Accident Impact Prediction based on a deep convolutional and recurrent neural network model

Mahya Qorbani[a*], Seyed Poyan Sajjadi[a*], Sobhan Moosavi[b] and Erfan Hassannayebi[a‡]

[a]Department of Industrial Engineering, Sharif University of Technology, Tehran, Iran
[b]Department of Computer Science and Engineering, Ohio State University, Columbus, Ohio

ARTICLE HISTORY
Compiled November 27, 2023

ABSTRACT
Traffic accidents pose a significant threat to public safety, resulting in numerous fatalities, injuries, and a substantial economic burden each year. The development of predictive models capable of real-time forecasting of post-accident impact using readily available data can play a crucial role in preventing adverse outcomes and enhancing overall safety. However, existing accident predictive models encounter two main challenges: reliance on either costly or non-real-time data and the absence of a comprehensive metric to measure post-accident impact accurately. To address these limitations, this study proposes a deep neural network model known as the *cascade* model. It leverages readily available real-world data from Los Angeles County to predict post-accident impacts. The model consists of two components: Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN). The LSTM model captures temporal patterns, while the CNN extracts patterns from the sparse accident dataset. Furthermore, an external traffic congestion dataset is incorporated to derive a new feature called the "accident impact" factor, which quantifies the influence of an accident on surrounding traffic flow. Extensive experiments were conducted to demonstrate the effectiveness of the proposed hybrid machine learning method in predicting the post-accident impact compared to state-of-the-art baselines. The results reveal a higher *precision* in predicting minimal impacts (i.e., cases with no reported accidents) and a higher *recall* in predicting more significant impacts (i.e., cases with reported accidents).

KEYWORDS
Accident prediction; LSTM; Machine learning; Deep learning

1. Introduction

Traffic crashes are a major public safety concern, leading to significant fatalities, injuries, and economic burdens annually. In 2020, the fatality rate on American roads saw a staggering 24% increase compared to 2019, the largest year-over-year rise since 1924, according to the National Safety Council (NSC). Understanding the causes and impacts of traffic accidents is crucial for evaluating highway safety and addressing negative consequences, including high fatality and injury rates, traffic congestion, carbon emissions, and related incidents. Despite ample existing literature, the development of accurate and efficient accident prediction models remains a challenging and fascinating research area.

Over the past decades, a wide range of predictive and computational methods have been proposed to predict traffic accidents and their impacts [1]. These models progressed from linear to nonlinear models and from traditional predictive regression models [2, 3, 4] to

*Both authors contributed equally
‡Corresponding author: hassannayebi@sharif.edu

today's most common data analytic and machine learning algorithms [5, 6, 7]. A majority of the studies treat the problem as a standard binary classification task [8, 9], with the output being a categorical event (accident or non-accident) and with no information on the post-accident consequences. Furthermore, existing studies that investigated post-accident impacts mostly failed to provide a compelling way to define and measure the "impact", which is a significant challenge. The commonly used metric to measure and illustrate the impact of accidents on traffic flow is "duration". However, to our understanding, this metric is not ideal because it can be influenced by various environmental factors, including the type of road and accessibility conditions. For example, if a road has accessibility issues, it may take longer to clear an accident scene, resulting in a longer duration. Furthermore, the datasets typically utilized for these studies are not publicly available and cannot be employed in real-time projects.

In recent years, hybrid neural networks that combine the strengths of multiple neural network components have garnered considerable attention in academic circles. To address the aforementioned issues and propose a practical solution for predicting the impact of accidents, we suggest the use of a LSTM-CNN cascade model. This approach harnesses the capabilities of recurrent neural networks (RNNs), specifically Long Short-Term Memory (LSTM), and convolutional neural networks (CNNs). LSTM is well-known for its ability to handle temporal relationships and capture long-term dependencies, while CNN excels at extracting spatial and temporal patterns from spatiotemporal event data, such as accident events [10]. Moreover, we introduce a novel metric that defines the impact of accidents on surrounding traffic flow, which addresses an additional challenge that existed in previous studies.

LSTM and CNN models are sequentially combined in terms of a "cascade" model. We specifically chose these two models because of their ability to handle temporal dependencies effectively. Furthermore, their combination offers a stronger capability to distinguish between accident and non-accident events, especially in imbalanced datasets, compared to using each model individually. Our model performs better in discriminating high-impact accidents. It achieves this by first distinguishing between accident and non-accident events and subsequently predicting the post-accident impact of the accident events. In more details, the cascade model utilizes a sequence of the last T time intervals (e.g., a two-hour interval) within each specific *region* (e.g., a zone specified by a square of size 5km $\times$ 5km on map) to predict the probability of an accident and its post-impact in that region for the next time interval (i.e., $T + 1$).

This study utilizes four complementary, easy-to-obtain, real-world datasets to build and evaluate the proposed prediction framework. The first dataset provides real-world accident data across the United States; it contains location, time, duration, distance, and severity. The second dataset offers detailed weather information such as weather conditions, wind chill, humidity, temperature, etc. The third dataset contains spatial attributes for each district (e.g., the number of traffic lights, stop signs, and highway junctions). The fourth dataset is a traffic congestion dataset that provides real-world congestion data across the United States. The congestion dataset is also used to develop a novel target feature, *gamma*, representing accident impact on its surrounding area. We propose a data-driven solution to define gamma as a function of three accident-impact-related features, *duration*, *distance*, and *severity*. Generally, non-accident intervals are far more prevalent in the real world compared to accident intervals (since accidents are relatively infrequent events). Therefore, working with such an imbalanced dataset poses challenges for model development and evaluation. To address this issue, we employ a random under-sampling method to mitigate the imbalanced nature of the dataset. Using these settings and through extensive experiments and evaluations, we demonstrate the effectiveness of the cascade model in predicting post-accident impact compared to state-of-the-art models that were evaluated using the same datasets. It is worth noting

that the datasets used in this research are generally easy to obtain, regarding how they have been collected (more details can be found in [11]) and the replicability of the data collection processes for other regions.

The main contributions of this paper are summarized as follows:

- **A real-world setup for post-accident impact prediction**: This paper proposes an effective setup to use easy-to-obtain, real-world data resources (i.e., datasets on accident events, congestion incidents, weather data, and spatial information) to estimate accident impact on the surrounding area shortly after an accident occurs. Also, this framework employs a data augmentation approach to obtain accurate feature vectors from heterogeneous data sources.
- **A data-driven label refinement process**: This study illustrates a novel feature to demonstrate post-accident impact on its surrounding traffic flow. This feature combines three factors, namely "severity", "duration", and "distance", to create a compelling feature that we refer to as *gamma* in this work.
- **A cascade model for accident impact prediction**: This paper proposes a cascade model to employ the power of LSTM and CNN to efficiently predict the post-accident impacts in two stages. First, it distinguishes between accident and non-accident events, then it predicts the intensity of impact for accident events.

In summary, the research contributions, both in terms of application and solution framework, are designed to directly connect to the potential reader, the distinguishing aspects and distinctiveness of the accident impact prediction method.

The remainder of the article is organized as follows. Section 2 provides an overview of related studies, followed by the description of data in Section 3. Section 4 describes our accident labelling approach, and then the cascade model is presented in Section 5. Section 6 illustrates experimentation design and results, and Section 7 concludes the paper by discussing essential findings and recommendations for future studies.

2. Related Work

Traffic accident analysis and prediction have been extensively studied over the past few decades. Previous work can be classified into two categories: *Accident Risk Prediction* and *Accident Impact Prediction*. In the following we overview some of the important studies in each category.

2.1. Accident Risk Prediction

These studies focused on predicting the risk of an accident itself, not consequences or any features of it. Two primary goals of these studies are 1) predicting the anticipated number of traffic accidents (i.e., accident rate prediction) [12, 13], or 2) predicting whether an accident will occur on a particular road section or geographical region or not (i.e., accident detection). In the first category, researchers applied regression models to predict a value as accident rate [14, 15, 16], while binary classification methods are used for the second category [17, 18, 19]. Some studies focused on temporal dependencies between accidents to predict the probability of accidents for the subsequent intervals [20, 21, 22]. In contrast, some others (e.g., [15, 23]) also sought to consider spatial dependencies.

As one of the latest studies, Kaffash et al. [24] designed an accident risk map using regression neural network tuned with self-organizing map technique which is able to estimate accident risk along 3008 points in a dual carriageway with more than 90% average accuracy. Yuan and Abdel-Aty [25] used a support vector machine (SVM) for real-time crash risk

evaluation. They used a classification and regression tree (CART) model to select the most important explanatory variables, and fed them to the SVM model with a radial-basis kernel function. They achieved 0.81 as their best model's Area Under the Curve (AUC) value, tested on I-70 freeway crash data from October 2010 to October 2011 provided by Colorado Department of Transportation, and real-time traffic data collected by 30 Remote Traffic Microwave Sensor (RTMS) radars. In [26], a novel feature selection algorithm based on a frequent pattern tree model was implemented to identify all the frequent patterns in the accident dataset, then ranked variables by their proposed metric called Relative Object Purity Ratio (ROPR). Finally, two classification models (k-nearest neighbor model and a Bayesian network) were used to predict real-time accident risk.

With the growing diversity and complexity of data, applying deep neural network (DNN) methods usually resulted in more accurate and robust predictions when compared to statistical methods or traditional machine learning models [27, 28]. Moosavi et al. [29] presented a deep neural network model (DAP) using multiple components, including a recurrent, a fully connected, and an embedding component for accident detection on their country-wide dataset. Li et al. [10] used a long short-term memory convolutional neural network (LSTM-CNN) to predict real-time crash risk on arterial roads using features such as traffic flow characteristics, signal timing, and weather condition. Their proposed model outperformed baseline models based on AUC value, sensitivity, and false alarm rate. Chen et al. [16] used a set of 300K accident records collected from GPS mobility data to predict a parameter g, defined as the summation of accident severity values in square areas of size $500m \times 500m$ during subsequent hour. They applied a stack denoising autoencoder model to extract latent features and then used a logistic regression model to predict the g parameter. Bao et al. [30] proposed a spatiotemporal convolutional long short-term memory network (STCL-Net) for predicting citywide short-term crash risk. Their results indicated that spatiotemporal deep neural network approaches perform better than other models to capture the spatiotemporal characteristics based on their multi-source dataset of New York City crashes. Ren et al. [14] collected extensive traffic accident data from Beijing in 2016 and 2017. They built a deep neural network model based on LSTM for predicting the number of traffic accidents (they called it "traffic accident risk index") occurring in a specific zone during the subsequent time interval. One limitation of their study is utilizing the traffic accident count for prediction and not using any other related data (such as traffic flow, human mobility, or road characteristics). In another study, Chen et al. [31] proposed a novel Stack Denoise Convolutional Auto-Encoder algorithm to predict the number of traffic accidents occurring at the city-level. They used two different datasets, one consisting of accident data and another one consisting of traffic flow collected by vehicle license plate recognition (VLPR) sensors. A summary of above studies can be found in Table 1.

