APOTS: A Model for Adversarial Prediction of Traffic Speed

Namhyuk Kim[1*], Junho Song[2*], Siyoung Lee[2], Jaewon Choe[2],
Kyungsik Han[2], Sunghwan Park[1], and Sang-Wook Kim[2†]
[1]*Hyundai Motor Company*, Seoul, Republic of Korea
{namhyuk.kim, sunghwan.park}@hyundai.com
[2]*Hanyang University*, Seoul, Republic of Korea
{hahadori, sylee88, dnjsdlgdi, kyungsikhan, wook}@hanyang.ac.kr

2022 IEEE 38th International Conference on Data Engineering (ICDE) | 978-1-6654-0883-7/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICDE53745.2022.00316

***Abstract*—Many global automakers strive to develop technologies towards the next-generation of intelligent transportation systems (ITS). One of the primary goals of ITS is predicting future traffic speeds to optimize a driver's route, which can lead to not only alleviating traffic flow but also increasing user satisfaction with an ITS service. While prior studies have applied deep learning models to traffic speed prediction and improved model performance, existing models did not well capture *abrupt speed changes*. In this paper, we propose a novel model, named as adversarial prediction of traffic speed (APOTS), based on *adversarial training*, *data augmentation*, and *hybrid deep learning modeling*. Through the experiments with real traffic data provided by Hyundai Motor Company, we demonstrate that APOTS effectively learns dynamics of traffic speed changes and predicts traffic speed up to 40% higher in accuracy than existing prediction models.**

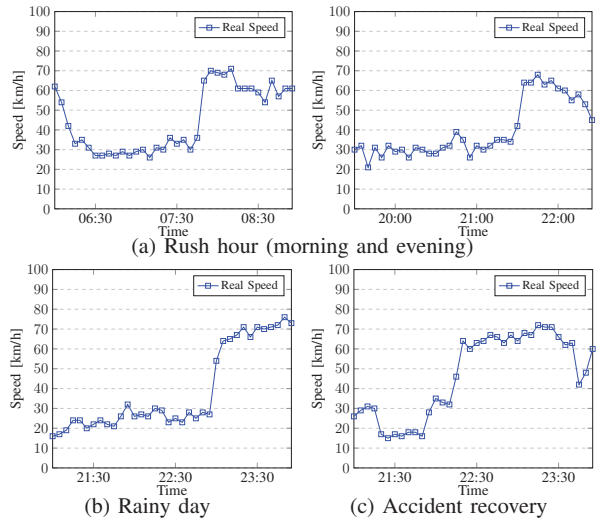
***Index Terms*—traffic speed prediction, adversarial training**

I. Introduction

Recently, many automobile manufacturing companies are striving to develop and secure technologies for establishing the next generation of intelligent transportation systems (ITS). Global leading automakers such as Hyundai Motor Company are also investing and developing technologies for convenient road guidance, such as the prediction of future traffic speed.

Traffic speed prediction refers to predicting a speed $\hat{s}_{t+\beta}$ at the prediction time $t+\beta$ from the present time t, based on a sequence of the past traffic speeds $S_{t-\alpha:t-1} = [s_{t-\alpha}, s_{t-\alpha+1}, \cdots, s_{t-2}, s_{t-1}]$ from $t-\alpha$ to $t-1$. Prior studies have tried to solve such prediction problems by the algorithms that learn time-series patterns. For instance, researchers predicted the traffic data through auto-regressive integrated moving average (ARIMA) [1], proposed a prediction model using support vector machines (SVMs) [2], [3], and employed k-nearest neighbor (KNN) [4] to predict short-term traffic speed.

As deep learning technology has shown its promise in many domains [5]–[8], a growing number of studies are proposing various deep learning models and demonstrating improved performance through their methods in traffic speed prediction. Studies proposed the models based on recurrent neural networks (RNNs) which consider the *sequential correlation* of speeds along time [9]–[15], convolutional neural networks (CNNs) which leverage the *spatio-temporal correlation* of speeds between the target and adjacent roads [16]–[18], or other deep learning models (e.g., stacked autoencoder, attention networks) [19]–[25].



(a) Rush hour (morning and evening)

(b) Rainy day (c) Accident recovery

Fig. 1: Case of abrupt changes in traffic speed.

However, while deep learning models have improved the performance of traffic speed prediction, it is still difficult to predict abrupt changes of speed in some situations as shown in Fig 1: sudden congestion during rush hour (Figs 1a), rainfall (Fig 1b), and traffic accident (Fig 1c) on the Gyeongbu Expressway passing through Seoul, South Korea. Accurately predicting such *abrupt speed changes* and suggesting an alternative route in these situations will help alleviate traffic flow and increase user satisfaction in a service of model application.

Based on this motivation, in this paper, we propose a novel deep-learning-based traffic speed prediction model named as *adversarial prediction of traffic speed* (APOTS), which has two goals: (1) define the situations that involve sudden speed changes and improve traffic speed prediction accuracy in those situations and (2) maintain the accuracy of traffic speed predictions in normal situations.

To achieve these goals, APOTS leverages the following two ideas. The first idea is the application of the *adversarial training* to capture changes of traffic speeds from the distribution of data. The generative adversarial network (GAN) [26] is one of the popular deep-learning techniques that has shown successful results of model improvement in many fields. In the adversarial

*The paper has two first authors having contributed equally to this work.
†Corresponding author.

