A TWO-STAGE MULTI-TASK LEARNING MODEL FOR PROACTIVE NON-RECURRENT TRAFFIC PREDICTION

Hao Wu
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
haowu3@cs.cmu.edu

Weiran Yao
Department of Civil and Environmental Engineering
Carnegie Mellon University
Pittsburgh, PA 15213
weiran@cmu.edu

Sean Qian, Ph.D., Corresponding Author
Department of Civil and Environmental Engineering
Heinz College of Information Systems and Public Policy
Carnegie Mellon University
Pittsburgh, PA 15213
seanqian@cmu.edu

Word Count: 4867 words + 5 table(s) × 250 = 6117 words

Submission Date: August 2, 2022

ABSTRACT

Research on short-term traffic prediction is intensive, but very few work focuses on alerting incidents/anomalies in advance, before an incident is reported anywhere, while predicting network speed under incidents. This is notoriously challenging because incidents and non-recurrent traffic data are generally rare, thus not naturally a good fit for training a machine learning model. Most work train models using both the recurrent and non-recurrent data altogether. Because the non-recurrent traffic patterns are rare, the overall traffic prediction can be biased and driven by common recurrent traffic conditions that is less important in traffic management practice. This paper proposes a two-stage model design that separates recurrent cases and non-recurrent use cases to promote performance in traffic speed prediction particularly for the more crucial non-recurrent traffic prediction. Moreover, additional features from multi-source data, probe data confidence score and vehicle-specific speeds, are incorporated to a novel multi-task sequence-to-sequence machine learning model to bolster model prediction performance. An experiment shows that this two-stage model can effectively pick up almost all incidents timely, being ahead of when incidents were actually reported in Waze. The speed prediction accuracy is also superior than other baseline models, particularly for speed prediction under those incidents. Features on speed by vehicle classes and confidence scores can effectively improve the prediction accuracy.

Keywords: emerging mobility, mobility service, accessibility, micromobility, reliability, multi-task learning, sequence model

INTRODUCTION

Operating transportation highway networks in real time is challenging. In 2019, U.S. roadways witnessed 8.8 billion hours of travel delay and 55 hours of delay per car commuter (*1, 2*), around half of which is non-recurrent (*3*) (25% by accidents, 15% by weather and 10% by work zones). Those planned and unplanned incidents (e.g. hazardous weather conditions, accidents, local events, etc.) on the highway networks can catastrophically impact mobility and safety. Mitigating non-recurrent impacts requires: accurate and ahead-of-curve real-time prediction, and proactive operational management. Unfortunately, both are not fundamentally addressed despite decades of research. This paper attempts to address the first challenge, namely to accurately predict network traffic conditions up to 30 minutes in advance, particularly foreseeing traffic break-down induced by incidents in a real-time manner. It is critical to clarify we do not intend to "predict" the occurrence of an incident before it actually occurs, since generally incident occurrence is rare and random. Rather, our goal is to provide timely real-time detection of traffic anomalies due to an incident that has not been reported to traffic management centers nor social media up to the present time. To this end, timeliness and accuracy are most crucial. We use the phrase "predict incident" thereafter for simplicity, but the goal is to predict an influential incident (or anormaly) exists that has occurred and will result in impacts in the near future, before the incident is being reported anywhere (including Waze).

Despite consideration of road incidents (accidents, severe weather impact, etc) in modeling, existing approaches of traffic speed prediction fail to distinguish between recurrent scenarios, where traffic flow follows periodic patterns, and non-recurrent scenarios, where traffic flow is disrupted by certain types of incidents. In fact, predicting incident induced impacts is more crucial and practically more helpful than predicting recurrent traffic conditions. A general consensus is that existing real-time traffic prediction works reasonably well for recurrent traffic, but not yet well for non-recurrent traffic. It is still unclear how to accurately predict the network impact of incidents, planned or unplanned, by learning from multi-source emerging traffic data. It is particularly challenging to work with an incident that occurs at a location/time where incidents were extremely rare (or nonexistent) but influential. Massive non-recurrent data collected over the years is likely noisy, biased, spatially and temporally sparse, siloed by its own sensing system, and not well exploited yet. Predicting those non-recurrent traffic impacts with a sufficient lead time, e.g. 30 minutes or 1 hour ahead, is notoriously difficult.

There are decades of research on machine-learning based models to estimate and predict traffic flow or travel time. Linear time series analysis has been widely recognized and used for traffic flow/speed prediction, such as Auto-Regressive Intergrated Moving Average (ARIMA) utilized by (*4*), (*5*), (*6*), for instance. (*7*) adopts Kalman filtering for traffic flow forecasting. Non-parametric regression model was also utilized in (*8*) and (*9*). Classical machine learning models, such as Support Vector machine model, is also used to predict traffic flow (*10*) as well as travel time (*11*). Compressed sensing is exploited by (*12*) to reduce the complexity of network, then support vector regression (SVR) is used for predicting travel speed on Nationwide traffic network in Singapore. (*13*) applied hidden Markov Model by incorporating traffic volume, lane occupancy, and traffic speed data in the model, using data from a 38 mile corridor of I-4 in Orlando, FL. In addition, (*14*) and (*15*) also used Markov chains to predict travel time on arterial routes. In those studies, oftentimes features used to train machines are limited to the road segment of its own. Very few spatio-temporal features were incorporated, which can drastically improve prediction performance particularly under incidents.

Recently, studies have taken spatial-temporal correlations into consideration when predicting link travel time or flows. (6) considered spatial correlations as function of distance and degree of neighbors when applying a multivariate autoregressive moving-average model to the forecasting of traffic speed, which was tested on a dataset collected via 25 loop-detectors at Athens, Greece. (16) discusses the extensions of time series prediction model such as ARIMA by considering correlations among neighbors and the utilization of LASSO for model selection. Patterns of the spatial and temporal prediction errors are inferenced through k-means clustering as well as PCA (17). (18) defined a graph based Coefficient of Determination (CoD) matrix and utilized a modified BFS algorithm to reduce the time complexity of calculating the CoD matrix. On top of that, a graph based lag-STARIMA is proposed and used for travel time prediction. Speed probes of the city of Berlin, Germany and Thessaloníki, Greece are used in experiments. (19) proposes a method using temporal Bayesian network, and applied it on a dataset of 500 probe vehicles in San Francisco, CA on network of more than 800 links. (20) introduces a space–time diurnal (ST-D) method in which link-wise travel time correlation at multiple lag time is utilized, the model was tested on a corridor of five links. (21) utilizes Gaussian process regression (GPR) model and graphic Lasso to forecast traffic flow, and applied the model to a road network of 31 links from Beijing. (22)proposes a KNN model to forecast travel time up to one hour ahead, the model uses redefined inter-segments distances by incorporating the grade of connectivity between road segments, and considers spatial-temporal correlations and state matrices to identify traffic state, the model is evaluated on a network of 1004 road segments in Beijing. (23) proposes a modified multivariate spatial-temporal autoregressive (MSTAR) model by leveraging the distance and average speed of road networks to reduce the number of parameters, the algorithm was tested on a road network of 502 links of which the traffic status are collected by loop detectors, the results remains accurate for up to one hour. Recent development in deep learning models in general has also accelerated the advancement of traffic prediction models. Several trials has been done using (deep) neural network to estimate short-term travel time, e.g., (24). In particular, (25) proposes a restricted Boltzmann Machine (RBM)- based RNN model with two layers, the output are binary for each link: congested or not. Matrix are used to represent transportation network, no spatial information is utilized. The model is tested using taxi GPS data of Shenzhen, China. (26) proposes a deep learning method to impute traffic flow data by taking into consideration a few spatial and temporal factors, such as weather and day of week. (27, 28) exploit the spatio-temporal relations of network traffic have been widely used to make accurate and ahead-of-curve traffic predictions on a network level.