2.2. Accident Impact Prediction

The other group of studies focused on analyzing and predicting the post-accident impact on the surrounding area. These studies aim to build or directly use features that best indicate the impact of accidents. Most of the existing studies used the duration or severity of the accident to define impact. They have employed various models for accident impact prediction, ranging from regression methods [32, 33, 34, 35] to neural network-based solutions [36, 37, 5]. ZHANG et al. [2] used the data of loop detectors and accidents recorded by police or traffic authorities to find traffic accident duration and defined duration of accident plus clearance time as the impact of an accident. They employed two models for impact prediction, multiple linear regression and artificial neural network (ANN). Yu et al. [38] obtained traffic incident duration occurred on a freeway from binary features like good or bad weather, day

Table 1.: A summary of previous works on accident risk prediction

Study	Input features	Predicted output	Model	Best modeling results
Moosavi et al. [29]	1.Weather conditions 2.Time of day 3.Location 4. Road features	Accident risk on different road segments	DNN with recurrent, fully connected and embedding components)	F1 score=0.59 for accident class F1 score=0.89 for non-accident class
Li et al. [10]	1.Traffic flow characteristics 2.Signal timing 3.Weather condition	Real-time crash risk	LSTM-CNN	False alarm rate=0.132 AUC=0.932
Parsa et al. [17]	1.Traffic 2.Network 3.Demographic 4.Land use 5.Weather feature	Occurrence of accidents	eXtreme Gradient Boosting (XGBoost)	Accuracy= close to 100%, Detection rate = varies between 70% and 83%, false alarm rate is less than 0.4% .
Yu and Abdel-Aty [25]	1.Crash data 2.Real-time traffic data	Crash occurrence	SVM with RBF kernel	AUC = 0.74 for all crash type, AUC = 0.80 for Multi-vehicle crashes, AUC = 0.75 for Single-vehicle crashes (evaluated on 30% of the whole dataset)
Ozbayoglu et al. [9]	1. Average velocity 2.Occupancy 3.The capacity difference between time t and t+1 4.Weekday/weekend 5.Rush hour	Crash occurrence	Nearest neighbor (NN), Regression tree (RT), feedforward neural network (FNN),	Accuracy=95.12 for NN Accuracy=97.59 for RT Accuracy=99.79 for FNN

or night, disabled vehicle, and peak hour. They compared results obtained from two models, ANN and SVM. The ANN model comprehensively provided better results for long-duration incident cases, while SVM performed better for short and medium-duration incidents. [39] implemented gradient boosting decision trees to predict freeway incident clearance time based on different explanatory variables. Compared to baseline models (SVM, NN, and random forest), their model resulted in superior performance in terms of interpretation power and prediction accuracy. [40] is one of the few accident impact prediction studies that focused on directly forecasting post-accident traffic flow; however, their data was limited to a single route rather than the entire road network of a city or an entire area. In one of the latest studies, [41] applied three machine learning methods (i.e., SVM, NN, random forest) to predict the duration of different traffic condition states after traffic accidents and considered the updating effect by adding newly acquired data to the prediction. The above studies are summarized in Table 2.

To the best of our knowledge, this study is the first that formulates post-accident impact on the surrounding area by exploiting a variety of signals such as duration, severity, and road blockage distance. This is an essential step toward providing a more comprehensive definition of impact, which addresses an important shortcoming in previous studies. Additionally, as opposed to those studies that only used data for a small region or a limited set of routes, we implement and evaluate the prediction model based on a large area (i.e., Los Angeles County), resulting in more generalizable outcomes. Lastly, this work is based on easy-to-obtain, public datasets which can be acquired from publicly available resources such as real-time traffic data providers[1], historical weather data providers[2], and open-street-map (OSM)[3]; thus other researchers can conveniently replicate our model and results for comparative purposes. More

[1]Examples of providers are Microsoft BingMaps and MapQuest

[2]Example of provider is Weather Underground, visit `https://www.wunderground.com/history`

[3]Visit `https://www.openstreetmap.org`

Table 2.: A summary of previous works on accident impact prediction

Study	Input features	Predicted output	Model	Best modeling results
Zhang et al. [2]	1.Temporal 2.Spatial 3.Environmental 4.Traffic 5.Accident details	Total duration and clearance time	Multiple linear regression and ANN	MAPE=27.1% for total duration, MAPE=49.8% for clearance time
Yu et al. [38]	1.Night 2.Casualties 3.Peak hour 4.Bad weather 5.Facility damage 6.Disabled Vehicle 7.Heavy tow truck 8.Lay-by occupied 9.Hazard material involved 10.Rollover vehicle involved	Incident duration	ANN and SVM	MAPE = 19%
Ma et al. [39]	1.Accident details 2.Temporal 3.Geographical 4.Environmental 5.Traffic 6.Operational	Incident clearance time	gradient boosting decision trees (GBDT)	MAPE=16% for clearance time less than 15min and MAPE=33% for clearance time more than 15min
Rose Yu et al. [40]	1.Accident type 2.Downstream post mile 3.Affected traffic direction 4.Traffic speed	The delay corresponding to the accident	Mixture Deep LSTM	MAPE = 0.97
Yu and Abdel-Aty [25]	1.Crash data 2.Real-time traffic data	Crash occurrence	SVM	AUC = 0.74 for SVM with RBF for all crash type
Lin and Li [41]	1.Accident details 2.Traffic details 3.Environmental 4.Air quality 5.Geographical	TAPI (derived from post-accident congestion level and its duration)	NN, SVM, RF	MAPE = 5.5%–53.8%
Wang et al. [36]	1.Vehicle type 2.Location 3.Time of day 4.Report mechanism	Vehicle breakdown duration	fuzzy logic (FL) and ANN	RMSE=24 for FL model RMSE=19.5 for ANN mode version one RMSE=24.1 for ANN mode version two

details about the process of collecting and building our datasets can be found in [11].

3. Dataset

This section describes all datasets used to build our accident impact prediction framework. In addition to describing original datasets, the processes of data cleaning, transformation, and augmentations are also described. Lastly, we briefly study accident duration distribution and compare it with findings by other researchers. It is worth noting that accident duration refers to the time-span that takes to clear the impact of an accident, and we believe it is highly correlated with accident impact. Hence further studying it would help to derive valuable insights when building the predictive model.

3.1. Data Sources

The data sources and study areas are related to the Los Angeles county, the most accident-prone county in the United states[4]. Four datasets are used to deliberate all factors involved in the aftermath of accident: *accident* dataset, *congestion* dataset, *point of interest (poi)*

[4]See `https://www.iihs.org/topics/fatality-statistics/detail/state-by-state` for more details.

dataset, and *weather* dataset. The following sub-sections describe these datasets in detail.

3.1.1. Accident Dataset

This study adopts the US-Accident dataset, a large-scale traffic accident dataset that has been collected from all over the United States [42] between 2016 and 2020. The dataset contains accident events collected from two sources: MapQuest Traffic and Microsoft BingMaps, and it covers 49 states of the United States. 73,553 accident records are extracted that cover four years from August 2016 to December 2020. Figure 1 shows the dispersion of accident location in the area of study that covers both freeway and urban arterial roads. Features of reported accidents are listed in Table 3. The reader can find a detailed description of features at `smoosavi.org/datasets/us_accidents`.

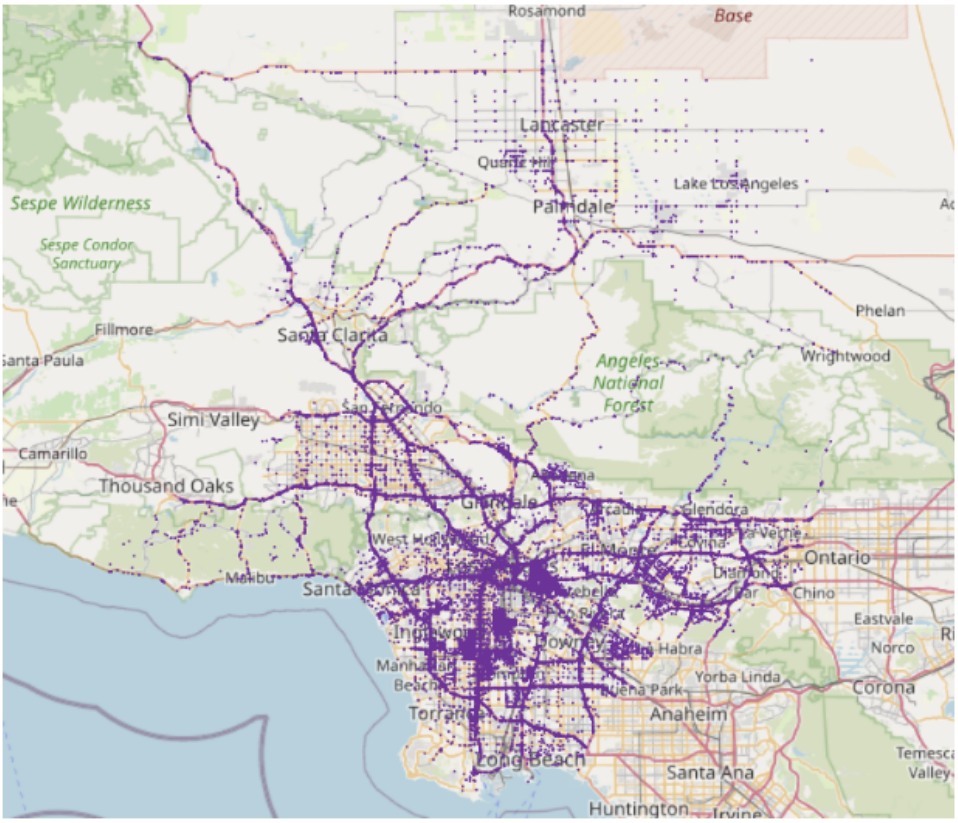


Figure 1.: Spatial dispersion of accident locations in Los Angeles county (2016-2020)

Table 3.: Accident dataset features

Category	Features
Geographical features	Start latitude, Start longitude, End latitude, End longitude, Distance, Street, Number, Airport code, City, Country, State, Zip code, Side, Amenity, Bump, Crossing, Give way, Junction, No-exit, Railway, Roundabout, Station, Stop, Traffic Calming, Traffic Signal, Turning Loop
Temporal features	Start time, End time, Time zone, Sunrise Sunset, Civil Twilight, Nautical Twilight, Astronomical Twilight
Other	ID, Severity, TMC

3.1.2. Congestion Dataset

Traffic congestion events are essential to predict and also to measure accident impact. The congestion dataset provided in [11] that includes such incidents over the same period and location is considered as the accident data. The dataset includes over 600,000 congestion events, collected between 2016 and 2020, where each event is recorded when traffic speed is slower than typical traffic speed. In addition to the basic details (such as the time and location of a congestion case), the dataset also offers other valuable details such as delay and average speed (the latter is in the form of natural language description of an event reported by a human agent). *"Severe delays of 22 minutes on San Diego Fwy Northbound in LA. Average speed ten mph."* is an example of the description attribute, that offers details such as the exact "delay" and "speed" of the congestion. One can easily extract such details from the free-form text via regular expression patterns. In this work, we use congestion data for accident impact labeling that is thoroughly discussed in section 4.

