A Novel Multi-Model Machine Learning Approach to Real-Time Road Accident Prediction and Driving Behavior Analysis

Diya Dinesh
Thomas Jefferson High School for Science and Technology
Alexandria
VA, USA
diyadinesh19@gmail.com

2021 International Symposium on Computer Science and Intelligent Controls (ISCSIC) | 978-1-6654-1627-6/21/$31.00 ©2021 IEEE | DOI: 10.1109/ISCSIC54682.2021.00023

***Abstract*—As the leading cause of death within the U.S, road accidents take over 38,000 lives every year. Efforts are being taken nationwide to reduce the accidents and fatalities. Previous studies on the use of computer science for road safety were centered around analysis of historical data and prediction. This study proposes a novel solution, including real-time updates and features to road safety, with the use of Artificial Intelligence and Deep Learning integrated with various APIs and statistical analyses through the RoadSafety application. This application consists of three features: accident risk prediction, landmark analysis, and driving behavior analysis. The accident risk prediction component consists of a fully connected feed-forward deep neural network that takes in location, weather, time, and road feature input to predict an accident risk level. The landmark analysis identifies, through usage of the Pearson correlation coefficient and recursive feature elimination, which types of locations/landmarks are best correlated with accident severity. The driving behavior analysis uses an object detection Core ML model and the pinhole projection formula to identify distance from the driver to an obstacle ahead. This feature also compares the driver's speed to the speed limit. All three features are integrated into an iOS application to provide drivers within D.C. with live updates on accident prone-zones, landmark indicators of high accident severity, and risky driving behaviors.**

Keywords- Accident Prediction, Driving behavior, Fully Connected Feed-Forward Deep Neural Network.

I. INTRODUCTION

Accidents take place on the road every day, causing millions of injuries and deaths across the world with over 90% of road crash fatalities occurring in low and middle income countries [1]. According to a 2018 study, a person dies due to a road accident almost every 24 seconds [2]. Particularly, in the U.S, over 38,000 deaths occur as a result of car accidents every year and the costs involved with these accidents reach over $871 Billion [2]. Not only do these accidents severely impact the economic, business, and public transportation industries in the U.S, they impact the lives of millions. Studies have shown that, in addition to the victims of the road accident, these incidents impact the mental and emotional health of the family and friends of these victims [3].

This study aims at reducing road accidents by providing a solution that utilizes technology, specifically artificial intelligence (AI). Previous studies have used AI in three spheres of research: accident frequency prediction [4,5,6], inclement weather effect on accident patterns [7,8,9], and accident severity prediction [10-16]. Studies have also been performed on effect of driver behavior on accidents [17, 18]. However, these studies are limited by one main factor, which is the lack of ability to predict these accidents, identify the accident-prone zones, and monitor and analyze driver behavior in real time.

Real-time application is essential for drivers to get live updates about how to stay safe while driving. The product of this study (RoadSafety) is a solution to this limitation. Based on a fully connected feed-forward deep neural network that analyzes location, road feature, time, and weather data, from the OpenWeatherMap API [19], RoadSafety produces live updates on accident risk to drivers. This application also combines risk-prediction with statistical landmark analysis including the Pearson correlation coefficient and recursive feature elimination on data from GooglePlaces [20] to warn users about locations and sites to be cautious around. In addition to this, RoadSafety monitors driver behavior in accident-prone zones by identifying obstacles with a CoreML [21] model and calculating distance through the pinhole projection formula. It also surveys speed through the CLLocationSpeed class on Swift [22] and compares it to the speed limit through the Roads API [23]. Based on these, RoadSafety provides real-time warnings and suggestions on driving style. All three novel features, accident risk prediction, landmark analysis, and driver behavior surveillance and notification, are integrated into an iOS application that is accessible for drivers. All capabilities of the app were rated as highly impactful by over 75% of respondents in a pre-study survey conducted on drivers across the U.S. The survey represented a diverse sample of drivers who have from two to over twenty-five years of experience driving.

