A Novel Short-term Post-accident Traffic Prediction Model

Farimasadat Miri
Computer Science
UOIT
farimasadat.miri@uoit.ca

Alireza A. Namanloo
Computer Science
UOIT
ali@ali-naman.com

Richard Pazzi
Computer Science
UOIT
richard.pazzi@uoit.ca

Miguel Vargas Martin
Computer Science
UOIT
miguel.vargasmartin@uoit.ca

***Abstract*—Traffic forecasting at appropriate times is vital for a variety of urban traffic control applications. There are a plethora of factors that can severely affect the performance of such forecasting models, especially those unpredictable events that cannot be learned through previous time steps in the model. One of the important factors that can change the traffic flow pattern abruptly, is accident occurrence. This issue affects more when the severity of the accident is high which would lead to a complete anomaly behavior in the traffic flow pattern. In this paper, we detected the approximate time that the accidents happen through anomaly detection around the accident record and then pulled out the ones that change the traffic pattern suddenly. To predict the traffic flow after a severe accident, we propose a hybrid CNN-LSTM model that takes spatio-temporal matrices as discrete time sequences for each accident. Furthermore, we leverage distances between neighboring nodes and accident location, coupled with static features that are pertained to the accidents. Moreover, we evaluate our model on PeMs dataset which is enhanced with the static features of the accidents from Moosavi's accident data set [1] [2]. Despite recent trials that are not able to forecast the results when an anomaly event happens, our results demonstrate that our proposed model can learn traffic patterns after the accidents and predict the traffic flow more accurately compared to current models.**

***Index Terms*—Traffic prediction, accident flow prediction, anomaly detection, CNN, LSTM**

I. Introduction

Urban traffic prediction has always been one of the hottest topics in the intelligent transportation system over the past decades. There are many benefits regarding urban traffic prediction that help many applications to work in the best version of themselves. For instance, having high accuracy in traffic prediction and discovering traffic congestion can help different applications such as Waze and Google maps direct drivers to less congested areas to get to their destination.

Many parameters are effective in traffic prediction such as rush hours, weather status, traffic status on working days. However, some are highly unpredictable such as accidents. It is important to note that, different accidents can have different effects on traffic congestion. Identifying the level of congestion can have a number of benefits for other applications such as resource allocation in a specific region when there is a hybrid architecture of fog and cloud in the smart city. Current traffic status in one observation point is highly dependent on the flows-in and flows-out of that point. In addition, it is highly dependent on previous patterns in previous days or weekdays at the same time. There are different approaches for traffic prediction: parametric, non-parametric, and deep learning methods [3].

Parametric models such as ARIMA (Auto Regressive Integrated Moving Average) [4] [5] [6] take fixed parameters and only used in linear systems. Despite having the benefits of using ARIMA, it cannot be highly reliable when anomaly patterns come into play, as it only looks into previous data to make predictions. Moreover, it is time-consuming to detect outliers. Non-parametric models can analyze noisy data and are better than parametric models. However, they only can rely on continuous data. Kalman Filter [7], Bayesian interference, and hidden Markov model are the ones that have tried to predict traffic in the past few decades.

Deep learning models have gained popularity in recent years mainly due to the capability of working high dimensional, non-linear and complex data. [8] [9] [10] [11]. One of the promising solutions that have achieved high accuracy compared to other deep learning methods for time series forecasting is LSTM (Long Short-Term Memory). Moreover, there are some hybrid methods such as CNN-LSTM to predict traffic flow [9]. There are other deep learning models such as attention-based that deal with long sequences and tries to keep the most important parts of a sequence [12] [13] [14].

In this paper, we focus on predicting the traffic flow after an accident happened. We propose a hybrid model of CNN-LSTM that not only takes the spatio-temporal correlation of traffic flow of neighboring locations in 1 hour before an accident time but considers different features at the accident location for having better accuracy. In other words, we propose a model that not only helps current traffic prediction methods but also helps us to predict real-time traffic flow before and after accident locations five and 10 minutes right away after an accident occurrence, through feeding the features of that accident to a neural network such as working days, working hours, traffic lights, weather, etc. The main contributions of this paper are listed below.

- Defining a new approach to detect anomaly subsequences in discrete sequences which leverages interpolation, regression, integral, and derivative.
- Providing a dataset that contains traffic flows before and after an accident in a spatio-temporal matrix, in which the record time of the accident has been shifted to the point that anomaly behavior has been observed. Then, adding static features to the dataset through merging two

different data sets from two different sources, PeMS and Moosavi accident datasets [1] [2].
- Traffic flow prediction through using a hybrid CNN and LSTM model named AT-LSTM that takes three inputs:the static features of each accident, distances between adjacent points and accident location, and traffic flow speed in closest locations and times before and after an accident.

The remainder of this paper is organized as follows:

- Related works in the scope of traffic prediction and accident duration prediction are discussed in Section II.
- Background about two well-known neural network models, LSTM and CNN, are explained briefly in Section III.
- Our model definition and required parameters before presenting the models are presented in Section IV.
- In Section V, we generate a new dataset based on two different sources of traffic and accident, where we define an anomaly detection algorithm to synchronize accident time with traffic speed.
- Evaluation and experimental results different models are explained in Section VI.
- Conclusion and future works are explained in Section VII.

