Hindawi
Journal of Advanced Transportation
Volume 2021, Article ID 5569143, 12 pages
https://doi.org/10.1155/2021/5569143

Research Article

A Dynamic Bayesian Network-Based Real-Time Crash Prediction Model for Urban Elevated Expressway

Xian Liu,[1,2,3] Jian Lu,[1,2,3] Zeyang Cheng,[1,2,3] and Xiaochi Ma[1,2,3]

[1]*Jiangsu Key Laboratory of Urban ITS, Southeast University, Nanjing 211189, China*
[2]*Jiangsu Province Collaborative Innovation Center of Modern Urban Traffic Technologies, Southeast University, Nanjing 211189, China*
[3]*School of Transportation, Southeast University, Nanjing 211189, China*

Correspondence should be addressed to Jian Lu; lujian_1972@seu.edu.cn

Received 24 January 2021; Revised 8 March 2021; Accepted 6 May 2021; Published 15 May 2021

Academic Editor: Eneko Osaba

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

Traffic crash is a complex phenomenon that involves coupling interdependency among multiple influencing factors. Considering that interdependency is critical for predicting crash risk accurately and contributes to revealing the underlying mechanism of crash occurrence as well, the present study attempts to build a Real-Time Crash Prediction Model (RTCPM) for urban elevated expressway accounting for the dynamicity and coupling interdependency among traffic flow characteristics before crash occurrence and identify the most probable risk propagation path and the most significant contributors to crash risk. In this study, Dynamic Bayesian Network (DBN) was the framework of the RTCPM. Random Forest (RF) method was employed to identify the most important variables, which were used to build DBN-based RTCPMs. The PC algorithm combined with expert experience was further applied to investigate the coupling interdependency among traffic flow characteristics in the DBN model. A comparative analysis among the improved DBN-based RTCPM considering the interdependency, the original DBN-based RTCPM without considering the interdependency, and Multilayer Perceptron (MLP) was conducted. Besides, the sensitivity and strength of influences analyses were utilized to identify the most probable risk propagation path and the most significant contributors to crash risk. The results showed that the improved DBN-based RTCPM had better prediction performance than the original DBN-based RTCPM and the MLP based RTCPM. The most probable risk influencing path was identified as follows: speed on current segment (V) (time slice 2)⟶V (time slice 1)⟶speed on upstream segment (U_V) (time slice 1)⟶Traffic Performance Index (TPI) (time slice 1)⟶crash risk on current segment. The most sensitive contributor to crash risk in this path was V (time slice 2), followed by TPI (time slice 1), V (time slice 1), and U_V (time slice 1). These results indicate that the improved DBN-based RTCPM has the potential to predict crashes in real time for urban elevated expressway. Besides, it contributes to revealing the underlying mechanism of crash and formulating the real-time risk control measures.

1. Introduction

Predicting road crashes in real time is a hotspot in road safety under the context of active traffic management (ATM) over the past two decades. Real-time crash prediction refers to the assumption that the occurrence probability of a crash on a specific road segment can be predicted within a very short precrash time interval by adopting instantaneous traffic flow characteristics [1–3]. The development of Intelligent Transportation System (ITS) and advanced transportation information systems (ATIS) is helpful for easily collecting traffic data in real time, promoting the effective and accurate assessment on crash risk on highways and expressways by use of RTCPMs [4–11].

In general, numerous RTCPMs studies establish a direct connection between traffic flow data (i.e., volume, speed, occupancy and their combinations) and crash data. In these models, the collinearity and correlation among dependent variables are avoided; thus, the independence of variables is guaranteed [12, 13]. However, road crash is a complex phenomenon involving coupling interdependency among multiple influencing factors. The concept of coupling

interdependency can be used to express the interaction between various risks. Although these complex system factors can exhibit many characteristics on their own, in reality these individual factors interact and couple with each other in even more complex ways in terms of coupling direction and coupling strength [14–16]. This interaction is called coupling interdependency, which can lead to an increased or a decreased risk of an accident [17]. Therefore, it is essential for RTCPMs to account for the interdependency among influencing factors. Additionally, the one-time interval of traffic data is frequently adopted for real-time crash prediction in a number of RTCPMs [6, 18]. However, for urban elevated expressway, the merging and lane-changing driving behaviours are frequent due to the dense-ramp setting. The traffic flow characteristic is prone to displaying dynamicity that varies over time, which is closely associated with crash risk [19]. Therefore, the dynamicity of traffic flow in the temporal dimension should be considered with the implementation of the RTCPM for predicting crashes on urban elevated expressway.

DBN, a particular form of Bayesian Network (BN), represents the dynamic evolution of some state space model through time [20]. It has been widely used to predict and assess the dynamically evolving process of risk in the field of maritime accidents, tunnel construction, ship-ice collision, etc. [21–23]. In order to express the dynamicity of traffic flow characteristics, some RTCPMs studies apply time-series traffic data consisting of several time intervals [24–27], which are proved to be feasible and robust. However, these researches ignore the investigation of interdependency among different traffic flow characteristics and simply connect each influencing factor to crash risk directly in the construction process of the graph structure. As the most critical step in the DBN construction, the interdependency of variables can be well assessed in the DBN model with the application of the structure learning algorithm. Besides, the neural network-based models (e.g., MLP) are also able to accommodate correlated dependent variables. However, the whole model should be rebuilt and recalibrated once the future new variables and knowledge from new data are input, whose tuning process can be highly resource-demanding [13].

Furthermore, considering the interdependency among influencing factors also helps to reveal the underlying mechanism of crash occurrences. The present study estimates the crash risk by quantifying the probability of crash occurrences. We hope this model can provide some real-time countermeasures to mediate risk when there is a high probability of crash. The formulation of countermeasures should be based on the identification of risk propagation path and significant risk contributors. However, there has been a dilemma between predictive and explanatory models: the models specialized in prediction are not the best in knowledge discovery, and vice versa [7, 28]. The DBN model has the advantage of implementing uncertainty analysis and probability reasoning and conducting bidirectional uncertainty investigation for prediction and diagnostic analyses. Combining with the sensitivity and strength of influences analysis methods can not only identify the most probable risk propagation path, but also can recognise the most sensitive contributors in the propagation path [29]. Once the most sensitive risk contributors in the whole propagation path are revealed, the references for the sequence and emphasis of mediation can be provided, which helps to formulate appropriate real-time countermeasures to cut off the risk propagation path and decrease the probability of crash occurrence.

The existing DBN-based RTCPMs mainly emphasize the dynamicity of traffic flow characteristics, lacking investigation on the coupling interdependency among traffic flow characteristics. The main contributions of this study are (1) to apply the DBN structure learning algorithm in an example to predict road crashes; (2) to compare the performances of two DBN-based RTCPMs (considering the interdependency and not considering the interdependency) and the MLP-based RTCPM; and (3) to identify the most sensitive risk contributors in the propagation path by the use of the sensitivity and strength of influence analyses.

The manuscript is organized into five sections. The remainder is organized as follows. In Section 2, the materials and methods are presented. Section 3 presents the results and discussions. Section 4 provides some concluding remarks.