II. APPROACH

This study focused on three goals: accident risk prediction, landmark analysis, and driving behavior surveillance. The first goal, accident risk prediction, is defined as the ability to predict the risk of an accident occurring within a four hour period of the day in a 5.86 km x 0.442 km grid cell within D.C, given the time, weather conditions, and location of the point of interest. The second goal, landmark analysis, is to analyze and determine which location types and landmarks correlate with a higher severity of accidents. The third, driving behavior surveillance, is to use two components of driving behavior, distance between

978-1-6654-1627-6/21/$31.00 ©2021 IEEE
DOI 10.1109/ISCSIC54682.2021.00023

the driver's car and an obstacle ahead as well as the speed of driving, to identify unsafe driving behaviors and warn users as needed. The final product of this study is a user-friendly iOS app that integrates all three goals and provides constant updates to the driver about accident-prone zones and risky driving behavior. This study chose to develop the models and analyze information for D.C. This was primarily done to limit the data for homogenization. D.C. was chosen since the entire region is considered an urban area and a previous study showed that more than 44% of accidents across the nation take place in urban areas [24].

III. PROCEDURE

A. Data Definition: Accident Risk Prediction

1) Accident: The information that is present for accidents was retrieved from the U.S. Accident 2020 updated dataset [25] and consists of time period, grid cell, road features, and accident severity data. Thedata was collected from 2016 – 2020 [25]. This study focused on training a model to predict accident risk within the District of Columbia.

The severity was calculated for each grid cell across a specified four-hour period as the sum of severities of all accidents that had taken place within the four hour period. Each accident's severity was situated on a scale from 0 to 5 based on the impact it had on traffic.

TABLE I. INPUT DATA DEFINITIONS FOR ACCIDENT RISK PREVENTION

Data	**Definition**	**Term**	**Attributes**
Grid Cell Coordinates	5.86 km x 0.442 km rectangular grid cell	G = {g1, g 2, …, gn} where gn is a grid cell	Start and End, Latitude and Longitude
Time Period	4-hour period within a day (6 total time periods)	T = {t(1),,t (2), …, t(n)}) where t(n) is the nth time interval in the day	Start Time
Road Features	Features that exist within a specified grid cell.	R = {r1, r2, , rn} where rn represents whether a road feature n exists in a grid cell	Amenity, Bump, Crossing, give way, Junction, No-exit, Railway, Roundabout, Station, Stop, Traffic Calming Traffic Signal, Turning Loop
Weather	Weather across a grid cell n for a specified time period m.	W = {w1, w2, …, wn}	Minimum Temperature (°C), Maximum Temperature (°C), Precipitation (cm), Snow (cm)
Lighting	"Lighting" time intervals.	L = [0,4,9,17,21,24]	Start and End Timings

2) Weather: Historical weather data collected by the National Oceanic and Atmospheric Administration [26] between 2016 and 2020 from the Washington D.C. Dulles Airport was used in the study. The dataset (W = {w1, w2, …, wn} where wn represents the weather data for a given grid cell gn and time period t(n)) consisted of information about the minimum temperature (°C), maximum temperature (°C), amount of precipitation (cm), and amount of snow (cm) for each day. In addition to this, the lighting for a time period t(n) within a grid cell was calculated based on general sunrise and sunset timings and assigned an interval of "lighting" time that it was located within (L = [0,4,9,17,21,24]) where each Ln value represents the bounds of one "lighting" time interval in hours).

3) Combined Data: Once both datasets were obtained, the data was rearranged by grid cell coordinates and time. The two datasets were then merged based on timestamps. The data was then split into three categories: One-hot encoded data, binary data, and continuous data.

- One-hot encoded data: Features that remained constant across each time period within a grid cell. The features in this data structure, the 'Name' and 'Lighting', were one-hot encoded due the fact that each data point could contain one of more than two options for each of these features
- Binary data: Features that remained constant across the grid cell regardless of time (time-invariant). This data structure mainly included information about the road features R.
- Continuous data: Features that changed over time within a grid cell (time-variant) and data that was stored as continuous data types rather than a binary number or interval. Mainly, these features included the weather attributes, W, for each time period within a grid cell.