3.1.3. POI Dataset

As previous studies have shown, road features have significant impact on determining the possibility of accidents [29]. As a result, road-network features must be considered when predicting and measuring the impact of accidents. This study utilizes a POI dataset taken from Open Street Map (OSM) that provides geographical features of each location, such as the number of railroads, speed bumps, traffic signals, and pedestrian crossings.

3.1.4. Weather Condition Dataset

Numerous studies had shown the significant impact of weather condition on accident prediction [43, 44, 45]. This research considers the weather data that had been collected from weather stations located in four airports in Los Angeles area (i.e., LAX [5] , BUR[6] , VNY[7], and KWHP[8]). The four airports listed above offer hourly weather characteristics for the period of study. Each record is represented by a vector $e = \langle$*Airport, Date, Hour, Temperature, Wind Chill, Humidity, Pressure, Visibility, Wind Speed, Wind Direction, Precipitation, Weather Conditions*$\rangle$, where the weather conditions and weather directions are categorical features (all their possible values are shown in Table 4).

Table 4.: Details of categorical weather features

Features	**Unique Values**
Weather Conditions	'Mostly Cloudy', 'Scattered Clouds', 'Partly Cloudy', 'Clear', 'Light Rain', 'Overcast', 'Heavy Rain', 'Rain', 'Haze', 'Patches of Fog', 'Fog', 'Shallow Fog', 'Thunderstorm', 'Light Drizzle', 'Thunderstorms and Rain', 'Cloudy', 'Fair', 'Mist', 'Mostly Cloudy / Windy', 'Fair / Windy', 'Partly Cloudy / Windy', 'Light Rain with Thunder'
Wind directions	'E', 'W', 'CALM', 'S', 'N','SE', 'NNE', 'NNW', 'SSE', 'ESE", 'NE', 'NW', 'WSW', 'ENE', 'SW', 'SSW', 'WNW'

[5]Los Angeles International Airport
[6]Hollywood Burbank Airport
[7]Van Nuys Airport
[8]Whiteman Airport

3.2. Preprocessing and Preparation

This section describes the required steps to preprocess different sources of data that are used in this work. Further, the proposal to build input data by combining different sources, to be used for modeling is decribed. In terms of preprocessing, we can elaborate the following steps:

- **Remove duplicated records**: Since the accident data is collected from two potentially overlapping sources, some accidents might be reported twice. Therefore, we first process the input accident data to remove duplicated cases.
- **Fill missing values using KNN[9] imputation**: The weather data suffers from missing values for some of features (e.g., Sunrise_Sunset). Based on two nearest neighbors, we impute missing values with mean (for numerical features) or mode (for categorical features). We use time and location to determine distance when finding neighbors.
- **Treat outliers**: For a numerical feature f, if $f_i > \mu_f + 3\sigma_f$ or $f_i < \mu_f - 3\sigma_f$, then we replace it with $\mu_f + 3\sigma_f$ or $\mu_f - 3\sigma_f$, respectively. Here μ_f and σ_f are mean and standard deviation of feature f, respectively, calculated on all weather records.
- **Omit redundant features**: We remove those features that satisfy either of the following conditions: 1) correlated features based on Pearson correlation; 2) categorical features with more than 90% data frequency on a specific value.
- **Discretize data**: After data cleaning, both accident and congestion data are first discretized in space and time. The temporal resolution is 2-hour intervals and spatial resolution is set to 5km×5km squares in uniform grids.

A vector like $l(s.t) \in R^m$ is used to represent the input data at time t in region s, applied for accident impact prediction[10]. In this way, we convert raw data to $l(s.t) \in R^{26}$ which consists of 5 categories of features that include 26 different features obtained from raw datasets. Table 5 describes details of features used to create an input vector. In addition, due to the rarity of accident events and data sparsity, we drop those zones for which the total recorded accidents is less than a certain threshold α during the two years. By setting $\alpha = 75$, which means dropping zones with less than 75 reported accidents in 2 years (i.e., the accidents which were reported in less than 0.8% of time intervals), the ratio of time intervals with at-least one reported accident (also known as accident-intervals) to all intervals increases by 4.8%. In section 5.2 an under-sampling method is discussed to balance accident to non-accident intervals ratio to a larger extent.

Table 5.: Selected features after preprocessing and preparation

Feature Category	Feature
Temporal	Day of Week, Part of Day (day/night), Sunrise/Sunset
Weather	Weather Condition
Accident	Severity, Accident Count, Duration, Distance
Congestion	Congestion Count, Congestion Delay
Spatial	Geohash Code, latitude, longitude, Amenity, Bump, Crossing, Give way, Junction, No-exit, Railway, Roundabout, Station, Stop, Traffic Calming, Traffic Signal, Turning Loop

[9]Nearest Neighborhoods

[10]This vector represents input for the predictive task, and later we describe labeling.

3.3. Data Augmentation

The initial accident dataset includes a variety of features. However, we only use highly accident-relevant features, and further augment the data with additional features described in section 3.1 to create the input feature vector. Road-network characteristics of regions are added to differentiate between various types of urban and suburban regions, congestion-related information is added to empower the model in finding latent patterns between accident and congestion events, and weather data collected from the nearest airport based on accident's occurrence time and location is used to further help the model by encoding weather condition data. Figure 2 summarizes how the input feature vector is built in this study.

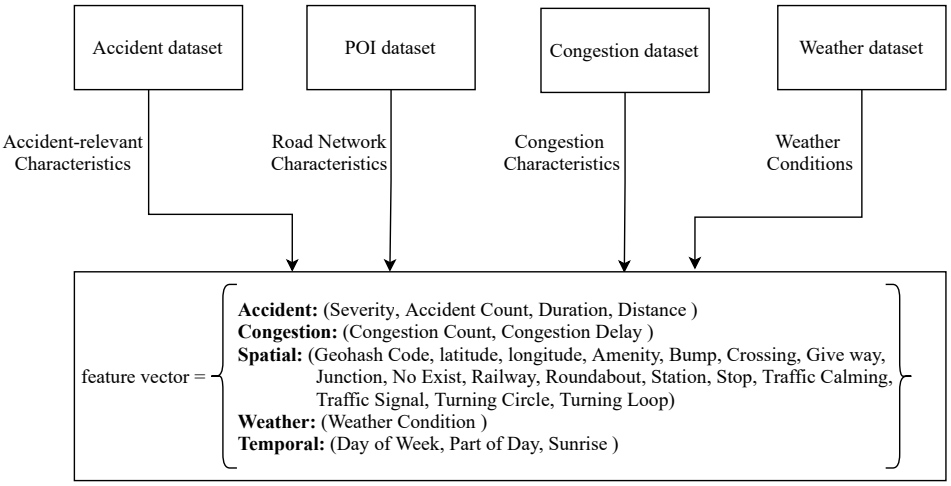


Figure 2.: Forming a feature vector at time t in region s based on heterogeneous data obtained from various data sources

To define a proper accident prediction task over geographical areas during different time intervals, it is required to build input data for accident-free time intervals. The data are discretized in space and time in order to build these non-accident (or accident-free) intervals (described in Section 3.2). When no accident is reported for an interval, accident-related features (i.e., severity, accident count, duration and distance) are set to zero, and remaining features (i.e., geographical, temporal, surrounding congestion and weather condition) are filled accordingly using available resources.

3.4. Accident Duration Distribution

Accident duration is an essential factor to determine impact. Thus, it is worth studying this factor to see how its distribution in our dataset is aligned with datasets that were used in the literature. Previous studies found that a log-normal distribution can best model accident duration [2, 46, 47].

This section investigates this phenomena using the input data and compares the research findings with [2]. First, we divide the duration dataset into K classes based on Doane's formula [46]:

$$K = log_2 n + log_2(1 + \frac{|g1|}{\sigma_{g1}})$$

Where n stands for the total number of observations and $g1$ refers to the estimated 3rd-

moment-skewness of the observations.

$$\sigma_{g_1} = \sqrt{\frac{6(n-2)}{(n+1)(n+3)}}$$

Class $C_i, i \in 1, 2, \cdots, K$ is specified by its upper bound $(min(D) + i \times \frac{max(D)-min(D)}{K})$ and lower bound $(min(D) + (i-1) \times \frac{max(D)-min(D)}{K})$ where D is accident durations. Then, the frequency of each class is calculated and summation of the squared estimate of errors between frequency of each class and probability density function of fitted distribution is calculated.

The sum of the squared estimate of errors (SSE) calculated for four selected distributions is shown in Figure 3. Log-normal and log-logistic distributions have the lowest SSE and therefore describe accident duration data the best. This is aligned with findings reported by Zhang et al. [2]. They used AIC (Akaike Information Criterion) and the BIC (Bayesian Information Criterion). They found log-normal and log-logistic distributions to be the first and the second best-fitted distributions, respectively.

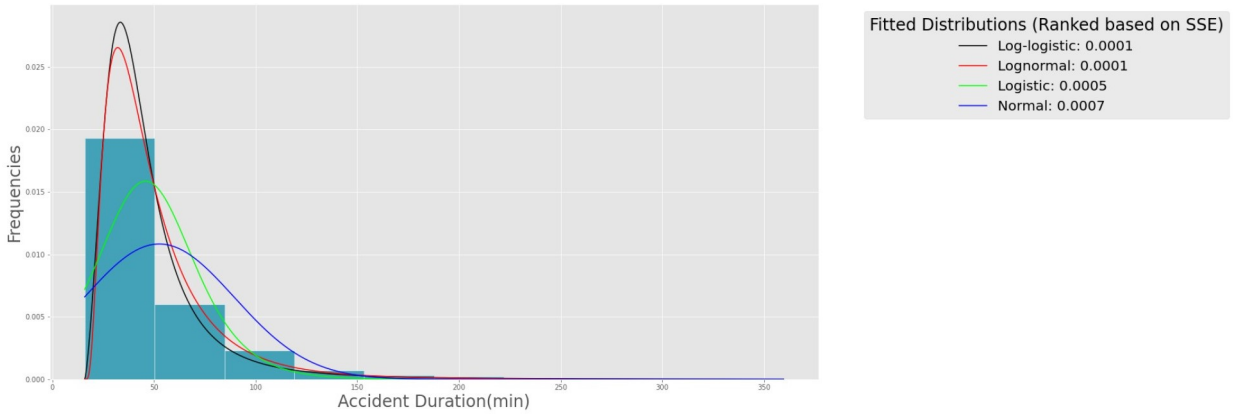


Figure 3.: Four best fitted distributions on accident duration data and their summation of squared estimate of errors

4. Label Development of Post-accident Impact

The previous section discussed data sources, pre-processing, and augmentation steps. However, we have not yet described "accident impact" in terms of a quantitative feature or attribute. To the best of our knowledge, there is no exact parameter to show the impact of the accident on its surrounding traffic flow in the real world. There are a number of studies addressing the effect of traffic congestion on accidents [48, 49, 50], while the effect of traffic accidents on traffic congestion has been rarely studied [51, 23]. Of those few studies, some researches used accident duration as an indicator for accident's impact, while some others studied the effects of speed changes in surrounding traffic flow as an indicator. We believe none of these approaches are comprehensive and genuine enough to describe how traffic flow would be affected by the occurrence of an accident.