2. Materials and Methods

2.1. Study Area and Data Preparation. The 40 segments of the Yan'an elevated expressway in Shanghai, China, sequentially linking up to each other along the westbound and eastbound expressway, were selected as the study areas (see Figure 1). All segments are three-lane with detectors spaced at an approximate distance of 300–500 m. Each segment has similar road geometry and on/off-ramp arrangement; thus, the road geometries and ramp locations were not considered as influencing variables on the crash risk. There were 82 crashes that happened on the Yan'an expressway during August and September 2018. The dates, times, and segment IDs of the crashes were collected. Based on the matched case-control design, three corresponding noncrash cases for each crash case were randomly matched for the same segment and occurrence time (246 noncrash cases in total). Besides, traffic flow characteristics and weather variables were also obtained as inputs of the RTCPM, aiming at classifying the crash and noncrash states based on the investigation of relationship between crash risk, traffic flow characteristics, and weather conditions.

The existing dual-loop detectors in study areas are available for providing the average speed (km/h) and the average volume of a single lane (pcu/h) for each segment. Hourly weather variables, including visibility (km) and weather type (rainy or sunny), were collected from the Shanghai Xujiahui Observatory, which is 7.5 km far from the Yan'an expressway. In this study, the Traffic Performance Index (TPI) varying between 0 and 1 was applied as an indicator to measure the magnitude of congestion degree, where 1 is a traffic jam state and 0 is a free flow state (equation (1)). Consider

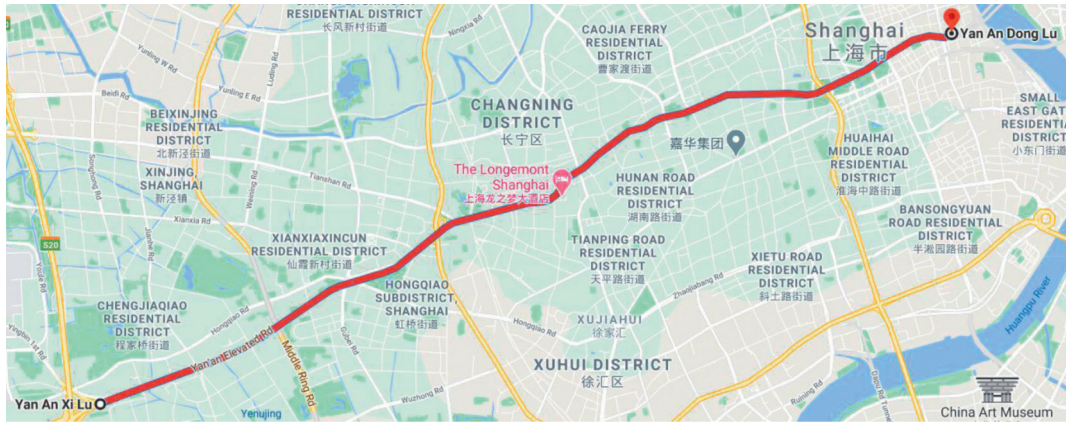

Figure 1: Yan'an elevated expressway, Shanghai, China.

$$\mathrm{TPI} = \frac{(V_{\max} - V_i)}{V_{\max}}, \tag{1}$$

where $V_{\max}$ is the maximum speed and V_i is the average speed at the ith time period.

The average speed data on the current, upstream, and downstream segments of the crash location and the TPI of the whole expressway were aggregated in 5-minute intervals. The evolution of traffic flow with time leading to a crash was a dynamic process; thus, the traffic flow characteristics of several time intervals before the crash should be combined to build the model. The intervals of 0–5 min (time slice 0), 5–10 min (time slice 1), and 10–15 min (time slice 2) prior to the crash were considered. The time slice 0 was excluded, because the crash warning system needs some time to recognise crash states, and the actual crash occurrence time and recorded time are not always completely consistent. Due to the raw weather data updated once an hour, the weather condition was regarded as a stable influencing variable across different time slices. Finally, the traffic flow and weather data corresponding to 82 crash cases and 246 noncrash cases were generated. In total, nine variables combining traffic flow characteristics on current, upstream, and downstream segments of the crash location with weather condition are shown in Table 1.

2.2. Random Forest (RF). The main purpose of constructing RTCPM is to evaluate crash risk in real time. High-dimensional variable space can increase the processing complexity of the RTCPM. Thus, Random Forest (RF), a widely used variable selection model, was implemented in this study to select influencing variables and reduce the redundancy of variables. Variable importance (VI) metric was used as the criterion to pick the mostly related variables [12, 30], which can be determined with the following steps.

(1) Sample N amount of data from the learning set to build a tree classifier by bootstrap sample technique, and the remaining samples of the learning set were not used in the growth of the tree. The left-out samples, an effective internal test data set, were called out-of-bag (OOB) data, which were adopted to obtain an unbiased error estimate. m number of variables were randomly selected from the original variable set M ($m < M$), and the best split variable in m at each tree node was adopted to split node. Each tree grew naturally without pruning. Repeat this step k times to construct RF consisting of k trees.

(2) Each tree classifier produced a classification result by voting for the binary target (crash or noncrash) based on OOB samples, and the classification error rate R_i was calculated consequently.

(3) Add random noise disturbance for the values of any variable in the OOB sample, and the new OOB sample was produced. Each tree that was implemented for crash/noncrash classification tests with the new OOB sample was used to calculate the classification error rate R_i'.

(4) VI was calculated as the increase in the mean of the classification error rate of trees after adding random noise disturbance. The calculation formula was shown in the following equation:

$$\mathrm{VI} = \frac{1}{k}\sum_{i=1}^{k}\left(R_i' - R_i\right). \tag{2}$$

2.3. Dynamic Bayesian Network (DBN). The Bayesian Network (BN) is a probabilistic graphical model that expresses the probability relationships among a set of variables that connect those variables in a directed acyclic graph (DAG). The BN has the advantages in learning causal relationships, predicting the consequences of intervention, and analyzing the most probable explanations of consequences. Some researchers have adopted BN to evaluate and analyze traffic accidents risk [31–33]. Most crashes did not happen based on a particular point in time, but they can be described through multiple traffic states among a series of time slices. The DBN is a kind of BN, which can couple time-series data to express the risk evolving process with time flowing forward [20]. With the application of the probabilistic inference, the critical step of BN generalization was to reveal the probabilistic dependencies of random variables, which are

Table 1: Information of the nine alternative variables.

Variable	Description
TPI	The average TPI of the whole expressway within 5 min interval
V	The average speed of current segment within 5 min interval (km/h)
U_V	The average speed of upstream segment within 5 min interval (km/h)
D_V	The average speed of downstream segment within 5 min interval (km/h)
Q	The volume of current segment within 5 min interval (pcu/h)
U_Q	The volume of upstream segment within 5 min interval (pcu/h)
D_Q	The volume of downstream segment within 5 min interval (pcu/h)
Visibility	The horizontal visibility within one hour (km)
Weather	The weather type in one hour, rainy or sunny

also expressed in time sequence in the DBN. Two types of dependencies existed in the DBN: dependencies within one time slice and dependencies among time slices. The DBN model consisted of observable evidence $X = \{x_1, x_2, \ldots, x_t\}$ and hidden variables $Y = \{y_1, y_2, \ldots, y_t\}$, which were traffic state variables and crash likelihood, respectively. When a Markov model and a BN were integrated to construct a DBN model, there were a transition model $P(x_t|x_{t-1})$, an observation model $P(y_t|x_t)$, and an initial state distribution $P(x_1)$. The joint probability distribution can be expressed as follows:

$$P(X, Y) = \prod_{t=2}^{t} P(x_t|x_{t-1}) \prod_{t=1}^{t} P(y_t|x_t) P(x_1). \quad (3)$$

There were three key steps to initialize a DBN model: (1) The ChiMerge algorithm was adopted to implement the discretization of continuous variables. (2) Structure learning was applied to present the graphic dependencies among variables. In this step, the DBN not only estimated the dependencies between variables within one time slice but also examined them among different time slices. The PC algorithm was used to build the structure of the BN within one slice among traffic state variables and crash likelihood. Then, the same variables among different slices were connected to build the structure of the DBN. (3) Parameter learning was conducted to learn the conditional probability distribution of variables within one time slice and across time slices. Parameter estimation was tested by the Expectation Maximization (EM) algorithm.