II. Related work

There are many papers about traffic prediction through time series models. To name a few, in [15] the authors used spatial correlation matrix to be used in LSTM for short-term traffic prediction. This matrix specifies the correlation between different observation points in traffic data. Each element in the matrix describes how much ith observation point affects jth observation point.

In [16], they defined network points that can include streets and highways. Each point can represent the traffic flow of a junction. In addition, each point such as point M in the traffic dataset has inflow sequences and outflow sequences. In-flow sequences are the neighboring points that have traffic flow to that point. And out-flow sequences, are the neighboring points that point M has traffic flow to them. After, they provided a matrix that includes spatial-temporal correlation between different observation points, in-flow and out-flow sequences. In [3], they have used temporal and hierarchical clustering for traffic speed prediction. First, they used clustering method to cluster similar traffic patterns, and based on the traffic patterns that have been gathered before, they found the nearest pattern to the current pattern. Then, they partitioned data into spatial and temporal speed vectors. In addition, two attention mechanisms have been proposed in this paper. In [17], they provided a novel architecture named STGCN, in which they formulate the traffic prediction problems on the graphs with convolutional structures to extract spatio-temporal features from graph structure. In their structure, they demonstrate two convolutional blocks that are spatio-temporal and each block includes three gates, two of them are temporal and another one is spatial.

All of the inputs are being processed through these blocks to detect spatio-temporal dependencies between inputs and finally, all of them are being integrated with the final layer to make a prediction. In [18], they predict the traffic flow with multiple inputs that are sequential. Weather features have been embedded in each time step to have an accurate prediction. So, each vector in the times series displays traffic times and weather features.

They put GRU in hidden layers instead of neurons and try to have both short and long-term predictions. Finally, they have a fully connected layer that is connected to the output layer. As it is clear, an anomaly in traffic pattern would not be just trained through historical data on a specific point in a normal day and unfortunately, most of the works do not pay enough attention to traffic behavior when unpredictable events happen, which might be lead to anomalous behavior in the traffic pattern and have a negative impact on their model.

Some other research works mainly focus on prediction for the incident duration. Overall, accident duration can be divided into 4 important phases [19]. The first one is accident detection or the time that the accident is reported. The second phase is the response time for the emergency team's arrival at the accident location. The third phase is the time that taken by the emergency team to clear the accident location. Finally, the last phase is the time duration that traffic gets back to normal situation.

In paper [20], they evaluate two different models for traffic accident duration. They used K-means clustering. In addition, they showed that traffic jams and peak hours are two features with a strong correlation. Finally, they proposed a simple ANN method for training the model. They have good accuracy for incidents longer than 60 minutes duration and have poor performance for accidents with less than 30 minutes duration.

There are numerous papers in the scope of traffic flow prediction and accident duration separately, that we discussed a few of them. However, to the best of our knowledge, there is a lack of research in the scope of short term traffic flow prediction after an accident. In this work, we predict traffic flow after an accident to fill the gap with existing works in predicting traffic after an unexpected event such as accident.

III. Background

Time series algorithms are useful in many research fields such as stock market prediction, weather prediction, and urban traffic prediction. Since we use three types of data for this work, we need to use appropriate models that satisfy spatial, temporal, and static data. CNN is widely used when we have any type of data that relates to geographical area or space, such as image classification problems. On the other hand, in many existing works, LSTM models have shown promising results for sequential time-series data types. To leverage the power of CNN on spatial data and LSTM on sequential data, we use the CNN-LSTM model that takes station distances (spatial) and traffic speed (spatio-temporal sequences) as the inputs for CNN and LSTM respectively. Then we flatten the output of CNN and LSTM, coupled with static features as the inputs of the final layer in our neural network. For a better understanding about CNN and LSTM separately, they have been briefly explained below.

A. LSTM

LSTM can resolve the vanishing gradient descent problem that RNN has. LSTM includes a memory cell and unlike RNN, it can decide to keep or remove the information to the next step through having gates. There are multiple equations for LSTM, in which there is an input gate, the forget gate, and the output gate. The reason that they have been called gates is because of the Sigmoid activation function. Through elementwise multiplication, the amounts of vectors that are needed for prediction is being used.

B. CNN

Another category of neural networks is defined as Convolutional Neural Network (CNN) which is one of the powerful algorithms for image recognition, classification, and segmentation. Recently, it has shown considerable advancements in time series classification and problems as well. It mainly consists of different layers. The first layer is a convolutional layer that applies kernel (filter) to the input. Its main task is, extracting important features for specific purposes such as edge detection in an image. Then, the result of applying a filter on the input leads to a feature map. The second layer is named the pooling layer that decreases the dimension of the feature map and keeps the most important features. The third layer is a fully connected layer that flattens the outputs from pooling layers and creates a single vector. It's worth mentioning that the main goal of the fully connected layer is using these features for classification through applying weights over the inputs that have been generated by previous layers. The last layer is a fully connected output layer that gives the final output or class of an image. Recent papers have demonstrated that using CNNs models can help us to classify time series classification as well, as they are resistant to noise and highly independent from time.