2375-026X/22/$31.00 ©2022 IEEE
DOI 10.1109/ICDE53745.2022.00316

training, a generator (G) learns the distribution of real data through a two-player minimax game, and a discriminator (D) determines whether the real data is generated by G. Instead of G, APOTS adopts a predictor (P), which learns the distribution of speeds during the situation that involves abrupt speed changes from D through the adversarial training.

The second idea is the use of *contextual information* that often influences the condition of road traffic. To effectively learn the distribution of speeds on sudden change, the model needs to identify unusual cases of abrupt speed changes. APOTS uses *additional data*, such as speed changes of the forward and backward road sections, weather condition, presence of car accidents or events (e.g., sports game, concert), and time of the day. Through this, APOTS can capture the distribution of speeds on those unusual cases and improve prediction accuracy with such types of contextual information.

In addition, APOTS provides three types of prediction models (i.e., LSTM, CNN, Hybrid (i.e., CNN+LSTM)) as P with promising performances, which can be flexibly employed by other studies and domains depending on their data availability and operating environments.

In summary, APOTS has the following three key features:

- *Adversarial training*: APOTS learns the distribution of real speed sequences to predict the speed at certain time accurately.
- *Additional data use*: APOTS exploits contextual information at given time to predict the future speed.
- *Predictor refinement*: APOTS employs LSTM, CNN, and hybrid models (CNN+LSTM) to learn the characteristics of data.

With the real-world data provided by Hyundai Motor Company, we demonstrate a greater performance of APOTS than existing prediction models. In particular, APOTS with the hybrid model (CNN+LSTM) showed 12.80 of MAPE, which is 40%, 31%, and 30% improvements in performance compared with the FC (fully connected model), LSTM, and CNN models, respectively.

II. Related Work

Traffic flow prediction is essential of transportation services such as public transportation guidance, vehicle navigation, and congestion avoidance [27]–[30]. Initially, a statistical model was proposed to regard traffic flow as time-series data and formulate its periodicity [31], [32]. Along this line, other models, such as the Bayesian model [32], queuing based model [33], cell transmission model [34], Vector Autoregressive model [35], and k-nearest neighbor model [4] were utilized. However, these methods are difficult to reflect the *nonlinear* dependencies of large-scale traffic flow data.

As deep learning has shown successful results in many domains [5]–[8], it has been also applied to traffic flow prediction and showed better performance by capturing non-linearity of temporal dependencies [19], [36]. Pack and Rilett [37] adopted fully connected networks, and Huang et al. [38] combined fully connected networks with a regression model. Lv et al. [19] employed an auto-encoder to learn the latent features from historical traffic flow data. Some studies [36], [39], [40] tried to use auxiliary data which plays an important role in increasing prediction performance. However, simply concatenating the auxiliary data to primary data lacks generality and misses the consideration of their correlations.

Moreover, whereas many of studies [5]–[8], [19], [36], [36]–[42] focus on the traffic speed prediction on general situations, to the best of our knowledge, there have not been studies focusing on the speed prediction of special situations in which abrupt speed changes occur. To address this issue, prior studies [19], [21] predicted the traffic speed by employing deep-learning models to learn non-linear correlations between the speed and a series of past speeds. However, since the models are trained for making the prediction to be close to real speeds with high frequency in given speed sequences [43], abrupt changes that occur less frequently (e.g., accidents, commute times as shown in Fig 1) are often difficult to be learned by the model.

In this paper, we present a new direction to overcome the limitation of capturing dynamics of traffic speeds. We propose a novel model for traffic speed prediction, named as *adversarial prediction of traffic speed* (APOTS), by incorporating the following three features: first, we adopt the adversarial training to mimic the distribution of a series of real speeds; second, we exploit additional types of contextual information – adjacent-speed data and non-speed data – that characterize the traffic situation; third, we employ LSTM, CNN [9]–[18], [22]–[24], and Hybrid model to take advantage of the data characteristics.

III. Adversarial Prediction of Traffic Speed

A. Architectural overview

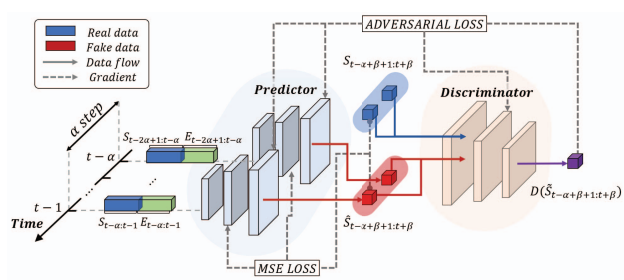


Fig. 2: Architectural overview of APOTS.

As shown in Fig 2, APOTS consists of two parts: predictor (P) and discriminator (D). For P, we use any existing deep-learning based traffic speed prediction model. We consider the most commonly used, fully-connected model (FC) and employ more-complicated one in Section IV-B.

We define $S_{i:j} = [s_i, s_{i+1}, \cdots, s_{j-1}, s_j]$, where $j - i + 1 = \alpha$, as a sequence composed of α speeds. P receives the speed sequence $S_{t-\alpha:t-1}$ and predicts the speed $\hat{s}_{t+\beta}$ at $t + \beta$ from the present t. In APOTS, P repeatedly predicts α speeds and delivers the predicted speeds in $\hat{S}_{t-\alpha+\beta+1:t+\beta}$ to D. In contrast, D receives $\hat{S}_{t-\alpha+\beta+1:t+\beta}$ from P and discriminates whether it is real or fake. In other words, D outputs $D(\hat{S}_{t-\alpha+\beta+1:t+\beta})$, the probability that the input speed

sequence $\hat{S}_{t-\alpha+\beta+1:t+\beta}$ came from the distribution of the real speed sequences (Fig 2).