2.4. ChiMerge Algorithm. The continuous variables are usually problematic in DBNs because it fails to capture the relationships between the continuous variables [34]. The classical way to deal with continuous variables in DBNs is to discretize the variables [35]. Discretization is the operation of dividing continuous variables into a small number of intervals, where each interval is mapped to a discrete symbol. There are two widely used simple methods, the equal-width intervals, which divides the variables between the minimum and maximum values into a number of intervals in equal size, and the equal-frequency intervals, where the interval boundaries are chosen based on the fact that each interval contains the same number of samples. However, both of the methods ignore the class of samples [36]. A good discretization has both the intrainterval uniformity and interinterval difference. ChiMerge algorithm performs merging operation by using the χ^2 statistic to test whether there are significant differences or similarities of relative class frequencies between adjacent intervals.

The ChiMerge algorithm is mainly consisted of several steps.

(1) Sort the samples according to their value.

(2) Calculate the χ^2 value for each pair of adjacent intervals with the following equation:

$$\chi^2 = \sum_{i=1}^{m} \sum_{j=1}^{n} \frac{\left(A_{ij} - E_{ij}\right)^2}{E_{ij}}, \quad (4)$$

where $m = 2$ (the 2 intervals being compared), $n = 2$ (number of classes, i.e., crash and noncrash), A_{ij} = number of samples in the ith interval, jth class, and E_{ij} = expected frequency of A_{ij}.

(3) Merge the pair of adjacent intervals with the lowest χ^2 value until all pairs of intervals with χ^2 values beyond χ^2 threshold. The χ^2 threshold is determined by a desired significance level (0.95 percentile level in this study) and the number of degrees of freedom (1 less than the number of classes). There are 2 classes (crash and noncrash); thus, the degree of freedom is 1. Finally, the χ^2 value is 3.841.

2.5. PC Algorithm. The PC algorithm is an efficient and classical algorithm used for structural learning in BN [37, 38]. The process of the PC algorithm mainly consists of three steps:

(1) Determine the skeleton of the graph by conditional independence tests. Let $X = \{x_1, x_2, \ldots, x_k\}$be a set of random variables and $V = \{v_1, v_2, \ldots, v_k\}$ be a set of nodes in a graph so that each node in V represents a random variable in X. Then, construct an undirected graph G where all nodes are connected to each other, and then the PC algorithm implements statistical tests to remove or maintain edges between adjacent nodes x_i and x_j given a conditioning x_γ in the graph by calculating the cross entropy $\mathrm{CE}(x_i, x_j \mid x_\gamma)$:

$$\mathrm{CE}\left(x_i, x_j|x_\gamma\right) = \sum_{x_\gamma} P\left(x_\gamma\right) \sum_{x_i,x_j} P\left(x_i, x_j|x_\gamma\right) \cdot \log \frac{P\left(x_i, x_j|x_\gamma\right)}{P\left(x_i|x_\gamma\right)P\left(x_j|x_\gamma\right)}. \tag{5}$$

The PC algorithm adopts G^2 test statistic, which equals $2n\mathrm{CE}(x_i, x_j|x_\gamma)$ with n indicating the sample size, to verify the independence. The result of this first step is the skeleton of the graph.

(2) Search the v-structures. If two variables x_i and x_j are not conditionally independent with given x_γ, then v_γ is determined as a collider node and a v-structure $v_i \longrightarrow v_\gamma \leftarrow v_j$ is drawn, and the other edges remain undirected $v_i - v_\gamma - v_j$.

(3) Confirm the directions of the rest of the edges. Combining with expert experience, some undirected edges between nodes are specified based on the principles where any cycle and any other v-configuration are not allowed.

2.6. Expectation Maximization (EM) Algorithm. The EM algorithm is a general algorithm to calculate maximal log likelihood and the performance has been proved to be effective in parameter learning of BN [39]. The basic theory of the EM algorithm is to learn the dependence among the nodes by iterating the process of parameters estimation [40]. The EM algorithm mainly consists of three steps:

(1) Initialize θ: Given a set of unknown parameters θ, the value of a log likelihood is maximized. The object function is

$$\ell(\theta; X) = \log P(X|\theta) = \log \sum_Y P(Y, X|\theta). \tag{6}$$

Introduce a distribution $Q(Y)$: an initialization distribution of θ is defined based on Jensen's inequality:

$$\ell(\theta; X) = \log \sum_Y Q(Y) \frac{P(Y, X|\theta)}{Q(Y)} \geq \sum_Y Q(Y) \log \frac{P(Y, X|\theta)}{Q(Y)}. \tag{7}$$

(2) E-Step: Calculate the distribution $Q(Y) = P(Y|X; \theta)$, which is viewed as the E-step.

(3) M-Step: Optimize the parameters based on the estimation of the joint probability distribution, which is viewed as the M-step.

$$\theta' = \arg\max_\theta \sum_Y Q(Y) \log \frac{P(Y, X|\theta)}{Q(Y)}. \tag{8}$$

θ' replaces θ. The iterations process would be repeated until a local optimum of the estimated parameters is achieved.

2.7. Multilayer Perceptron (MLP). The neural network, an effective function approximator, is often used to solve regression and prediction problems in various fields. A general multilayer perceptron model can be performed by the following 3 steps.

(1) Initialize the MLP model. Assume that the original function can be approximated by a set of basic functions:

$$F(x) = \sum_{i=1}^{m} w_i \varphi_i(x) + e, \tag{9}$$

where F is the original function, x is the input vector, m is the number of network synapses, w is the weight of synapses, φ is the basis function allocated on synapses, commonly used functions with S-shaped curves (such as tan$h$), and e is the error.

(2) Load the sample point pair (x, y) and calculate the error between the predicted and true values:

$$e = \sum_i w_i \varphi_i(x) - y, \tag{10}$$

where y is the true value of the sample.

(3) Adjust network synapse weights according to error feedback. The general calculation formula of the adjustment is

$$\Delta w = \varphi'(x) \sum_k w_k e_k, \tag{11}$$

where k is the layer after which the neuron to be adjusted is located. When the adjustment Δw is less than a preset threshold value η, this step would be terminated; otherwise, the weights would be updated and the process would go back to Step (2).

3. Results and Discussion

The DBN-based RTCPM was constructed based on the training dataset (involving 264 crash data and noncrash data) and validated based on the validation dataset (involving 64 crash data and noncrash data).

