graphs are a frequent technique for logistic regression. The synchronization among X_t and X_{t-k} that is not taken into consideration by delays 1 to k − 1 is known as the incomplete hysteresis at lag k.

- However, the ARIMA model cannot be applied for non-negative integer-valued data such as traffic accidents, since the underlying assumption of errors in the ARIMA model is normal distribution.
- To model traffic accident distribution and take account into time series temporal structure, this paper proposes an integer-valued autoregressive (INAR) count data

model with fusion of multi-sources data including speed, traffic flow, vehicle types, and weather information to forecast daily crash happening at study Freeway in China.
- Selection of NB or Poisson model was discussed according to probability integral transform histogram (PIT).
- Autocorrelation (ACF) and Partial Autocorrelation (PACF) plots were used to check time series stationarity.
- Prediction error of 0.7 was achieved according to RMSD.

2. Literature Review

Traditional statistical methods, machine learning, and deep learning methods can be used in accident prediction. The developed logistic regression model is used to explore the relationship between traffic flow and crash occurrence [17]. Yu et al. adopted an artificial neural network (ANN) and support vector machine (SVM) for accident duration prediction [18]. Bao et al. used the convolutional neural network (CNN) to conduct citywide short-term crash risk prediction [19]. Schlögl et al. considered accident prediction as a binary classification task, finding that tree-based models have a good predictive effect [20]. Ensemble learning methods in the machine learning realm, the count data model for conventional statistical regression, and deep learning methods are the ones that are usually used for crash prediction.

2.1. Ensemble Learning

Ensemble learning methods are always used as classifiers in accident prediction, especially two different tree-based methods, random forest and XGBoost. Random forest is a robust ensemble algorithm, which implements random selection of random forest based on Bagging and avoids overfitting effectively [21]. Compared with the neural network and support vector machine, the random forest algorithm has stable training results, fewer parameters, and good generalization performance.

Parsa et al. [22] used XGBoost to detect the occurrences of an accident based on real time data. XGBoost is a flexible and incredibly effective multilayer perceptron solution that raises the level of computer resources for enhanced forest techniques. It was created primarily to enhance the efficiency as well as computational efficiency of neural network models. The detection rate reached 79%, which shows that XGBoost has high performance in accident prediction.

Schlögl [23] used random forest and XGBoost in hourly road accident prediction and found that XGBoost performed better in accuracy but worse in sensitivity; the highest sensitivity was 0.69, which means that a considerable part of the accident was predicted incorrectly.

2.2. Count Data Model

Crash-frequency data are non-negative, integer, and random event counts [16]. With the satisfaction of underlying distribution assumptions, traffic crash data are widely explored through the count-data modeling approach, such as the Poisson model [16]. As an extension of the Poisson regression model, the negative binomial model is employed to overcome the possible over-dispersion [24], because of the excess zero in the crash data. Since the traffic crash data are usually collected in a certain time unit as a sequence, a time series model, such as Autoregressive, integrating the moving average (ARIMA) model and Seasonal-ARIMA, is commonly used to model time series count data and has enjoyed many applications for the past years as a time sequential analysis technique with serial correlation [16]. However, the ARIMA model cannot be applied for non-negative integer-valued data such as traffic accidents, since the underlying assumption of errors in the ARIMA model is normal distribution. Even if the information does not comprise any lower numbers, the ARIMA framework can theoretically still develop negative assumptions [16]. To model traffic accident distribution and take into account time series temporal structure, this paper proposes a time series count data model with fusion of multi-sources data as covariates to forecast the occurrences of crashes.

2.3. Machine Learning-Based Model

In addition to statistical modeling, machine learning techniques and deep learning algorithms are also popular in crash-frequency prediction. For predicting the occurrences of car crashes happening on highways, artificial neural network (ANN) and Bayesian neural networks are the common techniques. These multilevel network structures contain a series of nodes connected by the weight factors in a hierarchical manner: input layer, hidden layer, and output layer. The weights of neural networks are assumed to be fixed and are able to be tuned [25–27]. In general, multilevel neural networks tend to perform better approximations on crash forecasting, but the generalization of these models still needs to be examined [28]. Machine learning methods based on statistical learning theory, such as support vector machines, may have better generalization and model interpretability. Wu and Wang [29] compared the performance of support vector regression (SVR) and back propagation neural networks on road traffic accident prediction and concluded that SVR had higher estimation accuracy with a good generalization.

2.4. Other Commonly Used Prediction Methods

The AdaBoost-CNN method combines a number of base estimators through a weighted linear combination [30]. Unlike typical sequential base forecasting methods, such as the Long Short Term Memory model and Recurrent Neural Network, the AdaBoost-CNN method uses the convolutional neural network to extract crash characteristics. In the training process, weights that are assigned to samples will be updated after each round according to prediction accuracy [31]. A less representative sample, if it is related to crash prediction, will be assigned greater weight. Therefore, AdaBoost-CNN can deal with imbalanced data to some extent. However, since the CNN network cannot integrate time information, while crash occurrences are closely related to time stamp, the effectiveness in crash prediction is subject to specific cases.

To increase representative crash samples, the Synthetic Minority Over-sampling Technique (SMOTE) is also widely used to reduce the impact of data imbalance. However, the generated data are highly correlated to the existing sample and cannot contain time information, as well.

The newly initiated graph neural network has one generalized version that contains a spatial-temporal block. The spatial-temporal block consists of temporal gate convolution, spatial gate graph convolution, and a temporal attention mechanism. It can learn both the local and global features of a region's data. Unlike CNN, in the graph neural network process graph with directional arrows, the arrows can model the graph of the road network to construct the macro graph of regions. However, for our specific study, we focus on a bridge section of the G15 freeway that cannot include macro features of road maps.

3. Methodologies

3.1. XGBoost

XGBoost is a machine learning algorithm used for supervised learning problems. It is an inherited learning algorithm based on Boosting [10]. It can improve performance by reducing model bias; the tree generated each time is a new function to fit the last residual, and the calculated values of each leaf node are added as the final predicted value.

$$\hat{y} = \sum_{m=1}^{M} f_m(x), \quad f_m \in \mathcal{F} \tag{1}$$

where $\hat{y}$ is predicted value of the model, M is the number of trees, and f_m represents the model of the m^{th} tree. $\mathcal{F}$ is the set of all tree models. The model's target function can be written as:

$$\mathcal{L}^{(m)} = \sum_{i=1}^{N} L(y_i, \hat{y_i}^{(m-1)} + f_m(x_i)) + \Omega(f_m) \tag{2}$$

where Ω is a regulation used to control the complexity of the model, which can be written as:

$$\Omega(f) = \gamma J + \frac{1}{2}\lambda \sum_{j=1}^{J} b_j^2 \tag{3}$$

where γ and λ are the penalty coefficient for the model, ${b_j}^2$ represents L2 norm of leaf scores, and J denotes the number of leaves for classification and regression trees.

3.2. Support Vector Machine (SVM)

The support vector machine can map linearly inseparable data to high-dimensional data space through a kernel function and then classify them in the high-dimensional space. The basic idea of SVM is to solve for the separation hyperplane that can divide the training dataset with the largest geometric spacing. Use $wx + b = 0$ to indicate separation of hyperplanes.

The objective function that satisfies the constraints is:

$$\begin{cases} \min\limits_{\mathbf{w},b,\xi} \frac{1}{2}\|w\|^2 + C\sum\limits_{i=1}^{m} \xi_i \\ \text{s.t. } y_i\left(\mathbf{w}^T\mathbf{x}_i + b\right) \geq 1 - \xi_i,\ \xi_i \geq 0, \quad i = 1, 2, \ldots m \end{cases} \tag{4}$$

Characteristics only occur as a linear combination, which kernels may compactly encode in the double construction of the SVM. A reduced feature dataset is mapped into a greater dimensionality information source using the kernel function in SVM. Assume that even after retraining an SVM with a binary decision criterion, users properly deduce that the SVM model is training patterns. SVM models use kernel functions to solve high-dimensional problems. For issues involving function approximation, radial basis function (RBF) systems are a popular variety of artificial neural systems. The ubiquitous approach and quicker training pace of multilayer perceptron systems set them apart from other machine learning systems. Gaussian kernel, also known as radial inner basis function (RBF), is a common kernel function. It can be described as follows:

$$k(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|}{2\delta^2}\right) \tag{5}$$

where δ is the kernel parameter. The SVM decision function is:

$$f(x) = \operatorname{sgn}\left[\sum_{i=1}^{n} \alpha_i y_i k(x_i, x) + b\right] \tag{6}$$

The classification of samples can be obtained by substituting the new sample points into the decision function.