In this study, we propose a novel "accident impact" feature based on the delay caused by accidents. This is done by finding a function $\mathcal{F}$ from congestion dataset to estimate delay on accident dataset. In the following , detailed explanation of the process can be found.

4.1. Accident Impact: A Derived Factor

There are three attributes in our dataset that can be used to determine "impact":

- **Severity**: this is a categorical attribute that shows severity in terms of delay in free-flow traffic due to accidents. Although it seems to be a highly relevant feature, it suffers from skewness when looking at its distribution (see Figure 4), and being a coarse-grained factor (i.e., it is represented by just a few categorical values).

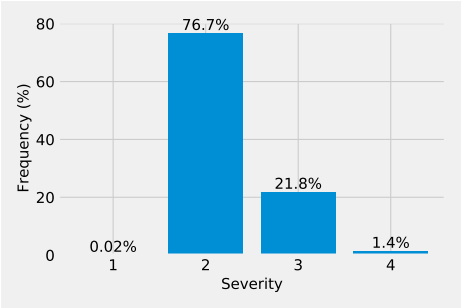


Figure 4.: Distribution of Severity values in accident dataset

- **Duration**: duration of a traffic accident shows the period from when the accident was first reported until its impact had been cleared from the road network. In this sense, the duration can be considered as another factor to determine impact. Please note that a long duration does not necessarily indicate a significant impact, as it could be related to the type of location and accessibility concerns. But generally speaking, duration is positively correlated with impact.
- **Distance**: distance shows the length of road extent affected by accident. Similar to duration, a long distance does positively correlate with higher impact. But a high impact accident may not necessarily result in impacting a long road extent.

All three attributes can, to some extent, represent the impact of an accident. However, each alone may not suffice to define the impact primarily. Thus, we propose to build a model that can estimate accident impact given these features as input.

4.2. Delay as a Proxy for Impact

To estimate accident impact, a function $\mathcal{F}$ is defined that maps the three input features to a target value γ which we refer it as "impact":

$$\gamma = \mathcal{F}(Severity, Duration, Distance)$$

Now the question is: how to find $\mathcal{F}$? While there is no straight-forward approach to estimate $\mathcal{F}$ based on accident data, if we choose to fit it on congestion data (see Section 3.1.2), we can use some extra signals from the input to estimate the desired function. For congestion events, our data offers a human-reported description of incidents that includes "delay" (in minutes) with respect to typical traffic flow (see Table 6 for some examples). To the best of our knowledge, the "description" is generated by traffic officials in a systematic way, therefore it is reliable and accurate. Suppose that the impact in congestion domain is represented by delay (which is a fair assumption given our definition of impact and what delay represents), we can fit $\mathcal{F}$ on congestion data using *delay* as target. Please note that "delay" is only available in the congestion dataset, and here the goal is to derive it for accidents and use it as target value.

Table 6.: Examples of congestion events and features to build function F

Event ID	Description	Severity	Duration (Minute)	Distance (Mile)
1	*Delays increasing and delays of nine minutes on Colorado Blvd Westbound in LA. Average speed five mph.*	Slow	49.6	2.42
2	*Delays of three minutes on Harbor Fwy Northbound between I-10 and US-101. Average speed 20 mph.*	Moderate	43.5	3.18
3	*Delays of eight minutes on Verdugo Rd Southbound between Verdugo Rd and Shasta Cir. Average speed five mph.*	Slow	41.6	0.55
4	*Delays of two minutes on I-5 I-10 Northbound between Exits 132 132A Calzona St and Exit 135A 4th St. Average speed 20 mph.*	Fast	41.6	2.26

After fitting $\mathcal{F}$ on congestion events, it is used to estimate impact (represented by γ) for accidents. It is worth noting that, according to our data, congestion and accident events share the same nature, given their attributes and sources that have been used to collect them. Thus, fitting a function like $\mathcal{F}$ on one and applying it on another is a reasonable design choice.

4.3. Estimating Delay

The reported delays of congestion events are extracted from their description and used as our γ variables (i.e., target values). The aim is to find $\mathcal{F}$ by fitting different functions on data and select the best one with lower Mean Squared Error (MSE) and Mean Absolute Error (MAE) values. The overall process is described in Figure 5. Two models are used to estimate $\mathcal{F}$:

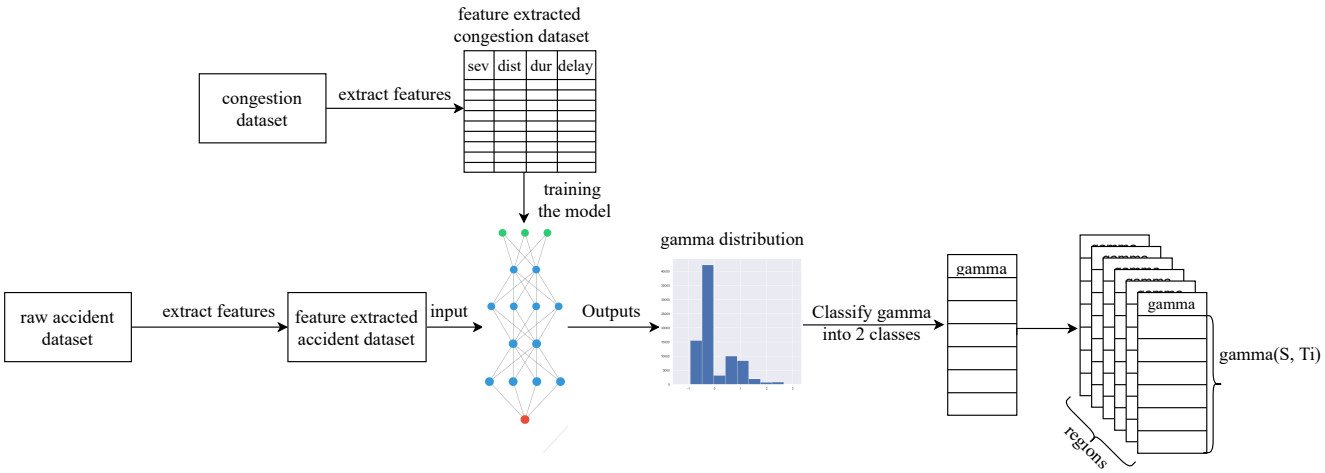


Figure 5.: The process of fitting $\mathcal{F}$ on congestion events and applying it on accidents events to predict impact (i.e., γ)

- **Linear regression (LR) model**: given our input features, a linear model seems a natural choice to estimate γ. So we use a linear regression model for this purpose.
- **Artificial Neural Network (ANN)**: to examine the impact of non-linearity to estimate γ, we also used a multi-layer perceptron network with four layers, each with three neurons, and a single neuron in the last layer. We used Adam optimizer [52] with

a learning rate of 0.0008 and trained the model for 200 epochs.

To train these models, in total 48,109 congestion records collected between June 2018 and August 2019 are used. The algorithm considers 85% of the data for the training and 15% for evaluation. Table 7 shows the results of different models to predict delay using the three input features. Based on these results, the non-linear model (i.e., ANN) is selected to estimate the delay (or γ).

Table 7.: Result of using ANN and LR models to predict delay (as a proxy for impact) on congestion data

Model	**ANN**	**LR**
MSE	**0.141**	0.153
MAE	**0.239**	0.255

In what follows, we discuss why it is reasonable to use the function fitted on congestion events and apply it to accident events. First, in our data, there is a resemblance between accidents and congestion events, given their attributes and sources used to collect them. Second, we define the reported delay of congestion as gamma, so with similar inputs from another type of traffic event, the output would be of the nature of delay, which in our opinion is a better indicator of the impact of a traffic accident in comparison to duration or speed changes.

After obtaining γ values for accident events, we categorize them into two different classes to show impact of accidents by more detectable labels: "medium severity" and "high severity"; a value of γ lower than the median is labeled as "medium severity" and a γ higher than median is labeled as "high severity". Although one could pick a finer-grained categorization of γ values (e.g., three or four classes), through the empirical studies, we found the choice of two categories best represent our data.

Please note that γ as a real value is not a proper target feature for accident impact prediction since there are numerous contributing factors determining the exact value of accident delay (in seconds or minutes) which cannot be collected in advance. Therefore, the aim is to classify γ into two classes (i.e., "medium severity" and "high severity") to deal with this natural drawback of accident impact prediction task.

5. Accident impact prediction methodology

This section describes the proposed model to predict accident impact. The algorithm design is inspired by spatiotemporal characteristics of the input. We leverage two major neural network components: convolutional neural network (CNN) and Long short-term memory (LSTM). While by the former we seek to efficiently encode all types of input attributes, especially the spatial ones, the latter component can efficiently encode temporal aspects of our data and leverage past observations to predict future accident impact. In the remainder of this section, we first briefly introduce some basic concepts, and then discuss the details of the proposed prediction model.

5.1. Basic concepts

5.1.1. Long short-term memory

The LSTM model proposed by Hochreiter and Schmidhuber [53] is a variant of the recurrent neural network (RNN) model. It builds a specialized memory storage unit during training

through a time backpropagation algorithm. It is designed to avoid the vanishing gradient issue in the original RNN. The key to LSTMs is the cell state, which allows information to flow along with the network. LSTM can remove or add information to the cell state, carefully regulated by structures called gates, including input gate, forget gate and output gate. The structure of a LSTM unit at each time step is shown in Figure 6. The LSTM generates a mapping from an input sequence vector $X = (X_1, X_t, \ldots, X_N)$ to an output probability vector by calculating the network units' activation using the following equations (t shows iteration index):

$$i_t = \sigma_g(W_{ix}X_t + W_{ih}h_{(t-1)} + W_{ic}c_{(t-1)} + b_i)$$

$$f_t = \sigma_g(W_{fx}X_t + W_{hf}h_{(t-1)} + W_{cf}c_{(t-1)} + b_f)$$

$$o_t = \sigma_g(W_{ox}X_t + W_{oh}h_{(t-1)} + W_{oc}c_t + b_o)$$

$$c_t = f_t \odot c_{(t-1)} + i_t \odot \sigma_c(W_{cx}X_t + W_{ch}h_{(t-1)} + b_c)$$

$$h_t = o_t \odot \sigma_g(c_t)$$

$$y_t = W_{yh}h_{(t-1)} + b_y$$

where X is the input vector, W and b are weight matrices and bias vector parameters, respectively, needed to be learned during training. σ_c,σ_g are sigmoid and hyperbolic tangent function, respectively, and $\odot$ indicates the element-wise product of the vectors. The forget gate f_t controls the extent to which the previous step memory cell should be forgotten, the input gate i_t determines how much update each unit, and the output gate o_t controls the exposure of the internal memory state. The model can learn how to represent information over several time steps as the values of gating variables vary for each time step.

