IEEE Open Journal of
Intelligent Transportation Systems

Received 15 June 2023; revised 17 October 2023; accepted 30 November 2023. Date of publication 4 December 2023; date of current version 29 December 2023.

Digital Object Identifier 10.1109/OJITS.2023.3339016

Transformer-Conformer Ensemble for Crash Prediction Using Connected Vehicle Trajectory Data

ZUBAYER ISLAM (Member, IEEE), MOHAMED ABDEL-ATY (Senior Member, IEEE), AND B M TAZBIUL HASSAN ANIK

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

CORRESPONDING AUTHOR: B M T. H. ANIK (e-mail: bmtazbiulhassan.anik@ucf.edu)

ABSTRACT Crash prediction is one of the important elements of real time traffic management strategies. Previous studies have demonstrated the use of infrastructure-based detector data and UAV video to predict a crash in the near future. The main limitation of such data is limited coverage. In this work, we have used connected vehicle trajectory data that can have wide coverage as well as provide insight into the trajectory that might lead to a crash. The trajectory data was provided by Wejo which collects data from the manufacturer and was spaced at 3 seconds. GPS locations and their associated time series features such as speed, acceleration and yaw rate were used to feed into an ensembled Transformer and Conformer model. A voting classifier was used to obtain the output of the final model which achieved a recall of 76% and the false alarm rate of 30%. This study showed how connected vehicle trajectory data can aid in getting insight into crashes. While most previous studies focus on using aggregated data to estimate crashes, the proposed work shows that trajectory data mining can also provide competitive results.

INDEX TERMS Transformer, conformer, connected vehicle, crash prediction, traffic safety.

I. INTRODUCTION

TRAFFIC injuries and fatalities result in considerable loss to the economy of an individual, their family and to a country. NHTSA projects that the number of crashes in 2022 will increase by 10% to 42,915 compared to that in 2020 [1] although most countries are aiming at vision zero action plan by 2030 to 2040. Thus, the need for mitigating crashes is endless to achieve zero fatalities within a couple of decades. A key component in this regard is the prediction of potential crashes. Since crash like situations develop within short-term turbulence of traffic flow [2] it is necessary to have real-time crash likelihood monitoring systems. While several researchers have actively worked on this arena, the ever-changing field of sensing technology means there are new ways of formulating the definition of real-time crash likelihood prediction. Most of the previous studies rely on infrastructure-based data such as loop and radar detectors, Bluetooth, drone, etc. Nowadays, geolocation data from car manufacturers are available in near real-time. Such data is able to provide granular information about each individual vehicle and as such, it will eventually be possible to identify or classify driving actions that can lead to a crash. Moreover, extended trajectory provides better insights about individual driver's driving patterns whether near a crash location or not. In this study, we have used individual vehicle data and the trajectory associated with it, to correlate it to crashes. Although some studies have been aimed at taxi and bus trajectory data, previous studies have always used some form of aggregation on it. For example, instead of using time series trajectory data, hard brakes or long stop events were used. Thus, trajectory data mining in crash likelihood prediction is a relatively new area of research that has been explored in this study. The paper is organized as follows: literature review, data preparation, methodology, results, and conclusion.

The review of this article was arranged by Associate Editor Guoyuan Wu.

II. LITERATURE REVIEW

A. MODELS IN CRASH LIKELIHOOD PREDICTION

Crash prediction has been studied by various researchers using aggregated data. Speed, volume, and its statistical features such as average, coefficient of variation, logarithm, etc. were used in all previous studies. The data is usually collected from roadside stationary sensors such as radar detections, microwave detectors, Bluetooth, etc. Therefore, the analysis is related to the spot speed rather than the actual speed at a crash location. It was reported that upstream and downstream detectors from a crash location can indicate speed and volume features that could lead to a crash [3]. Initially, case control logistic regression was studied [4] which was improved upon with log-linear and Bayesian logistic models [5], [6], [7]. Bayesian logistic regression model was used by Xu et al. [8] using four different congestion levels as dependent variable. Nevertheless, statistical approaches often come with significant assumptions related to data distribution and preparation [9], [10], [11], [12], [13], [14]. These prerequisites can often impair the accuracy of crash likelihood predictions. As such, machine learning and deep learning models are becoming more popular these days in crash prediction because of their better performance and absence of assumptions compared to statistical models. Thus far, support Vector Machine (SVM) [15], [16], Long Short-Term Memory (LSTM) [17], [18], Random Forest [19] have been used to predict crashes. Huang et al. [20] used Convolutional Neural Network (CNN) to predict crashes in Interstates and found better results than shallow models. Bao et al. [21] also used CNN to model citywide short-term crash likelihood prediction and reported that CNN was able to capture local spatial correlation. XGBoost [22], AdaBoost [23], Multilayer Perceptron (MLP) [24], [25], [26], Back Propagation Neural Net (BPNN) [27] have also been studied with competitive accuracy scores. Moreover, signal information was also used to predict crashes at intersections [28]. To add a further dimension to the crash prediction arena, Islam et al. [29] used variational autoencoder to balance crash samples before training since most previous studies have used matched case control models where a lot of the non-crash events need to be removed. Lately, transformer [30] and its variant, the conformer [31], have received significant attention among researchers because of their aptitude to capture context and dependencies in sequential data. So far, Transformer and Conformer have portrayed application prominence in several research domains such as natural language processing [32], computer vision, speech detection [31], time-series analysis [33], etc. However, despite their enormous contributions, to the best of the authors knowledge, no studies investigated the application of Transformer and Conformer in real-time crash.

B. DATA IN CRASH LIKELIHOOD PREDICTION

The data used in almost all studies is limited to roadside stationary sensors like detectors, signal controller, etc. Some research also takes into account the weather from nearby weather stations. The main limitation of such sensing is that it can only provide traffic features for a segment of the road. Often road segments can be a mile long in length and the data reported by roadside sensors cannot consider the various traffic parameter distribution within the segment length. The segments on the road that do not have a detector can often be a blind zone for the different machine learning models. On the other hand, vehicle-based location data can provide detailed data based on the frequency. This enables a system to reduce the so-called blind zones created by roadside sensors. Moreover, roadside sensing needs regular maintenance and as such the data availability is also hampered. Geolocation data from connected vehicles does not have a single point of failure that can lead to missing data. A segment of road will be traversed by multiple vehicles thereby sampled by different sensors [34]. Such data has been used in some previous studies to estimate queue length [35], lane changing behavior [36], stop-or-go decision [37], etc. Most studies are limited to using taxi data or large-scale bus data that cannot form a representative sample. Furthermore, taxis would exhibit certain pickup and drop-off patterns while buses can only travel fixed routes. The data used in this paper was provided by Wejo Data Services, Inc. which collects data from automotive OEMs (Original Equipment Manufacturers). Thus, it represents mostly non-commercial vehicles. The dataset contains trajectory data with an update interval of about 3 seconds. It covers four counties in Orlando, Florida: Orange, Seminole, Volusia and Osceola. The penetration rate was around 3%. Previous studies reported that penetration rate of 0.8% could provide an acceptable representation of traffic flow [38].