Defects that are close to the actual output are ignored by the linearity ε-insensitive loss function and are considered equal to zero. The separation among the actual quantity y as well as the boundaries are used to calculate the damage. The loss function is used in SVM models to quantify the subjective probability of a particular training dataset. A SVM model's attributes and effectiveness rely on how it determines the experimental deviation of the provided training set. As a result, in SVM algorithms, the error function selection is extremely important. To solve the regression problem via SVM, an ε-insensitive loss function is introduced with the basis of SVM classification. Similar to SVM, the fundamental idea of SVR is to obtain an optimal classification hyperplane, so that all training samples have the least distance from the optimal classification surface.

In the high-dimensional feature space, the decisive function is established as follows:

$$f(x) = w\varphi(x) + b \tag{7}$$

$f(x)$ is the predicted value returned by the regression function with $\varphi(x)$ as the nonlinear mapping function.

The Lagrangian duality model postulates a method for determining a boundary or resolving an efficiency issue (the primary issue) by taking a look at an unrelated efficient algorithm (duality issue). In SVM, a hyperplane is a determination border that distinguishes between the two categories. Depending on which side of the feature space a source of data falls, it might belong to a different category. The dataset's combination of multiple characteristics determines the hyperplane's dimensions. Lagrange function is introduced and converted into the dual form to find the hyperplane weight vector w and offset vector b:

$$\begin{gathered} \max_{\alpha,\alpha} \sum_{i=1}^{m} y_i(\hat{\alpha}_i - \alpha_i) - \epsilon(\hat{\alpha}_i + \alpha_i) - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}(\hat{\alpha}_i - \alpha_i)(\hat{\alpha}_j - \alpha_j)x_i^{\mathrm{T}}x_j \\ \text{s.t.} \sum_{i=1}^{m}(\hat{\alpha}_i - \alpha_i) = 0, 0 \le \alpha_i, \hat{\alpha}_i \le C \end{gathered} \tag{8}$$

In order to identify a combination of parameters for every part of the training process that minimizes the estimation error, these regression analyses require a technique. Effective optimizing strategies may be applied, since the equations are straight and well known. Then, the optimal regression function needed to be obtained is:

$$f(x) = \sum_{i=1}^{m}(\hat{\alpha}_i - \alpha_i)\kappa(x)(x, x_i) + b \tag{9}$$

3.3. Random Forest

Random forest is an ensembled classification algorithm which builds a large number of uncorrelated decision trees. For a classification problem, the result of random forest is a majority decision based on the votes of all classes in each subtree.

Each tree is grown as follows:

For N number of cases in the training set, the sample is taken with replacement from the original data for growing the tree. If there are M input variables, a number m << M is specified such that at each node, being the only tuning parameter, the number of variables m are selected randomly from M and the best split on m for node split. The value of m is fixed over the forest growing. Each tree is grown to the largest extent possible without pruning within the stage.

3.4. Time Series Count Data Model

The practice of counting discontinuous occurrences over a certain length of time yields counting information. Most of the temporal approaches presumptively presume that the mechanism that produces the occurrences is time varying (t). They lack recollection as a result. Suppose $\{Y_t : t \in N\}$ is a count time series and $\{X_t : t \in N\}$ is a time-varying r-dimensional covariate vector, where $X_t = \{X_{t1}, \ldots, X_{t,r}\}^T$. The multivariate time series' standard statistical system has the following basic principle:

$$g(\lambda_t) = \beta_0 + \sum_{k=1}^{p}\beta_k\widetilde{g}(Y_{t-i_k}) + \sum_{l=1}^{q}\alpha_l\left(g(\lambda_{t-j_l}) - \eta^T diag(e)X_{t-j_l}\right) + \eta^T X_t \tag{10}$$

where $g : \mathbb{R}^+ \to \mathbb{R}$ is the connection function, and $\widetilde{g} : \mathbb{N}_0 \to \mathbb{R}$ is the conversion function. The parameter η alludes to the consequence of covariates, and $\{Y_{t-i_1}, Y_{t-i_2,\ldots}, Y_{t-i_p}\}$ is the set of lagged observations to be regressed on, and $\{\lambda_{t-j_1}, \lambda_{t-j_2,\ldots}, \lambda_{t-j_q}\}$ are the lagged conditional means. $P = \{1, 2, \ldots, p\}$ and $Q = \{1, 2, \ldots, q\}$ are the ordering variables for the observable information's correlation processes [32].

Log-linear projections suggest how the samples rely on the degrees of explanatory data, going beyond simple statistical measures. They simulate the relationships and interactions between descriptive analysis. The log linear model of the count data time series which allows for negative covariate effects can be written as:

$$v_t = log(\lambda_t) = \beta_0 + \sum_{k=1}^{p}\beta_k log(Y_{t-k} + 1) + \sum_{l=1}^{q}\alpha_l v_{t-l} \tag{11}$$

Using the Poisson presumption in combination, $Y_t = y|\mathcal{F}_t - 1 \sim Poisson\ (\lambda_t)$, with the multivariate time series standard statistical model's basic context, it follows that:

$$P(Y_t = y|\mathcal{F}_t - 1) = \frac{\lambda_t^y exp(-\lambda_t)}{y!},\ y = 0, 1, \ldots \tag{12}$$

with the supposition $var(Y_t|\mathcal{F}_t - 1) = E(Y_t|\mathcal{F}_t - 1) = \lambda_t$.

The Poisson distribution can be replaced by the inverse probability density, particularly under the above-described variant description [32]. It is particularly helpful for univariate values with a standard deviation greater than the sample mean and an infinite positivity spectrum. With a significantly larger variability in the predictor variables due to heteroscedasticity in other data points, the impact of many variables that cause the relative frequency to rely on earlier experiences, the appearance of exceptions, and the appearance of surplus nulls in the regression analysis are all causes of overdispersion [32]. To tackle potential over-dispersion issues in some scenarios, in this case crash-frequency, the Negative Binomial distribution, instead of Poisson distribution, $Y_t = y|\mathcal{F}_t - 1 \sim Poisson\ (\lambda_t, \varnothing)$, is considered with the dispersion parameter $\varnothing$, satisfying:

$$P(Y_t = y|\mathcal{F}_t - 1) = \frac{\Gamma(\phi + y)}{\Gamma(y+1)\Gamma(\phi)}\left(\frac{\phi}{\phi + \lambda_t}\right)^{\phi}\left(\frac{\lambda_t}{\phi + \lambda_t}\right)^{y}, y = 0, 1, \ldots. \tag{13}$$

where the variance refers to $var(Y_t|\mathcal{F}_t - 1) = \lambda_t + \lambda_t^2/\sigma$.

Quasi maximum likelihood estimation for mixed Poisson assumptions is expressed as the following:

$$\Theta = \{\vartheta \in R^{p+q+r+1} : \beta_0 > 0,\ \beta_1, \ldots, \beta_p,\ \ldots, \alpha_p,\ \eta_1, \ldots,\ \eta_r \geq 0,\ \textstyle\sum_{k=1}^{p} \beta_k \sum_{k=1}^{p} \alpha_\iota < 1\} \tag{14}$$

The log-linear model with covariates the parameter space is:

$$\Theta = \left\{\vartheta \in R^{p+q+r+1} :\ \beta_0 > 0,\ |\beta_1|, \ldots, |\beta_p|, |\alpha_1|, \ldots, |\alpha_p|, |\textstyle\sum_{k=1}^{p} \beta_k + \sum_{k=1}^{p} \alpha_\iota| < 1\right\} \tag{15}$$

The intercept β_0 is positive, and all other parameters are non-negative. The quasi-maximum likelihood estimation for Negative Binomial distribution is maximized for an estimated dispersion parameter φ, which, given the estimated regression parameters $\varnothing$, are iterated until convergence [32].