The binary and one-hot encoded vectors were combined. The output data was calculated from the severity scoring for each period within each grid cell (gnt(m)) represents the data for grid cell gn across time period t(m)). The severity scoring was stored as a sum of all accident severities across gnt(m). Based on the distribution of the data, the severity scores were split into four groups as described by the severity bounds, [0, 0.01, 3, 5, 500000]. Then the severity level of risk was stored as a labels vector for each gnt(m). In summary, the input data, as is shown in Table I, consisted of time, grid cell, road feature, weather, and lighting information (i = (T, G, R, W, L)) and the output data consisted of the risk scoring for each datapoint i.

B. Data Definition: Landmark Analysis

The data for the landmark analysis consisted of the accident information and the landmark/location information. The accident information held the coordinates of all the grid cells across D.C. and the mean severity of accidents across these grid cells within a four-hour period. The landmark/location information was obtained from the GooglePlaces API. The midpoint coordinates of the grids were passed to the GooglePlaces API to obtain information about the types of landmarks nearby. These types were then stored along with the respective coordinates and included 97 different landmarks/locations.

C. Data Definition: Driving Behavior Surveillance

The data used for the driving behavior surveillance also consisted of two parts: live feed of the road and the speed information. The live feed video was provided through the iPhone camera on the iOS App. The features that the model aimed to identify included cars, buses, bicycles, motorbikes, trains, vehicles, and people. In terms of speed information, the Swift CLLocationSpeed class was used to find the speed. The speed limit for the road segment (S) across the driver's current grid cell was obtained from the Roads API.

D. Model/Analysis: Accident Risk Prediction

1) Train/Test data split: In order to simulate the real-world experience, the dataset was split into train and test data based on time. Train data consisted of all the data for all gnt(m) from 2016-2019. The 2020 data was then used as testing data to simulate a "real-time" data exposure.

2) Data generation: The total amount of data points across all time periods and grid cells in the train dataset reached over 2 million. This amount was too large for the model to train at once, so the data had to be split into batches as: one-hot encoded, continuous, and output [27]. The size of these batches was defined by the training method and the order and content of these batches were selected from a common random set of data points within the training data. To improve the model's performance, data-augmentation was used with the continuous data to introduce some noise and increase the data-set size. The data was augmented based on the distribution, based on the variance, of the total continuous data within the mini-batch. It was also ensured that the same number of data points were passed for each of the risk categories. This ensured that the model was exposed to a standardized set of input

3) Model Creation: A fully-connected feed-forward deep neural network was made to produce a risk scoring. The neural network model took discrete and continuous input. The two inputs were processed in parallel by linear transformations and passed to a hidden layer. The outputs of both the input and hidden layer was passed to a ReLU activation function. The output then passed through a sigmoid function and the model outputted a probability value for the applicability of each risk. The maximum probability value was then taken from this through the arg max method. Dropouts, at a 20% rate, were also applied at each layer to avoid overfitting. A visual of the model structure is shown in Figure 1 [29].

4) Model Training: The model trained for 100 epochs and the average loss was observed. The mean squared error loss function and the Adam optimizer [27] was used. For each batch of training data, a batch of test data was also created to observe the test loss alongside the train loss.