3.1. Variable Selection. Figure 2 shows the variable importance ranking which was determined by RF method. It is clear that the most important three variables were the TPI (0.151), V (0.144), and U_V (0.144) (VI >0.14). The relative importance of other variables was less than 0.12. Therefore, the TPI, V, and U_V were selected as influencing variables to construct the DBN model. It is surprising that the visibility and weather played limited roles. It is probably explained by the collection time of the crash data (from August to September), when there were more daylight and less visibility differences (mean = 21.32 km, SD = 10.90 km). According to the Horizontal Visibility Grading of Chinese Standard (GB/T 33673–2017), when the visibility is greater than or equal to 10 km, the visual field is considered as a good level. Therefore, the overall good visibility did not contribute a lot to crash risk in this study. In addition, a classified weather variable, the weather type (rainy/sunny), was used as the

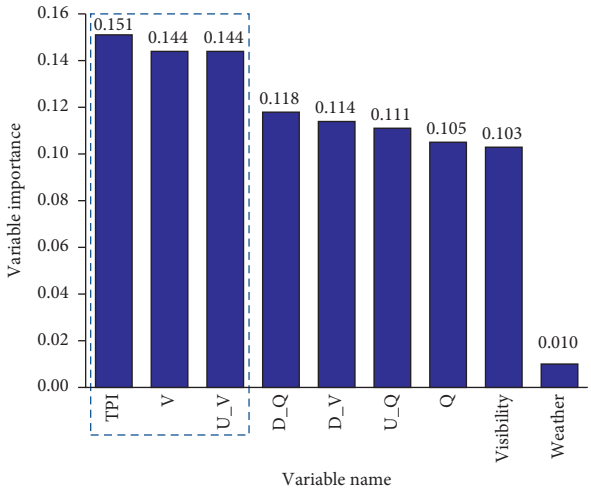

Figure 2: Variable importance ranking determined by Random Forest.

proxy to represent the weather condition in this study, rather than a quantized variable, rainfall. We assume that the relationship between the crash risk and rainfall might be more obvious than the weather type.

3.2. DBN Model Construction. The DBN models with and without considering the interdependence among traffic flow characteristics (TPI, V, and U_V) were both constructed based on the training dataset. The former model (the improved DBN-based RTCPM) was the main purpose, and the latter one (the original DBN-based RTCPM) was developed for comparison. Before constructing the graphical structure of the improved DBN-based RTCPM, the three traffic flow characteristics were discretized according to their corresponding crash cases using the ChiMerge algorithm. The number of discretization states of every variable was confined to 10 so that the calculation complexity in DBN models can be decreased. The discretization ranges of TPI, V, and U_V are presented in Figures 3–5, respectively. The results showed that the adjacent discretization intervals in every variable were characterized by distinguishable crash/noncrash ratio, indicating that the ChiMerge algorithm performs a good discretization.

After discretization, the PC algorithm and expert assessment were utilized to investigate the interdependency among traffic flow characteristics within one time slice. The dynamicity of traffic flow characteristics was reflected by connecting the same variables from time slice 2 to time slice 1. The dynamicity and interdependency determined the graphical structure of the improved DBN-based RTCPM (Figure 6). The original DBN-based RTCPM did not consider the interdependency among traffic flow characteristics, and its graphical structure was directly determined by connecting the traffic flow variables to crash risk within one time slice and connecting the same variables between two time slices (Figure 7).

Afterwards the parameter learning process was implemented using the EM algorithm. The initial states of the

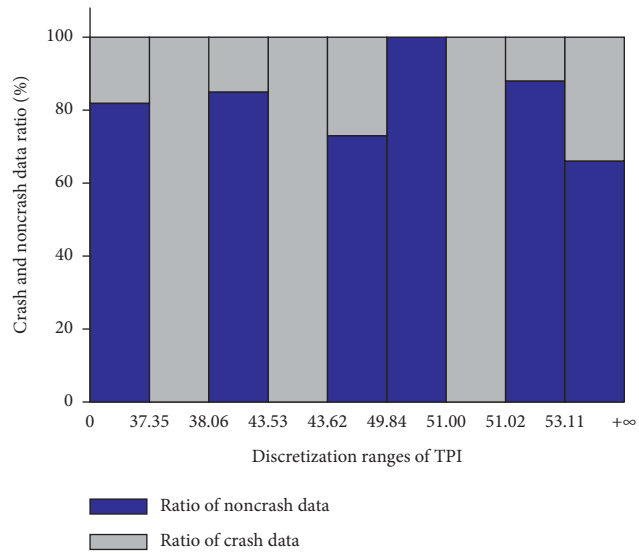

Figure 3: Discretization ranges of TPI.

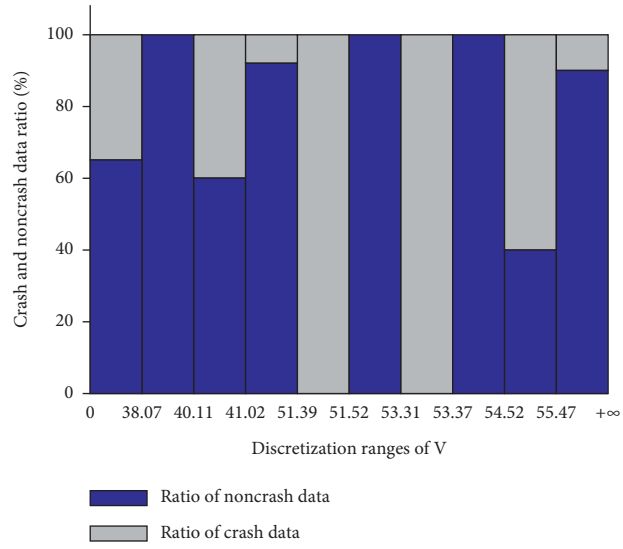

Figure 4: Discretization ranges of V.

improved DBN-based RTCPM and the original DBN-based RTCPM are presented in Figures 8 and 9, respectively. It was observed that their overall probabilities of a traffic flow state being associated with a crash were different (36% and 42%, respectively) when no new evidence was entered into the DBN. This difference suggested the importance of comparing the performance of the two types of DBN-based RTCPM.