Whenever precise probability approaches, such as maximum likelihood estimation, are technologically impractical, quasi-likelihood techniques are employed to determine the parameters of a predictive method [32]. Quasi-likelihood approximations are less asymptotically efficient than, say, probabilistic forecasting models, since incorrect probability is employed. For a vector of observations, the conditional quasi log-likelihood function is:

$$\iota(\vartheta) = \textstyle\sum_{t=1}^{n} log\ p_t(y_t; \vartheta) = \sum_{t=1}^{n} (y_t ln(\lambda_t(\vartheta)) - \lambda_t(\vartheta) \tag{16}$$

where $p_t(y_t; \theta)$ is the probability density function of Poisson distribution. The conditional score function is a $(p + q + r + 1)-$ dimensional vector as:

$$S_n(\theta) = \frac{\partial l(\theta)}{\partial \theta} = \sum_{t=1}^{n}\left(\frac{y_t}{\lambda_t(\theta)} - 1\right)\frac{\partial \lambda_t(\theta)}{\partial \theta} \tag{17}$$

The conditional information matrix refers to:

$$G_n(\theta; \sigma^2) = \textstyle\sum_{t=1}^{n} cov\left(\frac{\partial l(\theta; Y_t)}{\partial \theta}\middle|\mathcal{F}_{t-1}\right) = \sum_{t=1}^{n}\left(\frac{1}{\lambda_t(\theta) + \sigma^2}\right)\left(\frac{\partial \lambda_t(\theta)}{\partial \theta}\right)\left(\frac{\partial \lambda_t(\theta)}{\partial \theta}\right)^T \tag{18}$$

In this case, the Poisson distribution assumes that $\sigma^2 = 0$, while the Negative Binomial distribution occurs with the assumption $\sigma^2 = \frac{1}{\varphi}$.

3.5. Artificial Neural Network

Artificial neural network is analogous to human cognitive neurobiology in a mathematical way [33]. A typical ANN structure comprises a series of neurons, also called nodes, or units. Through the directed communication links, each neuron with an associated weight is connected to other neurons. These weight factors are considered as the parameters of the model. The ANN model usually establishes, in a hierarchical manner: one input layer, one or several hidden layers, and one output layer [33]. Figure 1 illustrates a basic neural network with three layers.

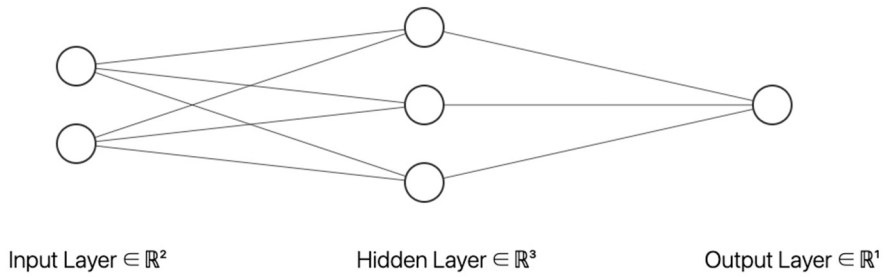


Figure 1. Layer neural network.

The process of predicting the accident number Y follows the formula:

$$Y = b_{2,1} + \sum_{j=1}^{n} w_j f\left(\sum_{i=1}^{m} W_{ij} y_{t-i} + b_{1,j}\right) \tag{19}$$

where m represents the number of nodes in the input layer; n is the number of hidden layer nodes; f is the activation function used in the hidden layers to transform the output into an acceptable range; $\{W_{ij}, i = 0, 1, \ldots, m; j = 1, 2, \ldots, n\}$ is a weight vector from the nodes in input layer to hidden layer; $\{w_j, j = 0, 1, \ldots, n\}$ is the weight vector from the hidden nodes to output nodes; $b_{2,1}$ and $b_{1,j}$ are the bias terms related to the nodes in output layer and hidden layer, respectively [34]. In this study, the ReLU (Rectified Linear Unit) activation function was employed and defined as:

$$f(x) = \begin{cases} 0, \text{ if } x < 0 \\ x, \text{ if } x \geq 0 \end{cases} \tag{20}$$

where x refers to the input value.

4. Model Design

4.1. Data Preparation

This study focuses on a 10 miles section on a interstate highway bridge in I-95 interstate highway in Pennsylvania, USA. The data used in this paper for the study section are from 1 January 2021–31 December 2021. North-South direction and South-North direction, re-spectively, contain 8638 samples.

An overview of variables considered is provided in Table 1.

Hourly data in the nine-month interval between July 2020 and March 2021, containing the number of car crashes, vehicle speed, traffic flow of three lanes, the ratio of truck to car, and holiday information, were collected for both directions of the study section. Daily data were computed based on the observed hourly data. In daily prediction, the traffic flow of three lanes was aggregated together, and weather data were also taken into account.

Table 1. Overview of variables.

Variable	Description	Type
Traffic Volume (veh/h) for Lane 1	To record traffic volume for the first lane	Continuous
Traffic Volume (veh/h) for Lane 2	To record traffic volume for the second lane	Continuous
Traffic Volume (veh/h) for Lane 3	To record traffc volume for the third lane	Continuous
Vehicle speed	Average speed of vehicle(km/h)	Continuous
Weather information	0: overcast 1: sunny 2: cloudy 3: rainy 4: snowy	Categorical
Holiday information	0: today is not a legal holiday in China 1: today is a legal holiday in China	Categorical
Truck to car ratio	The fraction of number of trucks to the number of passenger vehicles	Continuous
Crash in daily prediction	The number of car accidents on that day	Numerical, Ordinal
Crash in hourly prediction	0: no crash on a freeway segment within one hour 1: one or two crashes on a freeway segment within one hour	Categorical

Nine month data were used to train the model, the remaining three month data were set as the test set. A rolling base prediction structure was established. For each step, only the next day's crash number was predicted, and the next day crash number was added to the train set to subsequently update the model to predict the day after as a rolling base. The total prediction plot is shown as follows. The number of car accidents is the response variable, while other features are integrated as covariates into the model. In the modeling process, 70% of the whole dataset is set to be the training set, and the rest of dataset forms the test set. In addition, models are established and forecasted separately for each direction.

4.2. Model Performance Evaluation

4.2.1. Hourly Predictive Model

1. Accuracy: Refers to the proportion of correct predictions in all samples, which can be expressed by the formula:

$$Accuracy = \frac{True\ Positive + False\ Positive}{True\ Positive + True\ Negative + False\ Positive + False\ Negative} \quad (21)$$

2. Sensitivity: Represents the proportion of positive examples that are correctly classified, which measures the recognition ability of the classifier to positive examples. In the model established in this paper, the ability of correctly predicting accidents is very important; therefore, sensitivity is used as one of the indicators to measure the effect of the model, which can be expressed as follows:

$$Sensitivity = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (22)$$

3. ROC (Receiver Operating Characteristic Curve):

ROC curve is often used to evaluate the effect of binary classifiers. When the proportion of positive and negative samples in the test set changes, ROC does not change. In the accident dataset analyzed in this paper, there is an imbalance between positive

and negative samples. There are more negative samples than positive samples, and the proportion of positive and negative samples in the test data may also change with the partition of the dataset. The ROC curve can ensure the effectiveness of evaluation indicators under such circumstances.

4. AUC (Area Under Curve):

AUC is the area under the ROC curve, which can be used to evaluate the effect of binary classifier. AUC takes into account the ranking quality of sample prediction. AUC stands for the level or measurement of distinction, while ROC is a likelihood graph. It reveals how well the algorithm can differ across groups. The method is more appropriate at classifying 0 classes as 0, and classifying 1 event as 1, the greater the AUC. The larger AUC is, the better the classifier effect is. The closer AUC value is to 1, the better the model effect is. For random classifiers, the AUC value should be 0.5.

4.2.2. Daily Predictive Model

1. Average Prediction Error

The prediction error, also called residual, is used to measure how well the model predicts the response value. The average error of a sequence $\{y_1, y_2, \cdots, y_T\}$ obtained from the observed value over T times, with corresponding predicted values $\{\hat{y_1}, \hat{y_2}, \cdots, \hat{y_T}\}$, is computed as:

$$\text{Error } = \frac{|(\hat{y_1} - y_1)| + |(\hat{y_2} - y_2)| + \cdots + |(\hat{y_T} - y_T)|}{T} \tag{23}$$

The smaller the mean residual is, the more accurate the model is.

2. Root Mean Squared Deviation (RMSD)

The root mean square deviation is the standard deviation of the residuals, which estimates how spread out these prediction errors are. In other words, RMSD is a measure of how far from the estimated value of crashes to the true observed number of car accidents are. The RMSD of predicted values $\hat{y_t}$ at time t of a dependent variable y_t, with the independent variables observed over T times, is computed for T different predictions as the square root of the mean of the squares of the deviations:

$$\sqrt{\frac{\sum_{t=1}^{T} (\hat{y_t} - y_t)^2}{T}}. \tag{24}$$

The smaller RMSD means more accurate predicted values.

5. Model Results

5.1. Hourly Prediction

5.1.1. Model Performance

Accident prediction models were established for Nantung and Soochow, respectively, and then SVM, XGBoost, and Random Forest models were compared. Parameters were tuned to find the optimal parameters of the model, and the classification threshold was adjusted to examine the trade-off between accuracy and sensitivity. Parameters were tuned through a grid search with three-fold cross-validation with the criteria of the Gini impurity. How a thoroughly a clustering algorithm was divided is determined using a feature called Gini impurity. In essence, it assists us in selecting the ideal splitter so that we may create a clean decision tree. Gini impurities have values between 0 and 0.5. The easiest solution for model parameters tweaking is grid search. We essentially create a continuous grid within the feature subset region. Furthermore, utilizing cross-validation to calculate certain quality metrics, we attempt each possible permutation of the variables in this matrix.

