Accident Analysis and Prevention 148 (2020) 105844

Contents lists available at ScienceDirect

Accident Analysis and Prevention

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

Modeling pedestrians' near-accident events at signalized intersections using gated recurrent unit (GRU)☆

Shile Zhang *, Mohamed Abdel-Aty, Yina Wu, Ou Zheng

Department of Civil, Environmental and Construction Engineering, University of Central Florida, Orlando, FL, 32816, United States

ARTICLE INFO

Keywords:
Pedestrian safety
Surrogate safety measures (SSMs)
Extreme value theory (EVT)
Gated recurrent unit (GRU)

ABSTRACT

Pedestrian safety plays an important role in the transportation system. Intersections are dangerous locations for pedestrians with mixed traffic. This paper aims to predict the near-accident events between pedestrians and vehicles at signalized intersections using PET (Post Encroachment Time) and TTC (Time to Collision). With automated computer vision techniques, mobility features of pedestrians and vehicles are generated. Extreme Value Theory (EVT) is used to model PET and minimum TTC values to select the most appropriate threshold values to label pedestrians' near-accident events. A Gated Recurrent Unit (GRU) neural network is further used to predict these events. The established model reaches an AUC (Area Under the Curve) value of 0.865 on the test data set. Moreover, the proposed model can also be applied to develop collision warning systems under the Connected Vehicle environment.

1. Introduction

Globally, traffic accidents are estimated to be the eighth leading cause of deaths for people of all age groups. In the U.S., pedestrians make for 11.8 % of all road fatalities (Centers for Disease Control and Prevention, 2019). In 2019, 6590 pedestrian deaths were caused due to traffic accidents, ranked the highest for the last three decades (Governors Highway Safety Association, 2020). As accident reports can miss important details of the mechanisms of accidents, surrogate safety measures (SSMs) were widely used as a proactive approach for analyzing pedestrians' safety conditions (Tarko et al., 2009; Ismail et al., 2010; Zaki and Sayed, 2014). SSMs have multiple indicators, which are also referred to as traffic conflict indicators. The interaction between two road users, as a simultaneous arrival in a specific limited area, is defined as an "event" (Hydén (1987). As shown in Fig. 1, this pyramid describes the relationships between traffic events' severity and their frequency of occurrence. The events at the top of the pyramid are traffic accidents, such as fatal accidents, injury accidents, and PDO (Property Damage Only) accidents. Accidents are rare events and the higher the severity the lowest the frequency. Moreover, most of the interactions are safe passages, which are located at the bottom (also the most significant part) of the pyramid. Between safe passages and accidents are traffic conflicts, which can be used to describe the "near-accident" events (Hydén, 1987). Depending on the severity, the conflicts can be categorized into severe conflicts, slight conflicts, and potential conflicts. There are various indicators for traffic conflicts, such as Time to Accident (TA) (Hydén and Linderholm, 1984), Post Encroachment Time (PET) (Cooper, 1984; Ismail et al., 2010), Time to Collision (TTC) (Ismail et al., 2009; Tarko, 2019), Gap Time (GT) (Archer, 2004; Ismail et al., 2010), etc. The details of these indicators are listed in Table 1. Most of the indicators consist of continuous values during the whole interaction course, such as TTC, GT, and DST, while PET and TA use discrete values to describe the interaction.

A single indicator cannot accurately fully reflect the actual situation. For example, small TTC values can be observed when either one of the road users takes evasive actions at the urgent situations, or when a vehicle moves towards a pedestrian at a very low speed (Ni et al., 2016). The combination of different indicators (Ni et al., 2016; Kathuria and Vedagiri, 2020), or new indicators such as using motion predictions were proposed, to complement each other and better predict pedestrian safety conditions (Mohamed and Saunier, 2013). Among these insights to use multiple conflict indicators, it was found there was strong correlation between GT and PET, while no correlation between TTC and other indicators (Ismail et al., 2010). Kathuria and Vedagiri (2020)

☆ This paper has been handled by associate editor Tony Sze.

* Corresponding author.
E-mail addresses: shirleyzhang@Knights.ucf.edu (S. Zhang), m.aty@ucf.edu (M. Abdel-Aty), jessicawyn@Knights.ucf.edu (Y. Wu), ouzheng1993@gmail.com (O. Zheng).

https://doi.org/10.1016/j.aap.2020.105844
Received 28 July 2020; Received in revised form 8 October 2020; Accepted 11 October 2020
Available online 28 October 2020

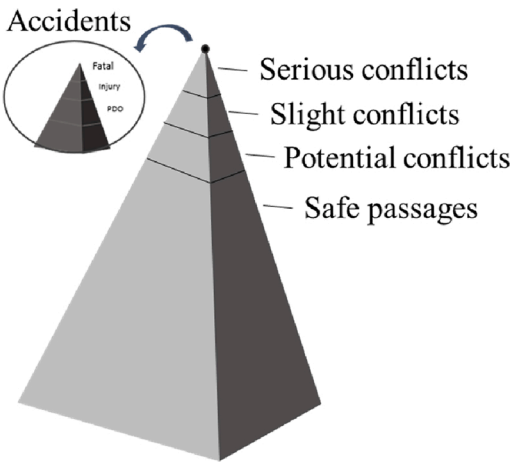


Fig. 1. The pyramid - interactions between road users as a continuum of events (Hydén, 1987; Tarko, 2012).

Table 1
Definitions of traffic conflict indicators (Kathuria and Vedagiri, 2020).

Indicator	Definition	Type of indicator
Time to Collision (TTC)	The time required for two road users to collide if they continue at their present speeds and on the same paths (Hayward, 1972)	A set of values continually calculated over time
Post Encroachment Time (PET)	The time difference between the moment an offending road user leaves the area of a potential collision and the moment of arrival of the conflicting road user (Cooper, 1984)	A discrete value
Gap Time (GT)	The time lapse between the first road user leaves the conflict zone and the second road user arrives if they continue with the same velocity and trajectory (Archer, 2004)	A set of values continually calculated over time
Deceleration to Safety Time (DST)	Deceleration required for the second road user to reach the conflict zone no earlier than the first user leaves it (Hupfer, 1997)	A set of values continually calculated over time
Time to Accident (TA)	The time that remains to an accident happening from the moment when one of the road users starts an evasive action (Hydén and Linderholm, 1984)	A discrete value

divided different interaction patterns according to the profiles of PET and TTC. Ni et al. (2016) used observers' rating-based approach to classify the interaction patterns into three severity categories, using the values of TTC and GT. However, most of the existing studies failed to do so, and used a single indicator or a static value to represent the interaction course instead.