IV. MODEL

A traffic network consists of several roads with different characteristics. Each road can consist of traffic signals such as roundabouts, no exit, junction, turning loop, etc. An accident can happen on the freeway roads and each freeway can have different lane types. In PeMS, each freeway has possible values of On-Ramp, Off-Ramp, Mainline, Freeway-Freeway Connector, and so on. They have been equipped with a number of sensors (detectors) that monitor traffic on the freeways. Having an accident in different freeways with different characteristics would have different impacts on the traffic flow. There are a variety of important factors that affect traffic speed after an accident happens that will be explained in the static features section.

1) Technical Definition: In this study, we define an accident that happened to the nearest detectors based on the time and location which have been gathered from two different datasets: Moosavi accident dataset [1] [2] and PeMS dataset. There are different lane types on the roads, and mainline type has the highest number of accidents compared to other lane types on the road. Moreover, it does not have many missing data compared to different lane types. So, we just evaluated all of the accidents that happened in the mainline for four consecutive years from 2016-2019 in a specific district of California. Moreover, there are a number of points before and after the accident that are adjacent to each other and there is a spatial correlation between them. Traffic speed on the accident location has a spatial correlation to the adjacent points. It can be concluded that traffic speed on the accident location, can be influenced by the traffic speed from the adjacent points. The points can be denoted as follows, in which negative numbers denote the detectors implemented before an accident and positive ones denote those implemented after an accident location. To put it more simply, The Detector(+1) and Detector(-1) are the closest detectors to the accidents in terms of distance. Also, Detector(-4) and Detector(+4) are the farthest ones to the accident location. It is important to note that accident location is between Detector(+1) and Detector(-1).

Detector(-4) $\leftrightarrow$ Detector(-3) $\leftrightarrow$ Detector(-2) $\leftrightarrow$ Detector(-1) $\leftrightarrow$ Detector(+1) $\leftrightarrow$ Detector(+2) $\leftrightarrow$ Detector(+3) $\leftrightarrow$ Detector(+4)

2) Spatio-temporal matrix definition : In order to predict the traffic speed in the closest accident locations, we need to know about the traffic speed of all the adjacent detectors before and after the accident location. Additionally, for each location, we also need traffic speed in different time intervals. Traffic speed around each station has been captured every 5 minutes. For our work, we monitored the traffic flow for 60 minutes (12 time slots) before an accident in the closest stations. We demonstrate the spatio-temporal correlations of traffic speeds in matrix S_D^T, in which each row represents traffic speed in one specific location in different time steps, while each column represents traffic speed in different locations at a specific time. $S_{d_i}^{t_j}$ is the traffic speed in detector $d_i \in D$ in time $t_j \in T$, in which T is the set of all time steps and D is the set of detectors. Also, in the matrix, σ is 12 to represent the number of time intervals. Finally each point has been normalized between $0 < S_{d_i}^{t_j} < 1$ through Z-score normalization.

TABLE II
WEATHER TYPES CATEGORIZATION

Weather Type	Severe	Moderate	Normal
Categories	Heavy Rain	Light snow	Fair
	Heavy Snow		Partly Cloudy
		Light rain	Scattered Clouds
	Stormy		Overcast

$$D = \{d_i | i \in [-4, 4], i \in Z\}$$

TABLE I
FEATURE TYPES FOR THE ACCIDENTS

Feature type	Categories					**# Categories**
Weather type	Severe		Moderate	Normal		3
Day type	Working days			Not working days		2
hours type	Midnight	Morning rush	Between rush	Evening rush	Night rush	5
Number of Lanes	2	3	4	5	6	5
Close to Junction	Yes			No		2
Close to Traffic Signal	Yes			No		2
Close to No-Exit	Yes			No		2

$$S_D^T = \left\{ \begin{matrix} S_{d_{(-4)}}^{t-\sigma} & S_{d_{(-4)}}^{t-\sigma+1} & \ldots & S_{d_{(-4)}}^{t-1} & S_{d_{(-4)}}^{t} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ S_{d_{(-1)}}^{t-\sigma} & S_{d_{(-1)}}^{t-\sigma+1} & \ldots & S_{d_{(-1)}}^{t-1} & S_{d_{(-1)}}^{t} \\ S_{d_{(+1)}}^{t-\sigma} & S_{d_{(+1)}}^{t-\sigma+1} & \ldots & S_{d_{(+1)}}^{t-1} & S_{d_{(+1)}}^{t} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ S_{d_{(+4)}}^{t-\sigma} & S_{d_{(+4)}}^{t-\sigma+1} & \ldots & S_{d_{(+4)}}^{t-1} & S_{d_{(+4)}}^{t} \end{matrix} \right\}$$

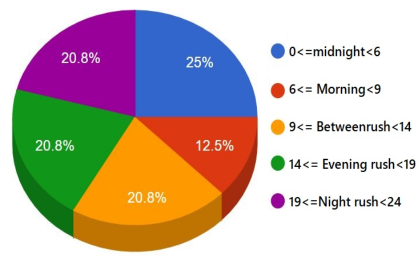

Fig. 1. Hours division

3) Distance between stations: The distance vector M contains distances between closest detectors D and an accident location. m_{d_i} denotes the distance between detector d_i and accident location.