Hyperparameter-tuning is completed for each single model independently. For XGBoost, the learning rate was 0.3, the number of estimators was 100, and max depth was 6. The optimal parameters of the random forest were 250 trees in the forest, the maximum depth with the value of 10, and 6 features for extraction when split. The SVM model used RBF kernel, penalty coefficient was 100, and γ is equal to 0.01.

The model effects are shown in Table 2:

Table 2. Model performance of hourly prediction.

Directions	Model	Threshold	Accuracy	Sensitivity	AUC
Nantung	SVM	NA	0.947	0.011	0.504
	XGBoost	0.01	0.656	0.734	0.717
	Random Forest	0.03	0.706	0.766	0.735
Soochow	SVM	NA	0.953	0	0.500
	XGBoost	0.008	0.801	0.762	0.782
	Random Forest	0.04	0.778	0.774	0.775

The ROC curve obtained by the random forest for hourly accident prediction for both Nantung and Soochow direction is shown in Figures 2 and 3, respectively. The diagonal dashed line is the reference line (random points). Area above the reference line represents good classification that is better than random and vice versa.

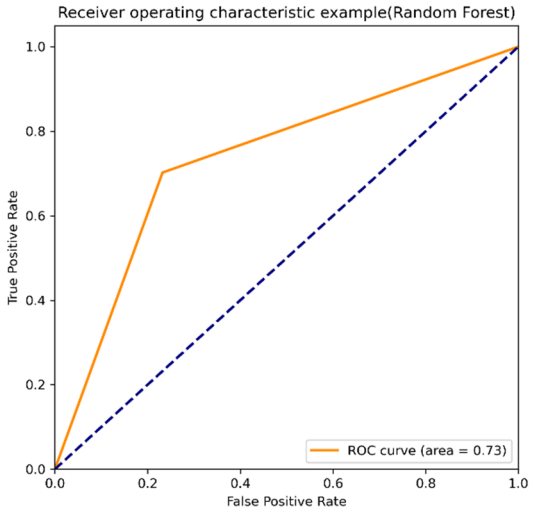


Figure 2. ROC in Nantung direction.

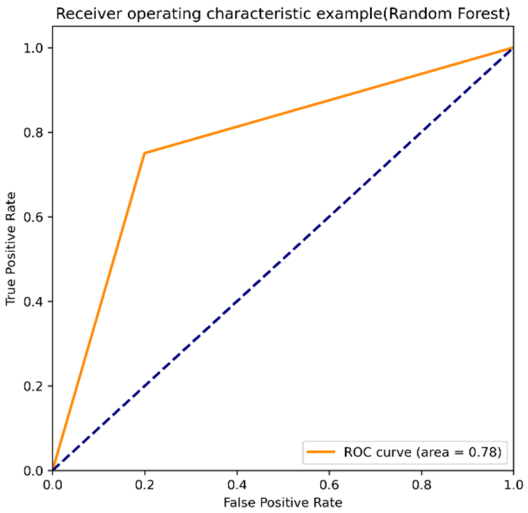


Figure 3. ROC in Soochow direction.

It can be seen from the table that the effect of SVM is poor. The sensitivity of the random forest is better than XGBoost. Since this paper focuses on the correct prediction

of accidents, random forest is a more suitable model. Figures 4 and 5 displayed below are the feature importance plots representing the importance of different features on the dependent variable, which is crash incidence in our study.

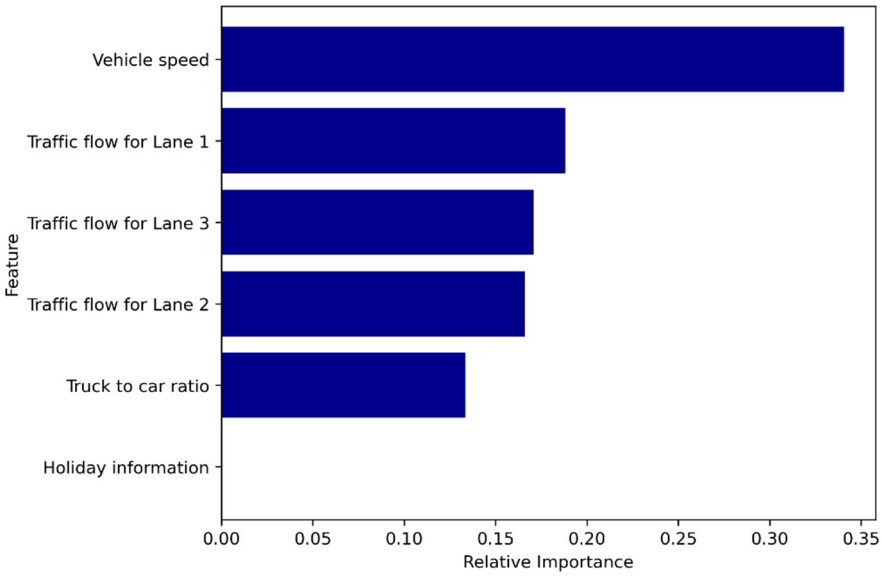


Figure 4. Feature relative importance in Nantung direction.

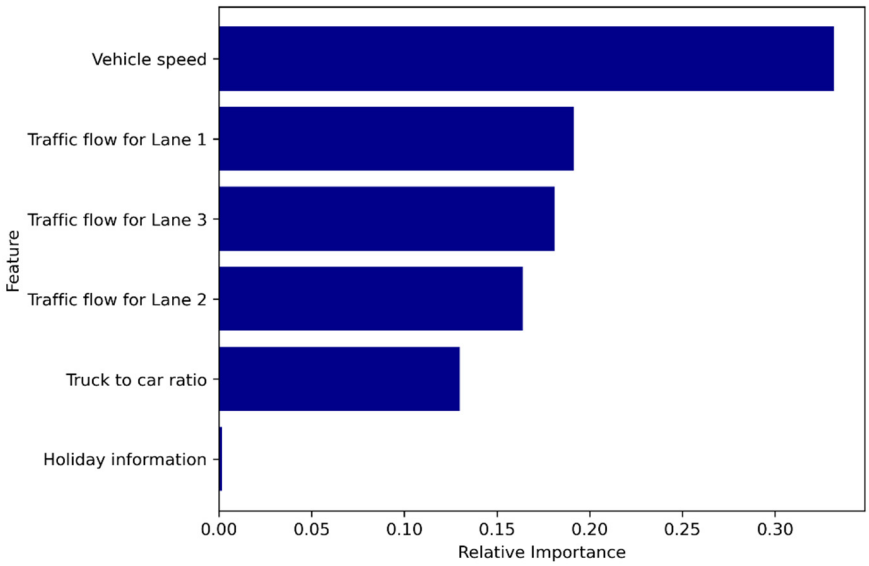


Figure 5. Feature relative importance in Soochow direction.

5.1.2. Feature Importance Plot

Although random forest cannot obtain the direction and absolute value of the regression coefficient like linear regression, it uses the decision tree as a weak classifier, which classifies data based on Gini coefficient, so the relative importance of variables can be identified through the Gini coefficient of each variable.

The ranking results of the importance of each variable can be returned. It can be found that:

1. The factor that has the greatest influence on short-term accidents is the average speed in this period, which accounts for a much higher proportion than other factors. This is because a usual traffic crash is related to congestion. When speed is higher, it means traffic flow is smoother. Crashes are less likely to take place. It seems contradictory

with common sense that the higher the speed, the more risk of road safety. The reason is that our study is designed for general road crash analysis, aggregating driving behavior, weather, traffic volume, speed, and other factors together, not for a specific vehicle.

2. Traffic flow in the three lanes has a similar influence on accident occurrence, and traffic flow in the first lane (overtaking lane) has the largest influence weight, which may be because the average speed of vehicles in the first lane is faster and accidents are more likely to occur.
3. Truck to car ratio also has an impact on traffic accidents, but smaller than vehicle speed.
4. Whether it was a holiday had little effect on whether the accident occurred.

5.2. Daily Prediction

Daily information, including vehicle speed, traffic flow, truck to car ratio, holiday, and weather condition, is treated as the covariate when modeling. In the process of prediction, a rolling based structure was adopted. Specifically, at each step of prediction, only the next day's accident number was estimated. Then, the next day crash number was appended to the training set in order to update the model and subsequently predict the day after as a rolling base.