To summarize, most traffic prediction studies train models using the recurrent and non-recurrent data altogether, without explicitly separating both. The predictions are not informative when the underlying traffic clearly has multiple modalities, such as when an incident occurs. Though those models can provide a prediction of traffic conditions regardless of incident occurrence, the accuracy under incidents that are yet to be reported (thus not known at a present time) is usually unsatisfactory. The overwhelming pattern of recurrent traffic (common conditions) can drive the prediction for non-recurrent traffic (rare conditions) inaccurate, which cannot be differentiated with the performance measures on all data including both recurrent and non-recurrent conditions. Without knowing the existence of an incident explicitly, the real-time traffic operational measures can be limited or too late before they can effectively influence traffic to mitigate any impact.

Another main issue that is not intensively investigated in the literature is that spatio-temporal features used in those studies are limited. With emerging data, we can potentially obtain more highgranular information that would improve the real-time prediction, in terms of both timeliness and accuracy. In addition to extra features to train model, we may also consider exploiting existing temporal information. Except for a few probabilistic time-series studies (29), most existing traffic prediction work only gives one-shot prediction for the future traffic. Prediction into the near future in time increments, e.g. 5-60min in 5min increments, would generally provide more information to detecting traffic anomalies in advance.

To address those issues, this paper proposes a two-stage model design that separates recurrent cases and non-recurrent use cases to promote performance in traffic speed prediction particularly for the more cruicial non-recurrent traffic prediction. It contributes to the literature in the following way:

- It designs a real-time two-stage model where the first stage is to predict the probability of incidents (namely anomalies) while no incident may have been reported up to the present time. The second stage would apply a machine trained using recurrent data to predict recurrent traffic, whereas a machine trained by non-recurrent data is used to predict non-recurrent traffic. This could substantially improve modeling performance, and meanwhile provides more practical insights to traffic operators in real world.
- It proposes a multi-task learning model, with a sequence-to-sequence encoder-decoder structure with attention mechanism, as the respective machine learning model for both recurrent and non-recurrent traffic.
- It adds additional features to the machine learning model to improve the model performance. Those features include probe data in multiple granularities (such as spatial segmentations, and temporal intervals)
- It also adds features in vehicle classes (trucks versus cars). Since truck flow can more effectively impact the overall traffic conditions than cars, we would expect truck data, if available, could help improve the model performance.
- Additional features regarding probe data confidence level are also added to the model. Those confidence level information conveys not only the data quality, but also a possible proxy of traffic volume information. Building those rough volume information by vehicle classes into the machine learning model may also capture special non-recurrent traffic patterns.
- Last but not least, we highlight the model performance under incidents and show that a balance of timeliess and accuracy can be achieved with proper model tuning process.

Definition of Incident - based on Waze & RCRS

METHODOLOGY

Data Processing

Data Sources

This research utilizes multi-modal data from the following sources to construct model to predict incident occurrence and traffic speed.

- Traffic speed data from vehicle probe data, such as INRIX, in both TMC and XD resolutions. TMC (Traffic Message Channel) and XD speed data from INRIX covers all highways and major arterials. Historical data are available in various temporal granularities, such as 5-min and 10-min. XD applies a finer level of spatial granularity than INRIX TMCs and covers some more road segments.
- Incident report from both PennDOT Road Condition reporting System (RCRS)
- Waze

- Records of weather condition from Weather Underground

The format of processed input and output is illustrated in Figure 1, where for each sample instance, the input sequence $\mathbf{X}$ comprises p time steps ($t-p, t-p+1, ..., t-1, t$) and the output sequence $\mathbf{Y}$ consists of q target vectors ($t+1, t+2, ..., t+q$). Every time step in the input or output sequence corresponds to a time slot with length/granularity l (e.g. $l = 5$ min). Each of the p feature vectors ($X_{t-p}, X_{t-p+1}, ..., X_{t-1}, X_t$) in input sequence encodes multi-sourced information gathered within a traffic network, including score, incident status, speed, weather and time. Each of the q target vectors ($Y_{t+1}, Y_{t+2}, ..., Y_{t+q}$) in output sequence contains ground truth values of speed S and incident status R in targeted road segments.

The data processing follows primarily the procedures specified in (30) to generate features for the machine learning model, with differences discussed in the sections below.

Input Features

Apart from information about incident status, weather, time ofday, slowdown speed and travel time index, as introduced in (30), this work also incorporates confidence score, which refers to the sample rate of probe vehicles for speed data inference.

For TMC/XD data, confidence score implies the source of the speed data field returned, where 30 means Real-time data; 20 implies historical data, and 10 implies reference speed. A number between any two categories (e.g., 25 or 15) means the provided speed is estimated from a mixture of both sources. Clearly, any data with a confidence score other than 30 is imputed by INRIX. Lastly, cvalue is a measure of the confidence for the real-time data, ranging from 0 to 100. The higher the cvalue is, the more confident we are about the speed measurement. In this sense, the confidence score may well be correlated with traffic flow volumes and therefore, traffic speed as well. In the input features, the confidence score is normalized into $[0, 1]$ through min-max scaling.

Moreover, since the TMC speed time series contain traffic speed record of different vehicles classes, input features in this work also incorporate vehicle-specific speed from TMC data, including speed of trucks, speed of personal vehicles, and speed of all vehicles combined. The motivation lies in the observation that the traffic flow can be severely limited when trucks are running at a relatively slow speed, which may well also impact the downstream traffic speed. Therefore, including speed of various types of vehicles may well contribute to the forecasting of traffic behavior. The raw vehicle-specific speed data is collected by averaging speeds of vehicles by type and is measured in Mph. During feature processing, the vehicle-specific speed data are normalized into $[0, 1]$ through min-max scaling.

Output Target

The output sequence includes time series of incident occurrences and and traffic speed in targeted road segments, each corresponding to one prediction task.

For incident occurrence ground truth R_t at time step t, each entry $r_{t;m}$ corresponds to a targeted road segment m and indicates whether m is hit by incident (1) or not (0) based on RCRS and Waze reports.

For speed ground truth, since two versions of speed data are available (TMC and XD), the speed target vectors have two versions S_{TMC} and S_{XD}, both of which have the same number of time steps q and time slot length l, and share the same set of targeted road segments. Therefore, depending on the type of speed ground truth, there are two versions of output sequences ($\mathbf{Y}_{TMC}$ and $\mathbf{Y}_{XD}$). In particular, for time step t and road segment m, S_{XD} has only one entry whereas S_{TMC}

is three entries denoting speeds of trucks, personal vehicles and all vehicles. Values in S_{XD} and S_{TMC} is not normalized but represented by the original scale, and the model is supposed to predict the exact number.