Vehicle based data have been used in the past studies to identify critical driving events like speeding, hard brakes, jerk, etc., which were found to be positively correlated to accidents [39]. The driving patterns of several drivers were found to be an indicator for potential crashes [40]. Only 167 drivers were used in a 14-month period making the penetration rate very low. Taxi geolocation data was used by Xie et al. [41] to study corridor intersection safety. Smartphone data was also used by some researchers to model crash frequency [42]. It was also noted in this study that hard brakes, congestion level, and speed variance were positively related to crash frequency. Hard braking events were also found to be correlated to crashes at work zone [43]. Bus GPS data was also used to identify real-time crash potential on arterials [17]. It can be noted in all these studies that researchers are more focused on using events to model crashes. GPS data has been used to calculate hazardous events like hard brake, jerk, etc. Thus, the full potential of connected vehicle trajectory data is still unexplored.

A summary of reviewed literatures on crash likelihood prediction are listed in Table 1, with details on best performing models and data sources.

C. CRASH LIKELIHOOD WITH UAV DATA

A comparable alternative to using connected vehicle trajectory data to predict crash likelihood is video-based vehicle trajectory data collected from unmanned aerial vehicle

TABLE 1. Summary of literature review on crash likelihood prediction.

Reference	Publication Year	Model (Best)	Data Sources
[4]	2004	Case Control Logistic Regression	Detector
[7]	2015	Bayesian Logistic Regression	Detector, Weather
[8]	2015	Bayesian Random Parameter Logistic Regression	Detector, Geometry
[9]	2017	Support Vector Machine	Automatic Vehicle Identification
[10]	2013	Support Vector Machine	Detector
[12]	2020	LSTM-CNN	Adaptive Signal Controller, Detector, Weather
[13]	2015	Random Forest	Archived Data Management System
[14]	2020	CNN	Radar Sensors
[17]	2019	Support Vector Machine	Detector
[18]	2020	Multilayer Perceptron	Simulation
[19]	2020	Multilayer Perceptron	Simulation
[21]	2019	Back Propagation Neural Net	Simulation
[22]	2018	Bayesian Conditional Logistic Regression	Adaptive Signal Controller, Detector, Weather
[23]	2020	Variational Autoencoder	Detector

(UAV) video. UAVs typically record traffic video from the "bird's-eye" perspective with high-resolution cameras, ensuring high-quality vehicle trajectory data at the frame level [44]. Several studies have attempted to extract vehicle trajectories from UAV videos in order to identify crash potential in moving traffic [45], [46], [47], [48]. The studies demonstrated that UAV recorded traffic videos have enormous potential for reliable crash likelihood prediction. Nonetheless, the same studies revealed several severe drawbacks to relying solely on UAV recordings for crash likelihood prediction. To begin, it is challenging to anticipate crash potential in real time using UAV video-based trajectory data. For instance, UAVs require a constant supply of power to function properly. Hence, any disruption in the power supply can cause UAVs to malfunction, ultimately delaying the real-time crash likelihood prediction work. Moreover, extreme weather can impair UAV real-time operation [49]. Secondly, UAVs can only capture short vehicle trajectories in space for a brief road stretch. So, if the goal is to estimate crash likelihood in real-time at the network level, UAVs have a significant disadvantage. While platooning many UAVs side by side along the road network can solve this problem, the installation and operation costs may skyrocket. In response to the shortcomings of UAVs, connected vehicles have reliable power supply systems and can operate in inclement weather. Furthermore, connected vehicles can generate extended trajectories suited for network-level crash likelihood prediction at no additional cost. Overall, connected vehicle trajectory data has enormous promise for predicting crash likelihood.

In this study, we employ connected vehicle trajectory data from Wejo to identify crash potential. As such, rather than depending solely on event data and brief trajectory data, this study is able to capture long-term vehicle dynamics (such as speed distribution, acceleration variation, and so on) that contribute to a crash. The contributions of this paper are summarized below:

1. Concatenation of state-of-the-art Transformer and Conformer algorithms to predict crash likelihood in real-time.

TABLE 2. Features from connected vehicle data.

Name	Description	Unit
Features from Raw Data		
dataPointId	Unique Id for each Data Point	-
journeyId	Unique Id for each journey (ignition start to ignition end)	-
latitude	North-South position of the vehicle	-
longitude	East-West position of the vehicle	-
speed	Speed of the vehicle at the instant the datapoint was captured	meter/s
heading	Direction of the vehicle at the instant the datapoint was captured	degrees
postalCode	Zip code of the GPS location	-
Features Computed from Raw Data		
isStopped	1 if speed is less than 5mph, else 0	-
acceleration	Positive rate of change of speed between two successive datapoints	meter/s*s
deceleration	Negative rate of change of speed between two successive datapoints	meter/s*s
yawRate	Rate of change of heading between successive datapoints	degrees/s
dayOfWeek	Temporal feature representing day of the week	-
hour	The hour at the instant GPS point was captured	-

2. Exploration of the aptness of connected vehicle trajectory data from Wejo in crash potential identification.

III. DATA PREPARATION

The dataset used in this paper was provided by Wejo. It contains vehicle specific data from several manufacturers. It mainly has non-commercial fleet data which better represents the vehicles on the roadways. Instantaneous data is sent from the vehicle to the cloud (V2C) in near real-time. The dataset consists of GPS location, heading, speed, postal code, journeyId and dataPointId. A brief description of the raw data features is shown in Table 2. The sampling rate of the dataset was limited to 3 seconds. Also, the dataset was limited to four counties in Florida. Within this region there are several expressways such as Interstate-4, Interstate-95, SR 408, SR 417 and SR 528, along with several arterials.

TABLE 3. Crash vs Wejo data statistics.

Time Period	Total Crashes	Crashes Covered by Wejo Data	Max Number of Journeys per Crash	Min Number of Journeys Per crash	Total Datapoints	Total Unique Journeys
11/11/2019-17/11/2019	2061	1181	21	1	1,967,050	6781
10/11/2020- 16/11/2020	1341	992				

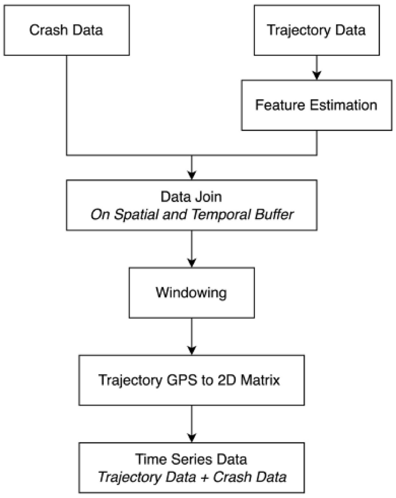


FIGURE 1. Data Processing Pipeline.