5.2.1. Model Performance

In both directions, combining the tables and the plots, the performance of a Negative binomial count data time series model outweighs support vector regression and artificial neural networks in predicting daily crash-frequency. Table 3 displays the model performance of daily prediction.

Table 3. Model performance of daily prediction.

Direction	Model	Avg_Error	RMSD
Nantung	Count Data Time Series with NB	0.70	0.96
	SVR	1.11	1.32
	ANN	1.77	2.42
Soochow	Count Data Time Series with NB	0.72	1.00
	SVR	1.01	1.29
	ANN	1.79	2.32

To mitigate the detrimental impact of variables, the logarithmic link function was used. The seasonality of the data was also considered as the regression on $\lambda t-7$, the unobserved conditional mean seven time units, which corresponds to a week back in time. The fitting distribution was decided through the synthesis result of scoring rules and diagnostic histograms.

Diagnostic plots of ACF, marginal calibration, and PIT were generated to compare the performance of fitting Poisson conditional distribution and a Negative binomial conditional distribution. Figure 6 shows diagnostic plots after model ftting in Nantung direction.

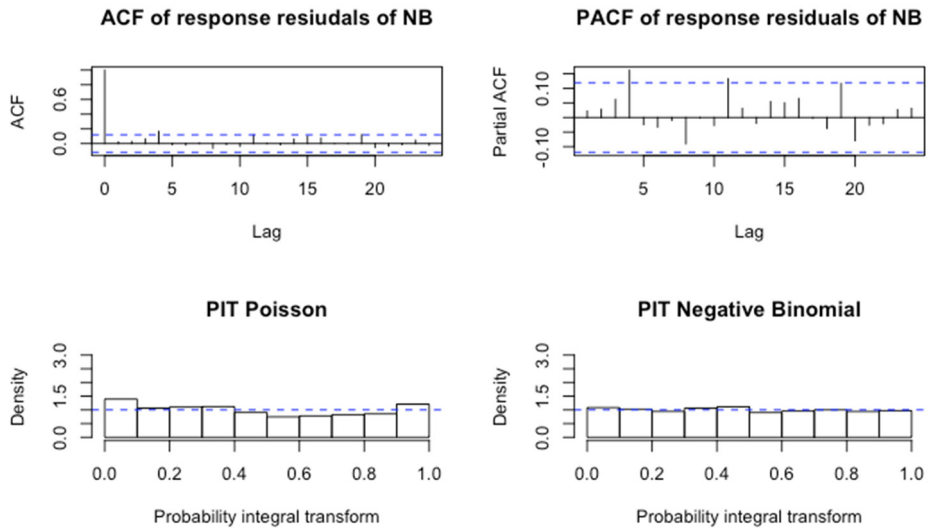


Figure 6. Diagnostic plots after model fitting in Nantung direction.

From the ACF plot of fitting Negative Binomial distribution, no obvious serial correlation was shown based on autocorrelation function. By comparing the probability integral transform plots, the Negative Binomial one appears more uniform, and thereby is more proper to be used in the fitting model. Besides, various metrics, namely, logarithmic, quadratic, spherical, rankprob, dawseb, normsq, and sqerro, are considered when comparing the suitability of Poisson and NB distribution. The smaller criterion points refer to better fitting of the distribution model. Table 4 shows the comparison between Poisson and NB model in Nantung direction.

Table 4. Comparison between Poisson and NB Model in Nantung direction.

	Logarithmic	Quadratic	Spherical	Rankprob	Dawseb	Normsq	Sqerror
Poisson	1.494	−0.308	−0.541	0.719	1.525	1.445	2.465
Negative Binomial	1.441	−0.319	−0.550	0.713	1.408	0.967	2.465

Scoring metrics shown in the above table indicate that the NB model is preferred. With the concerns of the over-dispersion issue of data distribution and the combined performance of measure metrics along with diagnostic graphs, the Negative binomial for the time series count data model was adopted.

The coefficient beta_1 and beta_7 corresponds to regression on the previous observation; alpha_7 corresponds to regression on values of the conditional mean seven units back in time for consideration of weekly periodicity. Table 5 shows the summary statistics for Negative Binomial Model in Nantung direction.

Table 6 shows the summary statistics for Negative Binomial Model in Soochow direction.

The fitted model for the number of crashes in Nantung direction Y_t in the time period t, is given by $Y_t|\mathcal{F}_t - 1 \sim NegBin(\lambda_t, 8.96)$ with:

$$\lambda_t = 8.03 \times 10^{-6} + 0.29Y_{t-1} + 0.18Y_{t-7} + \left(7.3 \times 10^{-6}, -0.01\ , -1.29\ , -0.02,\ 0.12\right)^T X_t, t = 1, \ldots, 271$$

Table 5. Summary statistics for Negative Binomial Model in Nantung direction.

	Estimate	Std.Error	CI(Lower)	CI(Upper)
Intercept	8.03×10^{-6}	0.089	−0.174	0.174
beta_1	0.285	0.094	0.101	0.469
beta_7	0.177	0.097	−0.013	0.367
alpha_7	0.538	0.099	0.344	0.731
traffic flow	7.30×10^{-6}	5.89×10^{-6}	-4.25×10^{-6}	1.89×10^{-5}
vehicle speed	−0.010	0.004	−0.018	−0.002
holiday	−1.290	0.404	−2.090	−0.503
weather	−0.016	0.056	−0.125	0.094
truck to car	0.123	0.071	−0.016	0.262

Table 6. Summary statistics for Negative Binomial Model in Soochow direction.

	Estimate	Std.Error	CI(Lower)	CI(Upper)
Intercept	1.04×10^{-5}	0.137	−0.268	0.268
beta_1	0.521	0.127	0.272	0.771
beta_7	0.239	0.139	−0.034	0.512
alpha_7	0.240	0.166	−0.086	0.565
traffic flow	1.35×10^{-6}	1.01×10^{-5}	-6.27×10^{-6}	3.32×10^{-5}
vehicle speed	−0.010	0.007	−0.024	0.003
holiday	−0.540	0.413	−1.35	0.269
weather	−0.060	0.077	−0.211	0.091
truck to car	−0.069	0.160	−0.383	0.245

The fitted model for the number of crashes in Nantung direction Y_t in the time period t is given by $Y_t|\mathcal{F}_t - 1 \sim NegBin(\lambda_t, 1.98)$ with:

$$\lambda_t = 1.04 \times 10^{-5} + 0.52Y_{t-1} + 0.24Y_{t-7} + \left(1.35 \times 10^{-6}, -0.01\ , -0.54, -0.06, -0.07\right)^T X_t, t = 1, \ldots, 271$$

From the tables above, it indicates that the previous crashes indeed influence the number of car accidents occurring in the following days. In the Nantung direction, it shows that vehicle speed and holiday covariates have negative effects on the number of crashes. To be specific, the crash-frequency on a non-holiday day is higher than the frequency on a holiday. The possible reason may be vehicles operating on the studied bridge are usually for daily use or business, and people who travel between Nantung and Soochow on holidays have a predilection for other transportation. Lower vehicle speed relates to more accidents. The possible explanation may be the vehicle speed is the past information. With the common sense that more severe car accidents may cause more severe traffic congestion and much lower vehicle speed, the model has learned the pattern of association between crashes and low vehicle speed, and thereby it indicates a negative relationship between the number of crashes and vehicle speed. However, in the Soochow direction, the covariates are not significant, as observed from the confidence intervals in the above tables. One possible explanation is that the covariates are the records in the past, while only the real-time information may highly affect real crashes. Another explanation may be that the influencing factors of crashes are various [7], and there may exist other factors influencing the number of car accidents. The covariates selected in this study may be limited, and these factors may only interpret part of the reason. Weather is negatively correlated with number of crashes. A possible explanation is that the study zone is part of

the freeway on the bridge. When the weather is getting fierce, regulations will be devised to cope with the potential risk, such as speed limit and traffic police interference. Such positive actions reduce crash incidence rate. Figure 7 is the prediction results in Nantung direction of NB Count Data Model. Figure 8 is the prediction results in Soochow direction of NB Count Data Model.