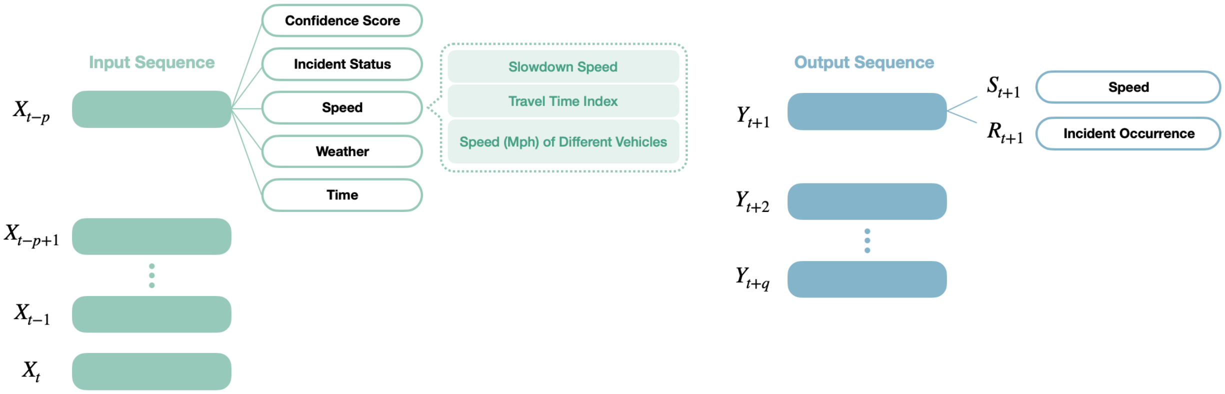


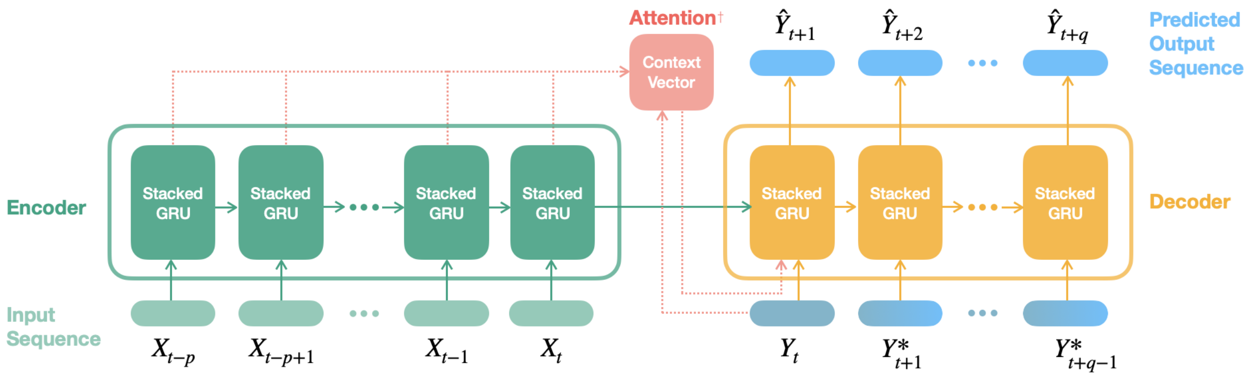
FIGURE 1 Input Sequence and Output Sequence.

Two-Stage Model Design

Similar to (30), due to the sequential nature of the input and output data in this work, the backbone of the two-stage model in this work is a encoder-decoder structure with gated recurrent units (GRUs) and attention mechanism as illustrated in Figure 2, which has been widely used in modeling sequential data for downstream tasks such as machine translation and text summarization.

The novelty of this work lies in that instead of having one encoder and one decoder to perform the single task of traffic speed prediction, the two-stage model consists of one encoder (E) followed by three decoders (D_{inc}, D_{rec}, D_{nonrec}) responsible for three downstream tasks (incident status prediction, speed prediction in recurrent case, speed prediction in non-recurrent case) respectively. As illustrated in Figure 3, after E encodes the input data into latent representations, at stage 1, D_{inc} predicts probability of incident occurrence $\hat{R} = \hat{P}(incident)$. At stage 2, D_{rec} and D_{nonrec} generates speed predictions in the recurrent scenario (S_{rec}) and the non-recurrent scenario ($\hat{S}_{nonrec}$) individually. Eventually, the two-stage model outputs $\hat{R}$ as incident occurrence prediction, and the expectation $\hat{S}_{rec} * (1 - \hat{P}_{nonrec}) + \hat{S}_{nonrec} * \hat{P}_{nonrec}$ as speed prediction $\hat{S}$, where $*$ denotes elementwise product.

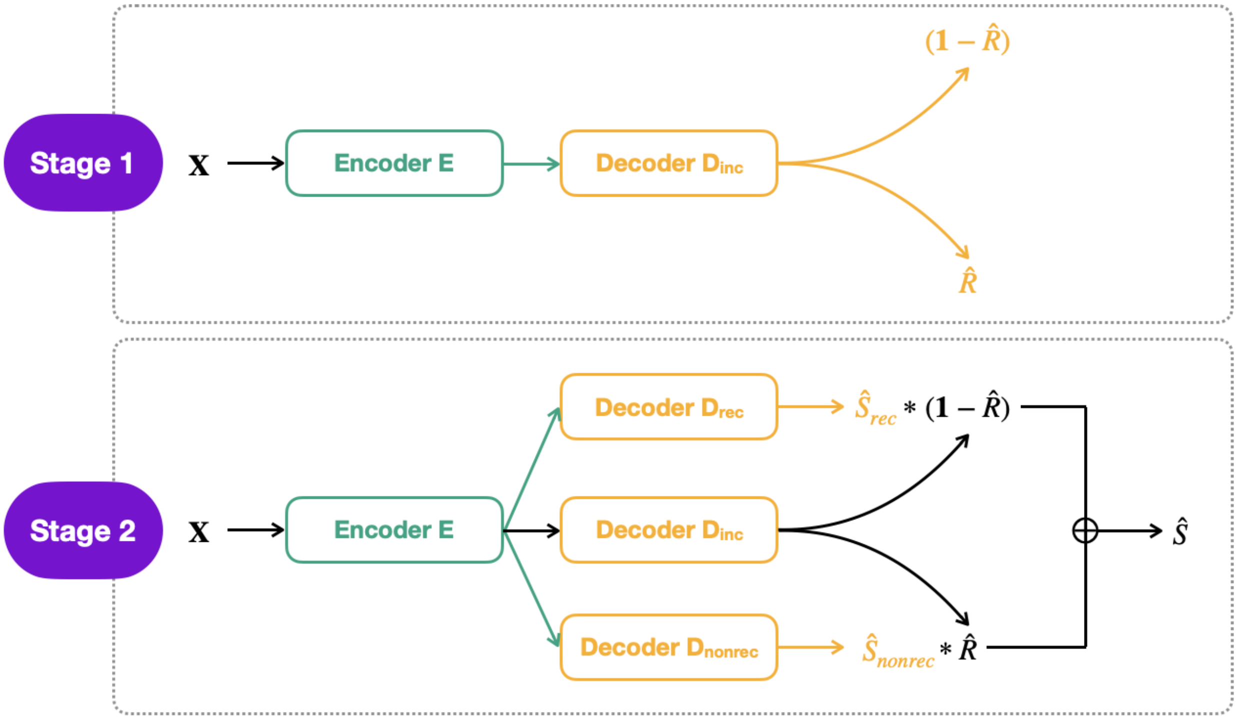
The motivation of two-stage design is that the traffic speed under recurrent scenarios and that under non-recurrent scenarios exhibit dramatically different value distributions, and therefore should be treated separately. Furthermore, rather than ensembling three individual encoder-decoder networks for the three tasks, the two-stage model leverages merely one encoder generating latent vectors that will be shared among three decoder modules. In this way, the computational overhead and model footprint will be significantly reduced. More importantly, the encoder E will not be trained to become heavily biased towards one particular type of tasks but become adept in extracting and encapsulating traffic dynamics in a collectively exhaustive manner, and therefore more versatile in fulfilling various sorts of downstream tasks revolving around traffic dynamics estimation.



* can be substituted with corresponding predicted output

† for simplicity, only the context vector associated with Y_t is illustrated here

FIGURE 2 Encoder-decoder Structure with Attention Mechanism.



* for simplicity, attention mechanism in each encoder-decoder pair is not shown

FIGURE 3 Two-stage Model Design.

Training Procedure

Training two-stage model is composed of two steps of pre-training steps and one fine-tuning step as demonstrated in Figure 4.