$$M = \left\{ \begin{matrix} m_{d_{(-4)}} & m_{d_{(-3)}} & \ldots & m_{d_{(+1)}} & \ldots & m_{d_{(+4)}} \end{matrix} \right\}$$

4) Static features: Apart from the distance and traffic speed of the adjacent locations, other factors can influence the traffic speed such as weather conditions, and the daytime(rush hours), we refer to these factors as static features. The full list of static features that can categorize an accident into different severity types is shown in Table I.

a) Weather conditions: Traffic speed and weather conditions are usually correlated. Average speed might be affected and goes lower when the weather is foggy, snowy, slippery, and wet. In our work, we categorized weather conditions into three different groups as denoted in table II.

b) rush hours and non-rush hours: It plays a really important role in traffic speed prediction. Based on the traffic pattern and rush hours in Los Angeles which has been recorded in tomtom.com [21], we categorized it into five different groups based on the traffic level percentage they tolerate which is shown in figure 1.

A. Problem Definition

We define the problem of predicting traffic speed after an accident by leveraging two different data sets, one consisting of traffic speeds and the other dataset containing accidents static features. For each accident, we use the traffic speed of 8 closest locations in 60 minutes (12 time intervals of 5 minutes).

In most traffic models traffic speed of a location is predicted based on the historical data gathered from the same location, this means the model can only predict traffic speed of observed locations. In contrast, our model learn traffic speed patterns after an accidents regardless of the a specific location, in a way that the traffic speed in the locations that do not exist in our dataset can also be predicted.

1) Problem formulation: Given X that includes spatio-temporal matrix S_D^T, consisting of traffic speeds, distance vectors and one hot encoded static features for each accidents, predict traffic speeds Y in t_1 and t_2 denoting the first and second time steps after an accident respectively in the closest station that is located before the accident location. As it is shown in the formula $S_{d_f}^T$ are the average speed of the vehicles before and after accident location in different times, $M_{d_{(n)}a}$ is the distance vector between detectors and accident location and finally $Statics$ is the static features we defined earlier.

$$X = \begin{pmatrix} S_D^T \\ M \\ Statics \end{pmatrix} \quad Y = \begin{pmatrix} S_{d_{(-1)}}^{t+1} & S_{d_{(-1)}}^{t+2} \end{pmatrix}$$

2) Using CNN and LSTM for traffic flow prediction: In this section, we try to predict traffic flow based on two well-known neural networks: LSTM [22] and CNN. Figure 2 demonstrates our model architecture. There are three different broad inputs with different nature and different shapes which have been fed into our network. Our inputs are traffic speed for each detector before and after accident location, distance vectors