We use $S_{t-\alpha:t-1}$, a series of speeds, as input to D, instead of individual speed. The reason can be explained as follows. Since P is trained by reducing the absolute difference between $s_{t+\beta}$ (i.e., the real (observed) speed) and $\hat{s}_{t+\beta}$ (i.e., the predicted speed) by the MSE loss, as the training progresses, the predicted speed becomes close to the real (observed) speed in the situation. In addition, we observed that in most cases the speed at a certain time is not that different from the speed at its neighboring time. Thus, it could happen P predicts the exactly same speed as the real (observed) speed at a given time. Then, the same speed from P is passed as fake to D and that from real data is pass as real to D; for D, discriminating the *same speeds with conflicting labels* causes degradation and transfers the wrong feedback to P; As a result, P's learning of the real speed distribution could fail. This problem similarly occurred in our CFGAN [44], GAN used in recommendation systems.

B. Adversarial training

Formally, the objective function of P is formulated as follows:

$$\begin{aligned} J^P =& \mathbb{E}_{s\sim p_{data},\hat{s}\sim p_\phi}[(s_{t+\beta}-\hat{s}_{t+\beta})^2]+ \\ & \mathbb{E}_{\hat{s}\sim p_\phi}[log(1-D(\hat{S}_{t-\alpha+\beta+1:t+\beta}))]. \end{aligned} \tag{1}$$

In Eq 1, $\mathbb{E}_{s\sim p_{\text{data}},\hat{s}\sim p_\phi}[(s_{t+\beta}-\hat{s}_{t+\beta})^2]$ indicates the MSE loss that represents the absolute difference between the predicted speed $\hat{s}_{t+\beta}$ and the real speed $s_{t+\beta}$, and $\mathbb{E}_{\hat{s}\sim p_\phi}[log(1-D(\hat{S}_{t-\alpha+\beta+1:t+\beta}))]$ does an adversarial loss that represents the difference between the distribution represented by the model parameter ϕ of P and that by the real speed sequences. $D(\hat{S}_{t-\alpha+\beta+1:t+\beta})$ indicates the probability, computed by D, that the predicted speed sequence $\hat{S}_{t-\alpha+\beta+1:t+\beta}$ came from the distribution of the real speed sequences (i.e., probability of real), and $1-D(\hat{S}_{t-\alpha+\beta+1\cdot t+\beta})$ does its exclusive probability (i.e., probability of fake). Since P tries to make $D(\hat{S}_{t-\alpha+\beta+1:t+\beta})$ to 1, P is trained by minimizing the objective function J^P [1].

The objective function of D is formulated as follows:

$$\begin{aligned} J^D =& \mathbb{E}_{s\sim p_{data}}[log\ D(S_{t-\alpha+\beta+1:t+\beta})]+ \\ & \mathbb{E}_{\hat{s}\sim p_\phi}[log(1-D(\hat{S}_{t-\alpha+\beta+1:t+\beta}))]. \end{aligned} \tag{2}$$

In Eq 2, D tries to make $D(S_{t-\alpha+\beta+1:t+\beta})$ equal to 1 with the real speed sequence $S_{t-\alpha+\beta+1:t+\beta}$ and make $(1-D(\hat{S}_{t-\alpha+\beta+1:t+\beta}))$ equal to 0 with the predicted speed sequence $\hat{S}_{t-\alpha+\beta+1:t+\beta}$ by P. That is, D is trained to maximize the objective function J^D.

IV. Refinements on a predictor

A. Exploiting additional data

We use additional data sequence $E_{t-\alpha:t-1}$ to better understand the situation in which $S_{t-\alpha:t-1}$ occurred, defined below:

$$E_{t-\alpha:t-1} = S^{Adj}_{t-\alpha:t-1} \oplus \bar{S}_{t-\alpha:t-1} \tag{3}$$

where $S^{Adj}_{t-\alpha:t-1}$ denotes adjacent-speed data and $\bar{S}_{t-\alpha:t-1}$ denotes non-speed data (both will be explained in more detail later). The final objective function of APOTS is defined as follows:

$$\begin{aligned} &\min_\phi \max_\theta J(D,P) = \\ &\quad \mathbb{E}_{s\sim p_{\text{data}},\hat{s}\sim p_\phi}[(s_{t+\beta}-\hat{s}_{t+\beta})^2]+ \\ &\quad \mathbb{E}_{s\sim p_{\text{data}}}[log\ D(S_{t-\alpha+\beta+1:t+\beta}|E_{t-\alpha:t-1})]+ \\ &\quad \mathbb{E}_{\hat{s}\sim p_\phi}[log(1-D(\hat{S}_{t-\alpha+\beta+1:t+\beta}|E_{t-\alpha:t-1}))]. \end{aligned} \tag{4}$$

Adjacent-speed data: The adjacent roads include the upstream and downstream roads of the target road as shown in Fig 3. The speed in the adjacent roads has a spatio-temporal correlation with the speed of the target road. For example, if congestion occurs due to an accident at a road, its following roads in the same direction will be likely to have the congestion soon.

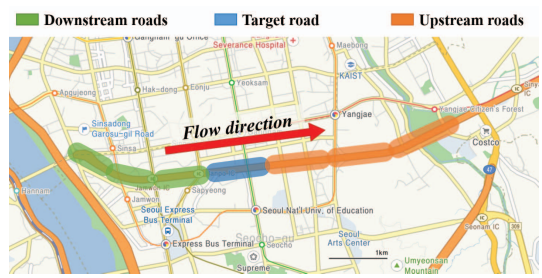


Fig. 3: Target road and adjacent roads in Gyeongbu highway.