Upon selecting the threshold values to identify pedestrians' near-accident situations, some studies investigated different threshold values of the SSMs indicators (Mahmud et al., 2017; Borsos et al., 2020). Among them, the Extreme Value Theory (EVT) was used. The EVT model could be employed to model the stochastic behavior of a process with unusually large or small levels (Coles et al., 2001). The fitted distributions of SSMs indicators could be used to estimate the frequency of accidents (Tarko, 2018; Zheng and Sayed, 2019b). Upon selecting the best fitted models, the desirable indicators and the threshold values could be decided (Songchitruksa and Tarko, 2006; Zheng and Sayed, 2019a; Borsos et al., 2020).

The interactions between pedestrians and vehicles are usually complicated, as either party can change the course or the speed suddenly (Ni et al., 2016; Kathuria and Vedagiri, 2020; Yue et al., 2020). Compared with other road sensors such as loop and Bluetooth sensors, video sensors show advantages with a more microscopic view. A few studies used video sensors to analyze SSMs indicators (Zaki and Sayed, 2014; Wu et al., 2020). With the development of automated video processing techniques, it is possible to generate trajectory profiles of the road users in time series (Ismail et al., 2010; Zaki and Sayed, 2014; Wu et al., 2020; Zhang et al., 2020). These can be better used to predict the pedestrians' situations, using the continuous values of pedestrians' mobility features.

Deep learning techniques have been applied to the transportation field with various applications, such as computer vision, time series prediction, classification, and optimization problems (Wang et al., 2019b; Gong et al., 2020; Li et al., 2020). The most popular types of deep learning techniques, neural network models, can employ multiple layers and convolutional layers to extract the high-level features from the data. For sequential data-related problems such as handwriting recognition, speech recognition, and natural language processing, the recurrent neural networks are used with internal memory cells to exhibit the temporal dynamic features from the input data (Graves et al., 2020; Cho et al., 2014). Recurrent neural networks have many variants such as long short-term memory neural network (LSTM), bidirectional recurrent neural network, gated recurrent unit (GRU), etc. (Hochreiter and Schmidhuber, 1997; Schuster and Paliwal, 1997; Cho et al., 2014). The LSTM neural networks can capture long-term dependencies of time series data by introducing a special memory unit (Hochreiter and Schmidhuber, 1997). Over the years, it is found that removing some components of the LSTM won't affect the performance of the model. GRU is proposed with a simpler memory unit, which makes it easier to train and implement (Cho et al., 2014). Thus, the training efficiency of GRU is improved as well. However, LSTM can remember longer sequences than GRU. GRU was employed in the transportation field to conduct traffic condition forecasting, and travel time prediction, etc. (Zhao et al., 2018, 2019). It is expected to be a more efficient option to model sequential data, especially trajectory data generated from videos.

This study uses GRU to predict the pedestrians' near-accident events at signalized intersections. With PET and TTC indicators generated from videos, Extreme Value Theory is used to select the best threshold values. The near-accident events of pedestrians are classified into three severity categories. With the sequential data generated from pedestrians' and vehicles' trajectories, the GRU model reaches an AUC (Area Under the Curve) value of 0.865 on the test data set. The proposed model can be used to warn drivers of the potentially dangerous situations involving pedestrians if to be implemented in the Connected Vehicle (CV) environment.

2. Data Collection

The video data used in this study are collected at two signalized intersections (intersection 1: (latitude: 28.800916, longitude: −81.27317); intersection 2: (latitude: 28.809715, longitude: −81.27317)) located in Seminole County, Florida using CCTV (closed-circuit television) cameras. All the videos are captured during 8:00−19:00 on sunny workdays during October and November 2019. A total of around 80-h videos with 150 pedestrians are collected. Automated computer vision techniques, such as object detection, object tracking, and perspective transformation are used to process the video data.

2.1. Video preprocessing

2.1.1. Object detection

The object detection model is used to classify different kinds of road users, i.e., human beings and vehicles. The detection model used in this study is Mask R-CNN (Region-based Convolutional Neural Network), the state-of-the-art automated object detection model (He et al., 2020; Ou

(a) pedestrians' trajectories (1st intersection) (b) vehicles' trajectories (1st intersection)

(c) pedestrians' trajectories (2nd intersection) (d) vehicles' trajectories (2nd intersection)

Fig. 2. Trajectories of the road users from videos.

Zheng, 2019). The Mask R-CNN model is adapted from Faster-RCNN (Ren et al., 2020). The detection model can classify different kinds of objects in a frame and generate a segmentation mask. It first scans the whole image and estimates the areas that are likely to contain an object, then classifies the objects in each crop of these areas. This mechanism ensures the good performance of the detection model, especially on small objects that are usually hard to detect (Ou zheng et al., 2019).

2.1.2. Object tracking

The object tracking model is used to take the initial sets of detection model and track the movements of each road user. The tracking method used is CSRT (DCF-CSR, Discriminative Correlation Filter with Channel and Spatial Reliability) tracker from OpenCV (Open Sources Computer Vision) library (Bradski, 2019; Ou Zheng, 2019; Ou Zheng and Xin, 2019). The detection model and the tracking model are used to generate the trajectories of the road users from videos.

2.1.3. Perspective transformation

As the image is a 2D plane, it is hard to generate the actual locations of the objects from images. To create a mapping between the image coordinates and the world coordinates, perspective transformation is conducted. As shown in Eq. (1), the image coordinate is $(u, v, 1)$, and the world coordinate is $(X, Y, 1)$, a homograph matrix $\boldsymbol{h}$ is the middle matrix to convert the coordinates from image plane to world plane. $\boldsymbol{h}$ matrix contains nine values in total, from h_1 to h_9.

To obtain the $\boldsymbol{h}$ matrix, a linear least-squares method is used (Naphade et al., 2019; Španhel et al., 2019; Tang et al., 2019). As shown in Eq. (2), the coordinates (u_i, v_i) from the image plane, which are the middle points of the bottom lines from the generated bounding boxes, and the coordinates (X_i, Y_i) from the world plane (GPS coordinates in decimal degrees), are used to form matrix $\boldsymbol{A}$. Each pair of points forms two rows of matrix $\boldsymbol{A}$. Singular value decomposition (SVD) method is used to obtain the solution with $h_9 = 1$. Namely, the SVD derives the solution for minimizing the value of $\|\boldsymbol{Ah}\|$. Using the inverse matrix of $\boldsymbol{h}$, the image coordinates (u_i, v_i) can be transformed to (X_i, Y_i). However, since the GPS coordinates are in decimal degrees, it will generate some inaccuracies when calculating distances between different objects due to the curvature of the Earth. To eliminate this effect, the coordinates in GPS are further transformed into a linear coordinate system UTM (Universal Transverse Mercator) system (zone:17).