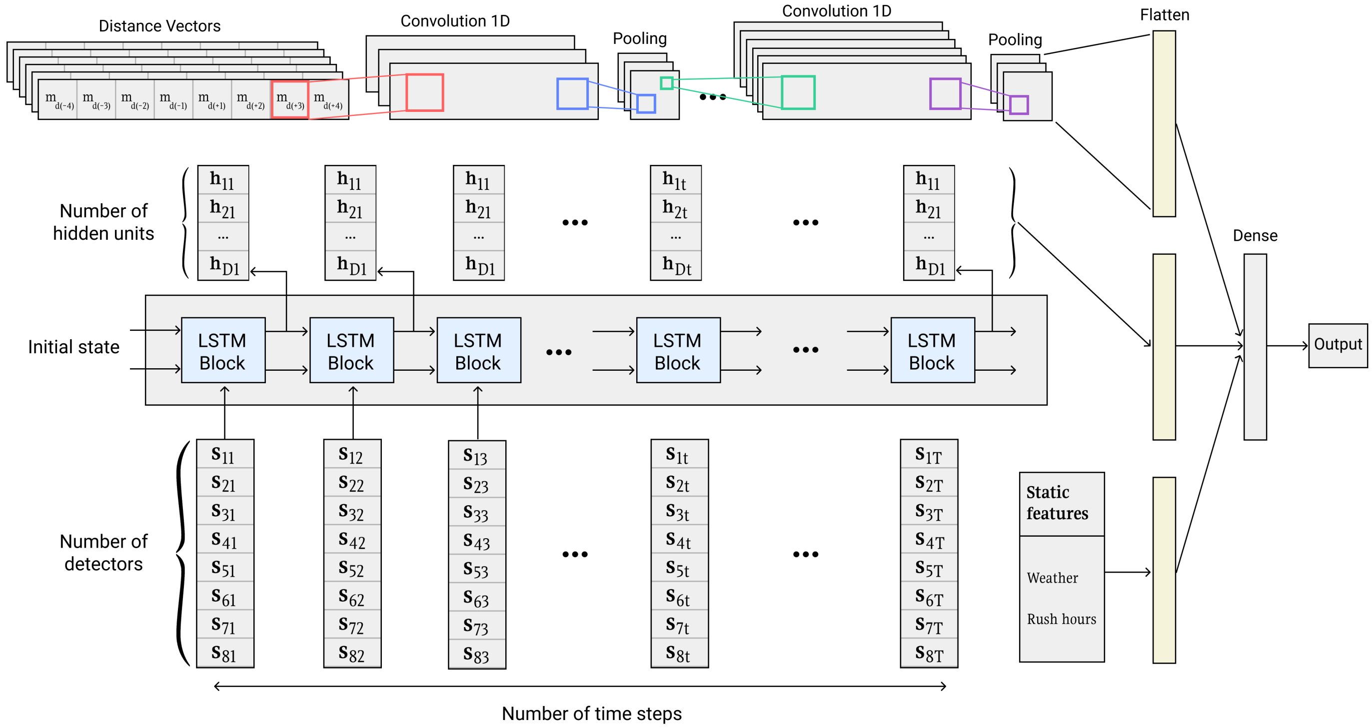

Fig. 2. The schematic architecture of our model: At the bottom, we can see the traffic speeds (spatio-temporal data) are fed into LSTM blocks, following by a number of hidden units to get the output. At the top, for each accident, the distances between detectors and accident location are given to a CNN model. Finally, the flatten output of CNN and LSTM, and static features are passed to a dense layer to get the final predictions.

and finally static parameters which are not temporal and mainly related to the accident point. Although, some static features can be defined as temporal features for each data point in our model, such as: weather conditions, working hours and non-working hours, however, in order to decrease the complexity of the model, we connected all static features to the accident point. The distance vector is given to the 1D-CNN to extract important stations that have a key role for traffic flow prediction. Moreover, spatio-temporal speed matrix is given to the LSTM to remember long-term dependencies between stations in different time steps. Then, static features and distance features are added to the end of our LSTM layer through flattening and finally, given to the dense layers before getting the output.

V. Data preparation

There are two separate data sets from two different sources that have been merged together in order to get the traffic flow and speed before and after the accident location in different time steps. The first one includes 4 million accidents all over the US. Each accident has a number of static features such as weather conditions, rush hours, day type, number of lanes, close to junction, traffic signal and no-exit. These accidents happened all over the US not just in freeways. The second data set is PeMS that includes traffic data that has been collected in real time in the state of California from approximately 39000 detectors which have been installed in freeways. There are 9 different districts division in California, and we mainly focus on accidents in district 7 (Los Angeles). Moreover, PeMS dataset utilizes incidents that have been gained from another source named TASAS (Traffic Accident Surveillance and Analysis System). Incidents are categorized into accidents, advisory, breakdown, closure, congestion, hazard, police, weather and others. However, there is no categorization by incident type. In order to get the traffic flow for each accident coupled with related static features, merging those data sets and extracting the accidents from incidents is needed. There are a number of challenges that we encountered: Firstly, as the accident times have been recorded by humans, there are delays in recording the time of the accident in both data sets. In order to match the same accident in both data sets, we get the 7 minutes time error and 50 meters distance error to find the same accident that has been recorded in both data sets. The reason for choosing 7-minutes and 50-meters thresholds is that, the probability of occurring more than one accident in a 7-minutes time frame and in 50 meters is very low. Clearly, lower thresholds resulted in more confidence in finding the same accidents, however we would lose more data for the training. Then, after getting the freeway and direction of each accident from incident dataset (PeMS), we compare the latitude and longitude of each accident with all of the latitude and longitude of detectors in the same freeway and the same direction in order to get the closest detectors that the accident happened. Then, to get the 4 detectors before and 4 detectors after an accident occurrence location, we sorted all the detectors with Absolute post-mile (ABS PM).

By observing traffic patterns for each accident, it can be clearly seen that traffic patterns are not the same for each accident. There are some cases that traffic patterns have not changed drastically and there are other cases that traffic patterns have completely changed. One of the pressing problems for urban traffic prediction is forecasting unpredictable events such as accidents. Our main focus is forecasting traffic flow after a severe accident that completely changes the traffic behavior, as current traffic prediction models have a reasonable accuracy for normal traffic and experience deficiency in forecasting severe and unpredictable events.

Each accident in different situations can have different impacts on traffic speed, which affect both the average of traffic speed that is decreased, and the amount of time it takes that traffic get back to the normal state.

Another challenging issue is about the delays for accidents record times. This delay is different for each record which varies from 5 minutes to even 1 hour. So, in this phase we shifted our data back and consider the accident event in a time that anomaly behavior has been observed. One of the main reasons that some accidents have not changed the traffic manner, is because of the accident location, such as right-hand shoulder of the freeway. Or it can be related to the severity of the accident and the time in which it happened. For example, if it happened in the weekend and non-rush hours, there is no specific peaks and troughs around accident time in data plot. Our approach is denoted below in order to detect anomaly around the accident record and eventually correct the delays for accidents time to the right time of the occurrence.

a) Linear interpolation: Firstly, we define function f by using linear interpolation method to get the missing values at positions in between the data points, because our data points are discrete and we have traffic speed every 5 minutes.