Considering availability, data during the following two time periods were used in this study: 11.11.2019-17.11.2019 and 10.11.2020-16.11.2020.

The trajectory data was joined with crash data from Signal Four Analytics (S4A). It provides the geolocation as well as crash severity, crash time, etc. Crash data was merged with connected vehicle data considering spatial and temporal buffer. For a particular crash location, the prior 5-minute interval was considered as the influence period and 300 feet was set as the spatial buffer. To specify further, if any Wejo vehicle traversed within a 300-foot radius of a location where a crash would happen in the following 0-5 minutes, it was regarded as a representative sample that contributed to the crash. It should be noted that between the two datasets (crash and trajectory), it was not possible to identify whether any of the Wejo vehicles were involved in the crash. Rather, our methodology aims to obtain the vehicle trajectories and their speed, acceleration profile, etc., to be indicative of the possible traffic features that led to a crash. We would also like to note that as the market penetration increases, it will ultimately be possible to do specific analysis of the vehicle involved in a crash. The overall pipeline is shown in Fig. 1.

Table 3 shows the statistics after the crash data was joined with Wejo data. Almost 2 million GPS datapoints were available after screening by time and location of crashes. In 2019, about 50% of the crash locations had representative Wejo vehicle data while around 80% for 2020. The max number of vehicles that passed through a crash location 5 minutes prior to a crash was 21 and minimum was 1.

The joining on spatial and temporal buffer was followed by windowing. In this step, the intent was to distinctively organize and associate traffic features with crash and non-crash events. To do so, a 30-second window was selected. Afterwards, we identified the time when a Wejo vehicle in its journey traversed through the crash influence zone, outlined by a 5-minute interval and a 300-foot radius. As the time was identified, the trajectory datapoints from the 30 seconds prior to that specific moment were attributed to the crash event. For instance, if a crash occurred at 10:50:15, with the crash influence time interval spanning from 10:45:15 to 10:50:15, and a Wejo vehicle drove past the crash spot at 10:48:30, then the vehicle's trajectory datapoints from 10:48:15 to 10:48:30 were associated with a crash, labeled as "1." The remaining datapoints of the Wejo vehicle were labeled as "0" (i.e., non-crash event). Fig. 2 illustrates the trajectory of a Wejo vehicle, from the start of ignition to the moment of the vehicle traverse through the crash influence zone. The red star represents the crash location. The green dots on the plot signify datapoints labeled non-crash, whereas the blue dots highlight the labeled crash.

After windowing, the trajectory data was then converted into a 2D matrix to extract the nature of the vehicle trajectory. This matrix was generated to obtain the nature of vehicle trajectory (whether straight linear motion or turning motion or lane changing motion) in the time window.

The speed, acceleration, yaw rate and heading profile during and before a crash is also shown in Fig. 3. The x-axis contains the order of the datapoints that are separated by 3 seconds. The instances from 40 to 50 were labelled as crash while any prior instance was non-crash. Each journey is highlighted in a different color. It is noted that variation in acceleration is more significant than in other cases of the journey. Same can be concluded about the yaw rate. This shows that the crash location is experiencing a high number of braking events and unusual swerving movements. The speed of most journeys also falls. The heading also shows notable changes indicating a probable turbulent situation.

IV. METHODOLOGY

This study, with the attention-based Transformer [30] and its Conformer [31] variant, presents ensemble Transformer-Conformer algorithm for real-time crash likelihood prediction using connected vehicle trajectory data. The detailed architecture of the Transformer-Conformer is discussed in this section. Fig. 4 depicts the proposed ensemble structure. Per the structure, both the Transformer and Conformer models were trained separately using the processed trajectory data. The modeling results of the two trained models were then concatenated following the technique of voting classifier [50] to finally use it for crash likelihood prediction. Additionally, the proposed algorithm

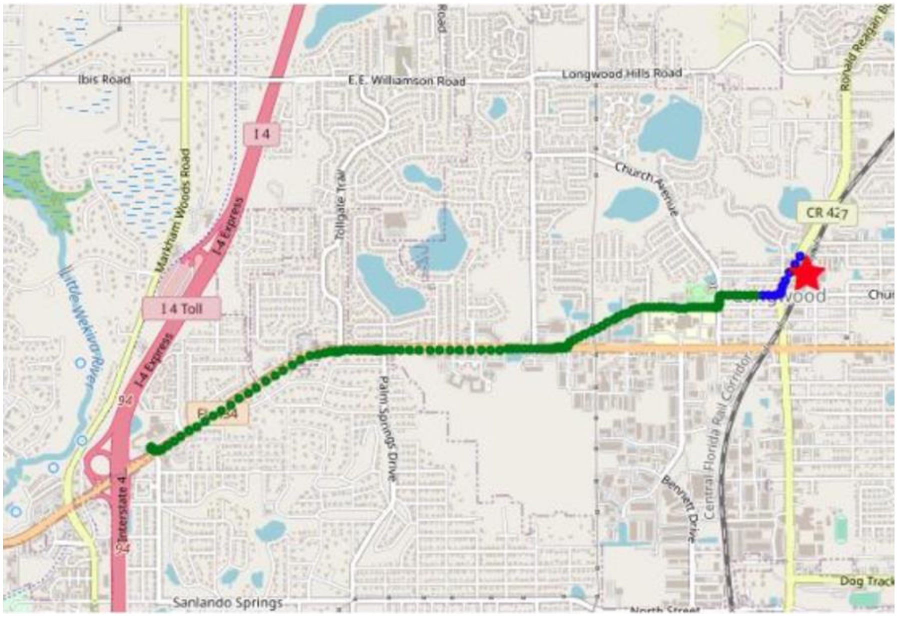


FIGURE 2. Journey Plot of a Wejo Vehicle Along with Crash Location.