To predict the future speed in the target road, we consider the speeds of the adjacent roads that have the spatio-temporal correlations to the target road. We define *adjacent-speed data* as the speed data of the *target* and the *adjacent* roads. We denote the adjacent-speed data of α speeds prior to present t for the m upstream and m downstream roads from target road h as follows:

$$S^{Adj}_{t-\alpha:t-1} = [S^{h-m}_{t-\alpha:t-1}, \cdots, S^{h}_{t-\alpha:t-1}, \cdots, S^{h+m}_{t-\alpha:t-1}] \tag{5}$$

where $S^{x}_{t-\alpha:t-1}$ is defined as a speed sequence composed of α speeds prior to the present t on the road x.

Non-speed data: Traffic speeds are related to weather, time, events, etc. For example, an accident might cause road congestion, and the weather (e.g., rainy, snowing), holidays, and commute times could affect the road speed. We thus classify non-speed data into three categories: event, weather, and time.

As with the time at which the speeds were measured, the values in weather and time data are measured by the same time interval. *Event data* include the information related to the *accident* and *construction*; if an accident occurs at time t, it is set as 1, otherwise 0. *Weather data* include the information related to *temperature* and *precipitation*; for each of them, the value measured at time t is recorded. *Time data* include the information related to *hour* and *day type*; for hour, a value between 0 and 23, depending on time t, is recorded (e.g., 7 for 7:35 AM); for day type including weekday, holiday, a day *before* holiday, and a day *after* holiday. For example, if

[1] As explained in Section III-A, the MSE loss is computed per predicted speed value, whereas the adversarial loss is computed per sequence of α speeds values. Therefore, in actual implementation, the MSE loss and the adversarial loss are learned at the ratio of $\alpha : 1$.

yesterday was Wednesday (i.e., weekday) and today is the Independence Day (i.e., holiday), yesterday is set with four values of $[1, 0, 1, 0]$. A naive predictor takes a sequence of α speeds as input. Likewise, we use the same form, a sequence of α values, for each factor in non-speed data. However, in the case of a day type, we normally have the same α values in the sequence; thus we use only one value for this case in our implementation.

B. Refining a predictor

APOTS is able to adopt any of Fully Connected (FC), Convolutional Neural Network (CNN), Long Short-Tearm Memory (LSTM [45]), and Hybrid (i.e., the combination of LSTM and CNN) models as predictor P. However, to consider the sequential, spatio-temporal correlation of traffic speeds, APOTS employs the hybrid model as P. Here, CNN captures the spatio-temporal correlation from the speed matrix of Eq 6 through the convolution layer, and LSTM captures sequential correlation of traffic speeds. Thus, the hybrid model is able to learn both spatio-temporal and sequential correlations of traffic speeds.

$$S^{Adj}_{t-\alpha:t-1} = \begin{bmatrix} S^{h-m}_{t-\alpha:t-1} \\ \vdots \\ S^{h}_{t-\alpha:t-1} \\ \vdots \\ S^{h+m}_{t-\alpha:t-1} \end{bmatrix} = \begin{bmatrix} S_{h-m,t-\alpha} & \cdots & S_{h-m,t-1} \\ \vdots & & \vdots \\ S_{h,t-\alpha} & \cdots & S_{h,t-1} \\ \vdots & & \vdots \\ S_{h+m,t-\alpha} & \cdots & S_{h+m,t-1} \end{bmatrix}. \quad (6)$$

V. Evaluation

This section reports the results of our extensive experiments to validate the effectiveness of APOTS. The experiments are designed to answer the following three key questions:

- Q1. How effective is the adversarial training for learning the abrupt change of traffic speed?
- Q2. How beneficial are the adjacent-speed and non-speed data employed additionally in APOTS?
- Q3. How does the accuracy of APOTS vary based on combinations of adversarial training, additional data, and predictors?

A. Experimental settings

We set the problem as predicting the traffic speed with $\alpha = 12$ (i.e., 1 hour) where the time between two adjacent speeds is five minutes, which reflects the requests from Hyundai Motor Company.

Dataset: We used a real-world dataset provided by Hyundai Motor Company. It includes logs of speeds, accidents, and constructions between July and October, 2018. Additionally, we crawled logs of temperatures and precipitations from Korea Meteorological Administration during the same time period.

We divided the whole speed sequence (122 days) into $35,350$ samples (12 speeds per hour) using a sliding-window technique [46]. We randomly selected 80% of the samples for training and the rest of 20% for testing. To avoid overfitting due to the overlaps between the training and test sets, we discarded the overlapped samples from the training set. We used 20% of the training samples as a validation set for hyper-parameter tuning.

TABLE I: Hyper-parameters of APOTS.