The first step of pre-training focuses on forecasting the probability of incident occurrence $\hat{R}$, where E and D_{inc} are trained with binary cross entropy loss computed against ground truth of incident status in the targeted road segments.

$$L_{inc} = -\frac{1}{N}\sum_{i=1}^{N} R_i \log \hat{R}_i + (1-R_i)\log(1-\hat{R}_i) \tag{1}$$

, where R_i and $\hat{R}_i$ are elements in incident occurrence ground truth R and model prediction $\hat{R}$, and N is the total number of elements in R or $\hat{R}$.

In the second pre-training step, the weights of pre-trained E is frozen while D_{rec} and D_{nonrec} are trained individually with masked recurrent speed ground truth S'_{rec} and non-recurrent speed ground truth S'_{nonrec}. S'_{rec} is the ground truth S where non-recurrent entries are 0, while S'_{nonrec} is the ground truth S where recurrent entries are 0. The loss function used in this step is mean squared error:

$$L_{rec} = \frac{1}{N_{rec}}\sum_{i=1}^{N}(s'_{rec_i} - \hat{s}'_{rec_i})^2 \tag{2}$$

$$L_{nonrec} = \boxed{\frac{1}{N_{nonrec}}}\sum_{i=1}^{N}(s'_{nonrec_i} - \hat{s}'_{nonrec_i})^2 \tag{3}$$

, where s'_{rec_i} and $\hat{s'}_{rec_i}$ are elements in the masked recurrent traffic speed ground truth S'_{rec} and model prediction $\hat{S'}_{rec}$, and N_{rec} is the total number of elements in S' or $\hat{S}'$. The same notation style applies to non-recurrent cases.

During fine-tuning, the two-stage model processes the input data with pre-trained encoder E and decoders (D_{inc}, D_{rec}, D_{nonrec}), and generates estimated future traffic speed $\hat{S}$ through two stages as delineated in Two-Stage Model Design. The traffic speed prediction $\hat{S}$ is then compared with ground truth speed data S, and the parameters of all modules in the two-stage model are updated with respect to the mean squared error loss.

$$L = \frac{1}{N}\sum_{i=1}^{N}(s_i - \hat{s}_i)^2 \tag{4}$$

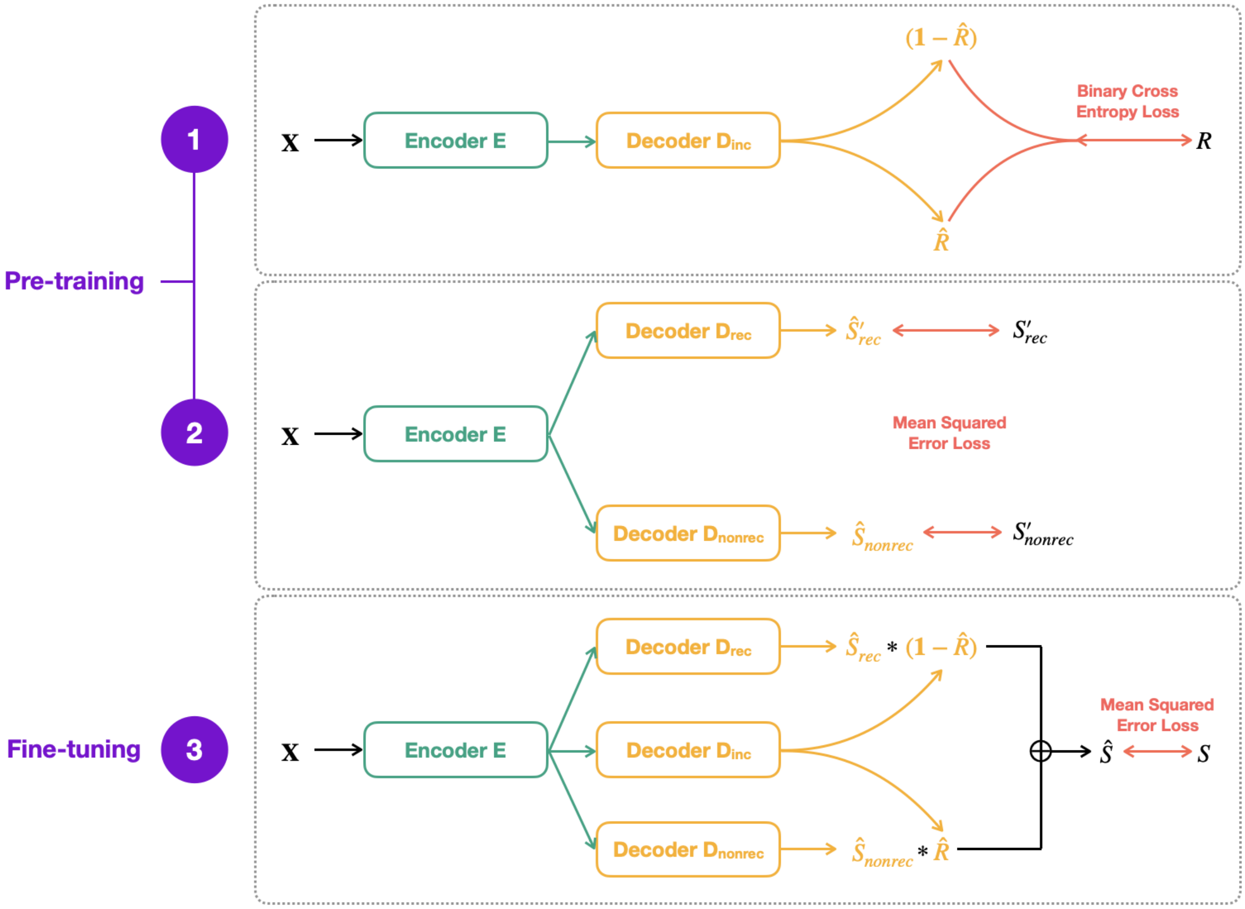
, where s_i and $\hat{s}_i$ are elements in traffic speed ground truth S and model prediction $\hat{S}$, and N is the total number of elements in S or $\hat{S}$.

EXPERIMENTS

Network settings and data

Experiments in this work focus on the dynamics in a traffic network within Cranberry Township, PA, U.S., where 70 road segments have been targeted for output and 316 peripheral road segments are selected to extract input features. Based on the selected input and output road segments, input and output dataset are constructed by sorting out multi-sourced information in a 5-min frequency from 5:30 to 20:59 per day from Feb 10, 2019 to Jul 23, 2019. For raw data with frequency shorter than 5 min, the data is aggregated to match 5-min frequency, whereas for time series that has missing entries or is collected in a frequency longer than 5 min, linear interpolation is used.

For each sample instance, the lengths of input sequence and output sequence are selected to be 7 and 6 respectively ($p = 7, q = 6$). Each component of input or output sequence corresponds to feature information or ground truth in a 5-min time slot ($l = 5$). Therefore, for each sample,



* for simplicity, attention mechanism in each encoder-decoder pair is not shown

FIGURE 4 Training Steps of Two-stage Model.

the task is to supposed to make a 30-min forecasting of incident status and traffic speed in 70 road segments based on the feature information in 35-min history. As mentioned in Output Target, there are two sources of speed data (TMC and XD). Consequently, there are two versions of target output sequence ($\mathbf{Y}_{TMC}$ and $\mathbf{Y}_{XD}$) that differ in speed data but share the same incident status data. Both versions of target output have been utilized in training and testing models.

For models that require training, the entire dataset is randomly split into training set (80%) and test set (20%), based on which models are trained until convergence and checkpoints with the best performance in test set are kept for inference and evaluation.

Baselines

Three linear baselines and one non-linear deep-learning based baseline have been experimented with in juxtaposition with the two-stage model.

Historical Average

The method of historical average generates speed prediction by averaging historical data with same "time of the day" profile.

Latest Observation

For each input sequence, the method of latest observation outputs the latest traffic speed data of the input as the estimation for the next 30 minutes (entire time span of output sequence).