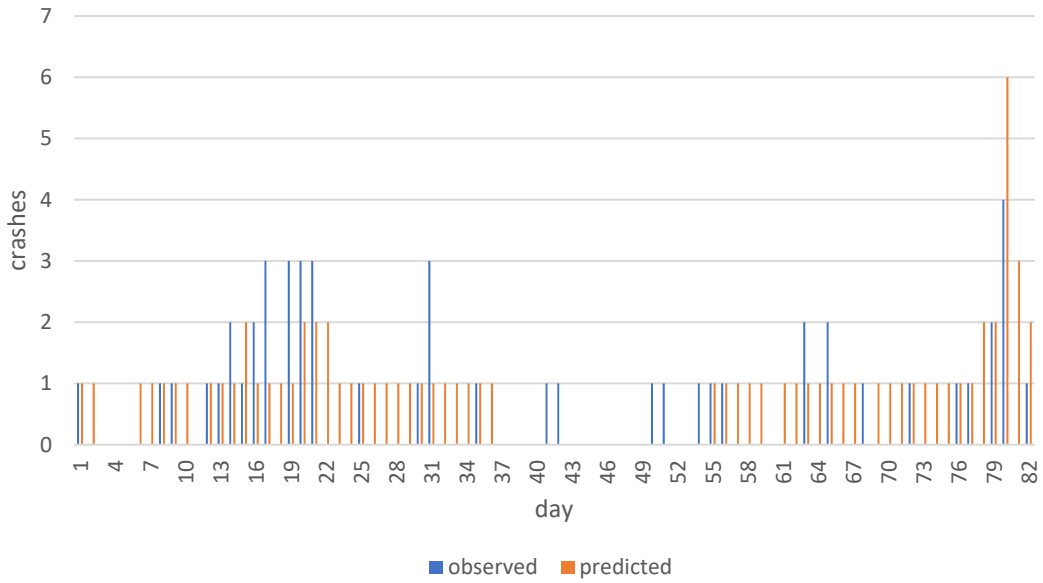


Figure 7. Prediction in Nantung direction of NB Count Data Model.

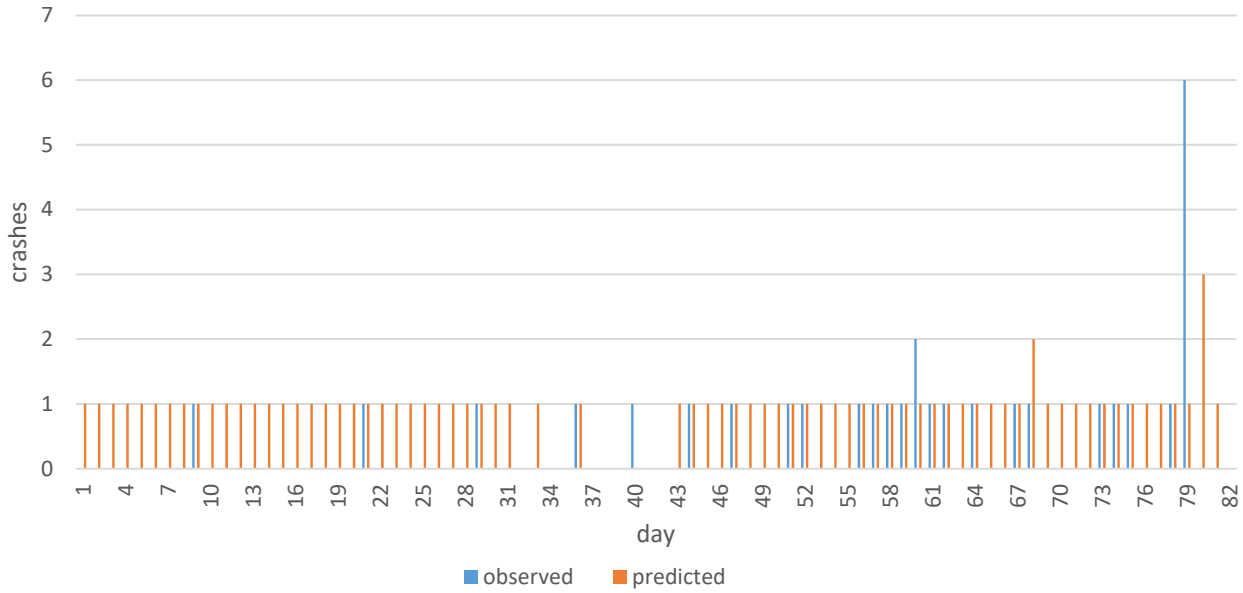


Figure 8. Prediction in Soochow direction of NB Count Data Model.

According to less than 3 months of rolling prediction results, the average error is 0.7 car accidents per day compared to the real crash number in the Nantung direction, while the average prediction error in the Soochow direction is 0.72. Overall, the root mean square deviation of prediction results in the Nantung direction is 0.96, while in the Soochow direction the root mean square deviation of predicted car crashes is 1.

5.2.2. Support Vector Regression Model

Through examining other scholars' work, the radial basis kernel function was selected, since it has a better generalization result and can realize nonlinear mapping. In addition, the radial basis kernel has less parameters, which influences the complexity of the model.

Owing to the force at the roller connections, the material is concurrently moved forward and squeezed by the blades. The process is finished as immediately as the product moves through the bearings. At each step of rolling-based prediction, the choice of parameter combination of the kernel coefficient and the penalty term was tuned through a grid search with 10-fold cross validation. Then, the optimal choice of the parameter combination was updated and trained in the next prediction.

According to less than 3 months of rolling based testing, the error of predicting crashes in the Nantung direction is 1.11 per day compared to the real accident number, while in the Soochow direction, the average error of crash prediction is 1.01 compared to the real crash cases. Overall, the root mean square deviation of prediction results in the Nantung direction is 1.32, while for the Soochow direction, the root mean square deviation is 1.29. Figure 9 shows the prediction results in Nantung direction of SVR.

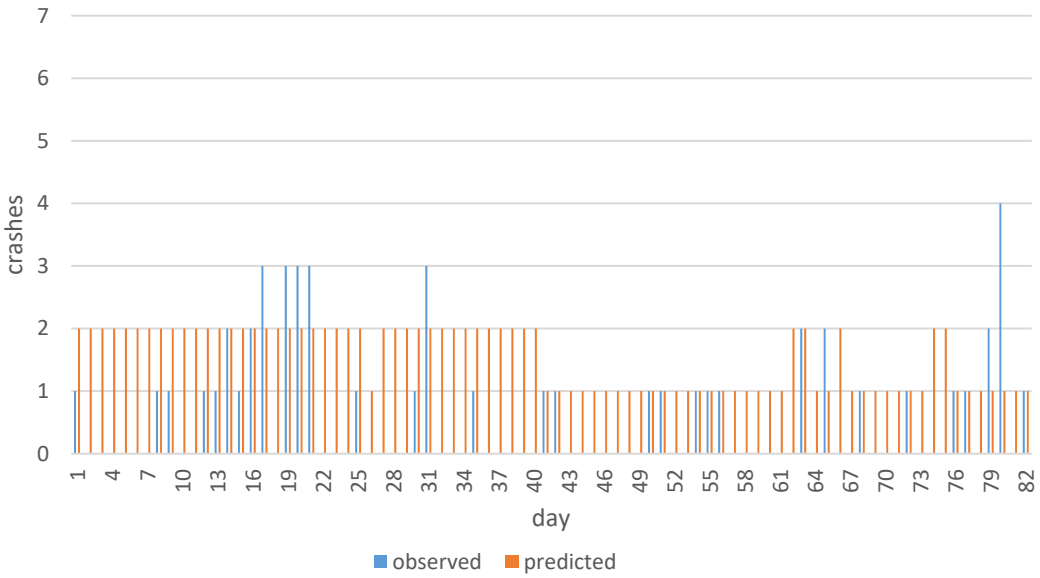


Figure 9. Prediction in Nantung direction of SVR.

Figure 10 shows the prediction results in Soochow direction of SVR.

5.2.3. Artificial Neural Network

Concerning the computational efficiency, the ReLU activation function was utilized. In order to improve the model performance, two hidden layers were considered: one with four nodes and another with one node. The covariates were fed into this artificial neural network as the input.

The performance of ANN in both directions is much poorer than the other two models. In the Nantung direction, the mean error of prediction is 1.77, and the root mean squared deviation is 2.42. In Soochow direction, the mean error is 1.79, and the root mean squared deviation is 2.32. Figure 11 shows the prediction results in Nantung direction of ANN.