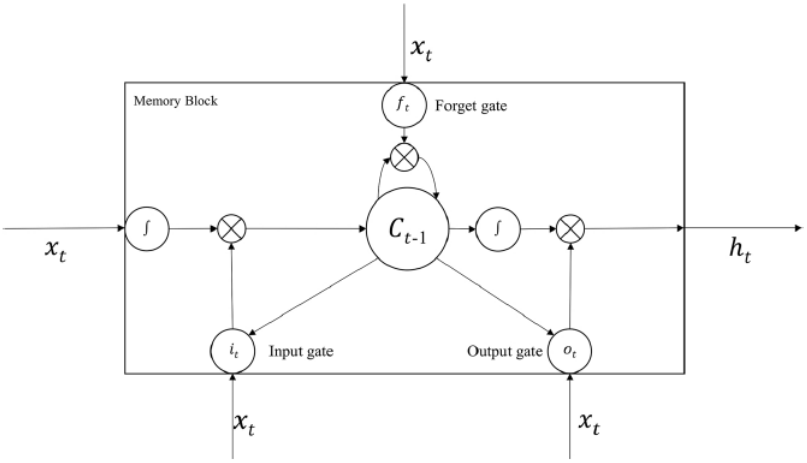


Figure 6.: LSTM unit structure

5.1.2. Convolution Neural Network

Convolution neural network (CNN) is a multi-layer neural network, which includes two important components for feature extraction: *convolution* and *pooling* layers. Figure 7 shows the overall structure of the convolution neural network. Its basic structure includes two special neuronal layers. The first one is convolution layer; the input of each neuron in this layer is locally connected to the previous layer, and this layer is to extract local features. The second layers is the pooling layer used to find the local sensitivity and perform secondary feature extraction [54]. The number of convolutions and pooled layers depends on the specific problem definition and objectives. Firstly, the model uses a convolutional layer to generate latent features based on the input (Figure 8a). Then, a sub-sampling layer is used on top of the convolutional output to extract more high-level features for the classification or recognition task (Figure 8b).

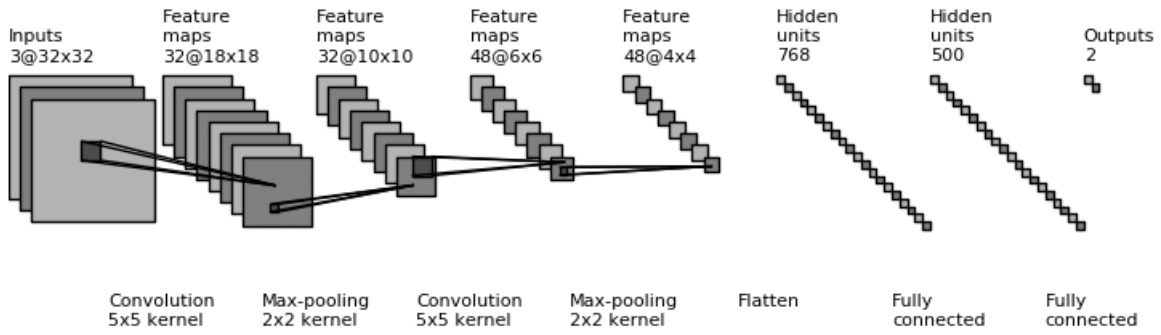
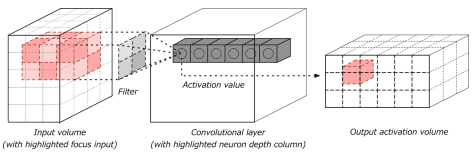


Figure 7.: Overall structure of the convolution neural network

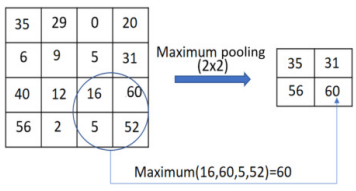
5.2. Model input and output

As described in section 3.2, we create a vector representation $l(s,t) \in R^{26}$ for an accident in geographical region s during time interval t. Each vector has a corresponding *Gammaclass* label that indicates the intensity of delay in traffic flow after an accident. For accident-free data vectors, we label them by 0, which means no congestion is likely to take place. While this might not be necessarily true in the real-world, it helps to simplify our problem formulation and label data vectors efficiently. The model predicts *gamma* for time $t+1$ in region s_i given sequence of w previous time intervals in region s_i. Mathematically speaking, $gamma_{(s_i,T+1)}$ is predicted given a sequence of $l(s_i,t), t \in \{T-w+1, T-w+2, \ldots, T\}$. Figure 9 shows the process of converting the dataset to a 3-D structure of $l(s_i,t)$ in time and space domain. *Gammaclass* is structured in the same order as $l(s_i,t)$. Figure 10 shows an arbitrary input sequence and its corresponding target in our 3-D data structure.

We choose accidents from February 2019 to August 2019 as the training time frame. This 27-week time frame includes 13,026 accident representations and 319,194 non-accident



(a) Structure of a convolution layer with input and output volumes



(b) Pooling layer operation

Figure 8.: Two main operations in CNN models to extract spatial latent patterns

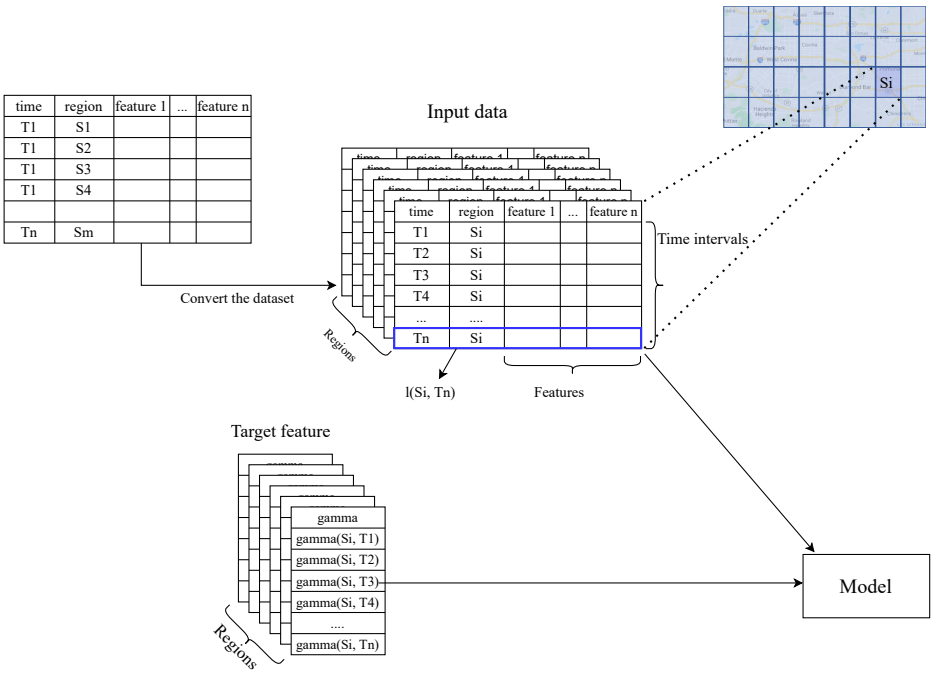


Figure 9.: Converting dataset into a 3-D structure of $l(s_i, t)$ and *Gammaclass* in time and space domains. Model inputs and outputs are extracted from this 3-D structure

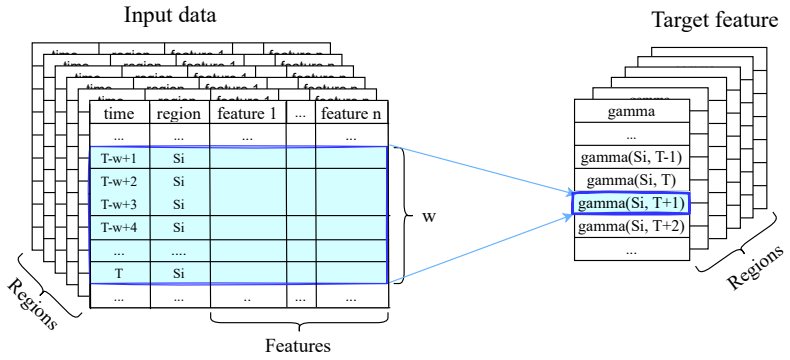
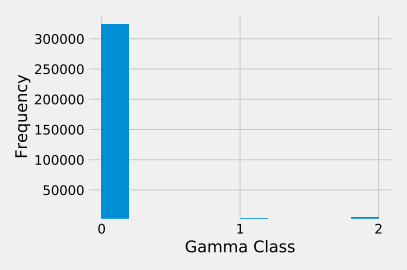
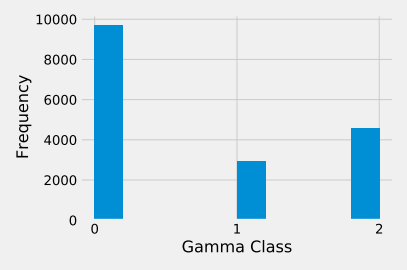


Figure 10.: A sequence of $l(s_i, t), t \in \{T - w + 1, T - w + 2, \ldots, T\}$ as model input and $Gammaclass_{(s_i, T+1)}$ as its corresponding target feature

representations. Frequency of each *gammaclass* is shown in Figure 11a, indicating that the data is highly imbalanced. Various methods exist to resolve the class imbalance problem in the context of classification or pattern recognition. Examples are (i) removing or merging data in the majority class, (ii) duplicating samples in minority classes, and (iii) adjusting the cost function to make misclassification of minority classes more costly than misclassification of majority instances. In this study, random under-sampling (RUS) method resulted in better outcomes. Hence we use it to mitigate the class imbalance problem of *gammaclasses*. Using this approach, ratio of accident to non-accident events has increased from 1/43.3 to 1/1.3. Figure 11b shows the frequency of each *gammaclass* after under-sampling. In addition, we assign a weight to each class in the loss function of models (and adjust them during training) to further address the imbalance issue. The following section describes class weighting in more details.



(a) Frequency of gamma classes before RUS



(b) Frequency of gamma classes after RUS

Figure 11.: Comparing frequency distribution of *gammaclass* before and after random under sampling

5.3. Model development

Since distinction between accidents and non-accidents is difficult due to complex factors that can affect traffic accident, and some factors that can not be observed and collected in advance (e.g., driver distraction), a model may not perform well on distinction of *gammaclass*es using single step prediction (i.e., just by using a single model). Hence, we propose a cascade model that includes two deep neural network components. It is a cascade model, meaning the output of the first model is served as input for the second model. The first component (or we can call it "layer" in our cascade design) focuses on detecting accidents from non-accident events; in other words, the first model detects if there will be an accident in the next two hours given w previous intervals information. If the first model classifies the input as an accident, then it goes to the second component. The second component focuses on accident impact prediction (i.e., *gammaclasses*) for accident events, detected by the previous layer. The structure of the proposed accident impact prediction model is shown in Figure 12. The structure of the two components in our design is described below.