LASSO Regression

LASSO regression refers to a linear regression method with L_1 regularization. In this work, each LASSO regression model (with weights $\mathbf{w}_i$) treats the flattened input sequence $\mathbf{X}_{flatten}$ as input, and one element $s_{t;m;v}$ (t denotes the index of time step; m denotes the index of road segments; v denotes vehicle type and is only used for $\mathbf{Y}_{TMC}$) of speed ground truth in the output sequence $\mathbf{Y}$ as target output. Each LASSO regression model (weights w) is trained to through optimization of the objective below, where α controls the strength of L_1 regularization and is fine-tuned through cross validation.

$$\min_{\mathbf{w}_i} \|s_{t;r;v} - \mathbf{X}_{flatten}\mathbf{w}_i\|_2^2 + \alpha_i\|\mathbf{w}_i\|_1 \tag{5}$$

Encoder-decoder-attention

(30) has proposed an encoder-decoder model based on GRU and attention mechanism that takes input feature sequence and forecasts target output sequence of traffic speed.

Tasks and Evaluation Metrics

For the task of traffic speed prediction, models are supposed to output speed in Mph for each of the 70 targeted road segments at each time step (and for each type of vehicle when $\mathbf{Y}_{TMC}$ is used). The predicted speed is evaluated by root mean squared error (RMSE) and mean absolute percentage error (MAPE) for both overall scenarios and non-recurrent scenarios:

$$RMSE(S,\hat{S}) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(s_i - \hat{s}_i)^2} \tag{6}$$

$$MAPE(S,\hat{S}) = \frac{1}{N}\sum_{i=1}^{N}\frac{\|s_i - \hat{s}_i\|}{s_i} \tag{7}$$

$$RMSE(S_{nonrec},\hat{S}_{nonrec}) = \sqrt{\frac{1}{N_{nonrec}}\sum_{i=1}^{N}(s_{nonrec_i} - \hat{s}_{nonrec_i})^2} \tag{8}$$

$$MAPE(S_{nonrec},\hat{S}_{nonrec}) = \frac{1}{N_{nonrec}}\sum_{i=1}^{N}\frac{\|s_{nonrec_i} - \hat{s}_{nonrec_i}\|}{s_{nonrec_i}} \tag{9}$$

, where s_i and $\hat{s}_i$ are elements in traffic speed ground truth S and model prediction $\hat{S}$, and N is the total number of elements in S or $\hat{S}$. *nonrec* denotes the non-recurrent portion of the data.

For the task of forecasting incident occurrence, only the two-stage model is evaluated due to its multi-task design. Accuracy and recall are adopted as the metrics to evaluate the predicted probability of incident in each of the 70 targeted road segments at each time step.

Results

In the following sections, baselines are denoted as HA (Historical Average), LO (Latest Observation), LASSO (LASSO Regression) and Seq2Seq (Encoder-decoder-attention).

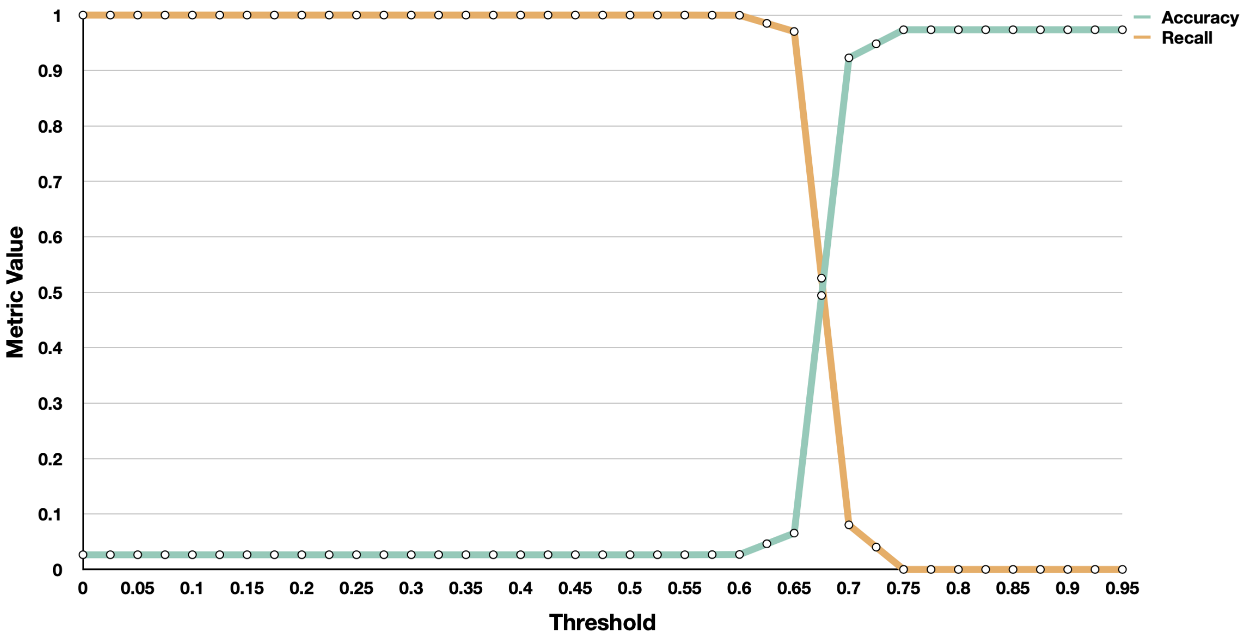


FIGURE 5 Effect of Threshold on Accuracy and Recall. Text

Incident Status Prediction

For the task of estimating incident occurrences, output from the first stage in the two-stage model($\hat{R}$) has achieved 2.59% accuracy. In spite of the extremely low accuracy, the recall of incident status prediction is 99.96%, which indicates high false positives (sensitivity) and high true positives (capability of capturing incident occurrence), while keeping the false alarms relatively low. Here the threshold of probability has been selected to be 0.1, above which the particular road segment at certain time slot is considered to be hit by an incident in the future. The low threshold aligns with the goal of this work to capture as many true incidents (true positives) as possible. In real-world practice, the threshold can be adjusted appropriately to achieve the balance between sensitivity and correctness. Figure 5 shows how accuracy and recall of two-stage model vary based on different values of threshold.

Apart from the trade-off between accuracy and recall, the timeliness of incident forecasting, before it is actually reported anything including Waze, is of prime significance. Figure 6 demonstrates an example of model inference made at 16:45 on Mar 5, 2019, 30 minutes after which a Waze report was recorded on 17:15. Based on $\hat{R}$, all 6 predicted probabilities exceed the threshold of 0.1, alarming that an incident could occur starting from 16:50 to 17:15 in road segment TMC 104-04441. In this case, the two-stage model has been able to foretell an road incident occurring 30 minutes before Waze report, while achieving more accurate traffic speed predictions than baselines such as Seq2Seq.

Speed Prediction

As illustrated in Table 1, two-stage model presents better overall speed prediction performance in terms of both RMSE and MAPE, for speed ground truth of both TMC and XD data.