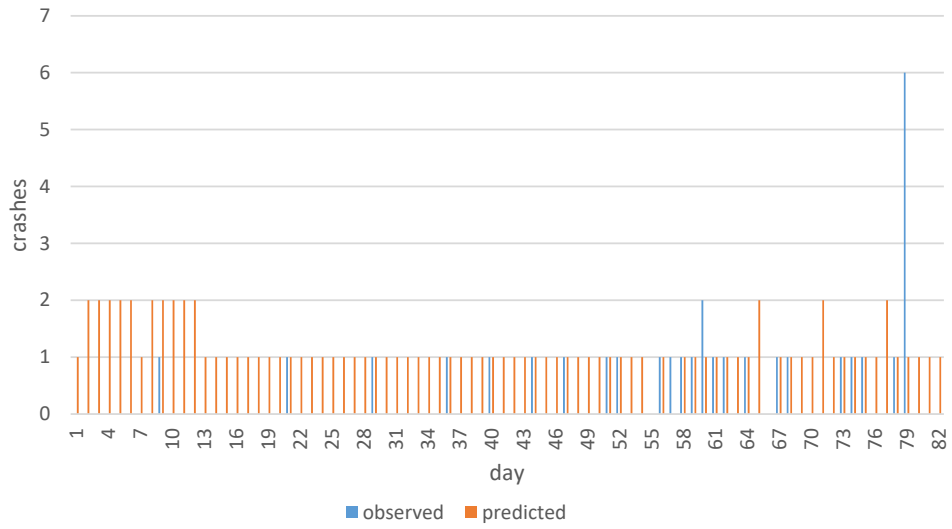


Figure 10. Prediction in Soochow direction of SVR.

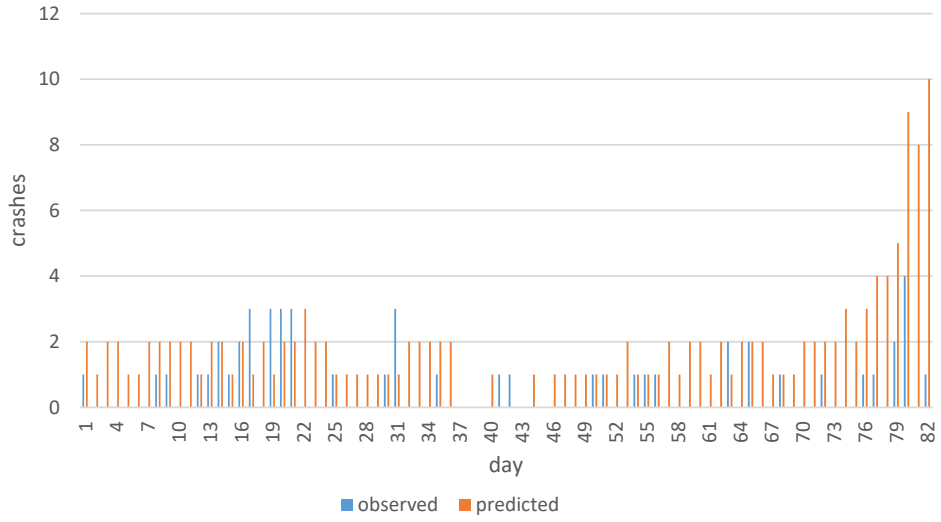


Figure 11. Prediction in Nantung direction of ANN.

Figure 12 shows the prediction results in Soochow direction of ANN.

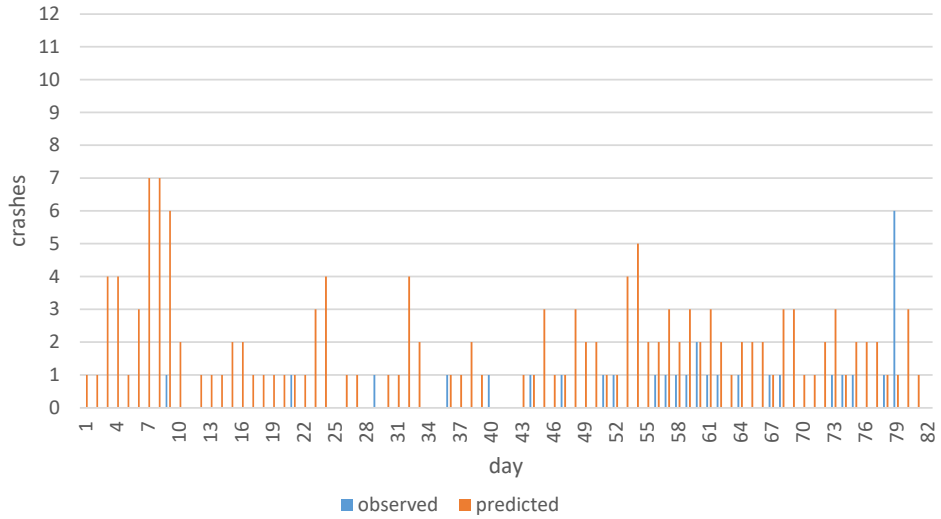


Figure 12. Prediction in Soochow direction of ANN.

6. Discussion and Conclusions

Traffic accident incidence results from various factors. Spatial and temporal structures are traffic accidents' inherent traits. Traffic accidents usually follow Poisson or Negative Binomial distribution. This study used a time series Negative Binomial count data model to capture traffic crash features, investigate influential factors, and predict future crash number. Daily crash incidence is difficult to forecast. This study achieved 0.7 error of RMSD for crash prediction. Results show that previous crash occurrence is the most significant factor of future crashes happening. Other traffic operation data, such as speed, traffic flow, weather, and vehicle types, may not play a significant role in crash prediction.

Although deep learning forecasting methods such as Long Short Term Memory model (LSTM), Bayesian Neural Network (BNNs), or machine learning methods, such as Support Vector Machine (SVM) as well as multivariate adaptive regression splines (MARS), achieved moderate accuracy in crash prediction, the difficulty of interpreting modeling structure compromises the real application of deep learning models in crash forecasting [13]. Linear regression is a statistical technique known as multimodal adaptable rehabilitation shafts (MARS). It is a non-parametric extrapolation approach that dynamically analyzes intermodulation distortion and interconnections among factors, acting as an augmentation of model parameters. For instance, even though the forecasting number is close to the real number, police officers do not know the specific reason leading to crashes. Therefore, duty arrangement is hard to allocate accordingly. In addition, due to the large scale of the modeling parameter, any small parameter change may influence model stability. Robustness of modeling is becoming an issue. The mathematical statistical model in time series count data distribution for crash forecasting is more robust, and the relationship between explanatory variables and traffic crash accidents is easy to interpret.

This paper compared the Negative Binomial time series model with the Poisson time series model and selected Negative Binomial time series model to account overdispersion. As an alternative, the Quasi-Poisson model may also be considered to account for overdispersion by adding a conditional variance of $\varphi\lambda$ as linearly and not quadratically increasing in the conditional mean of λ [34]. Additionally, zero-inflation hurdle GARCH processes can also be used as generalizations of time series count data model to deal with excess zeros, which is typical in real time traffic accidents for future analysis.

Hourly and daily crash prediction can help traffic police and the freeway control center to devise corresponding regulations to prevent crashes. However, since a crash is a small probability and random event, prediction of a crash incidence with high accuracy is very hard. Therefore, predicting crash risk may also be an alternative to predicting the real number. For actual crash number prediction, when the predicted value is larger than 0.5, then it will be rounded to 1. Roundness will also introduce some errors.

This paper compares several machine learning and time series forecasting methods in predicting short time traffic crashes. It shows that the count data time series Negative binomial model outperforms typical machine learning methods. The novelty of this research is to compare different forecasting methods for both machine learning and conventional time series models in crash prediction, and it proposes a count data time series model with loosening the strict assumption of ARIMA and introducing intervention and seasonal effect in INGARCH time series generalized linear regression. However, this research has some potential limitations. In the first place, it does not consider excess zero in crash data. For further improvement, the zero inflation or hurdle model structure can be involved. In the second place, it does not compare the results with a deep learning neural network, such as long short term memory model and spatio-temporal graph neural network. More studies will be conducted for future research.

Author Contributions: The authors confirm contributions to the paper are as follows: study conception and design: B.C. and Q.D., analysis and interpretation of results: B.C. and Q.D., draft manuscript preparation: B.C. and Q.D. Both authors are first author. All authors have read and agreed to the published version of the manuscript.