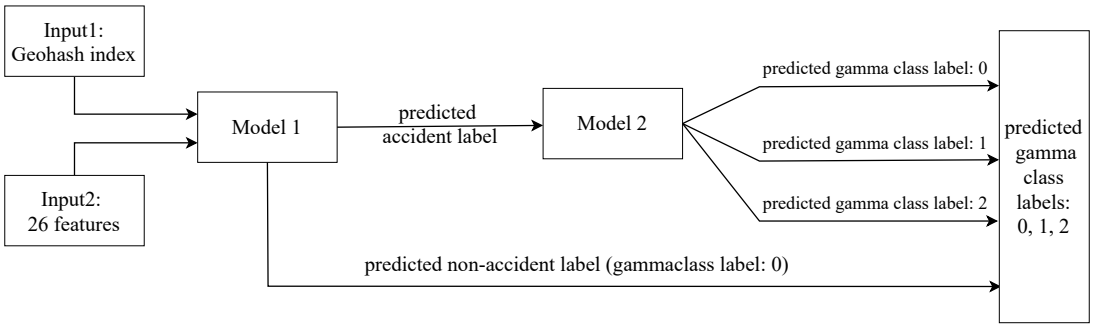


Figure 12.: Cascade model overview; the first model predicts whether the next interval would have an accident (i.e., label=1) or not (i.e., label=0). Next, predictions with label=1 are given to the second model that predicts the intensity of accident impact (i.e., *gammaclass*). Non-accident predictions of the first model are labeled as $gammaclass = 0$.

A) Label Prediction: The First model is a binary LSTM classifier that predicts whether the next interval would have an accident (i.e., label=1) or not (i.e., label=0). In accident prediction it is vital that if an accident is likely to happen, the model can predict it in advance. Therefore, in the first model the focus is on detecting accident events and we use a weighting mechanism for this purpose, such that the weight of the accident class is higher than the weight of non-accident class. Using grid-search, we found optimum weights to be 1 and 3 for non-accident and accident classes, respectively. The structure

of the first model is shown in Figure 13. For this model the input data consists of two components: a) index of a given zone s_i (i.e., $index_{s_i}$), and b) $l(s_i, t) - index_{s_i}$, for $t \in \{T - w + 1, T - w + 2, \ldots, T\}$. Here "subtraction" indicates using all features represented by $l(s_i, t)$ except the index of region s_i. The first component provides a distributed representation of that cell that encodes essential information in terms of spatial heterogeneity, traffic characteristics, and impact of other environmental stimuli on accident occurrence. It is fed to an embedding layer of size $|R| \times 20$, where R is the set of all grid-cell regions in input dataset. The second component provides other features as a set of w vectors, each of size 35. This component is fed into two LSTM layers with 12 and 24 neurons, respectively. The output of these layers is then concatenated and fed to two fully connected layers with 25 neurons each. Moreover, batch normalization and dropout layers are added to prevent vanishing gradients and overfitting, respectively.

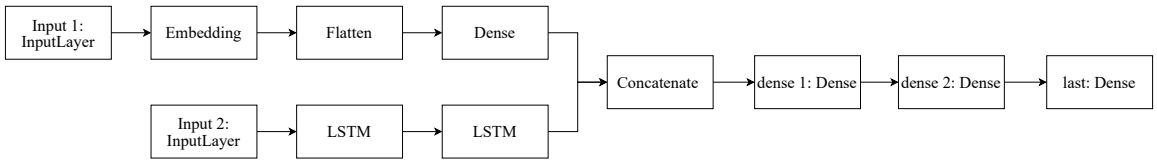


Figure 13.: Structure of the first model (layer) in the cascade model. Index of a given zone is fed into an embedding layer to extract latent features for that zone during training. The other features (e.g., weather condition, and congestion data) are fed into LSTM layers. The output of the aforementioned layers is then concatenated and inputted to fully connected layers to predict possibility of an accident.

B) Impact Prediction: The second model is a 3-class CNN classifier that predicts *gammaclass* for next interval. The weight of each class is assigned based on their frequency and importance in our problem setup. Optimum weights are found to be 0.7, 4.5 and 3.5 for $gammaclass = 0$, $gammaclass = 1$ and $gammaclass = 2$, respectively. The structure of the second model is shown in Figure 14. Input for this model is data predicted as an accident event (i.e., label=1) by the first model. The rationale behind considering $gammaclass = 0$ in the second model is that we assume the first model might misclassify some non-accident events, thus this can further ensure the quality of final outcome in case of any misclassifications.

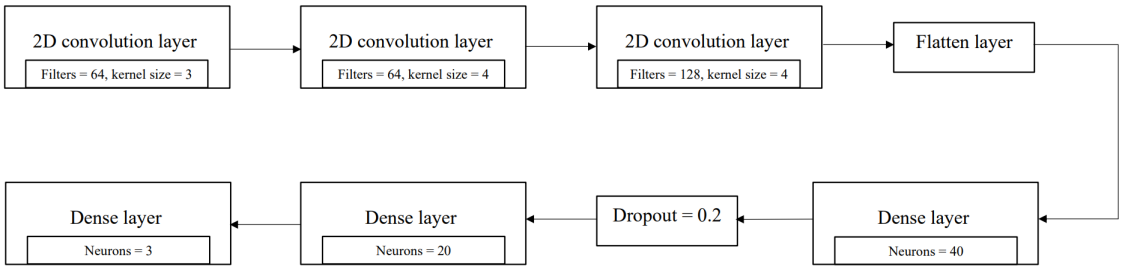


Figure 14.: Structure of the second model (layer) in the cascade model; three consecutive convolutional layers are used to extract spatial latent features from data and then three fully connected layers are used to convert the output of convolutional layers to probability vector for *gammaclasses*.

To the best of our knowledge, the proposed model is the first cascade model that can be applied in the real-world to classify time intervals as accident and no-accident, and then

predict severity of accidents using available data in real-time. Additionally, given the type of the used data, we believe this model can be used in real-world to serve and make compelling predictions.

6. Experiments and Results

In this section, we first describe our evaluation metrics, then provide details on experimental setup and lastly present our results followed by discussions.

6.1. Evaluation Metrics

Failing to predict an accident event (and the consequent impact on traffic flow) is more costly comparing to the case of falsely forecasting traffic delay in an area as a result of a false accident prediction. Hence, we focus on two objectives: 1) maximizing confidence when predicting a non-accident case (i.e., $gammaclass = 0$), and 2) minimizing possibility of failing to predict accident events (i.e., cases with $gammaclass = 1$ and $gammaclass = 2$). In terms of metrics, these objectives can be translated to 1) "precision" for $gammaclass = 0$, and 2) and "recall" for $gammaclass = 1$ and $gammaclass = 2$. Precision and recall for multi-class classification are formulated as below:

$$Precision_{class_i} = \frac{M_{ii}}{\sum_i M_{ij}}$$

$$Recall_{class_i} = \frac{M_{ii}}{\sum_j M_{ij}}$$

where M_{ij} is the number of samples with true class label i and predicted class label j. Therefore, we focus on precision for class "0" and recall for the other two classes for evaluation purpose. That being said, we still need to ensure reasonable recall for class "0" and acceptable precision for the other classes.

6.2. Experimental Setup

This section explains the experimental setup and the corresponding test runs. You can find the supplementary materials in our GitHub[11]. In this study, we used Keras, a Python-based Deep Learning library, to build the prediction models. We choose Adam [52] as the optimizer, given its characteristics to dynamically adjust the learning rate to converge faster and better. To find the optimal models' settings, we performed a grid search over choices of LSTM layers: {1, 2, 3}, number of neurons in recurrent layers: {12,18,24}, fully-connected layers: {1, 2}, and size of fully-connected layers: {12, 25, 50} for the first model; and a grid search over choices of convolutional layers: {1, 2, 3} and number of filters in each convolutional layer: {8, 16, 32, 64, 128} for the second model. The first model (i.e., LSTM) is trained for 150 epochs and the CNN model is trained for 25 epochs.

When building the input vectors, we can use past w intervals to build a vector and pass it to the cascade model to predict a label and a gamma class. But, what is the right choice of w? To answer this question, we ran an experiment on the test data to study the metrics introduced in the previous section on different classes for different choices of w. The results

[11] `https://github.com/mahyaqorbani/Accident-Impact-Prediction-using-Deep-Convolutional-and-Recurrent-Neural-Networks`

are shown in Figure 15. According to these results, a choice of 4 or 5 for w (which translates to having information from 8 to 10 hours before the accident) seems to be reasonable for this parameter. Here we choose to set $w = 4$, since it consistently provides reasonable results over all three gamma classes.

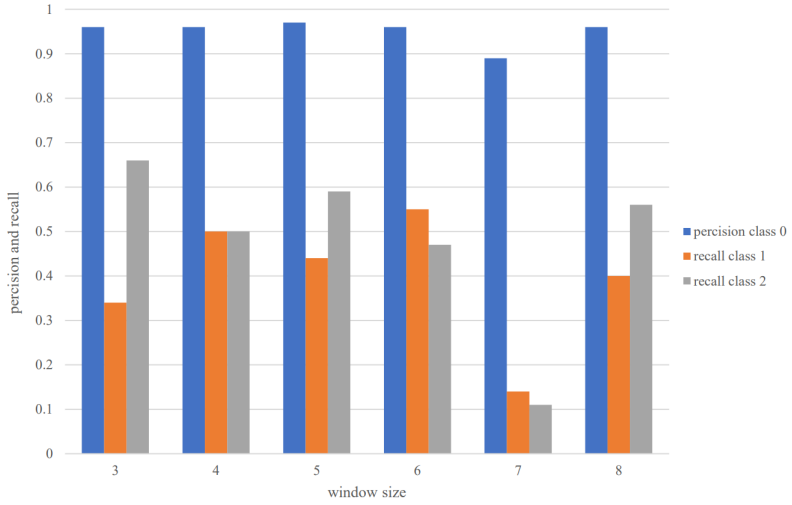


Figure 15.: Comparison of evaluation metrics (i.e., *recall* for classes 1 and 2, and *precision* for class 0) based on the cascade model for different choices of input length (or window size)

6.3. Baseline Models

In this section, random forest (RF), gradient boosting classifier (GBC), a convolutional neural network (CNN), and a long short-term memory (LSTM) are selected as the baseline algorithms. RF and GBC are traditional models that generally provide satisfactory results on a variety of classification or regression problems. Furthermore, as our proposed cascade model leverages LSTM and CNN components, the use of these two as standalone models seems to be a reasonable choice. We used Scikit-learn and Keras for the off-the-shelf implementation of the baseline models. As for hyperparameters, for the random forest we used 200 estimators, maximum depth=12, minimum sample split=2 and the rest of the parameters were set to defaults. For GBC, we used 300 estimators, learning rate=0.8, maximum depth=2, and the rest of the parameters were set to defaults. The baseline CNN and LSTM models share the same structure as in the proposed cascade model, except for the last layer of the LSTM that uses 3 neurons instead of 2 (since by this model we seek to predict three *gammaclass*es in a single step).