3.3. Model Validation and Comparison. The validation dataset was used to validate the DBN models. When no new evidence was entered into the DBN, the marginal probability of crash risk node of initial state of DBN model was set as the classification threshold for evaluating the model performance. And then, each validation dataset was entered individually in the models. The crash risks, i.e., the posterior

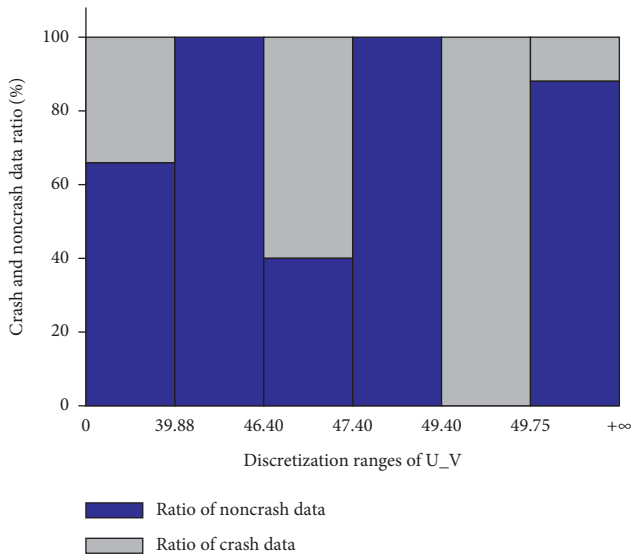

Figure 5: Discretization ranges of U_V.

probability of crash risk node, relating to the prone traffic condition, were calculated based on the prior probabilities. Several evaluation metrics based on the confusion matrix (Table 2) are presented in the following equations:

$$\text{overall\_accuracy} = \frac{T_{\text{crash}} + T_{\text{noncrash}}}{T_{\text{crash}} + F_{\text{crash}} + F_{\text{noncrash}} + T_{\text{noncrash}}}, \tag{12}$$

$$\text{sensitivity} = \frac{T_{\text{crash}}}{(T_{\text{crash}} + F_{\text{noncrash}})}, \tag{13}$$

$$\text{specificity} = \frac{T_{\text{noncrash}}}{(T_{\text{noncrash}} + F_{\text{crash}})}, \tag{14}$$

$$\text{precision} = \frac{T_{\text{crash}}}{(T_{\text{crash}} + F_{\text{crash}})}, \tag{15}$$

$$\text{recall} = \frac{T_{\text{crash}}}{(T_{\text{crash}} + F_{\text{noncrash}})}, \tag{16}$$

$$F - \text{measure} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}, \tag{17}$$

$$G - \text{means} = \sqrt{\text{sensitivity} * \text{specificity}}. \tag{18}$$

Besides the overall accuracy from equation (12), the sensitivity from equation (13), G-means from equation (17), and F-measure from equation (18) were used to compare the performance of two types of DBNs and MLP. For imbalanced classification, the overall accuracy metric is not sufficient due to its inability to examine the minor positive samples; thus, the sensitivity was chosen as the supplementary metric to examine the crash classification accuracy. The F-measure is the harmonic mean of precision and recall and represents the ability to detect crashes. Furthermore, the balanced classification ability can be reflected by G-means, which is the geometric mean of sensitivity and specificity. The comparison results are presented in Table 3.

Table 2: Confusion matrix.

	Predicted crashes	Predicted noncrashes
Actual crashes	T_{crash}	F_{noncrash}
Actual noncrashes	F_{crash}	T_{noncrash}

As illustrated by Table 3, all the models can reach a good classification accuracy. Among them, the improved DBN-based RTCPM showed the best overall classification accuracy, followed by the original DBN-based RTCPM and MLP-based RTCPM. For the crash detection ability, the sensitivity metric indicated that the improved DBN-based RTCPM performed the best, and the relatively poor performance was seen in the original DBN-based RTCPM. Furthermore, the F-measure also suggested that the improved DBN-based RTCPM had better crash prediction ability than the original DBN-based RTCPM and MLP-based RTCPM. With respect to the balanced classification ability, the G-means revealed that the improved DBN-based RTCPM achieved better than the other models. For all the comparisons, the results demonstrated that the improved DBN-based RTCPM can achieve desirable overall prediction performance. It is also demonstrated that this model had an effective ability to monitor crashes in real time. Meanwhile, the model can keep the balance between crash and no-crash prediction. In summary, the prediction performance of DBN-based RTCPM can be improved by accounting for the interdependence of traffic flow characteristics.

3.4. Sensitivity and Strength of Influences Analysis. Investigation of the interdependency among the traffic flow also contributes to revealing the underlying mechanism of crash occurrence, which is helpful for formulating the real-time risk control measures. The sensitivity and strength of influences analysis were implemented in a professional DBN analysis software, Genie, to identify the most significant contributors to crash risk and the most probable risk propagation path.

3.5. Sensitivity Analysis. The sensitivity analysis of Genie can be utilized to identify which node had greater contribution to the target node in DBN. Setting the crash risk as the target node, conducting sensitivity analysis on it, and the contribution degrees of traffic flow characteristics on crash risks are presented in Figure 10 in a descending order. The results showed that the TPI in time slice 2 was the most sensitive factor that results in crash risk, followed by V in time slice 2, TPI in time slice 1, etc.

3.6. Strength of Influences Analysis. The strength of influences analysis was utilized to identify the most probable risk propagation path based on the improved interdependency structure. The strength of influence is always calculated from the distance between the probability distributions of the

Table 3: Performance comparison of two types of DBNs and MLP.

	Overall accuracy	Sensitivity	*F*-measure	*G*-means
Original DBN-based RTCPM	0.750	0.313	0.385	0.529
Improved DBN-based RTCPM	0.766	0.688	0.595	0.738
MLP-based RTCPM	0.725	0.556	0.476	0.656

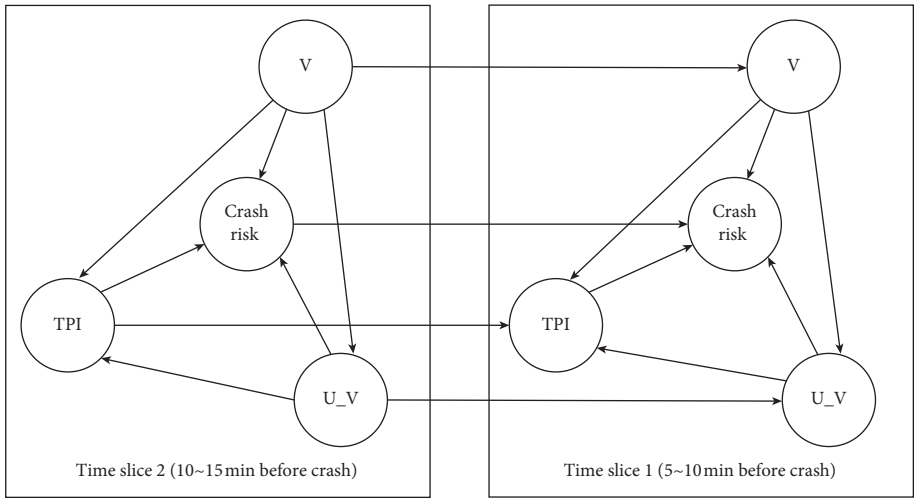

Figure 6: Graphical structural of the improved DBN-based RTCPM.