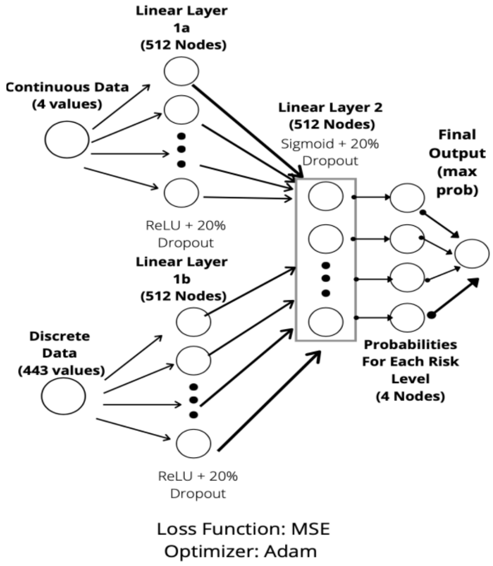

Linear Layer 1a (512 Nodes)
Continuous Data (4 values)
Linear Layer 2 (512 Nodes)
Sigmoid + 20% Dropout
Final Output (max prob)
ReLU + 20% Dropout
Linear Layer 1b (512 Nodes)
Discrete Data (443 values)
Probabilities For Each Risk Level (4 Nodes)
ReLU + 20% Dropout
Loss Function: MSE
Optimizer: Adam

Figure 1. Accident risk prediction model.

5) Application: The model was uploaded to the app and produced a risk rating given the inputs. The weather input was gathered through the OpenWeatherMap API, the time was found through the iPhone's local time, and the location data was obtained through a CLLocationManager object. The model provided constant updates to the driver on risks through the app in four levels: "no risk", "low risk", "medium risk", and "high risk".

E. Model/ Analysis: Landmark Analysis

1) Pearson Correlation Coefficient: A Pearson correlation coefficient was calculated using the formula (1), for the relationship between the landmark data and the mean severity of accidents occurring around each landmark. This coefficient value indicates how strongly each location acts as a predictor for the severity of accidents as well as the way in which the two factors are correlated (positive or negative relationship).

$$r= \Sigma(xi-\overline{x})(yi-\overline{y}) /\sqrt{}\Sigma(yi-\overline{y})^2*\Sigma(xi-\overline{x})^2 \qquad (1)$$

2) Recursive Feature Elimination (RFE): This method used the locations and accident severities for each grid cell. The RFE method ranks the features from 1 to 50 in terms of how well they predict accidents. RFE helps in the comparison of features amongst one another to indicate the best predictors. The features that ranked in the top 25 on the RFE method were taken into consideration for warning users in the iOS app.

F. Model / Analysis: Driving Behavior Surveillance

1) Distance Identification: There were two parts for the distance identification aspect of surveillance. The first was to identify all "obstacles" through a livestream of the road. These obstacles included cars, buses, bicycles, motorbikes, trains, vehicles, and people. The model used to identify these objects was an Apple CoreML adapted version of a "You Only Look Once" (YOLO) object detection machine learning model.

$$x/f = X/D \tag{2}$$

This model drew a bounding box, with a label, around each object. The distance from the camera to the "obstacle" ahead, is calculated using the coordinates of the bounding box. The pinhole projection formula was used to calculate the "real" distance between the driver's car and the obstacle ahead. The pinhole projection formula describes the relationship between a 3D object and its projection through a pinhole camera (such as the iPhone camera) onto a 2D plane. The formula (2) relates the focal length of the camera (f = ~ 50 mm), the height of the object in pixels in the detection (x), and the approximate height of the object in real life (X) to the distance between the car and the obstacle in real life (D). The average heights for each obstacle are illustrated in Table 4.

2) Speed Monitoring: The speed monitoring aspect calculated the speed limit and driver speed every 5 minutes. The speed limit was obtained for the driver's grid cell from the Roads API. The driver's speed was calculated using a CLLocationSpeed object in Swift. These two values were then compared to warn the driver.

G. Model / Analysis: App Integration

The features were integrated into the iOS app RoadSafety (Figure 2). The risk prediction model received location and time data from Swift classes and weather data from the OpenWeatherMap API. The rest was taken from the U.S. Accidents dataset. The GoogleMaps API provided landmark data for analysis. The CoreML model used the iPhone camera for obstacle detection. The user's speed was taken from the CLSpeed class and speed limit was taken from the Roads API.