b) Linear regression: On the other hand, we define function g using linear regression method to find an approximation of the data points in order to make the residual sum of squares between the points and linear interpolation as small as possible. In order to do that, linear regression uses the equation $y = a + bx$ in which a and b are achieved with the formula that have been shown below:

$$a = \frac{(\sum y)(\sum x^2) - (\sum x)(\sum x * y)}{n\sum x^2 - (\sum x)^2} \tag{1}$$

$$b = \frac{n(\sum x * y) - \sum x \sum y}{n\sum x^2 - (\sum x)^2} \tag{2}$$

c) Intersection detection : Then, we use Brent q method [23] to get the intersection points between both interpolation function f and linear function g. This would gives us a number of intervals that each one contains one bounded area between interpolation and regression functions of our data points.

d) Anomaly detection: After having bounded areas, we compare them and choose the one with the largest area size. Since accidents would lead to decrease the traffic speed, just the areas under the linear regression line are considered.

e) Detecting the correct time of accidents: Then, in order to get the correct time an accident happened, we compare the slope of a start intersection point of the largest area with the previous data points (previous time steps), until we get a slope that has a higher value of a threshold close to zero. Intuitively, we start from a time when the traffic speed is decreasing significantly, and we move back to find the nearest time when traffic speed is not decreasing suddenly. Algorithm 1 shows our proposed anomaly detection algorithm, and Figure 4 shows better intuition about our anomaly detection process.

Algorithm 1: Anomaly Detection to Correct Accident Delays

Input: a set of data points
$D = \{(t_i, v_i)| -12 \leq i < 12\}$
Output: anomaly start time $t_j \in T$

$f(x) \leftarrow$ linear interpolation on D
$g(x) \leftarrow$ linear regression on D
$h(x) = g(x) - f(x)$
$R \leftarrow$ set of root points of $h(v)$ using Brent's method
foreach $1 \leq i < size(R)$ **do**
  $A_i = \int_{(r_i)}^{r_(i+1)} (g(x) - g(x)) * d(x)$
$A_i^{largest} \leftarrow \max(abs(A_i)|A_i < 0)$
$l \leftarrow index(A_i^{largest})$
$t_l \leftarrow max(t_i|t_i < l)$
for $j = t_l$, $j{-}{-}$, *while* $j \geq -12$ **do**
  $slope = \frac{f(j)-f(j-1)}{j-(j-1)}$ **if** $slope > threshold$ **then**
    return t_j

VI. Experimental results and analysis

we utilized a system with following features:

- Intel(R) Xeon(R) Gold 6148 CPUs with 80 total cores @ 2.40GHz,
- 36 GB Ram
- 4 Nvidia GeForce 1080 Ti GPUs.

For our model evaluation, we used the PeMS dataset to get the traffic speed in district 7 in California from 2016 to 2019. Our evaluation metrics are MAE (Mean Absolute Error) and RMSE (Root Mean Square Error) for 5 and 10 minutes after an accident in the closest station before an accident location. Also, the input for LSTM has the shape of (number of accidents, 12, 8) that 12 is the number of 5-minute time steps which in total captures one hour before an accident, and 8 is the number of detectors that we considered them as features in LSTM. Also, we split training and testing to 0.85 and 0.15 respectively. The batch size is 30 and the number of epochs is 20. We run our experiments 10 times using Monte Carlo cross-validation method and averaged the final results. We saw the best performance when using 60 minutes before an accident compared 15, 30, and 45 minutes in terms of MAE and RMSE.