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \boldsymbol{h} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} = \begin{pmatrix} h_1 & h_4 & h_7 \\ h_2 & h_5 & h_8 \\ h_3 & h_6 & h_9 \end{pmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \tag{1}$$

$$\boldsymbol{A} * \boldsymbol{h} = \begin{bmatrix} 0 & 0 & 0 & -X_1 & -Y_1 & 1 & v_1X_1 & v_1Y_1 & v_1 \\ X_1 & Y_1 & 1 & 0 & 0 & 0 & -u_1X_1 & -u_1Y_1 & -u_1 \\ 0 & 0 & 0 & -X_2 & -Y_2 & 1 & v_2X_2 & v_2Y_2 & v_2 \\ X_2 & Y_2 & 1 & 0 & 0 & 0 & -u_2X_2 & -u_2Y_2 & -u_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \\ h_7 \\ h_8 \\ h_9 \end{bmatrix} = 0 \tag{2}$$

2.2. PET and TTC generations

After object detection, object tracking, and perspective transformation, an example of the generated trajectories of pedestrians and vehicles are shown on Google Maps© as in Fig. 2.

To estimate the near-accident events of pedestrians at the two studied locations, two surrogate safety indicators, PET and TTC during the interactions are used. The PET indicator is defined as the time difference between the moment when the first road user leaves the conflict point and the moment when the second road user reaches it (Allen et al., 1978). And the conflict points, which can be seen as potentially dangerous locations for pedestrians, are extracted from the actual trajectories of the conflicting pedestrians and vehicles. The potential

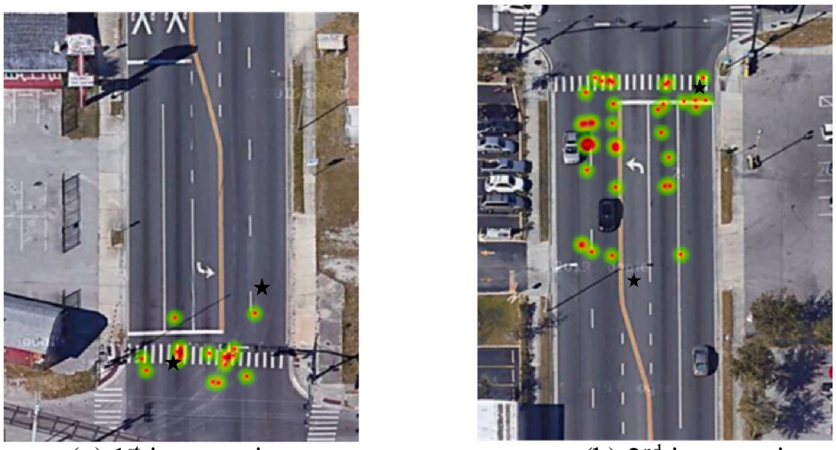


Fig. 3. Spatial distributions of pedestrian-vehicle conflicts points.

conflict points of PET values ranging from 0 to 10 s at the studied intersections are shown in Fig. 3. The black stars are the places where pedestrian-related accidents happened during the past ten years (2010–2020). There were two pedestrian-related accidents happened at each location. The TTC indicator is dynamically calculated using the time difference between the pedestrian and the vehicle before they reach the potential conflict points based on their current speed. Usually, the minimum TTC (mTTC) during an interaction course is used to represent the severity of the interaction (Ni et al., 2016; Kathuria and Vedagiri, 2020). Thus, in this study, both PET and mTTC are used as safety indicators to classify the severity levels of the pedestrian-vehicle interactions.

3. Methodology

3.1. Gated recurrent unit (GRU)

As trajectory data are in time series, the recurrent neural networks can be used to better handle sequential data. Different from traditional neural networks, the output of the recurrent neural network from the current time slice is the input of the next time slice. Suppose x_1, x_2, x_3 are the input vectors, h_1, h_2, h_3 are the hidden state vectors, and y_1, y_2, y_3 are the output vectors. At the time slice t, the hidden state vectors h_t are computed by the input vector x_t, the previous hidden state vector h_{t-1}, and the weights w_t^x and w_t^h. Thus, the output is produced by joining hidden layer vectors together with input from previous time slices. That's why the recurrent neural network can memorize the sequential information lying in the time series data.

However, when dealing with data in a long sequence, recurrent neural network models can suffer from vanishing gradient problems, thus mitigating the ability of the model to learn long-term information (Pascanu et al., 2020). The Gated Recurrent Unit (GRU) is proposed by Cho et al. (2014) to solve the long-term dependency problem of recurrent neural networks. The GRU model consists of two gates, reset gate, and update gate. The update gate controls the previous information that will be carried over to the current layer, while the reset gate decides the amount of information to forget. The equation of the update gate z_t and the reset gate r_t are shown in Eqs. (3) and (4), respectively. The input vector at time t is given by x_t. When x_t is fed into the network unit, it is multiplied by the weight W_z. And h_{t-1} generated from the last hidden layer is multiplied by its weight U_z. The two results are added together and a sigmoid function σ is used as the activation function to generate a probabilistic value between 0 and 1. So the values of update gate z_t and the reset gate r_t are generated. The weight vectors W_z, U_z, W_r, U_r are learned through the training process. $\widetilde{h_t}$ is the memory unit that can store the relevant information using the reset gate r_t. σ is the sigmoid function and $\bigotimes$ is the element-wise product function of two vectors. W_r and U_r are weight matrices that are learned through the process. The hidden layer output h_t at time t is calculated by Eq. (6). These equations are iteratively computed from the first time slice to the last time slice, and finally generate the output of GRU.

$$z_t = \sigma(W_z x_t + U_z h_{t-1}) \tag{3}$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1}) \tag{4}$$

$$\widetilde{h_t} = tanh(W_c x_t + r_t \bigotimes (U h_{t-1})) \tag{5}$$

$$h_t = (1 - z_t) \bigotimes h_{t-1} + z_t \bigotimes \widetilde{h_t} \tag{6}$$

Compared with Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997), the unit of the GRU is simpler to compute and implement (Cho et al., 2014). The training efficiency of GRU gets improved with the number of weights reduced. However, LSTM can remember longer sequences than GRU, as LSTM has a more sophisticated memory cell.