More importantly, Table 2 denotes that for TMC output, in non-recurrent scenarios, two-stage model is capable of achieving up to 6.7% lower RMSE and 1.18 % lower MAPE compared

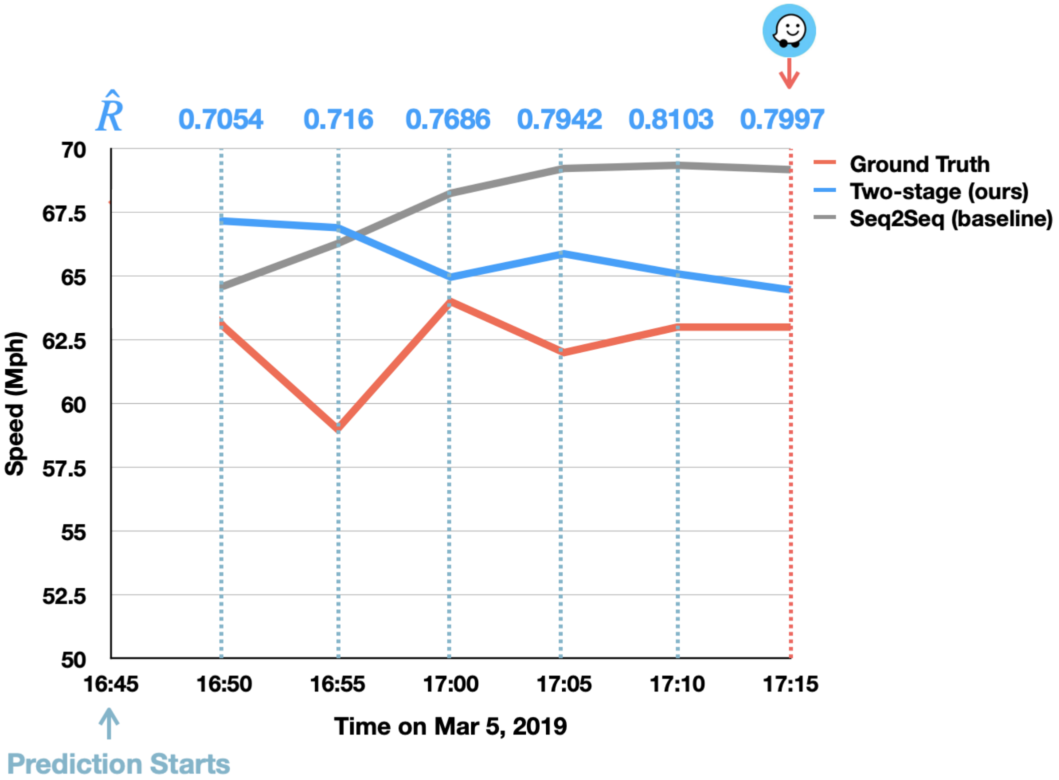


FIGURE 6 Prediction by Two-stage Model for TMC 104-04441.

with the best baseline, which demonstrates the great potential of two-stage model in dealing with non-recurrent traffic speed forecasting with TMC data as ground truth. It is worth noting that for XD output, LASSO has beaten all the other models in non-recurrent scenarios. One way to interpret the result is that XD speed data has been collected in one-min frequency, and therefore non-recurrent XD speed data tend to be noisy and volatile although despite aggregation and de-noising in data processing. Consequently, the two-stage model might have been overfitted and subdued to the linear counterpart LASSO.

In addition, both Table 1 and Table 2 include results of two-stage[†] where the output from stage 1 ($\hat{R}$) is replaced with the ground truth (R) during inference. In spite of ground truth R, there is only trivial improvement in speed prediction in both overall cases and non-recurrent cases for both versions of output. One interpretation is that the predicted incident status ($\hat{R}$) is very close to ground truth (R), which is not the case as in the former discussion of accuracy and recall. Another interpretation is that the proximity in the performance can be attributed to the great robustness of decoders D_{rec} and D_{nonrec}, which allows for slight deviation in incident prediction (stage 1) without compromising the performance of speed prediction (stage 2).

Ablation Study - Input Features

Add qualitative examples (success & failure)

To verify the effectiveness of incorporating confidence score, speed of trucks and speed of personal vehicles in input features, further experiments have been conducted, where two-stage models are trained and tested with all types of ablated input. The speed prediction results of ablation study have been summarized in Table 3 and 4. For the notation of "Input", the first "T" or "F" refers to

TABLE 1 Overall MAPE and MAPE in Traffic Speed Prediction

Output Version	Model	5 min	10 min	15 min	20 min	25 min	30 min
RMSE							
TMC	two-stage (ours)	**5.6350**	**5.5212**	**5.4600**	**5.4218**	**5.4586**	**5.7659**
	two-stage† (ours)	**5.6350**	5.5228	5.4613	5.4250	5.4627	5.7709
	Seq2Seq	5.9418	5.7158	5.7188	5.7348	5.7874	6.0864
	LASSO	7.5945	7.7482	7.7720	7.7713	7.7763	7.7765
	LO	7.5754	8.9260	9.5044	9.8100	9.9895	10.1173
	HA	7.9806	7.9806	7.9806	7.9806	7.9806	7.9806
XD	two-stage (ours)	**3.7423**	3.6299	3.5752	3.5738	3.6291	3.8708
	two-stage† (ours)	**3.7423**	**3.6263**	**3.5700**	**3.5676**	**3.6198**	**3.8645**
	Seq2Seq	3.9233	3.9620	3.9253	3.9300	3.9840	4.1850
	LASSO	6.1853	6.1857	6.1871	6.1884	6.1893	6.1914
	LO	5.0531	6.4538	7.0386	7.2898	7.4524	7.5756
	HA	7.0171	7.0171	7.0171	7.0171	7.0171	7.0171
MAPE							
TMC	two-stage (ours)	**13.52%**	**13.26%**	**13.01%**	**12.83%**	**12.96%**	**13.94%**
	two-stage† (ours)	**13.52%**	**13.26%**	13.02%	12.84%	12.97%	13.96%
	Seq2Seq	14.39%	13.72%	13.74%	13.80%	13.99%	14.95%
	LASSO	20.41%	20.96%	21.07%	21.07%	21.09%	21.08%
	LO	16.24%	20.78%	22.81%	23.89%	24.52%	24.94%
	HA	21.69%	21.69%	21.69%	21.69%	21.69%	21.69%
XD	two-stage (ours)	**8.11%**	7.93%	7.83%	7.84%	7.99%	8.58%
	two-stage† (ours)	**8.11%**	**7.92%**	**7.82%**	**7.82%**	**7.97%**	**8.57%**
	Seq2Seq	8.52%	8.72%	8.67%	8.70%	8.86%	9.39%
	LASSO	14.23%	14.23%	14.23%	14.24%	14.24%	14.25%
	LO	8.20%	12.35%	14.36%	15.34%	15.99%	16.49%
	HA	17.81%	17.81%	17.81%	17.81%	17.81%	17.81%

two-stage†: two-stage model where the output of the first stage ($\hat{R}$) is replaced with ground truth (R) during inference

inclusion or exclusion of confidence score feature ($Feat_c$), while the second "T" or "F" refers to inclusion or exclusion of vehicle-specific speed feature ($Feat_s$).

When $\mathbf{Y}_{TMC}$ is used as ground truth, both confidence score and vehicle-specific speed have effectively contributed to the speed prediction in overall scenarios and non-recurrent scenarios. As shown in Table 3 and 4, the two-stage model has achieved the worst RMSE and MAPE when trained on input data without either $Feat_c$ or $Feat_s$. By adding either type of information into input features, the speed prediction performance is improved, and reaches optimum when $Feat_c$ and $Feat_s$ are both incorporated in input features. When $\mathbf{Y}_{XD}$ is used as ground truth, although the two-stage model benefits from the incorporation of either $Feat_c$ or $Feat_s$, blending both features into input data in fact has undermined the speed prediction performance, which might have been