$$MAE = \frac{1}{n} \sum_{(i=1)}^{n} \mid y_{pred}^{i} - y_{actual}^{i} \mid$$

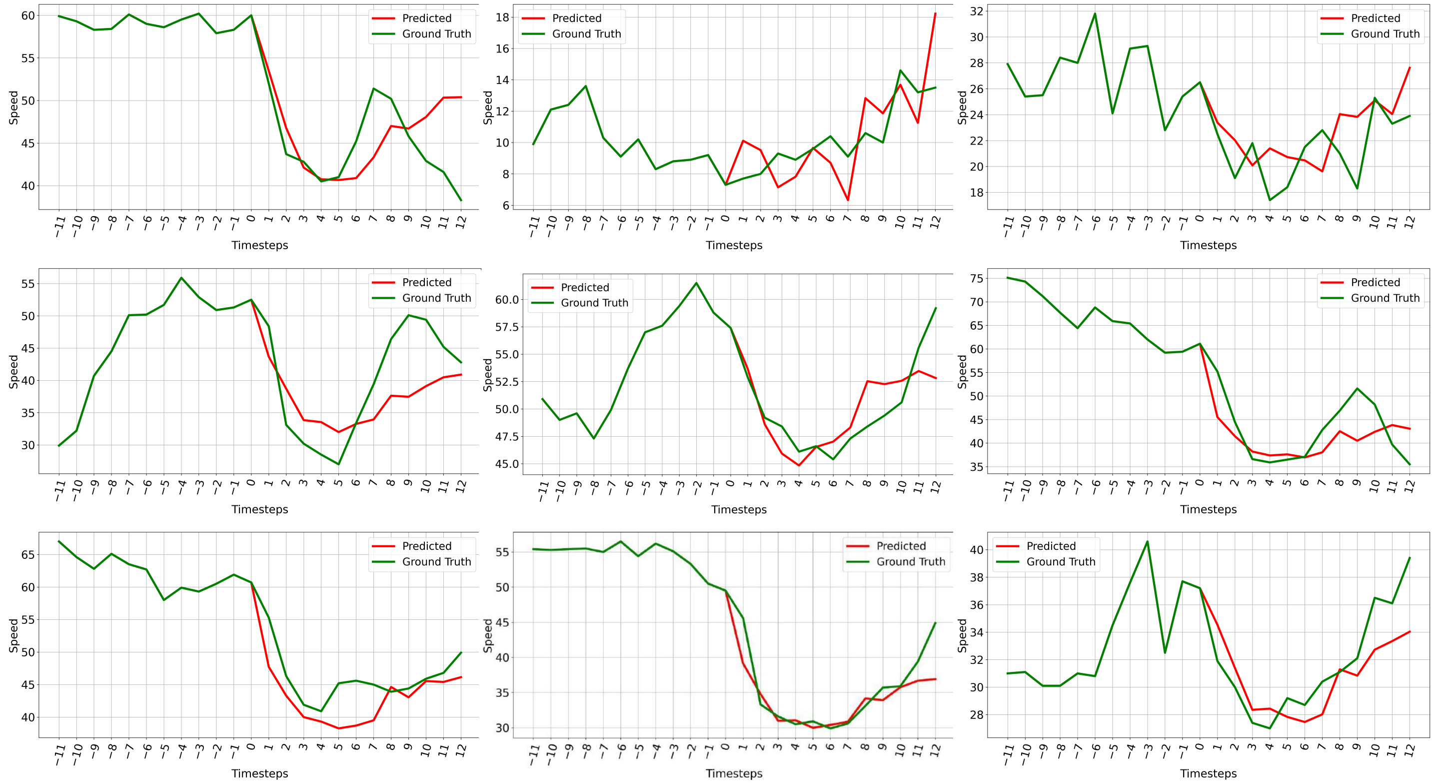

Fig. 3. A few examples to show our model performance in predicting traffic speed after an accident. The figures illustrate that not only our model can predict how much traffic speed will decrease after an accident, but also it can predict approximately when the traffic speed will start getting back to normal.

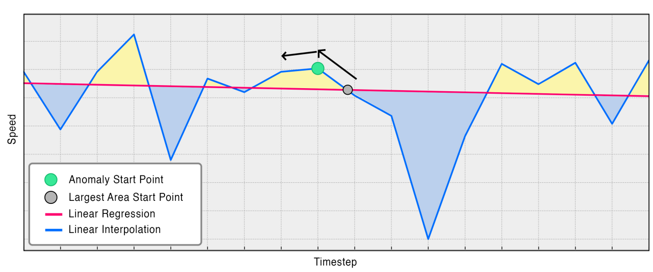

Fig. 4. Schematic view of anomaly detection

TABLE III
COMPARISON OF MODELS – PEMSD7(5/10 MINUTES)

Model	MAE	RMSE
LSTM	3.60 / 5.42	5.61 / 7.21
LSC	3.35 / 5.05	4.92 / 7.20
STGCN	5.37/9.17	7.35/11.98
AT-LSTM	**3.21 / 5.13**	**4.92 / 6.99**

$$RMSE = \sqrt{\frac{1}{n}\sum_{(i=1)}^{n}(\mid y_{pred}^{i} - y_{actual}^{i} \mid)^2}$$

It is important to note that our main goal is forecasting traffic speed after an accident. In other words, as we mainly focus on the anomaly behavior of traffic patterns, it is not reasonable to compare our model with current short-term and long-term traffic prediction methods, as they evaluate the traffic in an entire day that has been converted to stationary data. However, we extracted all the anomaly manners in the traffic data set, and through correcting accident delays in the anomaly starting point, we help the system to predict the unexpected events. We compare our model with one of the recent promising forecasting models named STGCN [17] to see their results in accident points. In table III, we demonstrate that our model performance is better in predicting the average traffic flow 5 and 10 minutes after an accident compared to other models. Furthermore, AT-LSTM(our main model) has a better performance compared to LSTM, mainly due to adding static features to the model for better understanding the situation. Also, LSC is defined by adding static features to LSTM model, which helps LSTM to increase the prediction. Figure 5 compares the forecasting results for 1000 accidents with the ground truth data sorted by traffic speed.

In figure 3, we showed a few charts that depict our forecasting. Although in this paper, our main focus is to predict the traffic speed only 5 and 10 minutes after an accident, in figure 3 we also included the predictions until 60 minutes to just show some examples how increasing the time intervals affect the prediction performance. However, our results in table III are for the first 5 and 10 minutes (the first two time steps after an accident).

Figure 5 shows the average speed forecasting till 10 minutes after an accident. There is a good convergence between real and predicted data, with a maximum speed difference of about 3 Km/h.