3.2. Threshold selection using extreme value theory (EVT)

To label the pedestrians' near-accident events, the threshold values of SSMs indicators need to be decided. Most of the existing studies use threshold values of mTTC and PET as static values, such as 3 s, 1 s, etc. (Ni et al., 2016; Zheng et al., 2018). Extreme Value Theory (EVT) can be used here for selecting the threshold values (Leadbetter et al., 2012; Wang et al., 2019a). EVT is a branch of statistical models that is capable of modeling the stochastic behaviors of extreme events. These extreme events usually have unusually large or small values, such as the values on the tail of a normal distribution. It has proven to be an effective tool to evaluate traffic safety by comparing model estimates of the SSMs indicators to the actual accident frequencies (Tarko and Songchitruksa, 2020; Farah and Azevedo, 2017; Wang et al., 2018; Orsini et al., 2019; Zheng and Sayed, 2019c; Borsos et al., 2020). Multivariate EVT models using multiple SSMs indicators were also proposed (Fu et al., 2020).

In this study, univariabte EVT models based on either negated PET or negated mTTC are used to select threshold values. EVT offers two approaches to model extreme events, the block maxima (BM) (or minima) approach using Generalized Extreme Value (GEV) distribution, and the Peak over Threshold (POT) approach using Generalized Pareto (GP) distribution. As the typical EVT model is used to estimate the minima of extreme values, here the minima of negated values of PET and mTTC during the pedestrian-vehicle interactions are considered. GEV models and GP models are used for comparison.

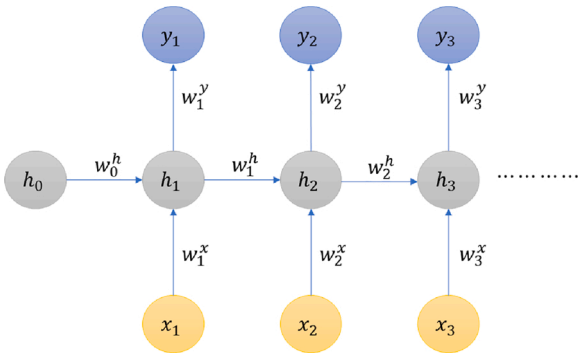


Fig. 4. The architecture of recurrent neural network.

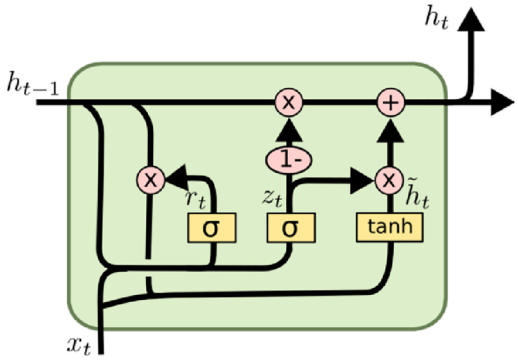


Fig. 5. Schematic of GRU unit (adapted from Cho et al. (2014)).

3.3. BM approach

For BM approach, the observations are aggregated into fixed interval (block). So, every interaction between a pair of pedestrian and vehicle can be a block, and the extremes values of PET or mTTC from each interaction are extracted from every interaction. Assume X is a variable with certain probability distribution. And $x_1, x_2, \ldots, x_n$ are independent random observations from. Let $M = \max(x_1, x_2, \ldots, x_n)$. When $n \to \infty$, the M will converge to a General Extreme Value (GEV) distribution that best illustrates the probabilities of those occurrences of extreme values. The standard GEV distribution is as follows:

$$G(x) = exp\left\{ -\left[1 + \varepsilon\left(\frac{x-\mu}{\sigma}\right)\right]^{-1/\varepsilon} \right\} \quad (7)$$

With $-\infty < \mu < \infty$, $\sigma > 0$ and $-\infty < \varepsilon < \infty$, three parameters are regarded as location parameter (μ), the scale parameter (σ), and the shape parameter (ε). With the threshold value of μ, scale parameter $\sigma_\mu > 0$ (depending on threshold μ), and shape parameter $-\infty < \varepsilon < \infty$. When the shape parameter ε is equal to 0, the GEV tends to a Gumbel distribution; when $\varepsilon > 0$, GEV tends to the Frechet distribution; when $\varepsilon < 0$, GEV tends to a Weibull distribution (Figs. 4 and 5).

Table 2
Modeling results for negated PET with different threshold values.

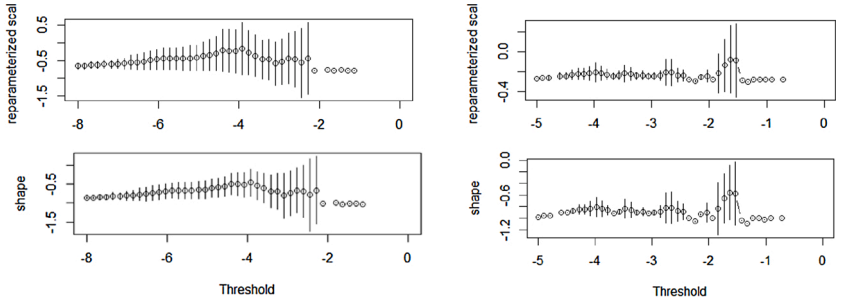
Model	Threshold (negated PET)	−7	−6	−5
	Sample size	73	67	34
GEV	**Location parameter μ (Standard error)**	−4.540 (0.182)	**−4.319 (0.168)**	−3.792 (0.138)
	Scale parameter σ (Standard error)	1.3644 (0.133)	**1.190 (0.124)**	0.840 (0.104)
	Shape parameter ε (Standard error)	−0.233 (0.097)	**−0.171 (0.110)**	−0.007 (0.139)
	Probability of accident (-*PET* > 0)	0.0017	**0.0034**	0.0101
	AIC (Akaike information criterion)	263.425	**228.786**	152.340
	BIC (Bayesian information criterion)	270.296	**235.400**	158.194
	Negative Log-Likelihood Value	128.713	**111.392**	73.170
	Kolmogorov-Smirnov test *p*-value	0.996	**0.992**	0.998
GP	**Scale parameter σ (Standard error)**	5.186 (0.299)	3.656 (0.490)	**2.741 (0.440)**
	Shape parameter ε (Standard error)	−0.828 (0.0452)	−0.688 (0.101)	**−0.628 (0.119)**
	Probability of accident (-*PET* > 0)	0.076	0.051	**0.047**
	AIC (Akaike information criterion)	269.362	219.566	**147.529**
	BIC (Bayesian information criterion)	273.943	223.976	**151.431**
	Negative Log-Likelihood Value	132.681	107.783	**71.764**
	Kolmogorov-Smirnov test *p*-value	0.500	0.726	**0.734**

3.4. POT approach

For POT approach, an event is identified as an extreme case if it exceeds a predefined threshold μ. With $\sigma > 0$ and $-\infty < \varepsilon < \infty$, the threshold excesses $(x - \mu)$ will converge to a GP distribution:

$$G(x) = 1 - \left[1 + \left(\frac{\varepsilon*(x-\mu)}{\sigma}\right)\right]^{-1/\varepsilon} \quad (8)$$

For POT approach, the threshold values need to be predefined. And two parameters, scale parameter σ and shape parameter ε are to be estimated. The parameter stability plots can be used to determine the threshold values. Fig. 6(a) shows the parameter stability plots of negated PET for reparameterized scale ($\sigma^* = \sigma - \varepsilon\mu$) and shape ($\varepsilon$). A threshold value of about –7 or –5 seems appropriate, as σ^* and ε parameters seem to be stable within the ranges near -–7 and –5, as the parameters are



(a) GP model for PET (b) GP model for mTTC

Fig. 6. Parameter stability plots of PET and TTC.