6.4. Results and Model comparison

This section provides the evaluation results on test data (from September 2019 to November 2019) to compare our proposal against the two traditional (i.e., RF and GBC) and deep-neural-network models (i.e., CNN and LSTM). A summary of the results is presented in Table 8 and Figure 16. From these results, we can see that the GBC model performed quite well on detecting non-accident events (i.e., *gammaclass*=0) based on precision for this class. However, its results to predict the other two classes are not satisfactory.

Further, the RF model does a better job in detecting accidents with medium or high impact. The LSTM model seems to perform better than the other models (except the cascaded

Table 8.: Accident impact prediction results based on precision for *gammaclass*=0 and recall for *gammaclass*es 1 and 2

models	**Precision class 0**	**Precision class 1**	**Precision class 2**	**recall class 0**	**recall class 1**	**recall class 2**
Gradient Boosting	0.92	0.14	0.04	0.81	0.14	0.23
Random Forest	0.93	0.13	0.05	0.69	0.31	0.30
LSTM	0.94	0.10	0.04	0.56	0.34	0.36
CNN	0.94	0.12	0.04	0.11	0.32	0.32
Cascade Model	**0.96**	0.10	0.04	0.31	**0.41**	**0.50**

model) in detecting accidents with medium or high impact, indicating the importance of taking into account the temporal dependencies to encode input data better. However, CNN's performance is not satisfactory as a single-step model.

While the proposed cascade model provides satisfactory results in terms of precision for *gammaclass*=0, it results in significantly higher recalls to predict the other two classes. These are important observations, because in the case of accident prediction, failing in predicting occurrence of an accident in a zone (i.e., low recall) can result in more serious outcome than falsely predicting accident in a non-accidental zone (i.e., low precision). Therefore, we generally care more about having higher recall to predict impact for accident events, and also a high precision to decide whether a time-interval is associated with an accident event or not. Please note that while high precision is still necessary for non-zero classes (i.e., cases with reported accident), a real-world framework must ensure acceptable recall for these cases.

Figure 16.: Comparing different models based on precision and recall to predict gamma-classes 0, 1, and 2.

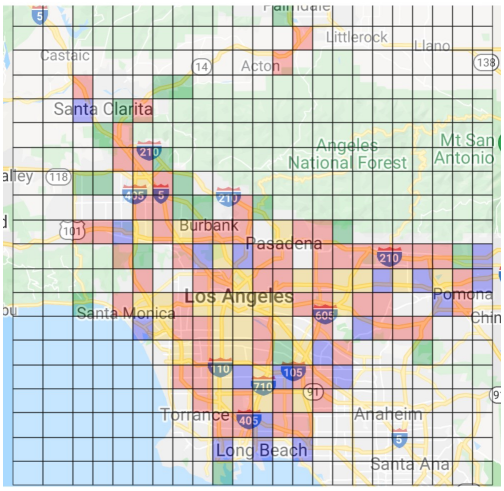
6.5. Influencing Factors Analysis

In this section, we conduct a few analysis to study the importance of different factors in our model design (i.e., components and input features). Such analysis can help to improve our input features and model design, all in order to build an effective framework for real-world applications.

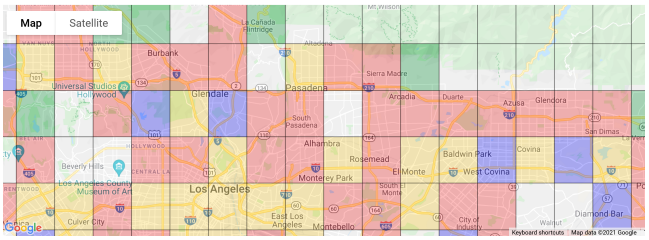
A) Latent Location Representations: As discussed in Sections 5.2 and 3.1, our dataset covers accident events in an extensive area containing urban road network and high-speed roads. During the model training, we derive latent representations for each region to encode essential spatiotemporal characteristics. The first analysis is to study quality of derived representations and understand how they can help to distinguish between regions (or zones) with different spatial characteristics. We believe that these latent representations can show how well the model has learned spatial factors involved in post-accident impacts. Therefore, we clustered latent (or embedding) vectors into four clusters and specified each cluster with a specific color on a map for better interpretation. Figure 17 shows the results of this analysis. Based on Figure 17a, most of the adjacent regions in our study area are found to be in the same cluster, indicating that they share the same accident-related characteristics, which is aligned with principals of urban design. As we move away from the center to the outskirts of the county area, we see more heterogeneity in clusters. This can be either a result of less accident reports (i.e., less data for model to train and properly distinguish between regions) or more heterogeneity in road network in suburbs. Figure 17b shows more details of different clusters. We can see that urban highways are mainly highlighted as yellow, while areas with sparse road network are highlighted as green. Blue and red regions are mostly urban regions with high density of spatial point of interests (e.g., intersections), and difference between them may be in intensity of traffic congestion caused by accidents.

In summary, the research findings in this section indicate the ability of our framework to build meaningful latent representations for different regions, that help to better predict accidents and their impact. Moreover, different clusters seem to reasonably represent regions with similar characteristics. The yellow cluster mostly represents regions with a concentrations of arterial highways (for which we can expect higher delays as result of an accident). The green cluster, in contrary, represents regions with marginal and low traffic flow, thus an accident on these regions will not cause as much delay as we should expect for regions in yellow cluster. For the regions in the red cluster, we can expect less delay (or impact) due to lower traffic flow, but higher delay in comparison to the regions in blue cluster. Also note that regions with sparse road network and low traffic flow (green regions) are mostly located in suburbs and far from downtown areas.

B) Ablation study: The second analysis is an ablation study to explore importance of each feature category when used individually for predicting accident impact. Here we study the following categories of features: weather, spatial, and accident information; and compare their results with the case of using all categories of features. Figure 18 shows the analysis results, and it reveals that removing all but one feature category would significantly degrade the prediction results. In comparison to the best model results, using each category alone as an input would result in greater Recall for $gammaclass$=2, lower Recall for $gammaclass$=1, and almost the same precision for $gammaclass$=0 (except for weather category). It is worth noting that using just weather features does not assist in identifying cases where $gammaclass$=0. This can either indicate that the quality of weather data in our dataset is not as good as it should be (e.g., it could be finer-grained spatially and temporally), or weather characteristics do not play an important role in the area of our study (throughout the year, we do not usually see significant shifts in weather condition in Los



(a) An overview of different clusters of latent representations of regions



(b) Closer look on Clustering of different regions and their spatial characteristics

Figure 17.: Representation of embedding layer output as different-colored regions on map. Regions represented by transparent color are removed due to lack of enough accident data.

Angeles area[12]).

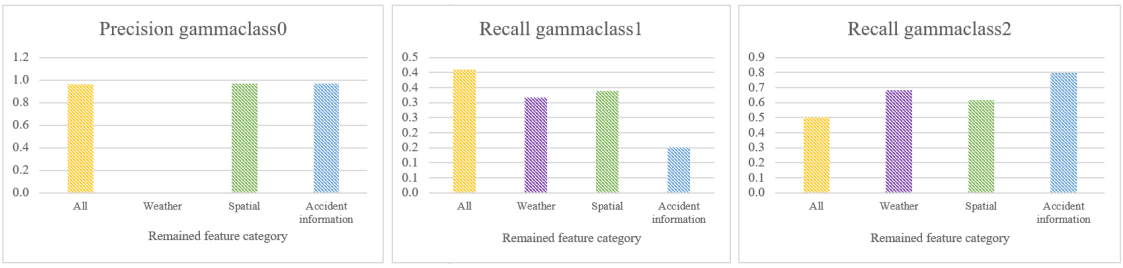


Figure 18.: Model performance based on evaluation metrics (Recall of *gammaclass*=1, 2 and precision of *gammaclass*=0) when using just one feature category (i.e., weather features, spatial features, and accident-related features) in comparison to the case of using all feature categories as input

7. Conclusion and Future Work

Traffic accidents are serious public safety concerns, and many studies have focused on analyzing and predicting these infrequent events. However, existing studies suffer from employing extensive data that may not be easily available to other researchers, or in real-time to be utilized for real-world applications. Additionally, they fail to establish an appropriate criterion for determining the impact of accidents. To overcome these limitations, this study proposes a cascade model that combines LSTM and CNN components for real-time traffic accident prediction. The model detects future accidents and assesses their impact on surrounding traffic flow using a novel metric called gamma. Four complementary datasets, including accident data, congestion data, weather data, and spatial data, are utilized to construct the input

[12]See `https://weatherspark.com/y/1705/Average-Weather-in-Los-Angeles-California-United-States-Year-Round` for more details.

data. Through extensive experiments conducted using data from Los Angeles county, we demonstrate that our proposed model outperforms existing approaches in terms of precision for cases with minimal impact (i.e., no reported accidents, $gammaclass$=0) and recall for cases with significant impact (i.e., reported accidents, $gammaclass$=1 and 2).

This study has several implications for future research. Firstly, we suggest expanding the framework to incorporate data from neighboring regions to predict accident impact for a target region, mitigating data sparsity. Additionally, the inclusion of satellite imagery data to enhance the model's ability to distinguish between regions and capture spatial information could also improve performance.

While our work makes important contributions, it is important to note its limitations. Our model currently only classifies accident impacts into three broad categories. Future research could benefit from more granular classifications or assigning probabilities to each accident's potential impact on road conditions. Additionally, the accuracy of our results may have been limited by the accessibility and quality of the weather data we used from a private source, which was gathered from only four airports. To enhance the accuracy of our model, future research could incorporate radar data or other more comprehensive weather data sources. Addressing these limitations will be essential to advancing our understanding of the factors that contribute to road accidents and improving road safety.

Overall, this study provides a promising approach to real-time traffic accident prediction using a novel metric for measuring accident impact. The proposed framework has the potential to be applied in real-world settings to enhance public safety and traffic management.

8. 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.

9. Data Availability

All data generated or analysed during this study are included in this published article, "Short and Long-term Pattern Discovery Over Large-Scale Geo-Spatiotemporal Data" [11]. The Weather Condition Dataset and POI Dataset are not available publicly, but you can refer to the paper mentioned above to get insightful information about how to collect such a dataset.

References

[1] Chao Wang et al. "Soft computing in big data intelligent transportation systems". In: *Applied Soft Computing* 38 (2016), pp. 1099–1108.

[2] Jie ZHANG, WANG Junhua, and FANG Shou'en. "Prediction of urban expressway total traffic accident duration based on multiple linear regression and artificial neural network". In: *2019 5th International Conference on Transportation Information and Safety (ICTIS)*. IEEE. 2019, pp. 503–510.