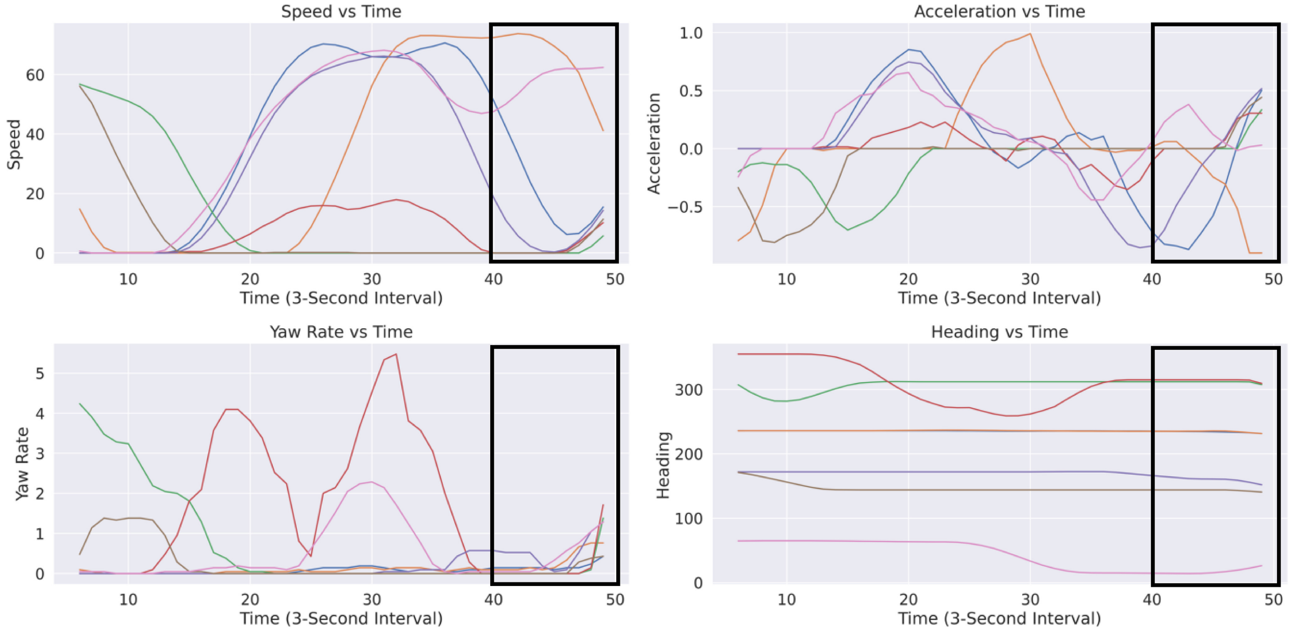


FIGURE 3. Speed, Acceleration, Yaw Rate, and Heading Profile Before and During Crash Location.

was tested with several deep learning algorithms including Convolution Neural Network (CNN) [51], [52], Long Short-Term Memory (LSTM) [53], [54], Particle Swarm Optimization (PSO) LSTM [55], Attention LSTM-CNN [56], and Deep Neural Network (DNN) [57], to conform its effectiveness of application.

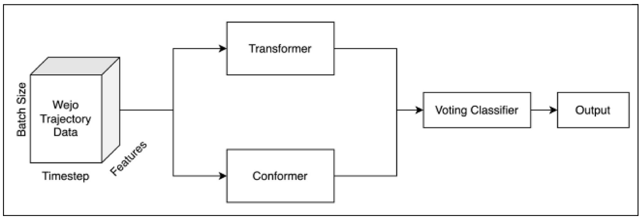


FIGURE 4. Transformer-Conformer Model Workflow.

A. TRANSFORMER

Originally, the Transformer algorithm was designed to perform NLP tasks leveraging attention-mechanism [30]. In this study, the functional domain of the original transformer is extended to predict crash likelihood in real-time. The model architecture of the Transformer is shown in Fig. 5.

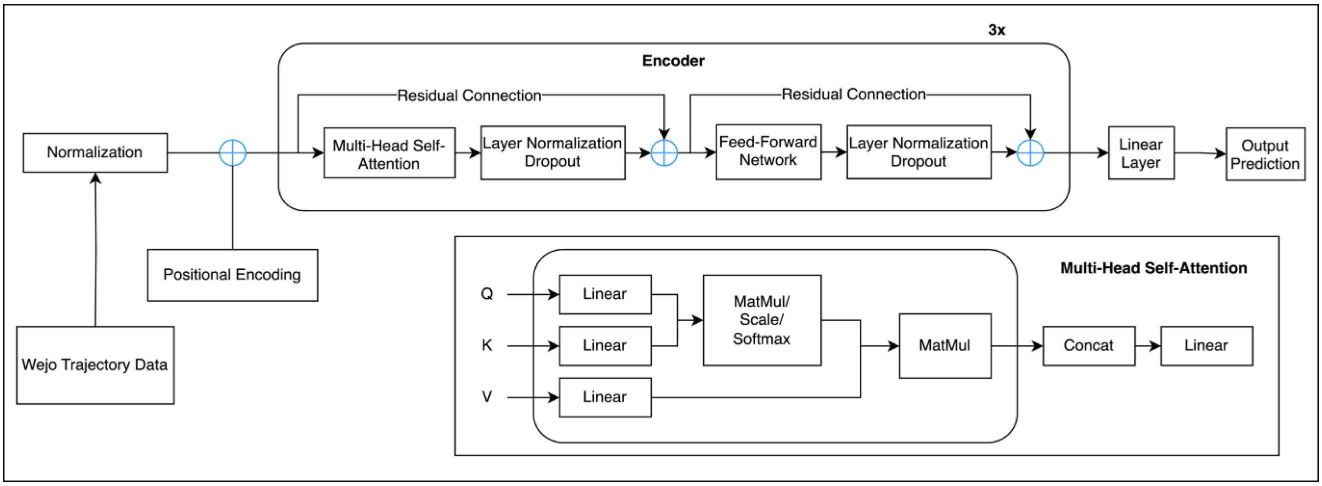


FIGURE 5. Transformer Architecture.

Unlike traditional recurrent neural networks (RNNs) and long-short term memory (LSTM) models, Transformer structurally favors an attention-based system by averting sequential processing. Hence, to capture the order of each sequence/input in the trajectory data, a positional encoding block was embedded in the transformer architecture. The normalized trajectory data with information on the order of the sequences were forwarded all at once to the encoder block as shown in Fig. 5. The encoder is the core block of the transformer architecture employed in this study. In total, three encoder blocks were stacked with a linear layer to make output predictions. Inside the encoder block, two key sub-blocks, namely multi-head self-attention, and feed-forward network, function. The multi-head self-attention executes the attention mechanism of the Transformer by concatenating the attention weights of n single-heads. Each single-head takes 3 inputs, namely query Q, key K, and value V, in total to calculate the attention weights that measure the relationship between sequences in the embedded trajectory data. The Q, K, and V vectors for each sequence are obtained through transformation. Fig. 5 also depicts how the multi-head attention sub-block functions. The attention mechanism of multi-head attention sub-block can be mathematically represented as follows:

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

where Q, K, and V are query, key, and value, vectors, respectively. T represents transpose, and d_k is the dimension of the key vector. The feed-forward network in Fig. 5 enables the model with the capacity to model more complex relationships by altering the representation obtained from the multi-head attention sub-block. In this study, CNN with convolution of kernel size 1 was incorporated as the feed-forward network. Also, in the encoder, each of the key components are followed by dropout, residual connection, and layer normalization, to prevent the model from overfitting, combat the problem of vanishing gradients, and keep the activations and gradients on a similar scale, respectively.