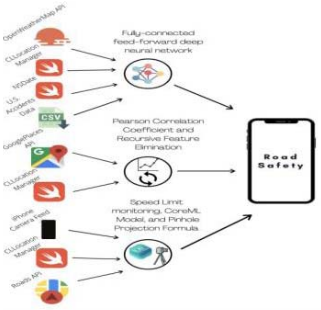
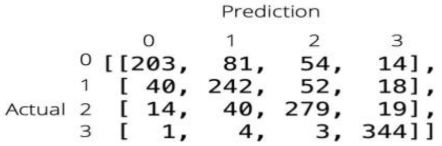

OpenWeatherMap API
CLLocation Manager
NSDate
U.S. Accidents Data
CSV
GooglePlaces API
G
CLLocation Manager
iPhone Camera Feed
CLLocation Manager
Roads API
Fully-connected feed-forward deep neural network
Pearson Correlation Coefficient and Recursive Feature Elimination
Speed Limit monitoring, CoreML Model, and Pinhole Projection Formula
Road Safety

Figure 2. Final Multi-Model Integration with RoadSafety App.

IV. RESULTS

A. Accident Risk Prediction

The MSE loss and a confusion matrix indicated the accident risk prediction accuracy. The MSE loss indicated how much error was present within the results the neural network gave on classification. The average training loss reached the ~ 0.07565 and the average test loss reached ~0.26836. The confusion matrix displayed the model's performance within each of the risk levels (Figure 3).



```
                Prediction
          0      1      2     3
     0 [[203,   81,   54,   14],
     1  [ 40,  242,   52,   18],
Actual 2  [ 14,   40,  279,   19],
     3  [  1,    4,    3,  344]]
```

Figure 3. Confusion matrix for risk levels

Overall, the top 5 most risky regions within D.C can be found in Table 2.

TABLE II. MOST ACCIDENT PRONE ZONES

Region Rank	Start Lat, Start Long (degrees)	End Lat, End Long (degrees)
1	38.863, -76.952	38.916, -76.948
2	38.863, -76.976	38.916, -76.972
3	38.791 , -77.009	38.844, -77.005
4	38.926 , -77.118	38.979, -77.114
5	38.934 , -77.030	38.987, -77.026

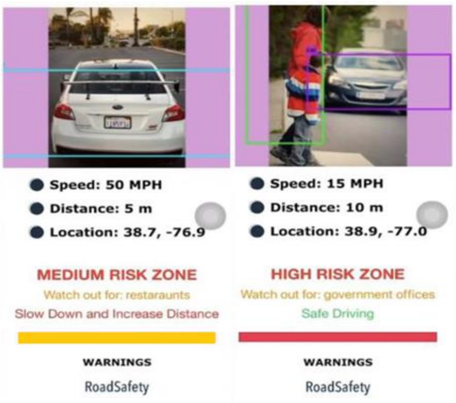
TABLE III. LANDMARK PREDICTOR FOR HIGH ACCIDENT SEVERITY

Feature	Pearson Correlation Coefficient	RFE Ranking

Church	0.1676709373	1
Place of Worship	0.1616376981	1
School	0.1102428725	1
Primary School	0.09926320258	1
Bakery	0.0831657766	1

TABLE IV. OBSTACLES AND COREML MODEL DETECTION PREDICTION

Obstacle	**Average Height (mm)**	**Precision (%)**
Bicycle	500	83.6
Bus	4419.6	87.054
Car	1524	86.558
Motorbike	762	83.169
Person	1752.6	78.844
Train	4025	86.559



- Speed: 50 MPH
- Distance: 5 m
- Location: 38.7, -76.9

MEDIUM RISK ZONE

Watch out for: restaraunts

Slow Down and Increase Distance

WARNINGS

RoadSafety

- Speed: 15 MPH
- Distance: 10 m
- Location: 38.9, -77.0

HIGH RISK ZONE

Watch out for: government offices

Safe Driving

WARNINGS

RoadSafety

Figure 4. RoadSafety App output.

B. *Landmark Analysis*

Multiple features were placed on the first rank from the landmark analysis. Upon consideration of how each rank 1 feature and the Pearson correlation coefficient, the top 5 landmark indicators of high severity were: churches, places of worship, schools, primary schools, and bakeries (Table 3).

C. *Driver Behavior Surveillance*

The accuracies and heights for each "obstacle" identification are displayed below in Table 4

D.IOS App

All the features were integrated into the app. A few examples of the final app are shown in Figure 4.