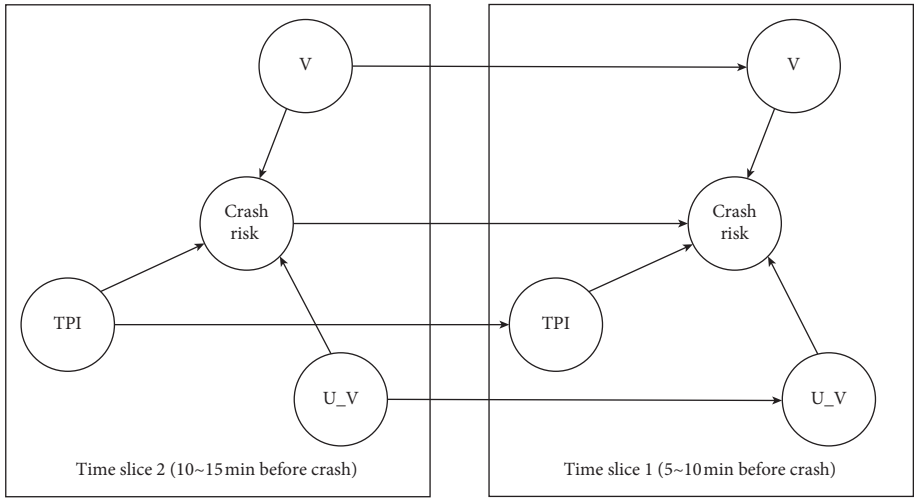

Figure 7: Graphical structural of the original DBN-based RTCPM.

child node conditional on the state of its parent node. As shown in Figure 11, the arcs have different values and thicknesses, presenting the strength of influence between connected nodes. The biggest accumulative value indicates that the most probable risk propagation path is V (time slice 2)⟶V (time slice 1)⟶U_V (time slice 1)⟶TPI (time slice 1)⟶crash risk on current segment.

Synthesizing the results of sensitivity and strength of influences analysis can be used to identify the most probable risk propagation path, as well as determine the most sensitive contributor in the propagation path. The results suggested that the sequence and emphasis of the real-time risk countermeasures should sequentially lay on V (time slice 2), TPI (time slice 1), V (time slice 1), and U_V (time slice 1).

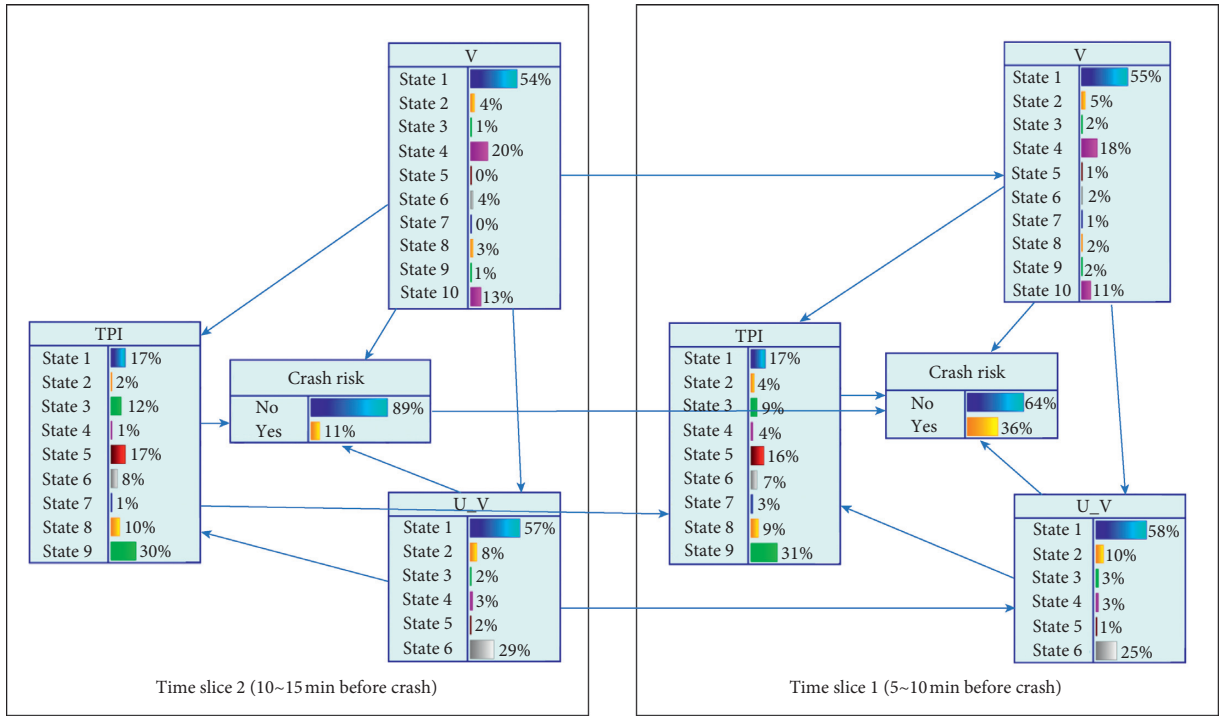

Figure 8: Initial state of the improved DBN-based RTCPM.

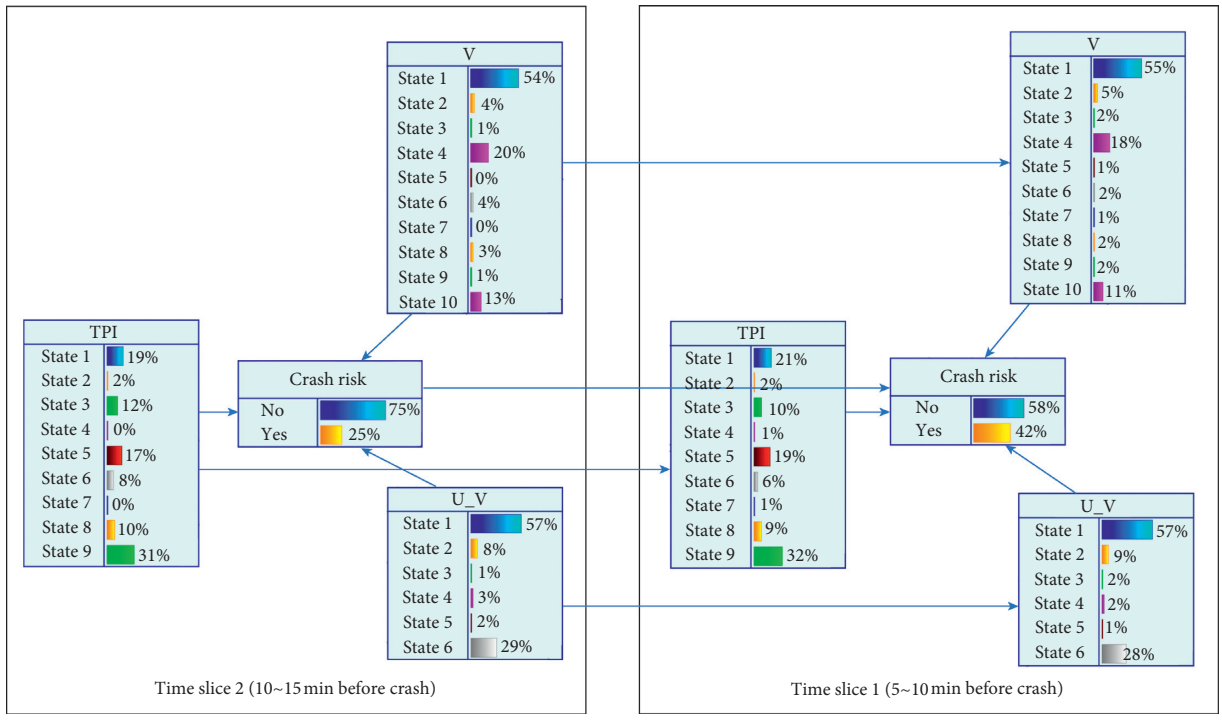

Figure 9: Initial state of the original DBN-based RTCPM.