B. CONFORMER

Conformer combines the strengths of convolutional neural networks (CNNs) and transformers. This combination allows the model to capture both local and global dependencies in the input data. The model architecture of the Conformer is shown in Fig. 6 (a). There were four conformer blocks stacked with a linear layer to make output predictions. A Conformer block is shown in Fig. 6 (b). The three main sub-blocks are a feed-forward module, a multi-head self-attention module and a convolution module. Due to these three sub-blocks, a Conformer is able to extract global and local features more effectively. A feed forward module, as shown in Fig. 6 (c), is stacked on a multi-head attention module followed by a convolution module. This is preceded by another feed-forward module. The feed-forward module consists of a layer normalization, linear layer and Swish activation function [58]. Some dropout layers and a linear layer follows. The first linear layer expands the input dimension four-fold and the last one brings it back to the original dimension. The multi-head attention module is visualized in Fig. 6 (d). It features Layer Normalization, multi-head attention with relation positioning embedding and a Dropout layer. Lastly, the convolution module in Fig. 6 (e) consists of Layer Normalization with pointwise Convolution and gated linear unit (GLU) Activation; one of the layers expands the dimension by 0.5 while the other doubles it. It is followed by a 1D depth convolution, batch normalization, Swish Activation and Pointwise convolution.

Therefore, the relationship between the input and output can be modelled as:

$$X = X + \frac{1}{2}FFN(X) \tag{2}$$

$$X = X + MHSA(X) \tag{3}$$

$$X = X + Conv(X) \tag{4}$$

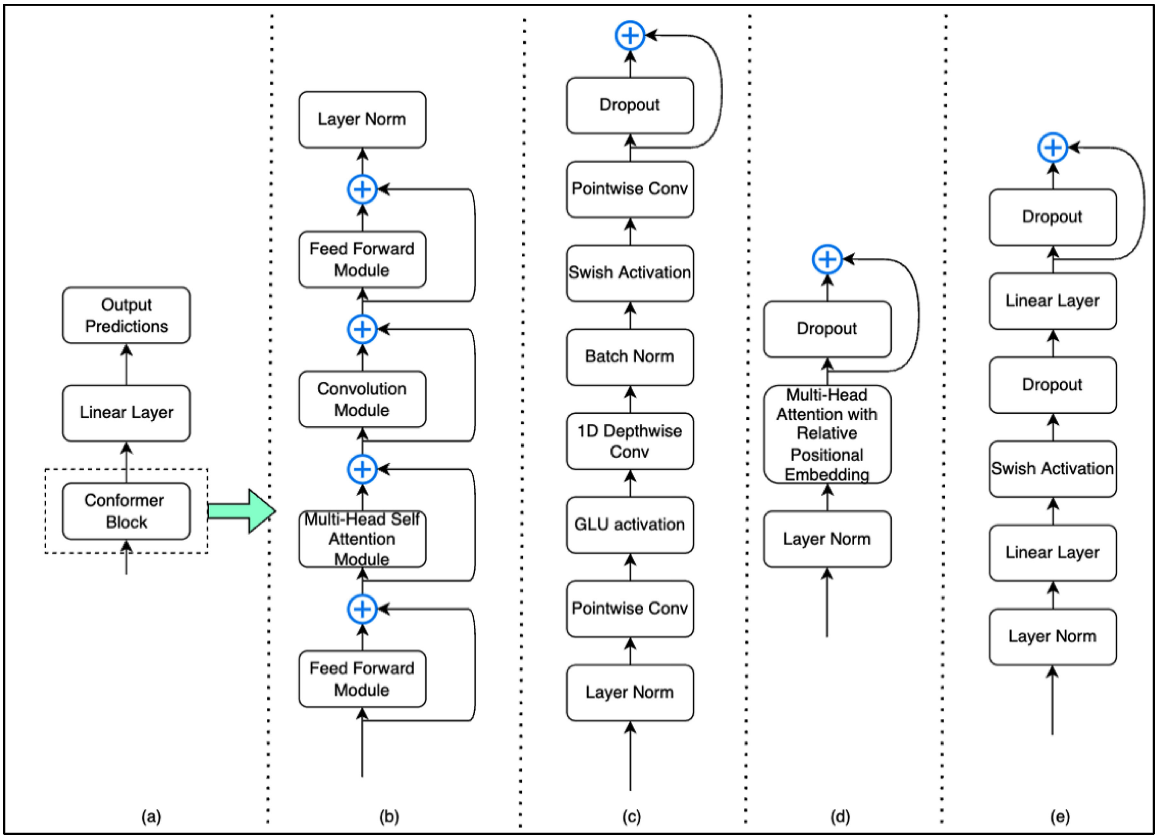


FIGURE 6. Conformer Architecture.

TABLE 4. Hyperparameter tuning.

Hyperparameters	*Transformer* Pool of Parameters **(Best Parameter)**	Hyperparameters	*Conformer* Pool of Parameters **(Best Parameter)**
Learning Rate	0.00001, 0.0001, 0.001 **(0.0001)**	Learning Rate	0.01, 0.001, 0.0001 **(0.001)**
Batch Size	500, 1000, 2500, 5000 **(1000)**	Linear Layers	1, 2, 4 **(1)**
No. of Epoch	10, 25, 50, 100, 200 **(25)**	No. of Epoch	10, 20, 30, 40, 100, 200 **(20)**
No. of Heads	3, 4, 5, 6 **(5)**	No. of Heads	30, 70, 100, 160 **(70)**
No. of Encoders	1, 3, 5, 7 **(3)**	No. of Conformer Blocks	2, 4, 6, 8, 16 **(4)**
Optimization Function	Adam, SGD **(Adam)**	Class weights	{0.5, 14}, {0.5, 10}, {0.5, 6}, {0.5, 4.3} **({0.5, 4.3})**

$$AY = LayerNorm\left(X + \frac{1}{2}FFN(X)\right) \tag{5}$$

where $FFN()$, $MHSA()$, $Conv()$, and $LayerNorm()$ denote feed-forward module, multi-head self-attention, convolution module and layer-normalization respectively.

V. RESULTS

The processed data set had 196,905 samples in total. Out of these total samples 7180 were related to a crash and were labeled as one while the others were related to non-crash. Since this is an extremely imbalanced data set, Synthetic Minority Oversampling Technique (SMOTE) was used to upsample the minority class. This has been used widely and has been accepted as an effective method for oversampling crashes [18], [29], [59]. The data set was split into an 80:20 ratio for training and testing purposes. It should be noted that only the train samples were oversampled and the performance of the model on the test data is reported. This ensures that the model is evaluated on real data and not synthetic data.

A critical aspect of ensuring dependable outcomes from model training is the fine-tuning of hyperparameters and the selection of suitable optimization functions. In this research, before the actual training, both the Transformer and Conformer models were fine-tuned using a variety of hyperparameters and optimization functions. Table 4 depicts the optimal parameter sets for the trained Transformer and Conformer models. After identifying the parameters, the models were trained accordingly and subsequently ensembled to finally assess its effectiveness on the test data.

For evaluating the model performance on the test dataset, several key metrics were used such as accuracy, recall, and false positive rate. The performance metrics are reported for the test data. While accuracy is usually studied for balanced datasets, parameters such as recall, false alarm rate, etc., are widely used to measure model performance for imbalanced

TABLE 5. Confusion matrix.