TABLE 2 Non-recurrent MAPE and MAPE in Traffic Speed Prediction

Output Version	Model	5 min	10 min	15 min	20 min	25 min	30 min
RMSE							
TMC	two-stage (ours)	**5.7822**	5.6641	5.6014	5.5670	5.6097	5.9119
	two-stage† (ours)	**5.7822**	**5.6618**	**5.5939**	**5.5595**	**5.6067**	**5.9047**
	Seq2Seq	6.1356	5.9267	5.9261	5.9379	5.9842	6.2766
	LASSO	7.0363	7.1983	7.2226	7.2207	7.2256	7.2228
	LO	7.7558	9.1557	9.7964	10.1029	10.3368	10.5160
	HA	19.1033	19.1033	19.1033	19.1033	19.1033	19.1033
XD	two-stage (ours)	6.8330	6.5209	6.3753	6.3273	6.4927	7.1226
	two-stage† (ours)	6.8330	6.5010	6.3417	6.3022	6.4533	7.0785
	Seq2Seq	9.0208	8.8264	8.6987	8.7100	8.7994	9.2113
	LASSO	**6.1925**	**6.1927**	**6.1940**	**6.1951**	**6.1960**	**6.1979**
	LO	9.9743	12.5913	13.7536	14.3201	14.6805	14.9339
	HA	16.6876	16.6876	16.6876	16.6876	16.6876	16.6876
MAPE							
TMC	two-stage (ours)	**14.40%**	**14.11%**	13.86%	13.68%	**13.81%**	14.89%
	two-stage† (ours)	**14.40%**	**14.11%**	**13.84%**	**13.67%**	**13.81%**	**14.86%**
	Seq2Seq	15.48%	14.79%	14.81%	14.83%	14.99%	16.01%
	LASSO	18.53%	19.07%	19.18%	19.18%	19.19%	19.18%
	LO	17.41%	22.19%	24.59%	25.81%	26.65%	27.34%
	HA	92.20%	92.20%	92.20%	92.20%	92.20%	92.20%
XD	two-stage (ours)	20.54%	19.78%	19.30%	19.14%	19.82%	21.95%
	two-stage† (ours)	20.54%	19.71%	19.25%	19.14%	19.74%	21.82%
	Seq2Seq	28.74%	28.30%	27.71%	27.79%	28.32%	30.17%
	LASSO	**14.32%**	**14.31%**	**14.31%**	**14.32%**	**14.32%**	**14.33%**
	LO	24.81%	37.89%	44.50%	48.15%	50.66%	52.54%
	HA	72.24%	72.24%	72.24%	72.24%	72.24%	72.24%

due to the noises present in XD speed data.

In summary, both types of features ($Feat_c$ and $Feat_s$) have proven useful in traffic speed prediction, regardless of the type of ground truth ($\mathbf{Y}_{TMC}$ or $\mathbf{Y}_{XD}$) and scenarios (overall or non-recurrent only). For XD target speed in particular, introducing both features together in input might bring about negative impact as seen in the ablation result, which can be attributed to the noisy nature of XD data available. In practice, the selection of input features should be decided based on the type of ground truth data that the model is supposed to fit.

As for incident status prediction, as previously pointed out, the threshold can be fine-tuned to accommodate the practical need of sensitivity and ability to capture incident occurrences, and Table 5 (threshold = 0.1) is listed here merely for reference.

TABLE 3 Overall MAPE and MAPE in Traffic Speed Prediction with Ablation in Input Features

Output Version	Input	5 min	10 min	15 min	20 min	25 min	30 min
RMSE							
TMC	TT	**5.6350**	**5.5212**	**5.4600**	**5.4218**	**5.4586**	**5.7659**
	TF	5.8277	5.5772	5.5450	5.5295	5.5638	5.8781
	FT	5.8249	5.6556	5.6249	5.6160	5.6490	5.9683
	FF	5.9036	5.6683	5.6216	5.6093	5.6375	5.9221
XD	TT	3.7423	3.6299	3.5752	3.5738	3.6291	3.8708
	TF	3.6425	3.5595	3.5214	3.5188	3.5636	3.8270
	FT	**3.6206**	**3.5345**	**3.4997**	**3.5032**	**3.5541**	**3.8110**
	FF	3.6896	3.6076	3.5636	3.5580	3.5921	3.8344
MAPE							
TMC	TT	**13.52**%	**13.26**%	**13.01**%	**12.83**%	**12.96**%	**13.94**%
	TF	14.13%	13.35%	13.24%	13.20%	13.33%	14.36%
	FT	13.98%	13.47%	13.36%	13.36%	13.51%	14.57%
	FF	14.31%	13.60%	13.45%	13.42%	13.55%	14.45%
XD	TT	8.11%	7.93%	7.83%	7.84%	7.99%	8.58%
	TF	7.84%	7.69%	7.62%	**7.63**%	**7.76**%	**8.39**%
	FT	**7.83**%	**7.68**%	**7.61**%	**7.63**%	7.77%	**8.39**%
	FF	7.98%	7.87%	7.78%	7.78%	7.89%	8.46%

CONCLUSION

This paper proposes a two-stage multi-task learning model that considers recurrent and non-recurrent scenarios separately while performing the prediction of both incident occurrence and traffic speed in a single iteration of inference. It leverages multi-source data in different granularities, particularly in TMC and XD segmentations, car and truck specific speeds, and confidence scores of probe speed data. Results of diverse experiments have demonstrated the effectiveness of two-stage model design in foretelling incident occurrence in a timely manner as well as accurately forecasting traffic speed, especially under non-recurrent situations. It demonstrats that it holds great potential to pick up all incidents before they are being reported anywhere including social media. This would give traffic operators much flexibility and sufficient lead time to engage proactive traffic mitigation measures. Moreover, results from ablation study further validate the usefulness of additional input features (confidence score and vehicle-specific speeds) in estimating near-future traffic speed, which were not used previously in the literature. It is also demonstrated that having both TMC and XD in the features would help improve the prediction results. Predicting TMC level speeds is generally more accurate than predicting XD level speeds, albeit at the price of coarser spatial granularity. XD has a finer resolution, but is generally more noisy and more challenging to predict.

There are various directions where the potentials of two-stage model can be further excavated. One major next step can be the ablation study on the number of time steps (p,q) and length of time intervals (l) of input and output sequences. We will also apply this model to other geographic regions to fully examine its replicability. In addition, other than taking expectations, there

TABLE 4 Non-recurrent MAPE and MAPE in Traffic Speed Prediction with Ablation in Input Features

Output Version	Input	5 min	10 min	15 min	20 min	25 min	30 min
RMSE							
TMC	TT	**5.7822**	**5.6641**	**5.6014**	**5.5670**	**5.6097**	**5.9119**
	TF	6.0091	5.7790	5.7492	5.7381	5.7811	6.0953
	FT	6.0218	5.8509	5.8182	5.8181	5.8567	6.1844
	FF	6.0796	5.8431	5.7946	5.7759	5.8127	6.1291
XD	TT	6.8330	6.5209	6.3753	6.3273	6.4927	7.1226
	TF	6.6751	6.3546	6.2882	**6.2704**	**6.2934**	**6.9280**
	FT	**6.6442**	**6.3359**	**6.2677**	6.2963	6.4466	7.1292
	FF	6.8138	6.4569	6.3011	6.3110	6.4070	7.0065
MAPE							
TMC	TT	**14.40%**	**14.11%**	**13.86%**	**13.68%**	**13.81%**	**14.89%**
	TF	15.30%	14.47%	14.34%	14.28%	14.46%	15.59%
	FT	15.06%	14.47%	14.39%	14.39%	14.59%	15.71%
	FF	15.29%	14.49%	14.33%	14.29%	14.47%	15.58%
XD	TT	20.54%	19.78%	19.30%	19.14%	19.82%	21.95%
	TF	19.90%	19.20%	18.99%	**18.88%**	**18.92%**	**21.07%**
	FT	**19.70%**	**19.07%**	**18.85%**	18.92%	19.44%	21.98%
	FF	20.30%	19.55%	19.07%	19.11%	19.46%	21.41%

TABLE 5 Performance in Incident Prediction with Ablation in Input Features (Threshold=0.1)

Output Version	Input	Accuracy	Recall
RMSE			
TMC	TT	2.59%	**99.96%**
	TF	2.59%	86.35%
	FT	**2.90%**	95.97%
	FF	2.59%	84.92%
XD	TT	2.28%	55.82%
	TF	**2.33%**	**60.27%**
	FT	2.31%	57.11 %
	FF	2.31%	57.92%

are a variety of ways to leverage outputs from three decoder to generate traffic speed prediction, which is well worth investigating.