Table 3
Modeling results for negated mTTC with different threshold values.

Model	Threshold (negated mTTC)	−5	−4	−3
	Sample size	110	96	63
GEV	**Location parameter μ (Standard error)**	−3.016 (-0.136)	−2.723 (0.126)	**−1.986 (0.105)**
	Scale parameter σ (Standard error)	1.231 (0.105)	1.038 (0.099)	**0.713 (0.080)**
	Shape parameter ε (Standard error)	−0.341 (0.093)	−0.287 (0.112)	**−0.271 (0.125)**
	Probability of accident (-*mTTC* > 0)	0.010	0.0076	**0.0056**
	AIC (Akaike information criterion)	358.857	287.411	**144.132**
	BIC (Bayesian information criterion)	366.959	295.105	**150.562**
	Negative Log-Likelihood Value	176.429	140.706	**69.066**
	Kolmogorov-Smirnov test *p*-value	0.754	0.677	**0.836**
GP	**Scale parameter σ (Standard error)**	4.694 (2e-8)	3.073 (0.288)	**2.475 (0.059)**
	Shape parameter ε (Standard error)	−0.994 (2e-8)	−0.820 (0.079)	**−0.906 (0.018)**
	Probability of accident (-m *TTC* > 0)	0.057	0.037	**0.077**
	AIC (Akaike information criterion)	345.640	254.126	**130.052**
	BIC (Bayesian information criterion)	351.040	259.254	**134.339**
	Negative Log-Likelihood Value	170.820	125.063	**63.026**
	Kolmogorov-Smirnov test *p*-value	0.641	0.993	**0.940**

independent with the threshold value. For negated mTTC, as shown in Fig. 6(b), the parameters show steady tendencies within the range near –3.

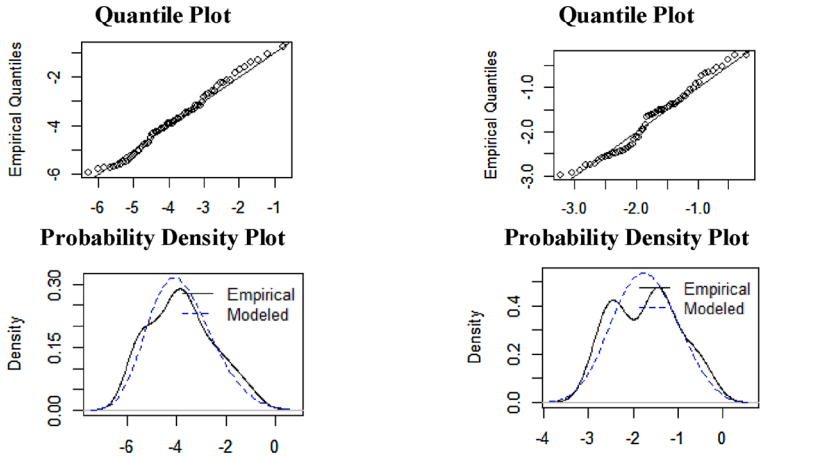
3.5. Modeling results

In this study, the PET or mTTC values for each interaction between a pair of pedestrian and vehicle can be the extreme values to model. The threshold values of negated PET are selected within the range (–7, –5), such as –7, –6, –5. The threshold values of negated mTTC are selected from –5, –4, and –3. Both GEV and GP models are established. For GEV model, three parameters, μ, σ, and, are to be estimated, while for GP model, two parameters, σ, and ε are estimated. As shown in Eqs. (7) and (8), suppose x is negated PET or negated mTTC generated from pedestrian-vehicle interactions, the crash occurrence can be regarded as extreme cases, with $x\ (-PET) = 0$ or $x\ (-mTTC) = 0$. The crash probability can be regarded as $Pr(x\ (-PET) > 0)$ or $Pr(x\ (-mTTC) > 0)$, which can be calculated using the respective model. The Kolmogorov–Smirnov test is used with the null hypothesis that the true distribution of the samples is drawn from the hypothesized distribution. If the *p*-value is greater than 0.05, we cannot reject the null hypothesis. The modeling results are shown in Tables 2 and 3 . All statistical analysis is done using Maximum Likelihood Estimate (MLE) method in R (v3.5.2) using "ExtRemes" and "evd" packages (Smith, 1985; Team, 2013; Gilleland and Katz, 2016). With the AIC and BIC as evaluating metrics, four models are selected, as marked in bold. It should be noted that GEV model with $threshold_{-PET} = -5$ is not selected, as the estimated standard error for shape parameter ε is too large, which means the model is not well fitted.

For MLE estimation, when the shape parameter $\varepsilon > -0.5$, maximum likelihood estimators are regular with the usual asymptotic properties; When $-1 < \varepsilon < -0.5$, maximum likelihood estimators do not hold the standard asymptotic properties; When $\varepsilon < -1$, maximum likelihood estimators are unlikely to be obtainable (Smith, 1985; Coles et al., 2001). Thus, among the four bold marked models in Tables 2 and 3, the estimators from two GP models are generally not reliable, with $\varepsilon = -0.628$ and $\varepsilon = -0.906$. The two GEV models are selected, with the diagnostic plots are shown in Fig. 7. The quantile plot compares between the quantile of empirical data and quantile of the fitted distribution, if the points are close to a good linearity, the model has a good fit. The density plot compares between the histogram of empirical data and the probability density function of the fitted model. For both models, the blue dotted lines (fitted distributions) almost cover the histograms of the empirical data (black lines). As both models have good fit, the threshold value for PET is selected as 6 s, and the threshold value for mTTC is selected as 3 s.