		Predicted	
		Non-Crash	Crash
Actual	Non-Crash	26395	11581
	Crash	323	1042

TABLE 6. Model comparison.

Model	Recall	False Alarm Rate
Transformer-Conformer (Proposed)	0.76	0.30
Transformer	0.74	0.36
Conformer	0.70	0.34
Attention LSTM-CNN	0.70	0.34
PSO-LSTM	0.66	0.47
LSTM	0.92	0.77
CNN	0.93	0.73
DNN	0.91	0.71

datasets. Recall indicates what percentage of correct positive predictions are made [60], [61], while the false positive rate indicates how often the model is likely miss-predict.

$$Sensitivity = \frac{TP}{TP + FN} \tag{6}$$

$$False\ Alarm\ Rate = \frac{FP}{FP + TN} \tag{7}$$

As stated earlier the Transformer-Conformer model was developed by concatenating the results of Transformer and Conformer using voting classifier technique. The ensemble model generated a recall of 76% and a false alarm rate of 30%. The confusion matrix is based on the test data shown in Table 5. It can be seen that out of the 1365 crashes in the test data, 1042 were correctly classified. Almost 11,600 samples were also classified as true which is the false alarm of the model. While this performance is lower than the traditional studies with detector data, it should be noted that the penetration rate of the CV vehicles used in this study was very low: about 3%. It is interesting to formulate the notion that only a percentage of the vehicles in the road can form a trajectory data pattern that can be indicative of a crash even though it may or may not directly participate in the crash.

A. MODEL COMPARISON

To perform a complete evaluation of the proposed model, the Transformer-Conformer model was compared with six different benchmark models. The results of the different models and associated performance metrics on test dataset are shown in Table 6. Compared to the benchmark models, the Transformer-Conformer model gives the best accuracy. While the recall is higher for most other models, it comes at the expense of high false alarms. The CNN and LSTM models individually perform much worse than the combined model. The performance of the DNN model is slightly better than the CNN or LSTM model with lower false alarm rate. The closest competitors to the Transformer-Conformer model are Transformer, Conformer, Attention LSTM-CNN, and PSO-LSTM. Nevertheless, they also performed poorly compared to the performance of Transformer-Conformer.

VI. CONCLUSION

In this paper the authors proposed the use of novel connected vehicle trajectory data in order to predict crash events in real time. An ensemble crash likelihood prediction model was built based on the trajectory data using Transformer and Conformer algorithms. To do so, both the Transformer and Conformer models were trained individually on the processed trajectory data before finally concatenating them using voting classifier technique. The proposed model achieved a recall of 76% with a false alarm rate of 30% showing that the model performs reasonably well given that the data set only represents 3% of the vehicles on the road.

The crash likelihood prediction model developed in this study offers wide-ranging applications in the real-world. These models are instrumental in improving Advanced Driver Assistance Systems (ADAS) that alert drivers of potential crashes and can even autonomously brake the vehicle. This technology is also crucial for the development of autonomous vehicles, ensuring they navigate safely. On a broader scale, Traffic Management Centers can use these models to optimize traffic flows in urban areas, identifying high-risk zones and managing congestion. Mobile apps can be devised to notify drivers about these zones, and insurance companies can recalibrate premiums based on the derived data. Additionally, city authorities can make infrastructural adjustments in areas identified as crash hotspots. However, to maximize its real-world efficacy, further refinements are imperative in areas like data privacy, data reliability, data accessibility, vehicle compatibility, infrastructure readiness, etc.

In essence, this research effectively formulated a crash prediction model aimed at advancing real-time traffic safety. However, several aspects warrant further exploration. To begin with, the data used to construct the prediction model had a relatively modest penetration rate of 3%. It would be beneficial for subsequent studies to incorporate data with a higher penetration and possibly delve into vehicle-level crash predictions. Furthermore, feature importance of the variables could be explored using SHAP or LIME to find compounding variables that result in a crash. Another assumption was that vehicles near a crash site implicitly indicated a collision. It's vital for subsequent studies to determine which connected vehicles, if any, were directly involved in the incident. Additionally, while this study offers a crash prediction tool, it doesn't delve into the statistical correlations between identified factors and the likelihood of a crash. Future endeavors should address the magnitude and significance of these relations. Finally, while making crash predictions, the study didn't account for crash types, despite the evident link between crash types and the way crash occurs. It would be advantageous for future research to consider crash types while making predictions.

REFERENCES

[1] (NHTSA, Washington, DC, USA). *Estimates.* Accessed: Oct. 30, 2022. [Online]. Available: https://www.nhtsa.gov/press-releases/early-estimate-2021-traffic-fatalities

[2] J. Lee, B. Park, and I. Yun, "Cumulative travel-time responsive real-time intersection control algorithm in the connected vehicle environment," *J. Transp. Eng.*, vol. 139, no. 10, pp. 1020–1029, 2013.

[3] M. Abdel-Aty, A. Pande, A. Das, and W. J. Knibbe, "Assessing safety on Dutch freeways with data from infrastructure-based intelligent transportation systems," *Transp. Res. Rec.*, vol. 2083, no. 1, pp. 153–161, 2008.

[4] 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," *Transp. Res. Rec.*, vol. 1897, no. 1, pp. 88–95, 2004.

[5] L. Wang, H. Gjoreski, M. Ciliberto, S. Mekki, S. Valentin, and D. Roggen, "Enabling reproducible research in sensor-based transportation mode recognition with the Sussex-Huawei dataset," *IEEE Access*, vol. 7, pp. 10870–10891, 2019.

[6] L. Wang, M. Abdel-Aty, J. Lee, and Q. Shi, "Analysis of real-time crash risk for expressway ramps using traffic, geometric, trip generation, and socio-demographic predictors," *Accid. Anal. Prevent.*, vol. 122, pp. 378–384, Jan. 2019.

[7] L. Wang, M. Abdel-Aty, Q. Shi, and J. Park, "Real-time crash prediction for expressway weaving segments," *Transp. Res. C, Emerg. Technol.*, vol. 61, pp. 1–10, Dec. 2015.

[8] C. Xu, W. Wang, P. Liu, and F. Zhang, "Development of a real-time crash risk prediction model incorporating the various crash mechanisms across different traffic states," *Traffic Injury Prevent.*, vol. 16, no. 1, pp. 28–35, 2015.

[9] M. Abdel-Aty and X. Wang, "Crash estimation at signalized intersections along corridors: Analyzing spatial effect and identifying significant factors," *Transp. Res. Rec.*, vol. 1953, no. 1, pp. 98–111, 2006.

[10] J. Lee, M. Abdel-Aty, and Q. Cai, "Intersection crash prediction modeling with macro-level data from various geographic units," *Accid. Anal. Prevent.*, vol. 102, pp. 213–226, May 2017.