	F (FC)	L (LSTM)	C (CNN)	H (Hybrid: L+C)
Hidden layers	4	2	3	CNN (3) LSTM (2)
Hidden nodes	512, 128, 256, 64	512, 512	128, 32, 64	CNN (128, 32, 64) LSTM (512, 512)
Learning rate	0.001	0.001	0.001	0.001
Filter size	–	–	3*3, 1*1, 3*3	3*3, 1*1, 3*3

Implementation: We implemented basic fully-connected model (F), long short term memory model [9] (L), convolution neural network model [47] (C), and the hybrid model of CNN with LSTM [24] (H). Since the codes of C, L, and H proposed in [9], [24], [47] were not publicly available, we re-implemented those models, keeping the philosophy of each model for speed prediction. Then, we applied APOTS to the models, having APOTS_F, APOTS_L, APOTS_C, and APOTS_H. The discriminator is implemented with five fully-connected layers. To tune the hyper-parameters of each model of APOTS, we performed a grid search by evaluating the accuracy on the validation set as shown in Table I. Our code and datasets are all available at https://tinyurl.com/33vnf4tp.

Metrics: To evaluate the models, we used three metrics; Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE). However, we explain the results with MAPE because of page limitations.

B. Results

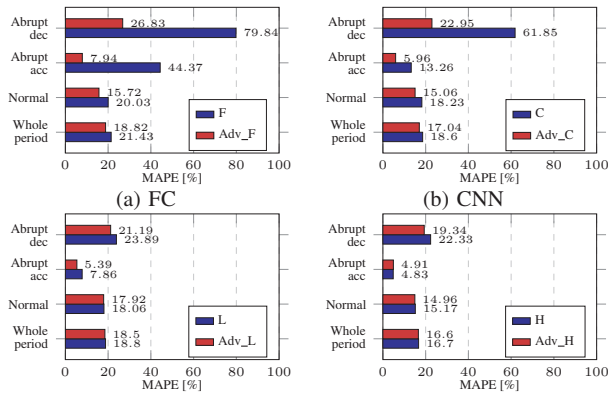
Q1) Effectiveness of adversarial training: To investigate the effectiveness of adversarial training, we compared the performance of F, C, L, and H with Adv_F, Adv_C, Adv_L, and Adv_H which is *applying only adversarial training without the additional data*. For this, we defined the situation of abrupt speed acceleration and deceleration where the present speed s_t changes more than θ compared with the past speed s_{t-1} as follow:

$$\frac{s_{t-1} - s_t}{s_{t-1}} \geq \theta, \quad where \quad \theta = 0.3, \quad (7)$$

$$\frac{s_{t-1} - s_t}{s_{t-1}} \leq \theta, \quad where \quad \theta = -0.3. \quad (8)$$

Here, we set the θ to ± 0.3 since the maximum changes of speed in our dataset were $\pm 30\%$.

Fig 4 shows the effect of adversarial training. The models with adversarial training (Adv_F, Adv_C, Adv_L, and Adv_H) showed more accurate predictions than the original models. In particular, Adv_F shows the MAPE of 18.82 in the whole period, which is lower than F that shows MAPE of 21.43. Moreover, in the cases of abrupt speed acceleration and deceleration, Adv_F shows MAPEs of 7.94 and 26.83, which are much lower than F that shows MAPEs of 44.37 and 79.84, respectively. The other models are less effective than F, however, they also shows the better accuracy of MAPE with adversarial training in the case of abrupt speed change.



(a) FC (b) CNN

(c) LSTM (d) Hybrid model

Fig. 4: Effect of adversarial training.

From these results, we confirm that the distribution learning through adversarial training significantly better captures abrupt changes in traffic data as well as contributed to more accurate predictions.

Q2) Effectiveness of additional data (with F only): We compared the MAPEs by adding (1) adjacent-speed data, (2) non-speed data, and (3) both, to the speed data of the target road. To confirm the direct effectiveness of additional data, we compared the performance of *only predictors without adversarial training*. To avoid the effect caused by different architectures of a predictor, the size of the input to a predictor was fixed to that of (3) for all evaluations; thus for the input of (1) or (2), the rest was filled with 0.

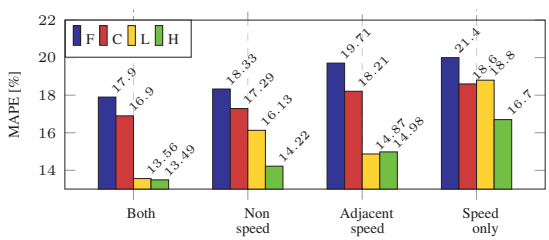


Fig. 5: Effect of additional data

Fig 5 shows the effect of employing additional data. The accuracy of all predictors is improved in the case of using additional data.Regarding F, the MAPE obtained with each of the adjacent-speed data and non-speed data is 19.71 (8% gain) / 18.33 (14% gain), lower than the MAPE of 21.43 obtained without the adjacent-speed data and non-speed data. The MAPE with both the adjacent-speed and non-speed data is the lowest, 18.07 (15% gain). The other predictors, C, L, and H, are also shows similar results to F. This indicates that additional contextual information contributes to improving the accuracy in speed prediction.

Additionally, we compare the effect of three factors in non-speed data (events, weather, and time) to examine the impact of each factor on the performance of APOTS_H. We measured the prediction accuracy by including each factor to the model one by one. In Table II, the time information had the greatest impact on the accuracy (20.12% gain), followed by the weather (3.73% gain), while the event showed little effect.

TABLE II: Performance of non-speed data for APOTS_H.