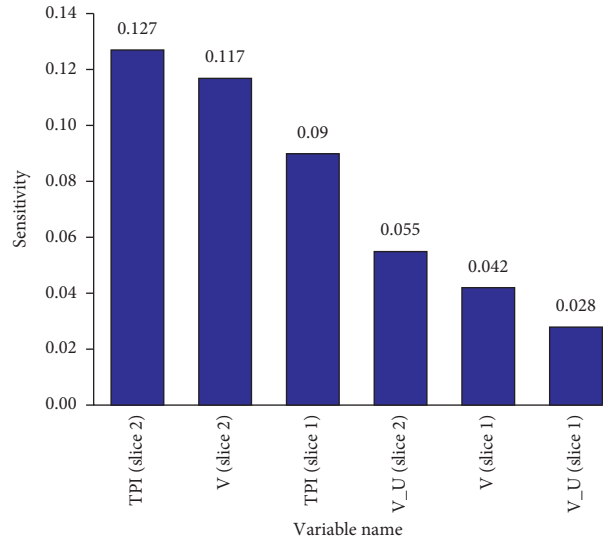

Figure 10: Results of sensitivity analysis.

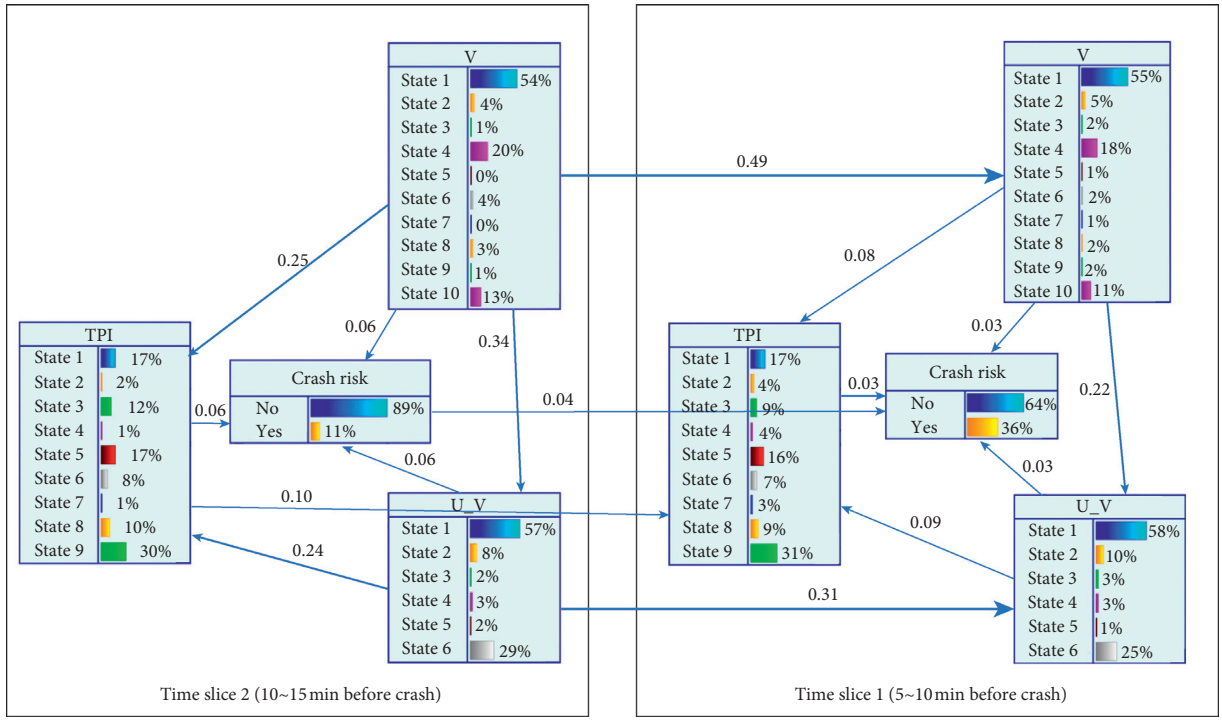

Figure 11: Results of strength of influences analysis.

4. Conclusions

This study aimed to build a RTCPM for urban elevated expressway by using the DBN model to capture the dynamicity and coupling interdependency among traffic flow characteristics before crash occurrence. The model was built and validated adopting traffic flow data collected on the Yan'an elevated expressway. Based on the DBN-based RTCPM, the sensitivity and strength of influences analysis were utilized to identify the most probable risk propagation path and the most sensitive contributors to crash risk. The main conclusions are as follows:

(1) In model construction process, interdependency in the DBN model was determined by the PC algorithm and expert experience, and the dynamicity of traffic flow characteristics was expressed by adopting data in time slices. By validation, the improved DBN-based RTCPM got an overall accuracy of 76.6%, with a crash prediction accuracy of 68.8% and a crash/noncrash balanced classification accuracy of 73.8%. The results indicated that the model can achieve an effective crash prediction for urban elevated expressway.

(2) Comparisons of the original DBN-based RTCPM and MLP-based RTCPM suggested that the improved DBN-based RTCPM can identify the interdependency among traffic flow characteristics before crash occurrences. The comparison results also indicated that the improved DBN-based RTCPM was more suitable for the prediction of real-time crashes for urban elevated expressway.

(3) According to the results of sensitivity and strength of influences analysis, the most probable risk propagation path is V (time slice 2)⟶V (time slice 1)⟶U_V (time slice 1)⟶TPI (time slice 1)⟶crash risk on current segment, and the most sensitive contributor to crash risk in this path is V (time slice 2), followed by TPI (time slice 1), V (time slice 1), and U_V (time slice 1). The results suggested that the formulation of the real-time risk countermeasures should sequentially focus on this sequence in the propagation path.

There would be two extensions in future research. On the one hand, the model was built and validated on the same urban elevated expressway; thus, the transferability of the model to another urban elevated expressway has not been discussed. On the other hand, the specific real-time risk countermeasures such as variable speed limit (VSL) can be investigated to improve crash risk.

Data Availability

The research data are available in the .CSV format file. They are available from the corresponding author upon request.

Conflicts of Interest

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

Acknowledgments

This study was supported by the National Natural Science Foundation of China (Grant no. 52072071).

References

[1] C. Lee, B. Hellinga, F. Saccomanno et al., "Real-time crash prediction model for application to crash prevention in freeway traffic," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1840, no. 1, pp. 67–77, 2003.

[2] M. Abdel-Aty, N. Uddin, A. Pande, M. F. Abdalla, and L. Hsia, "Predicting freeway crashes from loop detector data by matched case-control logistic regression," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1897, no. 1, pp. 88–95, 2004.

[3] M. Abdel-Aty, N. Uddin, A. Pande et al., "Split models for predicting multivehicle crashes during high-speed and low-speed operating conditions on freeways," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1908, no. 1, pp. 51–58, 2005.

[4] Q. Cai, M. Abdel-Aty, J. Yuan et al., "Real-time crash prediction on expressways using deep generative models," *Transportation Research Part C-Emerging Technologies*, vol. 117, 2020.

[5] P. Li, M. Abdel-Aty, and J. Yuan, "Real-time crash risk prediction on arterials based on LSTM-CNN," *Accident Analysis and Prevention*, vol. 135, 2020.