[11] M. J. Ahsan, S. Roy, and A. S. Huq, "An in-depth estimation of road traffic accident cost in bangladesh," in *Proc. Int. Conf. Sustain. Develop. Technol. 4th Ind. Revolut.*, 2021, pp. 1–5.

[12] T. Hasan, M. Abdel-Aty, and N. Mahmoud, "Multivariate approach to peak-period models for crash types and severities in variable speed systems on freeways," *Transp. Res. Rec.*, Aug. 2023, Art. no. 3611981231189735.

[13] T. Hasan, M. Abdel-Aty, and N. Mahmoud, "Freeway crash prediction models with variable speed limit/variable advisory speed," *J. Transp. Eng. A Syst.*, vol. 149, no. 3, 2023, Art. no. 4022159.

[14] A. Faden, M. Abdel-Aty, N. Mahmoud, T. Hasan, and H. Rim, "Multivariate Poisson-Lognormal models for predicting peak-period crash frequency of joint on-ramp and merge segments on freeways," *Transp. Res. Rec.*, Jul. 2023, Art. no. 3611981231178797.

[15] F. Basso, L. J. Basso, F. Bravo, and R. Pezoa, "Real-time crash prediction in an urban expressway using disaggregated data," *Transp. Res. C, Emerg. Technol.*, vol. 86, pp. 202–219, Jan. 2018.

[16] R. Yu and M. Abdel-Aty, "Utilizing support vector machine in real-time crash risk evaluation," *Accid. Anal. Prevent.*, vol. 51, pp. 252–259, Mar. 2013.

[17] P. Li, M. Abdel-Aty, and J. Yuan, "Using bus critical driving events as surrogate safety measures for pedestrian and bicycle crashes based on GPS trajectory data," *Accid. Anal. Prevent.*, vol. 150, Feb. 2021, Art. no. 105924.

[18] P. Li, M. Abdel-Aty, and J. Yuan, "Real-time crash risk prediction on arterials based on LSTM-CNN," *Accid. Anal. Prevent.*, vol. 135, Feb. 2020, Art. no. 105371.

[19] L. Lin, Q. Wang, and A. W. Sadek, "A novel variable selection method based on frequent pattern tree for real-time traffic accident risk prediction," *Transp. Res. C, Emerg. Technol.*, vol. 55, pp. 444–459, Jun. 2015.

[20] T. Huang, S. Wang, and A. Sharma, "Highway crash detection and risk estimation using deep learning," *Accid. Anal. Prevent.*, vol. 135, Feb. 2020, Art. no. 105392.

[21] J. Bao, P. Liu, and S. V. Ukkusuri, "A spatiotemporal deep learning approach for citywide short-term crash risk prediction with multi-source data," *Accident Analysis & Prevention*, vol. 122, pp. 239–254, Jan. 2019.

[22] S. Sun, B. Zhou, and S. Zhang, "Analysis of factors affecting injury severity in motorcycle involved crashes," in *Proc. COTA*, 2020, pp. 4207–4219.

[23] J. Ke, S. Zhang, H. Yang, and X. Chen, "PCA-based missing information imputation for real-time crash likelihood prediction under imbalanced data," *Transportmetrica A, Transp. Sci.*, vol. 15, no. 2, pp. 872–895, 2019.

[24] Z. E. Abou Elassad, H. Mousannif, and H. Al Moatassime, "Class-imbalanced crash prediction based on real-time traffic and weather data: A driving simulator study," *Traffic Injury Prevent.*, vol. 21, no. 3, pp. 201–208, 2020.

[25] Z. E. Abou Elassad, H. Mousannif, and H. Al Moatassime, "A proactive decision support system for predicting traffic crash events: A critical analysis of imbalanced class distribution," *Knowl. Based Syst.*, vol. 205, Oct. 2020, Art. no. 106314.

[26] Y. Peng, S. Liu, and D. Z. Yu, "An improved car-following model with consideration of multiple preceding and following vehicles in a driver's view," *Physica A Stat. Mech. Appl.*, vol. 538, Jan. 2020, Art. no. 122967.

[27] J. Wang, Y. Kong, and T. Fu, "Expressway crash risk prediction using back propagation neural network: A brief investigation on safety resilience," *Accid. Anal. Prevent.*, vol. 124, pp. 180–192, Mar. 2019.

[28] J. Yuan and M. Abdel-Aty, "Approach-level real-time crash risk analysis for signalized intersections," *Accid. Anal. Prevent.*, vol. 119, pp. 274–289, Oct. 2018.

[29] Z. Islam, M. Abdel-Aty, Q. Cai, and J. Yuan, "Crash data augmentation using variational autoencoder," *Accid. Anal. Prevent.*, vol. 151, Mar. 2021, Art. no. 105950.

[30] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 5998–6008.

[31] A. Gulati et al., "Conformer: Convolution-augmented transformer for speech recognition," 2020, *arXiv:2005.08100*.

[32] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.

[33] M. Xu et al., "Spatial-temporal transformer networks for traffic flow forecasting," 2020, *arXiv:2001.02908*.

[34] J. K. Mathew, J. C. Desai, R. S. Sakhare, W. Kim, H. Li, and D. M. Bullock, "Big data applications for managing roadways," *Inst. Transp. Eng. J.*, vol. 91, no. 2, pp. 28–35, 2021.

[35] H. Tavafoghi, J. Porter, C. Flores, K. Poolla, and P. Varaiya, "Queue length estimation from connected vehicles with low and unknown penetration level," in *Proc. IEEE Int. Intell. Transp. Syst. Conf. (ITSC)*, 2021, pp. 1217–1224.

[36] Y. Ali, M. M. Haque, and Z. Zheng, "Assessing a connected environment's safety impact during mandatory lane-changing: A block maxima approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 6, pp. 6639–6649, Jun. 2023.

[37] Y. Ali, M. M. Haque, Z. Zheng, and M. C. Bliemer, "Stop or go decisions at the onset of yellow light in a connected environment: A hybrid approach of decision tree and panel mixed logit model," *Anal. Methods Accid. Res.*, vol. 31, Sep. 2021, Art. no. 100165.

[38] C. M. Day et al., "Detector-free optimization of traffic signal offsets with connected vehicle data," *Transp. Res. Rec. J. Transport. Res. Board*, vol. 2620, no. 1, pp. 54–68, 2017.

[39] A. Arun, M. M. Haque, A. Bhaskar, S. Washington, and T. Sayed, "A systematic mapping review of surrogate safety assessment using traffic conflict techniques," *Accid. Anal. Prevent.*, vol. 153, Apr. 2021, Art. no. 106016.

[40] J. Jun, J. Ogle, and R. Guensler, "Relationships between crash involvement and temporal-spatial driving behavior activity patterns: Use of data for vehicles with global positioning systems," *Transp. Res. Rec.*, vol. 2019, no. 1, pp. 246–255, 2007.