	S	SE	SW	ST
MAPE	16.60	16.60	15.98	13.26
Gain	–	–	3.73%	20.12%
	SEW	**SET**	**SWT**	**SEWT**
MAPE	16.58	13.04	12.90	12.80
Gain	–	21.44%	22.28%	22.89%

S: Speed of target road, E: Event, W: Weather, T: Time

Q3) Model performance (with APOTS): We verify the accuracy of Prophet, F, L, C, and H under APOTS as well as the effect of additional data on different models. Here, we employ the Prophet as baseline, which is time-series prediction model based on statistics from Facebook. Prophet made predictions using the information about the day before, the day after, and the day of holidays. We set the upper and lower window values as 1 and the scale value as a default value suggested by Prophet. Table III summarizes the results. The gain in the Table III is defined as bellow:

$$Gain = \frac{E_a - E_b}{E_b} \times 100. \tag{9}$$

The gain of the column indicates the accuracy improvement by the predictor with the adversarial training. The gain of the row indicates the accuracy improvement by the predictor employing additional data together with the speed data in training. The gain of the diagonal indicates the accuracy improvement by the predictor employing both additional data and adversarial training. All predictors show the best performance when using *both adversarial training and additional data*.

The accuracy of all models increases when adversarial training was applied. The results of applying adversarial training shows meaningful difference from the result without adversarial (in MAPE, $t(7)=3.04,\ p<0.05$). F w/ Adv. shows the MAPE of 18.82 that improves accuracy the most (12.06% gain compared with F w/o Adv), followed by C w/ Adv. with the MAPE of 17.04 (8.39% gain compared with C w/o Adv). The L and H w/ Adv. show a slight decrease in MAPE (1.60% and 0.60% gain compared with L and H w/o Adv., respectively).

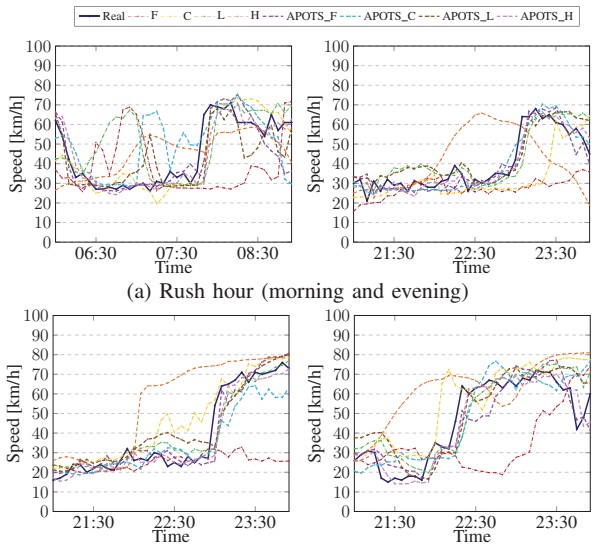
The accuracy of all models increases when additional data (adjacent-speed and non-speed data) are employed. The results of using additional data shows meaningful difference from the result without additional data (in MAPE, $t(7)=9.12,\ p<0.05$). The average of gain is 20.54% over all predictors. In particular, L w/o Adv. and L w/ Adv. using speed and additional data show the MAPEs of 13.50 and 13.40, which the improvement of accuracy is the largest (28.19% and 27.57% gain compared with those using speed data only), respectively.

In overall, APOTS_H (Speed+Add. data & w/ Adv.) shows the best perfomance, 12.80 of MAPE, which is achieving 87.5%, 40%, 32%, and 31% gain over the existing methods, Prophet, F, L, and C (marked * in Table III). Prophet shows the MAPE of 102.42, which is the worst performance among the models. This result may be due to the limitation of prediction method based on statistics, which is unable to capture the

TABLE III: Performance variations of APOTS based on combinations of *adversarial training* and *additional data* according to the predictors. The bold indicates the performance of APOTS using both adversarial training and additional data.

		Prophet	F (FC)			L (LSTM)			C (CNN)			H (Hybrid: L+C)		
			w/o Adv.	w/ Adv.	Gain	w/o Adv.	w/ Adv.	Gain	w/o Adv.	w/ Adv.	Gain	w/o Adv.	w/ Adv.	Gain
MAE	**Speed only**	23.62	6.90	6.02	12.76%	5.95	5.74	3.58%	5.55	5.22	5.87%	4.91	4.75	3.09%
	Speed +Add. data	23.58	5.25	**4.32**	17.65%	4.03	**3.94**	2.41%	5.41	**3.94**	27.21%	3.94	**3.71**	5.78%
	Gain	0.17%	23.88%	28.15%	37.39%	32.23%	31.40%	33.78%	2.52%	24.62%	29.01%	19.66%	21.89%	24.43%
RMSE	**Speed only**	26.70	11.03	9.96	9.72%	10.03	9.91	1.22%	8.06	8.00	0.73%	7.02	6.88	2.11%
	Speed +Add. data	26.67	6.72	**6.01**	10.57%	6.51	**6.04**	7.15%	6.73	**5.93**	12.02%	5.87	**5.68**	3.26%
	Gain	0.11%	39.14%	39.71%	45.51%	35.11%	39.01%	39.78%	16.47%	25.97%	26.52%	16.41%	17.39%	19.08%
MAPE	**Speed only**	102.42*	21.40*	18.82	12.06%	18.80*	18.50	1.60%	18.60*	17.04	8.39%	16.70	16.60	0.60%
	Speed +Add. data	102.61	17.90	**14.50**	18.99%	13.50	**13.40**	0.74%	16.90	**13.90**	17.75%	13.50	**12.80***	5.19%
	Gain	-0.19%	16.36%	22.95%	32.24%	28.19%	27.57%	28.72%	9.14%	18.43%	25.26%	19.16%	22.89%	23.35%

nonlinearity of traffic speed data. In addition, Prophet shows no significant difference in accuracy when holiday information is employed or not. This result is due to our dataset that contains a small number of holidays (only 7 days). From these results, we confirm *the synergy between the adversarial training and additional data* and the importance of using the *contextual information* in speed predictions.