[3] Ross D Austin and Jodi L Carson. "An alternative accident prediction model for highway-rail interfaces". In: *Accident Analysis & Prevention* 34.1 (2002), pp. 31–42.

[4] Oj Oyedepo and Oladapo Makinde. "Accident Prediction Models for Akure – Ondo Carriageway, Ondo State Southwest Nigeria; Using Multiple Linear Regressions". In: *African Research Review* 4 (2010).

[5] Chien-Hung Wei and Ying Lee. "Sequential forecast of incident duration using artificial neural network models". In: *Accident Analysis & Prevention* 39.5 (2007), pp. 944–954.

[6] Pooja Salahadin Seid Yassin. "Road accident prediction and model interpretation using a hybrid K-means and random forest algorithm approach". In: *SN Applied Sciences* 1576 (2020).

[7] Sobhan Sarkar et al. "Application of optimized machine learning techniques for prediction of occupational accidents". In: *Computers & Operations Research* 106 (2019), pp. 210–224.

[8] Lu Wenqi, Luo Dongyu, and Yan Menghua. "A model of traffic accident prediction based on convolutional neural network". In: *2017 2nd IEEE International Conference on Intelligent Transportation Engineering (ICITE)*. IEEE. 2017, pp. 198–202.

[9] Murat Ozbayoglu, Gokhan Kucukayan, and Erdogan Dogdu. "A real-time autonomous highway accident detection model based on big data processing and computational intelligence". In: *2016 IEEE International Conference on Big Data (Big Data)*. IEEE. 2016, pp. 1807–1813.

[10] Pei Li, Mohamed Abdel-Aty, and Jinghui Yuan. "Real-time crash risk prediction on arterials based on LSTM-CNN". In: *Accident Analysis & Prevention* 135 (2020), p. 105371.

[11] Sobhan Moosavi et al. "Short and Long-Term Pattern Discovery Over Large-Scale Geo-Spatiotemporal Data". In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. KDD '19. Anchorage, AK, USA: Association for Computing Machinery, 2019, pp. 2905–2913. ISBN: 9781450362016. DOI: 10.1145/3292500.3330755. URL: https://doi.org/10.1145/3292500.3330755.

[12] Ciro Caliendo, Maurizio Guida, and Alessandra Parisi. "A crash-prediction model for multilane roads". In: *Accident Analysis & Prevention* 39.4 (2007), pp. 657–670.

[13] Alameen Najjar, Shun'ichi Kaneko, and Yoshikazu Miyanaga. "Combining satellite imagery and open data to map road safety". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 31. 1. 2017.

[14] Honglei Ren et al. "A deep learning approach to the citywide traffic accident risk prediction". In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE. 2018, pp. 3346–3351.

[15] Zhuoning Yuan, Xun Zhou, and Tianbao Yang. "Hetero-convlstm: A deep learning approach to traffic accident prediction on heterogeneous spatio-temporal data". In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2018, pp. 984–992.

[16] Quanjun Chen et al. "Learning deep representation from big and heterogeneous data for traffic accident inference". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 30. 1. 2016.

[17] Amir Bahador Parsa et al. "Toward safer highways, application of XGBoost and SHAP for real-time accident detection and feature analysis". In: *Accident Analysis & Prevention* 136 (2020), p. 105405.

[18] Yi Lin et al. "Automated traffic incident detection with a smaller dataset based on generative adversarial networks". In: *Accident Analysis & Prevention* 144 (2020), p. 105628.

[19] Kemal Polat and S Savaş Durduran. "Automatic determination of traffic accidents based on KMC-based attribute weighting". In: *Neural Computing and Applications* 21.6 (2012), pp. 1271–1279.

[20] Filippo Maria Bianchi et al. "Recurrent neural networks for short-term load forecasting: an overview and comparative analysis". In: (2017).

[21] Yongxue Tian and Li Pan. "Predicting short-term traffic flow by long short-term memory recurrent neural network". In: *2015 IEEE international conference on smart city/SocialCom/SustainCom (SmartCity)*. IEEE. 2015, pp. 153–158.

[22] Amir Bahador Parsa et al. "Applying Deep Learning to Detect Traffic Accidents in Real Time Using Spatiotemporal Sequential Data". In: *arXiv preprint arXiv:1912.06991* (2019).

[23] Xi Wang et al. "Graph Convolutional Network-based Model for Incident-related Congestion Prediction: A Case Study of Shanghai Expressways". In: *ACM Transactions on Management Information Systems (TMIS)* 12.3 (2021), pp. 1–22.

[24] Neda Kaffash Charandabi, Amir Gholami, and Ali Abdollahzadeh Bina. "Road accident risk prediction using generalized regression neural network optimized with self-organizing map". In: *Neural Computing and Applications* (2022), pp. 1–14.

[25] Rongjie Yu and Mohamed Abdel-Aty. "Utilizing support vector machine in real-time crash risk evaluation". In: *Accident Analysis & Prevention* 51 (2013), pp. 252–259.

[26] Lei Lin, Qian Wang, and Adel W Sadek. "A novel variable selection method based on frequent pattern tree for real-time traffic accident risk prediction". In: *Transportation Research Part C: Emerging Technologies* 55 (2015), pp. 444–459.

[27] Gyanendra Singh et al. "Deep neural network-based predictive modeling of road accidents". In: *Neural Computing and Applications* 32.16 (2020), pp. 12417–12426.

[28] Athanasios Theofilatos, Cong Chen, and Constantinos Antoniou. "Comparing machine learning and deep learning methods for real-time crash prediction". In: *Transportation research record* 2673.8 (2019), pp. 169–178.

[29] Sobhan Moosavi et al. "Accident risk prediction based on heterogeneous sparse data: New dataset and insights". In: *Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. 2019, pp. 33–42.

[30] Jie Bao, Pan Liu, and Satish V Ukkusuri. "A spatiotemporal deep learning approach for citywide short-term crash risk prediction with multi-source data". In: *Accident Analysis & Prevention* 122 (2019), pp. 239–254.

[31] Chao Chen et al. "Sdcae: Stack denoising convolutional autoencoder model for accident risk prediction via traffic big data". In: *2018 Sixth International Conference on Advanced Cloud and Big Data (CBD)*. IEEE. 2018, pp. 328–333.

[32] Asad J Khattak, Joseph L Schofer, and Mu-Han Wang. "A simple time sequential procedure for predicting freeway incident duration". In: *Journal of Intelligent Transportation Systems* 2.2 (1995), pp. 113–138.

[33] A Garib, AE Radwan, and HJJoTE Al-Deek. "Estimating magnitude and duration of incident delays". In: *Journal of Transportation Engineering* 123.6 (1997), pp. 459–466.

[34] Srinivas Peeta, Jorge L Ramos, and Shyam Gedela. "Providing real-time traffic advisory and route guidance to manage Borman incidents on-line using the Hoosier helper program". In: (2000).

[35] Bin Yu and Zhengfeng Xia. "A methodology for freeway incident duration prediction using computerized historical database". In: *CICTP 2012: Multimodal Transportation Systems—Convenient, Safe, Cost-Effective, Efficient*. 2012, pp. 3463–3474.

[36] WenQun Wang, Haibo Chen, and MARGARET C Bell. "Vehicle breakdown duration modeling". In: *Journal of Transportation and Statistics* 8.1 (2005), p. 75.

[37] Eleni I Vlahogianni and Matthew G Karlaftis. "Fuzzy-entropy neural network freeway incident duration modeling with single and competing uncertainties". In: *Computer-Aided Civil and Infrastructure Engineering* 28.6 (2013), pp. 420–433.

[38] B Yu et al. "A comparison of the performance of ANN and SVM for the prediction of traffic accident duration". In: *Neural Network World* 26.3 (2016), p. 271.

[39] Xiaolei Ma et al. “Prioritizing influential factors for freeway incident clearance time prediction using the gradient boosting decision trees method”. In: *IEEE Transactions on Intelligent Transportation Systems* 18.9 (2017), pp. 2303–2310.

[40] Rose Yu et al. “Deep learning: A generic approach for extreme condition traffic forecasting”. In: *Proceedings of the 2017 SIAM international Conference on Data Mining*. SIAM. 2017, pp. 777–785.

[41] Yunduan Lin and Ruimin Li. “Real-time traffic accidents post-impact prediction: Based on crowdsourcing data”. In: *Accident Analysis & Prevention* 145 (2020), p. 105696.

[42] Sobhan Moosavi et al. “A countrywide traffic accident dataset”. In: *arXiv preprint arXiv:1906.05409* (2019).

[43] Fen Xing et al. “Hourly associations between weather factors and traffic crashes: non-linear and lag effects”. In: *Analytic methods in accident research* 24 (2019), p. 100109.

[44] Fanny Malin, Ilkka Norros, and Satu Innamaa. “Accident risk of road and weather conditions on different road types”. In: *Accident Analysis & Prevention* 122 (2019), pp. 181–188.

[45] Grigorios Fountas et al. “The joint effect of weather and lighting conditions on injury severities of single-vehicle accidents”. In: *Analytic methods in accident research* 27 (2020), p. 100124.

[46] David P Doane. “Aesthetic frequency classifications”. In: *The American Statistician* 30.4 (1976), pp. 181–183.

[47] Alexander Skabardonis, Pravin Varaiya, and Karl F Petty. “Measuring recurrent and nonrecurrent traffic congestion”. In: *Transportation Research Record* 1856.1 (2003), pp. 118–124.

[48] Yisheng Lv, Shuming Tang, and Hongxia Zhao. “Real-time highway traffic accident prediction based on the k-nearest neighbor method”. In: *2009 international conference on measuring technology and mechatronics automation*. Vol. 3. IEEE. 2009, pp. 547–550.

[49] Chao Wang, Mohammed Quddus, and Stephen Ison. “A spatio-temporal analysis of the impact of congestion on traffic safety on major roads in the UK”. In: *Transportmetrica A: Transport Science* 9.2 (2013), pp. 124–148.

[50] Santiago Sánchez González, Felipe Bedoya-Maya, and Agustina Calatayud. “Understanding the Effect of Traffic Congestion on Accidents Using Big Data”. In: *Sustainability* 13.13 (2021), p. 7500.

[51] Zhenjie Zheng et al. “Determinants of the congestion caused by a traffic accident in urban road networks”. In: *Accident Analysis & Prevention* 136 (2020), p. 105327.

[52] Diederik P Kingma and Jimmy Ba. “Adam: A method for stochastic optimization”. In: *arXiv preprint arXiv:1412.6980* (2014).

[53] Sepp Hochreiter and Jürgen Schmidhuber. “Long short-term memory”. In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[54] Jiangfeng Wang et al. “Prediction Model for Traffic Congestion Based on the Deep Learning of Convolutional Neural Network”. In: *CICTP 2017: Transportation Reform and Change—Equity, Inclusiveness, Sharing, and Innovation*. American Society of Civil Engineers Reston, VA, 2018, pp. 2494–2505.