4. Experiments and results

The generating rate of the trajectories from videos is 30 records per second. The input variables are the mobility features of the interacting pedestrians and vehicles, such as the traveling courses, speed, distance between the road user and the conflict point. The feature vectors from the trajectory data are used as input for the prediction of the pedestrians' near-accident events. As mentioned above, the threshold values of PET and mTTC are 6 s and 3 s, respectively. So, the categories of the dependent variables are defined accordingly: when PET value is smaller than 6 s, and mTTC value is smaller than 3 s, the interaction is defined as



(a) GEV model with $threshold_{-PET} = -6$ (b) GEV model with $threshold_{-mTTC} = -3$

Fig. 7. Diagnostic plots for selected GEV models.

Table 4
Descriptive statistics of variables.

Variable	Description	Range	Unit
Course (ped)	The traveling directions	(10.95, 347.31)	degree
Course (veh)	The traveling directions	(0.19, 359.99)	degree
Speed (ped)	The traveling speed	(0, 6.39)	ft./s
Speed (veh)	The traveling speed	(0, 50.83)	ft./s
Distance to conflict point (ped)	The distance between the pedestrian and conflict point	(0, 154.73)	ft.
Distance to conflict point (veh)	The distance between the vehicle and conflict point	(0, 220.34)	ft.
Near-accident events*	Category of events (Safe; Slight conflict; Series conflict)	(safe: 16,799; slight: 5179; serious: 3280)	–

Note: *Dependent predictor.

"serious conflict"; when one of the indicators is smaller than the threshold value, the interaction is defined as "slight conflict"; when neither of the two indicators is smaller than the threshold values, the case is regarded as "safe". The descriptive statistics of the variables used are listed in Table 4.

The pedestrians' and vehicles' features before reaching the conflict points are generated to feed into the model. There are 25,258 records in the data set, with a ratio of 5.1:1.6:1 between the targeted classes "safe", "slight conflict", and "serious conflict". Eighty percent of the data are used as the training data set, and twenty percent of the data is used as the test data set. An over-sampling method SMOTE (Synthetic Minority Over-Sampling Technique, Chawla et al. (2002)) is used on the training data set to increase the number of records for the two minority classes, i.e., "slight conflict", and "serious conflict". SMOTE is a popular over-sampling method, which can generate new minority class records by interpolating between several minority class examples that lie together. The ratio of the three classes in the training data set after using SMOTE is 1:1:1, while the test data include only the original records.

The proposed neural network, GRU model consists of two GRU layers, one dense layer and a dropout layer. The GRU layers are set to learn the sequential information from the data, and dropout layer is used to avoid overfitting. To better learn sequential data, the GRU layers use feature vectors that are stacked from the previous time slices as input. Thus, the input data of the first layer of the GRU model is in three dimensions (number of samples, number of time slices (= 3), number of independent variables (= 6)).

The proposed GRU model is well trained before getting overfitted. Hyperparameters of the model mainly include batch size, unit number, the learning rate, and decay value of the optimization function ("Adam" optimizer (Kingma and Ba, 2014)). The definitions of these hyperparameters are listed as below:

(1) Batch size: the number of samples used for each iteration during the training process.
(2) Unit number: the number of units for each layer in the neural network. A larger unit number will result in a faster learning process.
(3) Learning rate: learning rate controls how quick in updating the weights at each iteration in the optimization algorithm. A larger learning rate makes the model to learn faster. But a too large learning rate will result in an unstable learning process.
(4) Decay value: decay is used to adjust the learning rate in the optimization algorithm. It's a parameter used to avoid overfitting by taking a ratio of the learning rate.
(5) Epoch number: the number of training epochs. The more epochs, the greater capacity the model has. However, a too large epoch number will lead to overfitting, which will mitigate the performance of the model.

Table 5
Hyperparameter tuning.

Hyperparameter	Tuning range	Selected
Batch size	100, 500, 1000	100
Unit number	256, 128, 64	128
Learning rate	0.05, 0.01, 0.005	0.01
Decay value	0, 0.001, 0.005	0

Table 6
Confusion matrix for binary classification.

Actual	Classified	
Positive	True Positive (TP)	False Negative (FN)
Negative	False Positive (FP)	True Negative (TN)

The most appropriate hyperparameters need to be decided for a neural network, to achieve the best performance. As shown in Table 5, the tuning range is the range out of which the most appropriate value is selected. After comparing the model's performance under different hyperparameter values, the batch size is selected as 100, and the unit number within the GRU memory unit is 128, etc. The epoch number for the training process is 26.

4.1. Evaluating metrics and results

The evaluating metrics, such as accuracy, precision, recall, and specificity are illustrated as below:

(1) Accuracy: the f fraction of the correctly classified samples among all the samples.
(2) Precision: the fraction of correctly classified samples among the samples classified as positive.
(3) Recall: or sensitivity, the proportion of actual positive samples that are classified correctly.
(4) Specificity: the proportion of actual negative samples that are correctly classified..
(5) False alarm rate (FAR): one minus specificity, the proportion of the actual negative samples that are classified wrongly.
(6) AUC: Area Under the Curve. The ROC curve (Receiver Operating Characteristic curve) is used as a comprehensive metric to evaluate the model's performance. This curve plots two parameters, recall and false alarm rate (FAR), at different classification thresholds. The AUC value, which ranges from 0.5 to 1, is the area under the ROC curve.

The evaluating metrics are usually computed from the confusion matrix, as shown in Table 6. For a binary classification problem, the samples can be labeled as positive and negative, so the metrics can be computed as from Eqs. (9)–(13). However, in this study, the samples are in three classes, so the average values of the metrics calculated from each class are calculated and used for evaluation.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{9}$$

$$Precision = \frac{TP}{TP + FP} \tag{10}$$

$$Recall = \frac{TP}{TP + FN} \tag{11}$$

$$Specificity = \frac{TN}{TN + FP} \tag{12}$$

$$FAR\ (false\ alarm\ rate) = 1 - specificity = \frac{FP}{TN + FP} \tag{13}$$

Table 7
Experiment results.

Model	Training set AUC	Test set				
		Precision	Recall	FAR	Accuracy (with cross-validation)	AUC
GRU	**0.876**	**0.831**	**0.856**	**0.138**	**0.878** (±0.0048)	**0.865**
LSTM	0.868	0.814	0.818	0.168	0.861 (±0.0069)	0.856
SVM	0.785	0.702	0.560	0.488	0.580 (±0.0040)	0.781

Compared with the LSTM model with the same architecture, i.e., two stacked LSTM layers, a dense layer, and a dropout layer, with "Adam" optimizer, the GRU model performs better. This may be due to the reason that the structure of the data set is not very complicated. Besides, a machine learning model, SVM (Support Vector Machine), with an "rbf" kernel is used for comparison. The modeling results on the test data set are shown in Table 7. It can be found that GRU achieves the best performance with the accuracy of 0.878 and AUC value as 0.865. The accuracy values are validated using the k-fold cross validation (k equals ten here). The data set is shuffled and split into ten groups, with one group as the validation set and the other groups as the training data set. The process is repeated until every group has acted as the validation set. Taking the GRU model for an instance, the accuracy value is ranging from 0.8732 to 0.8828 across all the ten groups (the whole data set).