[41] K. Xie, X. Wang, H. Huang, and X. Chen, "Corridor-level signalized intersection safety analysis in Shanghai, China using Bayesian hierarchical models," *Accid. Anal. Prevent.*, vol. 50, pp. 25–33, Jan. 2013.

[42] J. Stipancic, L. Miranda-Moreno, N. Saunier, and A. Labbe, "Surrogate safety and network screening: Modelling crash frequency using GPS travel data and latent Gaussian spatial models," *Accid. Anal. Prevent.*, vol. 120, pp. 174–187, Nov. 2018.

[43] J. Desai, H. Li, J. K. Mathew, Y.-T. Cheng, A. Habib, and D. M. Bullock, "Correlating hard-braking activity with crash occurrences on interstate construction projects in Indiana," *J. Big Data Analyt. Transp.*, vol. 3, no. 1, pp. 27–41, 2021.

[44] J. Bock, R. Krajewski, T. Moers, S. Runde, L. Vater, and L. Eckstein, "The inD dataset: A drone dataset of naturalistic road user trajectories at German intersections," in *Proc. IEEE Intell. Veh. Symp. (IV)*, 2020, pp. 1929–1934.

[45] X. Gu, M. Abdel-Aty, Q. Xiang, Q. Cai, and J. Yuan, "Utilizing UAV video data for in-depth analysis of drivers' crash risk at interchange merging areas," *Accid. Anal. Prevent.*, vol. 123, pp. 159–169, Feb. 2019.

[46] C. Wang, C. Xu, and Y. Dai, "A crash prediction method based on bivariate extreme value theory and video-based vehicle trajectory data," *Accid. Anal. Prevent.*, vol. 123, pp. 365–373, Feb. 2019.

[47] R. Yu, L. Han, and H. Zhang, "Trajectory data based freeway high-risk events prediction and its influencing factors analyses," *Accid. Anal. Prevent.*, vol. 154, May 2021, Art. no. 106085.

[48] C. Yuan, Y. Li, H. Huang, S. Wang, Z. Sun, and H. Wang, "Application of explainable machine learning for real-time safety analysis toward a connected vehicle environment," *Accid. Anal. Prevent.*, vol. 171, Jun. 2022, Art. no. 106681.

[49] C. Yuan, Y. Li, H. Huang, S. Wang, Z. Sun, and Y. Li, "Using traffic flow characteristics to predict real-time conflict risk: A novel method for trajectory data analysis," *Analyt. Methods Accid. Res.*, vol. 35, Sep. 2022, Art. no. 100217.

[50] E. Bauer and R. Kohavi, "An empirical comparison of voting classification algorithms: Bagging," *Boosti. Variants*, vol. 36, pp. 105–139, Jul. 1999.

[51] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," 2015, *arXiv:1511.08458*.

[52] S. M. Shahid, S. Ko, and S. Kwon, "Performance comparison of 1D and 2D convolutional neural networks for real-time classification of time series sensor data," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, 2022, pp. 507–511.

[53] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[54] Y. Jeong, S. Kim, and K. Yi, "Surround vehicle motion prediction using LSTM-RNN for motion planning of autonomous vehicles at multi-lane turn intersections," *IEEE Open J. Intell. Transp. Syst.*, vol. 1, pp. 2–14, 2020.

[55] Y. Zhang and S. Yang, "Prediction on the highest price of the stock based on PSO-LSTM neural network," in *Proc. 3rd Int. Conf. Electron. Inf. Technol. Comput. Eng. (EITCE)*, 2019, pp. 1565–1569.

[56] Q. Du, W. Gu, L. Zhang, and S.-L. Huang, "Attention-based LSTM-CNNs for time-series classification," in *Proc. 16th ACM Conf. Embedded Netw. Sensor Syst.*, 2018, pp. 410–411.

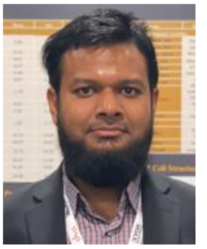
[57] Z. Wang, W. Yan, and T. Oates, "Time series classification from scratch with deep neural networks: A strong baseline," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2017, pp. 1578–1585.

[58] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for activation functions," 2017, *arXiv:1710.05941*.

[59] A. Ji and D. Levinson, "Injury severity prediction from two-vehicle crash mechanisms with machine learning and ensemble models," *IEEE Open J. Intell. Transp. Syst.*, vol. 1, pp. 217–226, 2020.

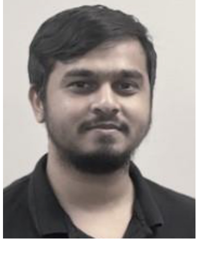
[60] M. A. Raihan, B. Anik, F. R. Ashik, M. M. Hasan, and S. Mahmud, "Motorcycle helmet use behavior: What does the data tell us?" Data Inf. Technol., Highways Safety Human Fact., Washington, DC, USA, Rep. TRBAM-23-03573, 2023.

[61] S. Zhang and M. Abdel-Aty, "Real-time pedestrian conflict prediction model at the signal cycle level using machine learning models," *IEEE Open J. Intell. Transp. Syst.*, vol. 3, pp. 176–186, 2022.



ZUBAYER ISLAM (Member, IEEE) received the M.S. degree in computer science and the Ph.D. degree in transportation engineering from the University of Central Florida, Orlando, FL, USA, where he is currently a Postdoctoral Scholar. He worked extensively on developing an application for data collection, cloud computing, and CV testing and used machine learning to estimate the vehicles' movements. He has contributed to computer vision applications. He was also part of the team which won the USDOT safety visualization challenge. He is involved in developing Android mobile applications, such as indoor and outdoor localization applications. His research interests include smart and safe transportation sensing, intelligent transportation systems, machine learning, and deep learning.

MOHAMED ABDEL-ATY (Senior Member, IEEE) is currently a Trustee Endowed Chair with the University of Central Florida (UCF). He is a Pegasus Professor and the Chair of the Civil, Environmental and Construction Engineering Department, UCF, where he is leading the Future City Initiative. He is also the Director of Smart and Safe Transportation Lab. He has authored or coauthored more than 750 papers, more than 400 in journals. As of June 2023, his Google Scholar citations are 28757 and H-index is 93. His main expertise and interests are in the areas of traffic safety, simulation, big data and data analytics, ITS, and CAV. He was the recipient of the nine Best Paper awards from ASCE, TRB, and WCTR, for his extraordinary research contributions. He is currently the Editor-in-Chief of *Accident Analysis and Prevention*.



B M TAZBIUL HASSAN ANIK received the B.Sc. degree in civil engineering from the Bangladesh University of Engineering and Technology, Dhaka, Bangladesh, in 2021. He is currently pursuing the Ph.D. degree with the University of Central Florida (UCF), where he is a Doctoral Research Assistant. He is a member of the Smart and Safe Transportation Lab (UCF-SST). His research interests include deep learning, causal machine learning, data science, traffic safety analysis, and intelligent transportation systems.