[6] C. Xu, W. Wang, P. Liu, R. Guo, and Z. Li, "Using the Bayesian updating approach to improve the spatial and temporal transferability of real-time crash risk prediction models," *Transportation Research Part C: Emerging Technologies*, vol. 38, pp. 167–176, 2014.

[7] R. Yu, X. Wang, K. Yang, and M. Abdel-Aty, "Crash risk analysis for Shanghai urban expressways: a Bayesian semi-parametric modeling approach," *Accident Analysis & Prevention*, vol. 95, pp. 495–502, 2016.

[8] Y. Guo, T. Sayed, L. Zheng, and M. Essa, "An extreme value theory based approach for calibration of microsimulation models for safety analysis," *Simulation Modelling Practice and Theory*, vol. 106, p. 102172, 2021.

[9] Y. Guo, T. Sayed, and L. Zheng, "A hierarchical bayesian peak over threshold approach for conflict-based before-after safety evaluation of leading pedestrian intervals," *Accident Analysis & Prevention*, vol. 147, p. 105772, 2020.

[10] Y. Guo, T. Sayed, and M. Essa, "Real-time conflict-based Bayesian Tobit models for safety evaluation of signalized intersections," *Accident Analysis & Prevention*, vol. 144, p. 105660, 2020.

[11] Y. Guo, P. Liu, Y. Wu, and J. Chen, "Evaluating how right-turn treatments affect right-turn-on-red conflicts at signalized intersections," *Journal of Transportation Safety & Security*, vol. 12, no. 3, pp. 419–440, 2020.

[12] M. Abdel-Aty, A. Pande, A. Das, and W. J. Knibbe, "Assessing safety on Dutch freeways with data from infrastructure-based intelligent transportation systems," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2083, no. 1, pp. 153–161, 2008.

[13] M. Hossain and Y. Muromachi, "A Bayesian network based framework for real-time crash prediction on the basic freeway segments of urban expressways," *Accident Analysis & Prevention*, vol. 45, pp. 373–381, 2012.

[14] D.-f. Chen, F. Luo, and Y. Feng, "Analysis of flight safety risk coupling based on fuzzy sets and complex network," in *Proceedings of 2013 International Conference on Management Science and Engineering*, pp. 329–334, Islamabad, Pakistan, November 2013.

[15] T. Liu, F. Luo, and G. Yao, *Mathematical Analysis of Air Traffic Control Safety Risk Coupling*, 2012.

[16] F. Zheng, M.-g. Zhang, J. Song, and F.-z. Chen, "Analysis on risk of multi-factor disaster and disaster control in oil and gas storage tank," *Procedia Engineering*, vol. 211, pp. 1058–1064, 2018.

[17] J. Wang, J. Wu, X. Zheng, D. Ni, and K. Li, "Driving safety field theory modeling and its application in pre-collision warning system," *Transportation Research Part C: Emerging Technologies*, vol. 72, pp. 306–324, 2016.

[18] C. Xu, W. Wang, and P. Liu, "A genetic programming model for real-time crash prediction on freeways," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 574–586, 2013.

[19] J. Sun, J. Sun, and P. Chen, "Use of support vector machine models for real-time prediction of crash risk on urban expressways," *Transportation Research Record*, vol. 2432, no. 1, pp. 91–98, 2018.

[20] N. E. Fenton and M. Neil, *Risk Assessment and Decision Analysis with Bayesian Networks*, Taylor & Francis, Milton Park, ML, USA, 2013.

[21] M. Jiang and J. Lu, "Maritime accident risk estimation for sea lanes based on a dynamic Bayesian network," *Maritime Policy & Management*, vol. 47, no. 5, pp. 649–664, 2020.

[22] B. Khan, F. Khan, and B. Veitch, "A dynamic bayesian network model for ship-ice collision risk in the Arctic waters," *Safety Science*, vol. 130, 2020.

[23] X. Wu, H. Liu, L. Zhang, M. J. Skibniewski, Q. Deng, and J. Teng, "A dynamic Bayesian network based approach to safety decision support in tunnel construction," *Reliability Engineering & System Safety*, vol. 134, pp. 157–168, 2015.

[24] M. Hossain and Y. Muromachi, "A real-time crash prediction model for the ramp vicinities of urban expressways," *Iatss Research*, vol. 37, no. 1, pp. 68–79, 2013.

[25] A. Roy, M. Hossain, and Y. Muromachi, "Enhancing the prediction performance of real-time crash prediction models: a cell transmission-dynamic bayesian network approach," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2672, no. 38, pp. 58–68, 2018.

[26] A. Roy, R. Kobayashi, M. Hossain et al., "Real-time crash prediction model for urban expressway using dynamic bayesian network," *Journal of Japan Society of Civil Engineers Ser D3*, vol. 72, no. 5, pp. I_1331–I_1338, 2016.

[27] J. Sun and J. Sun, "A dynamic Bayesian network model for real-time crash prediction using traffic speed conditions data," *Transportation Research Part C: Emerging Technologies*, vol. 54, pp. 176–186, 2015.

[28] M. Hossain, M. Abdel-Aty, M. A. Quddus, Y. Muromachi, and S. N. Sadeek, "Real-time crash prediction models: state-of-the-art, design pathways and ubiquitous requirements," *Accident Analysis & Prevention*, vol. 124, pp. 66–84, 2019.

[29] Y. F. Wang, T. Qin, B. Li, X. F. Sun, and Y. L. Li, "Fire probability prediction of offshore platform based on Dynamic Bayesian Network," *Ocean Engineering*, vol. 2017, 18 pages, Article ID 2525481, 2017.

[30] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[31] A. Karimnezhad and F. Moradi, "Road accident data analysis using Bayesian networks," *Transportation Letters The International Journal of Transportation Research*, vol. 9, no. 1, pp. 12–19, 2015.

[32] T. Tang, S. Zhu, Y. Guo et al., "Evaluating the safety risk of rural roadsides using a bayesian network method," *International Journal of Environmental Research and Public Health*, vol. 16, no. 7, 2019.

[33] X. Zou and W. L. Yue, "A bayesian network approach to causation analysis of road accidents using netica," *Journal of Advanced Transportation*, 2017.

[34] D. Geiger and D. Heckerman, "Learning Gaussian networks," *Uncertainty Proceedings 1994*, pp. 235–243, 1994.

[35] S. Monti and G. F. Cooper, "A multivariate discretization method for learning bayesian networks from mixed data," arXiv preprint arXiv:1301.7403, 2013.

[36] R. Kerber, "ChiMerge: discretization of numeric attributes," in *Proceedings of the 10th National Conference on Artificial Intelligence*, San Jose, CA, USA, July 1992.

[37] M. Kalisch and P. Buehlmann, "Estimating high-dimensional directed acyclic graphs with the PC-algorithm," *Journal of Machine Learning Research*, vol. 8, pp. 613–636, 2007.

[38] F. Musella, "A PC algorithm variation for ordinal variables," *Computational Statistics*, vol. 28, no. 6, pp. 2749–2759, 2013.

[39] P. B. G. Lindsay, "Alternative EM methods for nonparametric finite mixture models," *Biometrika*, vol. 88, no. 2, pp. 535–550, 2001.

[40] S. L. Lauritzen, "The EM algorithm for graphical association models with missing data," *Computational Statistics & Data Analysis*, vol. 19, no. 2, pp. 191–201, 1995.