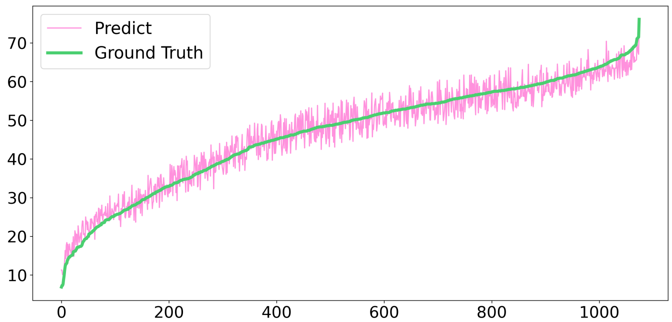

Fig. 5. speed prediction (accidents sorted by speed)

VII. Conclusion and future work

In this paper, we focused on short-term traffic flow forecasting after an accident. We leveraged different inputs namely spatio-temporal matrix of traffic speed, distance vector, and static features. Moreover, using a hybrid CNN and LSTM model, and based on our results, we showed that traffic patterns after an accident can be learned from each other. Besides, since the accident's record time was different in two different datasets, we defined an anomaly detection algorithm to approximate correct time of the accident occurrence. Finally, our results showed better short-term traffic flow forecasting compared to existing traffic models in unexpected events. This model might be combined with current traffic flow prediction models as it helps them have better forecasting of unexpected events. For future work, we integrate our model with long-term traffic prediction methods, so they can predict better in unpredictable situations.

References

[1] S. Moosavi, M. H. Samavatian, S. Parthasarathy, R. Teodorescu, and R. Ramnath, "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.

[2] S. Moosavi, M. H. Samavatian, S. Parthasarathy, and R. Ramnath, "A countrywide traffic accident dataset," *arXiv preprint arXiv:1906.05409*, 2019.

[3] D. Liu, L. Tang, G. Shen, and X. Han, "Traffic speed prediction: an attention-based method," *Sensors*, vol. 19, no. 18, p. 3836, 2019.

[4] B. M. Williams and L. A. Hoel, "Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results," *Journal of transportation engineering*, vol. 129, no. 6, pp. 664–672, 2003.

[5] Q. T. Tran, Z. Ma, H. Li, L. Hao, and Q. K. Trinh, "A multiplicative seasonal arima/garch model in evn traffic prediction," *International Journal of Communications, Network and System Sciences*, vol. 8, no. 4, p. 43, 2015.

[6] S. R. Chandra and H. Al-Deek, "Predictions of freeway traffic speeds and volumes using vector autoregressive models," *Journal of Intelligent Transportation Systems*, vol. 13, no. 2, pp. 53–72, 2009.

[7] M. G. Karlaftis and E. I. Vlahogianni, "Statistical methods versus neural networks in transportation research: Differences, similarities and some insights," *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 3, pp. 387–399, 2011.

[8] X. Ma, Z. Tao, Y. Wang, H. Yu, and Y. Wang, "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data," *Transportation Research Part C: Emerging Technologies*, vol. 54, pp. 187–197, 2015.

[9] S. Du, T. Li, X. Gong, and S.-J. Horng, "A hybrid method for traffic flow forecasting using multimodal deep learning," *arXiv preprint arXiv:1803.02099*, 2018.

[10] S. Zhang, Y. Yao, J. Hu, Y. Zhao, S. Li, and J. Hu, "Deep autoencoder neural networks for short-term traffic congestion prediction of transportation networks," *Sensors*, vol. 19, no. 10, p. 2229, 2019.

[11] X. Ran, Z. Shan, Y. Fang, and C. Lin, "A convolution component-based method with attention mechanism for travel-time prediction," *Sensors*, vol. 19, no. 9, p. 2063, 2019.

[12] V. Mnih, N. Heess, A. Graves *et al.*, "Recurrent models of visual attention," *Advances in neural information processing systems*, vol. 27, pp. 2204–2212, 2014.

[13] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," *arXiv preprint arXiv:1508.04025*, 2015.

[14] Y. Qin, D. Song, H. Chen, W. Cheng, G. Jiang, and G. Cottrell, "A dual-stage attention-based recurrent neural network for time series prediction," *arXiv preprint arXiv:1704.02971*, 2017.

[15] Z. Zhao, W. Chen, X. Wu, P. C. Chen, and J. Liu, "Lstm network: a deep learning approach for short-term traffic forecast," *IET Intelligent Transport Systems*, vol. 11, no. 2, pp. 68–75, 2017.

[16] M. Fouladgar, M. Parchami, R. Elmasri, and A. Ghaderi, "Scalable deep traffic flow neural networks for urban traffic congestion prediction," in *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2017, pp. 2251–2258.

[17] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," *arXiv preprint arXiv:1709.04875*, 2017.

[18]

[19] Y. Chung, "Development of an accident duration prediction model on the korean freeway systems," *Accident Analysis & Prevention*, vol. 42, no. 1, pp. 282–289, 2010.

[20] B. Yu, Y. Wang, J. Yao, and J. Wang, "A comparison of the performance of ann and svm for the prediction of traffic accident duration," *Neural Network World*, vol. 26, no. 3, p. 271, 2016.

[21] T. Website. Los angeles traffic report. [Online]. Available: https://www.tomtom.com/

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

[23] R. P. Brent, "Algorithms for minimization without derivatives, chap. 4," 1973.