(a) Rush hour (morning and evening)

(b) Rainy day (c) Accident recovery

Fig. 6: Predictions in speed with APOTS on the real traffic situations ($\alpha = 12$).

In addition, Fig 6 shows the result of predictions in four cases (rush hours in the morning and evening in Fig 6a, a rainy day in Fig 6b, and an accident recovery in Fig 6c). APOTS_F, APOTS_C, APOTS_L, and APOTS_H show better predictions than F, C, L, and H in all situations. In other words, the predictors with APOTS_(P) well capture the abrupt changes of real speed immediately, while P does not capture them well. Here, APOTS_(P) indicates the predictor w/ Adv. and using Speed+Add. data, whereas P indicates the without both. From the result, we confirm that *the adversarial training helps capture abrupt changes* in real traffic situations via a deep understanding of true speed distributions.

VI. Conclusions

Limitations and future work: In this paper, although APOTS was reported to outperform single models (F, CNN, LSTM, and hybrid), there is still more room for its improvement. As a future work, we plan (1) to demonstrate the superiority of APOTS through comparative experiments with other basic models (e.g., cGAN [48]) and state-of-the-art models [41], [42], (2) to compare its performance with existing systems (e.g., Route Guidance System [39], Geograph Information System [40]), and (3) to improve the performance of APOTS by utilizing additional data used in other studies [39]–[42] (e.g., traffic amount, traffic demands, traffic inflow and outflow).

In this paper, we proposed APOTS, a novel adversarial-training-based traffic speed prediction model. It (1) learns the distribution of real speeds, (2) uses additional contextual information, and (3) employs CNN and LSTM for reflecting data characteristics. We validated the effectiveness of APOTS through the extensive evaluations with real-world data from Hyundai Motor Company. The results demonstrated that the distribution of real traffic speed data (*adversarial training*), additional information about the road situation (*exploiting additional data*), and the CNN+LSTM model (*refining a predictor*) all contribute to improving the accuracy of the prediction model significantly.

Acknowledgement

This work was supported by (1) Hyundai Motor Company (T001600121010077), (2) Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2020-0-01373, Artificial Intelligence Graduate School Program (Hanyang University)), and (3) the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No. 2018R1A5A7059549).

REFERENCES

[1] M. S. Ahmed and A. R. Cook, *Analysis of Freeway Traffic Time-series Data by using Box-Jenkins Techniques*. No. 722, 1979.

[2] Y. Zhang and Y. Liu, "Traffic forecasting using least squares support vector machines," *Transportmetrica*, vol. 5, pp. 193–213, 2009.

[3] M. T. Asif, 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," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 2, pp. 794–804, 2013.

[4] S. Cheng and F. Lu, "Short-term traffic forecasting: A dynamic st-knn model considering spatial heterogeneity and temporal non-stationarity.," in *Proc. of Int'l Conf.on EDBT*, pp. 133–140, 2018.

[5] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time series," *The Handbook of Brain Theory and Neural Networks*, vol. 3361, p. 1995, 1995.

[6] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. of IEEE Conf. on CVPR*, pp. 1–9, 2015.

[7] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "Lstm: A search space odyssey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, pp. 2222–2232, 2016.

[8] D.-K. Chae, J.-S. Kang, S.-W. Kim, and J. Choi, "Rating augmentation with generative adversarial networks towards accurate collaborative filtering," in *Proc. of The Web Conference*, pp. 2616–2622, 2019.

[9] 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.

[10] Y.-y. Chen, Y. Lv, Z. Li, and F.-Y. Wang, "Long short-term memory model for traffic congestion prediction with online open data," in *Proc. of IEEE Int'l Conf. on ITSC*, pp. 132–137, 2016.

[11] D. Kang, Y. Lv, and Y.-y. Chen, "Short-term traffic flow prediction with lstm recurrent neural network," in *Proc. IEEE Int'l Conf. on ITSC*, pp. 1–6, 2017.

[12] R. Yu, Y. Li, C. Shahabi, U. Demiryurek, and Y. Liu, "Deep learning: A generic approach for extreme condition traffic forecasting," in *Proc. of SIAM Int'l Conf. on SDM*, pp. 777–785, 2017.

[13] Y. Tian, K. Zhang, J. Li, X. Lin, and B. Yang, "Lstm-based traffic flow prediction with missing data," *Neurocomputing*, vol. 318, pp. 297–305, 2018.

[14] K. Zhang, L. Zheng, Z. Liu, and N. Jia, "A deep learning based multitask model for network-wide traffic speed predication," *Neurocomputing*, 2019.

[15] K. Niu, H. Zhang, T. Zhou, C. Cheng, and C. Wang, "A novel spatio-temporal model for city-scale traffic speed prediction," *IEEE Access*, vol. 7, pp. 30050–30057, 2019.

[16] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Graph convolutional recurrent neural network: Data-driven traffic forecasting," *arXiv preprint arXiv:1707.01926*, 2017.

[17] Y. Wu, H. Tan, L. Qin, B. Ran, and Z. Jiang, "A hybrid deep learning based traffic flow prediction method and its understanding," *Transportation Research Part C: Emerging Technologies*, vol. 90, pp. 166–180, 2018.

[18] X. Dai, R. Fu, E. Zhao, Z. Zhang, Y. Lin, F.-Y. Wang, and L. Li, "Deep-trend 2.0: A light-weighted multi-scale traffic prediction model using detrending," *Transportation Research Part C: Emerging Technologies*, vol. 103, pp. 142–157, 2019.