5. Conclusions and discussions

This paper uses a GRU model to predict the pedestrians' near-accident events at the signalized intersections. With PET and mTTC indicators generated from videos, the EVT approach are used to fit the distributions of PET and mTTC values during the pedestrian-vehicle interactions. With the selected threshold values, the near-accident events of pedestrians are classified into three severity categories. A GRU model is further used to predict the pedestrians' near-accident events, reaching an accuracy of 0.878 and the AUC value of 0.865. The proposed model can be used to warn drivers of the potentially dangerous events involving pedestrians and to be implemented in the Connected Vehicle (CV) environment.

Among the contributions of this study are that two SSMs indicators are combined to classify the pedestrians' near-accident events. And the most appropriate threshold values are selected using GEV (General Extreme Value) distribution and Generalized Pareto (GP) distribution. A recurrent neural network, GRU, is used to better utilize information lying in the time-series data. Compared with LSTM, GRU is simpler and more efficient, but with better performance on this data set. However, the limitation of the study is that two time-based SSM indicators (PET and minimum TTC) were used, without using any speed-based SSM indicators such as Delta-V (Shelby, 2020). In future work, video data from more intersections, and more safety indicators can be used to test the robustness of the proposed model.

Authorship statement

The authors confirm contribution to the paper as follows: data collection: Shile Zhang, Yina Wu, Ou Zheng; analysis and interpretation of results: Shile Zhang, Mohamed Abdel-Aty; draft manuscript preparation: Shile Zhang, Mohamed Abdel-Aty, Yina Wu, Ou Zheng. All authors reviewed the results and approved the final version of the manuscript.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This study is sponsored by Advanced Transportation and Congestion Management Technologies Deployment (ATCMTD) grant from Federal Highway Administration (FHWA). The authors also acknowledge Florida Department of Transportation (FDOT) and the project manager Mr. Jeremy Dilmore, P.E. All results and opinions are those of the authors only and do not reflect the opinions or positions of the sponsors

References

Allen, B.L., Shin, B.T., Cooper, P.J., 1978. Analysis of Traffic Conflicts and Collisions.
Archer, J., 2004. Methods for the Assessment and Prediction of Traffic Safety at Urban Intersections and Their Application in Micro-simulation Modelling. Royal Institute of Technology.
Borsos, A., Farah, H., Laureshyn, A., Hagenzieker, M., 2020. Are collision and crossing course surrogate safety indicators transferable? A probability based approach using extreme value theory. Accid. Anal. Prev. 143, 105517.
Bradski, G., 2019. Opencv (open Source Computer Vision Library).
Centers for Disease Control and Prevention, 2019. Road Traffic Injuries and Deaths—A Global Problem.
Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P., 2002. Smote: synthetic minority over-sampling technique. J. Artif. Int. Res. 16 (1), 321–357.
Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y., 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv 1406, 1078.
Coles, S., Bawa, J., Trenner, L., Dorazio, P., 2001. An Introduction to Statistical Modeling of Extreme Values Springer.
Cooper, P., 1984. Experience with traffic conflicts in canada with emphasis on " post encroachment time" techniques. NATO Advance Study Inst Ser F. J. Comput. Syst. Sci. Int. 5.
Farah, H., Azevedo, C.L., 2017. Safety analysis of passing maneuvers using extreme value theory. Iatss Res. 41 (1), 12–21.
Fu, C., Sayed, T., Zheng, L., 2020. Multivariate bayesian hierarchical modeling of the non-stationary traffic conflict extremes for crash estimation. Anal. Methods Accid. Res., 100135
Gilleland, E., Katz, R.W., 2016. Extremes 2.0: an extreme value analysis package in r. J. Stat. Softw. 72 (8), 1–39.
Gong, Y., Abdel-Aty, M., Yuan, J., Cai, Q., 2020. Multi-objective reinforcement learning approach for improving safety at intersections with adaptive traffic signal control. Accid. Anal. Prev. 144, 105655.
Governors Highway Safety Association, 2020. Pedestrian Traffic Fatalities by State: 2019 Preliminary Data.
Graves, A., Mohamed, A.-R., Hinton, G., Year. Speech recognition with deep recurrent neural networks. In: Proceedings of the 2013 IEEE international conference on acoustics, speech and signal processing, pp. 6645-6649.
Hayward, J.C., 1972. Near-miss Determination Through Use of a Scale of Danger. Highway Research Record, p. 384.
He, K., Gkioxari, G., Dollár, P., Girshick, R., Year. Mask r-cnn. In: Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2980-2988.
Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Comput. 9 (8), 1735–1780.
Hupfer, C., 1997. Computer Aided Image Processing to Modify Traffic Conflicts Technique. Transportation Department, University of Kaiserslautern, Kaiserslautern.
Hydén, C., 1987. The Development of a Method for Traffic Safety Evaluation: the Swedish Traffic Conflicts Technique. Bulletin Lund Institute of Technology, Department, p. 70.
Hydén, C., Linderholm, L., 1984. The Swedish Traffic-conflicts Technique. International Calibration Study of Traffic Conflict Techniques, pp. 133–139.
Ismail, K., Sc, M., Sayed, T., Eng, P., Saunier, N., Lim, C., 2009. Automated analysis of pedestrian-vehicle conflicts using video data. Transp. Res. Rec. 2140.
Ismail, K., Sayed, T., Saunier, N., 2010. Automated analysis of pedestrian–vehicle conflicts: context for before-and-after studies. Transp. Res. Rec. 2198 (1), 52–64.
Kathuria, A., Vedagiri, P., 2020. Evaluating pedestrian vehicle interaction dynamics at un-signalized intersections: a proactive approach for safety analysis. Accid. Anal. Prev. 134, 105316.
Kingma, D.P., Ba, J., 2014. Adam: a method for stochastic optimization. arXiv preprint arXiv 1412, 6980.
Leadbetter, M.R., Lindgren, G., Rootzén, H., 2012. Extremes and Related Properties of Random Sequences and Processes. Springer Science & Business Media.
Li, P., Abdel-Aty, M., Yuan, J., 2020. Real-time crash risk prediction on arterials based on lstm-cnn. Accid. Anal. Prev. 135, 105371.
Mahmud, S.M.S., Ferreira, L., Hoque, M.S., Tavassoli, A., 2017. Application of proximal surrogate indicators for safety evaluation: a review of recent developments and research needs. IATSS Res. 41 (4), 153–163.
Mohamed, M.G., Saunier, N., 2013. Motion prediction methods for surrogate safety analysis. Transp. Res. Rec. 2386 (1), 168–178.
Naphade, M., Tang, Z., Chang, M.-C., Anastasiu, D.C., Sharma, A., Chellappa, R., Wang, S., Chakraborty, P., Huang, T., Hwang, J.-N., 2019. The 2019 ai city challenge. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops 452–460.