V. DISCUSSION

As the results of this study demonstrate, the approach yields a comprehensive application that can be used to monitor driving behaviors and accident prone zones. The low MSE values of the risk prediction model indicate a high accuracy. As the confusion matrix shows, the risk prediction model tends to err on the cautious side, so users of the app can be reassured that the warnings are accurate.

As the landmark analysis results also showed, the top 5 locations can be used as indicators of higher risk of accidents. This was implemented in the iOS app to warn users by raising the current risk score by a category if located next to a predictor landmark. The driving activity of the users is also monitored through the RoadSafety app. As stated, the obstacle detection model has a high accuracy and the distance to the obstacle can be accordingly calculated. This monitoring helps provide constant updates on risky driving, especially within accident prone zones.

However, there are a few limitations to consider during the implementation. The first limitation is to the dataset. Since this study chose to analyze datasets defined for D.C, the data for both the model and analysis was limited. Therefore, these results may not be as applicable to other locations. In the future, the app can be extended to other locations. Another limitation is the number of obstacles that this app considers. Since the app only monitors for certain types of vehicles and people, the app will not identify animals crossing the street or road signs that come in the way of the driver. In future, this study may be improved by adding more driver safety features. This includes monitoring and warning on severely inclement weather. Other features such as monitoring drivers for drowsiness may also be added. The main difficulty to accomplish these improvements would be to collect data that can easily be integrated with the current set of processed data.

VI. CONCLUSION

Road crashes, fatalities, and disabilities are increasingly becoming an international concern. The U.S. is the most affected high-income country with road accidents as the leading cause of death for people between the ages of 1 and 54. However, these accidents are preventable by ensuring that all drivers travel with caution and responsibility. Few studies have been conducted on predicting accidents to improve driver safety, but none of these studies provide real-time updates and a thorough analysis of driving patterns. The result of this study, an iOS application called RoadSafety, is novel in its ability to act as an aid to drivers by providing constant updates on accident risk and driving behaviors. The application's accident risk prediction successfully warns users of the risk level of encountering an accident. The landmark analysis identifies nearby locations that may be correlated with accident risk. The driving behavior surveillance monitors the user's speed and distance from obstacles to show warnings and suggestions. In the future, this app will be improved by expanding this research to other cities within the U.S. and adding more driver safety features.

REFERENCES

[1] Road Safety. (n.d.). WHO. https://www.who.int/data/gho/data/themes/road-safety

[2] Road Safety Facts. (2018). Association for Safe International Road Travel. https://www.asirt.org/safe-travel/road-safety-facts/

[3] Hobbs, M., Mayou, R., Harrison, B., & Worlock, P. (1996). A randomised controlled trial of psychological debriefing for victims of road traffic accidents. *BMJ*, *313*(7070), 1438-1439. https://doi.org/10.1136/bmj.313.7070.1438

[4] Chang, L.-Y. (2005). Analysis of freeway accident frequencies: Negative binomial regression versus artificial neural network. Safety Science, 43(8), 541-557. https://doi.org/10.1016/j.ssci.2005.04.004

[5] Chang, L.-Y., & Chen, W.-C. (2005). Data mining of tree-based models to analyze freeway accident frequency. Journal of Safety Research, 36(4), 365-375. https://doi.org/10.1016/j.jsr.2005.06.013

[6] Caliendo, C., Guida, M., & Parisi, A. (2007). A crash-prediction model for multilane roads. Accident Analysis & Prevention, 39(4), 657-670. https://doi.org/10.1016/j.aap.2006.10.012

[7] Jaroszweski, D., & McNamara, T. (2014). The influence of rainfall on road accidents in urban areas: A weather radar approach. Travel Behaviour and Society, 1(1), 15-21. https://doi.org/10.1016/j.tbs.2013.10.005

[8] Wang, Y., & Zhang, W. (2017). Analysis of Roadway and Environmental Factors Affecting Traffic Crash Severities. Transportation Research Procedia, 25, 2119-2125. https://doi.org/10.1016/j.trpro.2017.05.407