Funding: This work was sponsored by Shanghai Pujiang Program (20PJ1418400).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data is unavailable due to privacy, sensitiviety and security reason.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations and Expansions

Abbreviations	Acronyms
ARIMAX	Autoregressive integrated moving average with exogenous variables
ARIMA	Autoregressive integrated moving average
SPF	Safety performance functions
LSTM	Long short-term memory
GBRT	Gradient boosting regression trees
MVNB	Multivariate negative binomial
NB	Negative binomial
INAR	Integer-valued autoregressive
PIT	Probability integral transform
ACF	Autocorrelation
PACF	Partial Autocorrelation
ANN	Artificial neural network
CNN	Convolutional neural network
SVM	Support vector machine
SVR	Support vector regression
RBF	Radial basis function
ReLU	Rectified Linear Unit
AUC	Area Under Curve
ROC	Receiver operating characteristic curve
RMSD	Root Mean Squared Deviation
MARS	Multivariate adaptive regression splines
GARCH	Generalized AutoRegressive Conditional Heteroskedasticity

References

1. NHTSA. *Motor Vehicle Traffic Crashes as a Leading Cause of Death in the United States, 2012–2014*; Traffic Safety Facts, DOT HS 812 297; Department of Transportation, National Highway Traffic Safety Administration: Washington, DC, USA, 2016.
2. Zhang, Z.; Yang, W.; Wushour, S. Traffic Accident Prediction Based on LSTM-GBRT Model. *J. Control Sci. Eng.* **2020**, *2020*, 4206919. [CrossRef]
3. Sabey, B.E.; Staughton, G.C. *Interacting Roles of Road Environment Vehicle and Road User in Accidents*; Hrvatsko Drustvo za Ceste: Zagreb, Croatia, 1975.
4. Treat, J.R. Tri-Level Study of the Causes of Traffic Accidents: An overview of final results. *Proc. Am. Assoc. Automot. Med. Annu. Conf.* **1977**, *21*, 391–403. Available online: http://www.safetylit.org/citations/index.php?fuseaction=citations.viewdetails&citationIds[]=citjournalarticle_97133_19 (accessed on 25 January 2022).
5. Hossain, M.; Muromachi, Y. A Bayesian network based framework for real-time crash prediction on the basic freeway segments of urban expressways. *Accid. Anal. Prev.* **2011**, *45*, 373–381. [CrossRef] [PubMed]
6. Zheng, Z.; Ahn, S.; Monsere, C.M. Impact of traffic oscillations on freeway crash occurrences. *Accid. Anal. Prev.* **2010**, *42*, 626–636. [CrossRef] [PubMed]
7. Formosa, N.; Quddus, M.; Ison, S.; Abdel-Aty, M.; Yuan, J. Predicting real-time traffic conflicts using deep learning. *Accid. Anal. Prev.* **2020**, *136*, 105429. [CrossRef]
8. Yuan, J.; Abdel-Aty, M.; Gong, Y.; Cai, Q. Real-Time Crash Risk Prediction using Long Short-Term Memory Recurrent Neural Network. *Transp. Res. Rec. J. Transp. Res. Board* **2019**, *2673*, 314–326. [CrossRef]
9. Sabey, B.E.; Staughton, G.C. Bayesian hierarchical modeling traffic conflict extremes for crash estimation: A non-stationary peak over threshold approach. *Anal. Methods Accident Res.* **2019**, *24*, 100–106.
10. Wang, X.; Feng, M. Freeway single and multi-vehicle crash safety analysis: Influencing factors and hotspots. *Accid. Anal. Prev.* **2019**, *132*, 105268. [CrossRef]
11. Iranitalab, A.; Khattak, A. Comparison of four statistical and machine learning methods for crash severity prediction. *Accid. Anal. Prev.* **2017**, *108*, 27–36. [CrossRef]

12. Christoforou; Cohen, S.; Karlaftis. Identifying crash type propensity using real-time traffic data on freeways. *J. Saf. Res.* **2011**, *42*, 43–50. [CrossRef]
13. Dong, C.; Shao, C.; Li, J.; Xiong, Z. An Improved Deep Learning Model for Traffic Crash Prediction. *J. Adv. Transp.* **2018**, *2018*, 3869106. [CrossRef]
14. Liwei, H.U.; Zhang, T.; Guo, F.; Chen, Z. Traffic accident split rate of vehicle types prediction and prevention strategies study based on gray BP neural network. *J. Wuhan Univ. Technol.* **2018**, *10*, 388–392.
15. He, M.; Guo, X.C. 2e application of BP neural network principal component analysis in the forecasting the road traffic accident. In Proceedings of the Second International Conference on Intelligent Computation Technology and Automation, Zhangjiajie, China, 10–11 October 2009.
16. Quddus, M.A. Time series count data models: An empirical application to traffic accidents. *Accid. Anal. Prev.* **2008**, *40*, 1732–1741. [CrossRef]
17. Abdel-Aty, M.; Uddin, N.; Pande, A.; Abdalla, M.F.; Hsia, L. Predicting Freeway Crashes from Loop Detector Data by Matched Case-Control Logistic Regression. *Transp. Res. Rec. J. Transp. Res. Board* **2004**, *1897*, 88–95. [CrossRef]
18. Yu, R.; Abdel-Aty, M. Utilizing support vector machine in real-time crash risk evaluation. *Accid. Anal. Prev.* **2013**, *51*, 252–259. [CrossRef] [PubMed]
19. Bao, J.; Liu, P.; Ukkusuri, S.V. A spatiotemporal deep learning approach for citywide short-term crash risk prediction with multi-source data. *Accid. Anal. Prev.* **2019**, *122*, 239–254. [CrossRef] [PubMed]
20. Schlögl, M.; Stuetz, R.; Laaha, G.; Melcher, M. A comparison of statistical learning methods for deriving determining factors of accident occurrence from an imbalanced high resolution dataset. *Accid. Anal. Prev.* **2019**, *127*, 134–149. [CrossRef]
21. Zolfaghari, M.; Golabi, M.R. Modeling and predicting the electricity production in hydropower using conjunction of wavelet transform, long short-term memory and random forest models. *Renew. Energy* **2021**, *170*, 1367–1381. [CrossRef]
22. Parsa, A.B.; Movahedi, A.; Taghipour, H.; Derrible, S.; Mohammadian, A. (Kouros) Toward safer highways, application of XGBoost and SHAP for real-time accident detection and feature analysis. *Accid. Anal. Prev.* **2020**, *136*, 105405. [CrossRef]
23. Schlögl, M. A multivariate analysis of environmental effects on road accident occurrence using a balanced bagging approach. *Accid. Anal. Prev.* **2019**, *136*, 105398. [CrossRef]
24. Lord, D.; Mannering, F. The statistical analysis of crash-frequency data: A review and assessment of methodological alternatives. *Transp. Res. Part A Policy Pract.* **2010**, *44*, 291–305. [CrossRef]
25. Abdelwahab, H.T.; Abdel-Aty, M. Artifificial Neural Networks and logit models for traffific safety analysis of toll plazas. *Transp. Res. Rec.* **2002**, *1784*, 115–125. [CrossRef]
26. Riviere, C.; Lauret, P.; Ramsamy, J.M.; Page, Y. A Bayesian Neural Network approach to estimating the Energy Equivalent Speed. *Accid. Anal. Prev.* **2006**, *38*, 248–259. [CrossRef] [PubMed]
27. Xie, Y.; Lord, D.; Zhang, Y. Predicting motor vehicle collisions using Bayesian neural network models: An empirical analysis. *Accid. Anal. Prev.* **2007**, *39*, 922–933. [CrossRef]
28. Wu, D.; Wang, S. Comparison of road traffic accident prediction effects based on Svr and BP neural network. In Proceedings of the 2020 IEEE International Conference on Information Technology, Big Data and Artificial Intelligence (ICIBA), Chongqing, China, 6–8 November 2020. [CrossRef]
29. Roshandel, S.; Zheng, Z.; Washington, S. Impact of real-time traffic characteristics on freeway crash occurrence: Systematic review and meta-analysis. *Accid. Anal. Prev.* **2015**, *79*, 198–211. [CrossRef]
30. Taherkhani, A.; Cosma, G.; McGinnity, T.M. AdaBoost-CNN: An adaptive boosting algorithm for convolutional neural networks to classify multi-class imbalanced datasets using transfer learning. *Neurocomputing* **2020**, *404*, 351–366. [CrossRef]
31. Freund, Y.; Schapire, R.E. Experiments with a new boosting algorithm. In Proceedings of the Thirteenth International Conference on Machine Learning, Bari, Italy, 3–6 July 1996; pp. 148–156.
32. Liboschik, T.; Fokianos, K.; Fried, R. tscount: An *R* Package for Analysis of Count Time Series Following Generalized Linear Models. *J. Stat. Softw.* **2017**, *82*, 1–51. [CrossRef]
33. Guo, K.; Hu, Y.; Sun, Y.; Qian, S.; Gao, J.; Yin, B. Hierarchical Graph Convolutional Networks for Traffic Forecasting. In Proceedings of the The Thirty-Fifth AAAI Conference on Artificial Intelligence, Virtual, 2–9 February 2021.
34. Vanajakshi, L.; Rilett, L. A comparison of the performance of artificial. neural networks and support vector machines for the prediction of traffic speed. In Proceedings of the IEEE Symposium on Intelligent Vehicle, Parma, Italy, 14–17 June 2004. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.