Ni, Y., Wang, M., Sun, J., Li, K., 2016. Evaluation of pedestrian safety at intersections: a theoretical framework based on pedestrian-vehicle interaction patterns. Accid. Anal. Prev. 96, 118–129.
Orsini, F., Gecchele, G., Gastaldi, M., Rossi, R., 2019. Collision prediction in roundabouts: a comparative study of extreme value theory approaches. Transp. A Transp. Sci. 15 (2), 556–572.
Ou zheng, M.a.-A., Yina, Wu, Cai, Qing, 2019. A-r-c-i-s: Ucf-sst Automated Roadway Conflicts Identify System (a.R.C.I.S). UCF-SST.
Ou Zheng, Y.W., 2019. Opencv c++ Implementation of Drone View Car Tracker.
Ou Zheng, Y.W., Xin, Gu, 2019. Python Implementation of Computer Vision Tools.
Pascanu, R., Mikolov, T., Bengio, Y., Year. On the difficulty of training recurrent neural networks, pp. 1310-1318.
Ren, S., He, K., Girshick, R., Sun, J., Year. Faster r-cnn: Towards real-time object detection with region proposal networks. In: Proceedings of the Advances in neural information processing systems, pp. 91-99.
Schuster, M., Paliwal, K.K., 1997. Bidirectional recurrent neural networks. IEEE Trans. Signal Process. 45 (11), 2673–2681.
Shelby, S.G., Year. Delta-v as a measure of traffic conflict severity. In: Proceedings of the 3rd International Conference on Road Safety and Simulati. September, pp. 14-16.
Smith, R.L., 1985. Maximum likelihood estimation in a class of nonregular cases. Biometrika 72 (1), 67–90.
Songchitruksa, P., Tarko, A.P., 2006. The extreme value theory approach to safety estimation. Accid. Anal. Prev. 38 (4), 811–822.
Španhel, J., Bartl, V., Juránek, R., Herout, A., 2019. Vehicle re-identification and multi-camera tracking in challenging city-scale environment. Proc. CVPR Workshops.
Tang, Z., Naphade, M., Liu, M.-Y., Yang, X., Birchfield, S., Wang, S., Kumar, R., Anastasiu, D., Hwang, J.-N., 2019. Cityflow: a city-scale benchmark for multi-target multi-camera vehicle tracking and re-identification. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 8797–8806.
Tarko, A.P., 2012. Use of crash surrogates and exceedance statistics to estimate road safety. Accid. Anal. Prev. 45, 230–240.
Tarko, A.P., 2018. Estimating the expected number of crashes with traffic conflicts and the lomax distribution – a theoretical and numerical exploration. Accid. Anal. Prev. 113, 63–73.
Tarko, A., 2019. Measuring Road Safety With Surrogate Events. Elsevier.
Tarko, A.A., Davis, G., Saunier, N., Sayed, T., 2009. Surrogate Measures of Safety.
Tarko, A.P., Songchitruksa, P., Year. Estimating the frequency of crashes as extreme traffic events. In: Proceedings of the 84th Annual Meeting of the Transportation Research Board.
Team, R.C., 2013. R: a Language and Environment for Statistical Computing. Vienna, Austria.
Wang, C., Xu, C., Xia, J., Qian, Z., Lu, L., 2018. A combined use of microscopic traffic simulation and extreme value methods for traffic safety evaluation. Transp. Res. Part C Emerg. Technol. 90, 281–291.
Wang, C., Xu, C., Dai, Y., 2019a. A crash prediction method based on bivariate extreme value theory and video-based vehicle trajectory data. Accid. Anal. Prev. 123, 365–373.
Wang, Y., Zhang, D., Liu, Y., Dai, B., Lee, L.H., 2019b. Enhancing transportation systems via deep learning: a survey. Transp. Res. Part C Emerg. Technol. 99, 144–163.
Wu, Y., Abdel-Aty, M., Zheng, O., Cai, Q., Zhang, S., 2020. Automated safety diagnosis based on unmanned aerial vehicle video and deep learning algorithm. Transp. Res. Rec., 0361198120925808
Yue, L., Abdel-Aty, M., Wu, Y., Zheng, O., Yuan, J., 2020. In-depth approach for identifying crash causation patterns and its implications for pedestrian crash prevention. J. Safety Res. 73, 119–132.
Zaki, M.H., Sayed, T., 2014. Automated analysis of pedestrians' nonconforming behavior and data collection at an urban crossing. Transp. Res. Rec. 2443 (1), 123–133.
Zhang, S., Abdel-Aty, M., Yuan, J., Li, P., 2020. Prediction of pedestrian crossing intentions at intersections based on long short-term memory recurrent neural network. Transp. Res. Rec., 0361198120912422
Zhao, J., Gao, Y., Qu, Y., Yin, H., Liu, Y., Sun, H., 2018. Travel time prediction: based on gated recurrent unit method and data fusion. IEEE Access 6, 70463–70472.
Zhao, L., Song, Y., Zhang, C., Liu, Y., Wang, P., Lin, T., Deng, M., Li, H., 2019. T-gcn: a temporal graph convolutional network for traffic prediction. IEEE Trans. Intell. Transp. Syst. 1–11.
Zheng, L., Sayed, T., 2019a. Comparison of traffic conflict indicators for crash estimation using peak over threshold approach. Transp. Res. Rec. 2673 (5), 493–502.
Zheng, L., Sayed, T., 2019b. From univariate to bivariate extreme value models: approaches to integrate traffic conflict indicators for crash estimation. Transp. Res. Part C Emerg. Technol. 103, 211–225.
Zheng, L., Sayed, T., 2019c. A full bayes approach for traffic conflict-based before–after safety evaluation using extreme value theory. Accid. Anal. Prev. 131, 308–315.
Zheng, L., Ismail, K., Sayed, T., Fatema, T., 2018. Bivariate extreme value modeling for road safety estimation. Accid. Anal. Prev. 120, 83–91.