[19] Y. Lv, Y. Duan, W. Kang, Z. Li, and F.-Y. Wang, "Traffic flow prediction with big data: a deep learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, pp. 865–873, 2014.

[20] A. Artikis, M. Weidlich, F. Schnitzler, I. Boutsis, T. Liebig, N. Piatkowski, C. Bockermann, K. Morik, V. Kalogeraki, J. Marecek, *et al.*, "Heterogeneous stream processing and crowdsourcing for urban traffic management.," in *Proc. of Int'l Conf.on EDBT*, pp. 712–723, 2014.

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

[22] J. Wang, Q. Gu, J. Wu, G. Liu, and Z. Xiong, "Traffic speed prediction and congestion source exploration: A deep learning method," in *Proc. of IEEE 16th Int'l Conf. on DM*, pp. 499–508, 2016.

[23] X. Cao, Y. Zhong, Y. Zhou, J. Wang, C. Zhu, and W. Zhang, "Interactive temporal recurrent convolution network for traffic prediction in data centers," *IEEE Access*, vol. 6, pp. 5276–5289, 2017.

[24] Z. Lv, J. Xu, K. Zheng, H. Yin, P. Zhao, and X. Zhou, "Lc-rnn: A deep learning model for traffic speed prediction.," in *Proc. of IJCAI*, pp. 3470–3476, 2018.

[25] A. Dillen, G. Buroni, Y.-A. Le Borgne, K. Determe, and G. Bontempi, "Mobi-aid: A big data platform for real-time analysis of on board unit data.," in *Proc. of Int'l Conf.on EDBT*, 2020.

[26] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. of Conf. NIPS*, pp. 2672–2680, 2014.

[27] Y. Zheng, Y. Liu, J. Yuan, and X. Xie, "Urban computing with taxicabs," in *Proceedings of International Conference on Ubiquitous Computing*, 2011.

[28] J. Zhang, F.-Y. Wang, K. Wang, W.-H. Lin, X. Xu, and C. Chen, "Data-driven intelligent transportation systems: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12.

[29] H. Yuan and G. Li, "Distributed in-memory trajectory similarity search and join on road network," in *Proceedings of IEEE International Conference on Data Engineering*, pp. 1262–1273, IEEE, 2019.

[30] D. A. Tedjopurnomo, Z. Bao, B. Zheng, F. Choudhury, and A. Qin, "A survey on modern deep neural network for traffic prediction: Trends, methods and challenges," *IEEE Transactions on Knowledge and Data Engineering*, 2020.

[31] 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, 2003.

[32] A. Pascale and M. Nicoli, "Adaptive bayesian network for traffic flow prediction," in *Proceedings of the IEEE Statistical Signal Processing Workshop (SSP)*, 2011.

[33] N. Vandaele, T. Van Woensel, and A. Verbruggen, "A queueing based traffic flow model," *Transportation Research Part D: Transport and Environment*, vol. 5, 2000.

[34] L. Muñoz, X. Sun, R. Horowitz, and L. Alvarez, "Traffic density estimation with the cell transmission model," in *Proceedings of the American Control Conference*, 2003.

[35] 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, 2009.

[36] J. Lemieux and Y. Ma, "Vehicle speed prediction using deep learning," in *Proceedings of the IEEE Vehicle Power and Propulsion Conference (VPPC)*, 2015.

[37] D. Park and L. R. Rilett, "Forecasting freeway link travel times with a multilayer feedforward neural network," *Computer-Aided Civil and Infrastructure Engineering*, vol. 14, 1999.

[38] W. Huang, G. Song, H. Hong, and K. Xie, "Deep architecture for traffic flow prediction: deep belief networks with multitask learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, 2014.

[39] Z. Liang and Y. Wakahara, "Real-time urban traffic amount prediction models for dynamic route guidance systems," *EURASIP Journal on Wireless Communications and Networking*, 2014.

[40] W.-p. Fei, G.-h. Song, F. Zhang, Y. Gao, and L. Yu, "Practical approach to determining traffic congestion propagation boundary due to traffic incidents," *Journal of Central South University*, 2017.

[41] H. Yuan, G. Li, Z. Bao, and L. Feng, "Effective travel time estimation: When historical trajectories over road networks matter," in *Proceedings of ACM SIGMOD International Conference on Management of Data*, pp. 2135–2149, 2020.

[42] H. Yuan, G. Li, Z. Bao, and L. Feng, "An effective joint prediction model for travel demands and traffic flows," in *Proceedings of IEEE International Conference on Data Engineering*, pp. 348–359, 2021.

[43] T. Kneib, "Beyond mean regression," *Statistical Modelling*, vol. 13, pp. 275–303, 2013.

[44] D.-K. Chae, J.-S. Kang, S.-W. Kim, and J.-T. Lee, "Cfgan: A generic collaborative filtering framework based on generative adversarial networks," in *Proc. of ACM Int'l Conf. on CIKM*, pp. 137–146, 2018.

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

[46] C.-S. J. Chu, "Time series segmentation: A sliding window approach," *Information Sciences*, vol. 85, no. 1-3, pp. 147–173, 1995.

[47] X. Ma, Z. Dai, Z. He, J. Ma, Y. Wang, and Y. Wang, "Learning traffic as images: A deep convolutional neural network for large-scale transportation network speed prediction," *Sensors*, p. 818, 2017.

[48] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.