[9] Eisenberg, D. (2004). The mixed effects of precipitation on traffic crashes. Accident Analysis & Prevention, 36(4), 637-647. https://doi.org/10.1016/S0001-4575(03)00085-X

[10] Ihueze, C. C., & Onwurah, U. O. (2018). Road traffic accidents prediction modelling: An analysis of Anambra State, Nigeria. Accident Analysis & Prevention, 112, 21-29. https://doi.org/10.1016/j.aap.2017.12.016

[11] Kumar, S., & Toshniwal, D. (2015). A data mining framework to analyze road accident data. Journal of Big Data, 2(1). https://doi.org/10.1186/s40537-015-0035-y

[12] Lin, L., Wang, Q., & Sadek, A. W. (2015). A novel variable selection method based on frequent pattern tree for real-time traffic accident risk prediction. Transportation Research Part C: Emerging Technologies, 55, 444-459. https://doi.org/10.1016/j.trc.2015.03.015

[13] Mannering, F. L., Shankar, V., & Bhat, C. R. (2016). Unobserved heterogeneity and the statistical analysis of highway accident data. Analytic Methods in Accident Research, 11, 1-16. https://doi.org/10.1016/j.amar.2016.04.001

[14] Chen, C., Fan, X., Zheng, C., Xiao, L., Cheng, M., & Wang, C. (n.d.). SDCAE: Stack Denoising Convolutional Autoencoder Model for Accident Risk Prediction Via Traffic Big Data [Paper presentation]. 2018 Sixth International Conference on Advanced Cloud and Big Data (CBD).

[15] Moosavi, S., Samavatian, M. H., Nandi, A., Parthasarathy, S., & Ramnath, R. (n.d.). Short and Long-term Pattern Discovery Over Large-Scale Geo-Spatiotemporal Data [Paper presentation]. 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining.

[16] Najjar, A., Kaneko, S., & Miyanaga, Y. (2017). Combining Satellite Imagery and Open Data to Map Road Safety. Thirty-First AAAI Conference on Artificial Intelligence, 31(1). https://ojs.aaai.org/index.php/AAAI/article/view/11168

[17] Matawaha, J. A., Jadaan, K., & Freeman, B. (2019). Analysis of Speed Related Behavior of Kuwaiti Drivers Using the Driver Behavior Questionnaire. Periodica Polytechnica Transportation Engineering, 48(2), 150-158. https://doi.org/10.3311/PPtr.13167

[18] Moosavi, S., Omidvar-Tehrani, B., Craig, R. B., Nandi, A., & Ramnath, R. (n.d.). Characterizing Driving Context from Driver Behavior [Paper presentation]. 25th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems.

[19] Weather API [Computer software]. (n.d.). https://openweathermap.org/api

[20] Google Places API [Computer software]. (n.d.). https://cloud.google.com/maps-platform/places

[21] CoreML Object Detection Model [Computer software]. (n.d.). https://developer.apple.com/documentation/coreml

[22] Swift [Computer software]. (n.d.). https://developer.apple.com/swift/

[23] Roads API [Computer software]. (n.d.). https://developers.google.com/maps/documentation/roads/overvie w

[24] National Highway Traffic Safety Administration. (2005). *Traffic Safety Facts* [Infographic].

[25] Moosavi, S., Samavatian, M. H., Parthasarathy, S., Teodorescu, R., & Ramnath, R. (2019). Accident Risk Prediction based on Heterogeneous Sparse Data [Paper presentation]. 27th ACM SIGSPATIAL, International Conference on Advances in Geographic Information Systems.

[26] NOAA Weather Dataset. (n.d.). National Weather Service. https://www.weather.gov/

[27] Ioffe, S., & Szegedy, C. (n.d.). Batch Normalization: Accelerating Deep Network Training b y Reducing Internal Covariate Shift. arXiv.

[28] Kingman, D. P., & Ba, J. L. (2015). Adam: A Method for Stochastic Optimization. ICLR.