SOURCE CODE

The code for this study can be found on https://github.com/HaoWoo96/Traffic-Prediction.

ACKNOWLEDGEMENTS

This research is supported by US Department of Transportation Exploratory Advanced Research Award 693JJ321C000013. The contents of this paper reflect the views of the authors only, who are responsible for the facts and the accuracy of the information presented herein.

AUTHOR CONTRIBUTION STATEMENT

Hao Wu: study design, data processing, data analysis, interpretation of results, manuscript preparation. Weiran Yao: study design, literature view, data processing, interpretation of results, manuscript preparation. Sean Qian: Study conception and design, literature review, data collection, interpretation of results, manuscript preparation.

REFERENCES

1. Lasley, P., 2021 URBAN MOBILITY REPORT, 2021.
2. Hampshire, R. C., P. Hu, R. Schmitt, J. Schwarzer, S. Jahanmir, W. H. Moore, et al., Pocket Guide to Transportation 2022, 2022.
3. FHWA, Reducing Non-Recurring Congestion, 2022.
4. Pace, R. K., R. Barry, J. M. Clapp, and M. Rodriquez, Spatiotemporal autoregressive models of neighborhood effects. *The Journal of Real Estate Finance and Economics*, Vol. 17, No. 1, 1998, pp. 15–33.
5. Kamarianakis, Y. and P. Prastacos, Space–time modeling of traffic flow. *Computers & Geosciences*, Vol. 31, No. 2, 2005, pp. 119–133.
6. Kamarianakis, Y. and P. Prastacos, Forecasting traffic flow conditions in an urban network: comparison of multivariate and univariate approaches. *Transportation Research Record: Journal of the Transportation Research Board*, , No. 1857, 2003, pp. 74–84.
7. Guo, J., W. Huang, and B. M. Williams, Adaptive Kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification. *Transportation Research Part C: Emerging Technologies*, Vol. 43, 2014, pp. 50–64.
8. Smith, B. L., B. M. Williams, and R. K. Oswald, Comparison of parametric and nonparametric models for traffic flow forecasting. *Transportation Research Part C: Emerging Technologies*, Vol. 10, No. 4, 2002, pp. 303–321.
9. Rahmani, M., E. Jenelius, and H. N. Koutsopoulos, Non-parametric estimation of route travel time distributions from low-frequency floating car data. *Transportation Research Part C: Emerging Technologies*, Vol. 58, 2015, pp. 343–362.
10. Cong, Y., J. Wang, and X. Li, Traffic Flow Forecasting by a Least Squares Support Vector Machine with a Fruit Fly Optimization Algorithm. *Procedia Engineering*, Vol. 137, 2016, pp. 59–68.
11. Wu, C.-H., J.-M. Ho, and D.-T. Lee, Travel-time prediction with support vector regression. *Intelligent Transportation Systems, IEEE Transactions on*, Vol. 5, No. 4, 2004, pp. 276–281.
12. Mitrovic, N., M. T. Asif, J. Dauwels, and P. Jaillet, Low-dimensional models for compressed sensing and prediction of large-scale traffic data. *Intelligent Transportation Systems, IEEE Transactions on*, Vol. 16, No. 5, 2015, pp. 2949–2954.
13. Qi, Y. and S. Ishak, A Hidden Markov Model for short term prediction of traffic conditions on freeways. *Transportation Research Part C: Emerging Technologies*, Vol. 43, 2014, pp. 95–111.
14. Ramezani, M. and N. Geroliminis, On the estimation of arterial route travel time distribution with Markov chains. *Transportation Research Part B: Methodological*, Vol. 46, No. 10, 2012, pp. 1576–1590.
15. Yeon, J., L. Elefteriadou, and S. Lawphongpanich, Travel time estimation on a freeway using Discrete Time Markov Chains. *Transportation Research Part B: Methodological*, Vol. 42, No. 4, 2008, pp. 325–338.
16. Kamarianakis, Y., W. Shen, and L. Wynter, Real-time road traffic forecasting using regime-switching space-time models and adaptive LASSO. *Applied stochastic models in business and industry*, Vol. 28, No. 4, 2012, pp. 297–315.

17. Asif, M. T., J. Dauwels, C. Y. Goh, A. Oran, E. Fathi, M. Xu, M. M. Dhanya, N. Mitrovic, and P. Jaillet, Spatiotemporal patterns in large-scale traffic speed prediction. *Intelligent Transportation Systems, IEEE Transactions on*, Vol. 15, No. 2, 2014, pp. 794–804.

18. Salamanis, A., D. D. Kehagias, C. K. Filelis-Papadopoulos, D. Tzovaras, and G. A. Gravvanis, Managing Spatial Graph Dependencies in Large Volumes of Traffic Data for Travel-Time Prediction, ????

19. Hunter, T., A. Hofleitner, J. Reilly, W. Krichene, J. Thai, A. Kouvelas, P. Abbeel, and A. Bayen, Arriving on time: estimating travel time distributions on large-scale road networks. *arXiv preprint arXiv:1302.6617*, 2013.

20. Zou, Y., X. Zhu, Y. Zhang, and X. Zeng, A space–time diurnal method for short-term freeway travel time prediction. *Transportation Research Part C: Emerging Technologies*, Vol. 43, 2014, pp. 33–49.

21. Sun, S., R. Huang, and Y. Gao, Network-scale traffic modeling and forecasting with graphical lasso and neural networks. *Journal of Transportation Engineering*, Vol. 138, No. 11, 2012, pp. 1358–1367.

22. Cai, P., Y. Wang, G. Lu, P. Chen, C. Ding, and J. Sun, A spatiotemporal correlative k-nearest neighbor model for short-term traffic multistep forecasting. *Transportation Research Part C: Emerging Technologies*, Vol. 62, 2016, pp. 21–34.

23. Min, W. and L. Wynter, Real-time road traffic prediction with spatio-temporal correlations. *Transportation Research Part C: Emerging Technologies*, Vol. 19, No. 4, 2011, pp. 606–616.

24. Cui, Z., K. Henrickson, R. Ke, and Y. Wang, Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting. *IEEE Transactions on Intelligent Transportation Systems*, Vol. 21, No. 11, 2019, pp. 4883–4894.

25. Ma, X., H. Yu, Y. Wang, and Y. Wang, Large-scale transportation network congestion evolution prediction using deep learning theory. *PloS one*, Vol. 10, No. 3, 2015, p. e0119044.

26. Duan, Y., Y. Lv, Y.-L. Liu, and F.-Y. Wang, An efficient realization of deep learning for traffic data imputation. *Transportation research part C: emerging Technologies*, Vol. 72, 2016, pp. 168–181.

27. Polson, N. G. and V. O. Sokolov, Deep learning for short-term traffic flow prediction. *Transportation Research Part C: Emerging Technologies*, Vol. 79, 2017, pp. 1–17.

28. Cui, Z., R. Ke, Z. Pu, X. Ma, and Y. Wang, Learning traffic as a graph: A gated graph wavelet recurrent neural network for network-scale traffic prediction. *Transportation Research Part C: Emerging Technologies*, Vol. 115, 2020, p. 102620.

29. Salinas, D., V. Flunkert, J. Gasthaus, and T. Januschowski, DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, Vol. 36, No. 3, 2020, pp. 1181–1191.

30. Yao, W. and S. Qian, Learning to Recommend Signal Plans under Incidents with Real-Time Traffic Prediction. *Transportation Research Record: Journal of the Transportation Research Board*, Vol. 2674, No. 6, 2020, pp. 45–59.