Learning Congestion Propagation Behaviors for Traffic Prediction

Sun Yidan, He Peilan, Jiang Guiyuan and Siew-Kei Lam

***Abstract*— Traffic prediction is a challenging task as the traffic flow is influenced by many seasonal, stochastic, and structural factors. In addition, the spatial and temporal distribution of traffic flow can induce direct and indirect congestion propagation patterns. While existing works have attempted to model spatial-temporal graphs to capture the spatial correlations and temporal dependencies, they fail to consider congestion propagation behavior among road segments. In this paper, we propose a novel traffic prediction model that takes into account the congestion propagation tendencies to improve prediction accuracy. A novel diffusion graph convolution network model is developed to capture the spatial traffic correlations while considering the congestion propagation behavior. Our model also jointly learns the importance of seasonal traffic speed correlations, road contextual information (structural information), and stochastic factors (external factors) through an attention layer. Experimental results on real-world data-set demonstrate the superiority of our method over state-of-the-art traffic prediction techniques, and confirm the significance of congestion propagation behavior in traffic prediction.**

I. Introduction

Traffic prediction has become an essential enabler for intelligent transportation systems due to the advancement in traffic data collection. Short-term traffic prediction provides valuable information for many applications such as route planning [1] and public transportation management. However, achieving accurate prediction is challenging as the traffic flow is influence by seasonal factors (e.g., time of congestion, impact of weekends), stochastic factors (e.g., impact of weather and special events), and structural factors (e.g., Point of Interests (POIs), road characteristics).

Many works employ spatial and temporal correlations to predict future traffic states of a road based on seasonal factors (i.e., its historical traffic flow and the neighboring roads' traffic state [2]). These works often rely on graph convolution techniques [2] to capture spatial correlations on graph-structured road networks, and recurrent neural networks [3] to model temporal correlations. Existing works also take into account stochastic and structural factors for traffic prediction [4]. POIs, e.g., business buildings and schools, create periodic traffic demands. Road characteristics, e.g., road types (primary/highway) with different throughput capacities, also directly impact traffic speed. In addition, adverse weather conditions (e.g., heavy rain/snow) affect travel demands by influencing travel decisions, e.g., travel mode and departure time. It is noteworthy that the significance of these factors evolves with time, and they also impact each other. For example, POIs such as shopping malls and tourism areas attract higher traffic demands during holidays with opposing trends on workdays. Therefore, a dependable traffic prediction model must effectively incorporate the heterogeneous factors and capture their inter-dependencies.

Besides the factors mentioned above, traffic congestion also exhibits specific propagation behavior due to the spatial and temporal distribution of the traffic flow [5]. Congestion emerges on road segments where the traffic flow exceeds the road capacity, and the congested road segments are likely to affect its neighboring road segments. If the traffic flow continues to increase (e.g., during daily peak hours), the congestion propagation may exhibit a domino effect where the congestions propagate to other spatially connected road segments. This will last until the travel demand diminishes, and a reverse propagation effect can be observed. While it is evident that congestion propagation can negatively impact the traffic states among a cluster of road segments, it is often neglected in state-of-the-art traffic prediction works. In this paper, we propose a Congestion Propagation aware Traffic Prediction (`CPTP`) model, which jointly considers spatial-temporal correlations of seasonal factors, structural factors, and stochastic factors, while taking into account congestion propagation tendencies among road segments. Our contributions are summarized as follows:

- We develop a deep learning module to infer traffic states based on heterogeneous feature components: diverse seasonal temporal dependencies, structural factors (POIs and road characteristics), and stochastic factors (weather conditions and holidays). The evolving importance of each feature component is learned using an attention fusion layer.
- We improve traffic prediction accuracy by considering congestion propagation behavior while learning spatial correlations. Propagation probability matrices are used to capture propagation behavior in both temporal and spatial dimensions. Then, a novel diffusion graph convolution network (DGCN) is developed to learn spatial correlations taking into account congestion propagation.
- We evaluate the performance of our proposed method with real-world dataset of Singapore. The results show that our method significantly outperforms state-of-the-art methods. They also confirm the significance of congestion propagation behavior in traffic prediction.

This research project is supported in part by the National Research Foundation Singapore under its Campus for Research Excellence and Technological Enterprise (CREATE) programme with the Technical University of Munich at TUMCREATE.

Y. Sun, P. He, G. Jiang, S.K. Lam are with the School of Computer Science and Engineering, Nanyang Technological University, 639798, Singapore. (e-mail: {ysun014,phe002}@e.ntu.edu.sg, {gyjiang,assklam}@ntu.edu.sg.)

II. RELATED WORK

Deep learning-based traffic prediction models have mostly attempted to capture spatial and temporal traffic correlations [6] to achieve better prediction. Typically, graph convolution networks are used to model spatial dependencies on topological traffic network [2][7], while recurrent networks have demonstrated their effectiveness in modeling temporal dependencies [3][8][9]. As different traffic impact factors exhibit varying importance on the traffic state, attention mechanism has been introduced to determine the significance among the diverse spatiotemporal feature components [4][10][11]. However, unlike our work, most of the existing techniques fail to consider the impact of congestion propagation behavior on the traffic states for traffic prediction. The work that is most related to ours is [12], which builds a congestion diffusion model for traffic prediction. They apply a path distance-based weight matrix to implicitly incorporate the congestion diffusion process into the traffic prediction model. However, using a stationary weight matrix limits the ability to express the evolving propagation patterns. Hence, their method is unable to effectively exploit the congestion propagation behavior for traffic prediction.

The majority of the work in **congestion propagation** (or evolution) relies on statistical analyses using historical data for regularity mining [13]. These works employ tree [5] or graph [14] structures to construct congestion propagation relationships, wherein frequent congestion propagation patterns are mined for various applications. The related works to ours include [5], which used frequent sub-trees to build Dynamic Bayesian Network (DBN) for congestion prediction. [15] also mined sub-tree patterns to improve mining speed and storage efficiency. The work in [16] discovered frequent propagation patterns and used Markov chains to predict congestion propagation probabilities. [17] exploits tree structures to detect historical congestion propagation rather than use them for prediction. [14] proposed graph-based propagation patterns and predict patterns in the near future. The works above typically relied on tree and graph patterns, which can only be used by limited prediction models, such as probabilistic graph models (DBN, Markov chains). Our work employs deep learning-based prediction models that can incorporate more complex feature components to improve prediction accuracy.

III. PROPOSED APPROACH

In this section, we introduce our **C**ongestion **P**ropagation aware **T**raffic **P**rediction (`CPTP`) model, which is capable of effectively exploiting the heterogeneous feature components while taking into account dynamically changing congestion propagation behavior. The model consists of eight modules as shown in Fig. 1:

1) Feature extraction of temporal dependencies from traffic speed time-series.
2) Feature extraction of structural factors.
3) Feature extraction of external factors.
4) Learn the importance of those above three latent representations using an attention fusion layer.

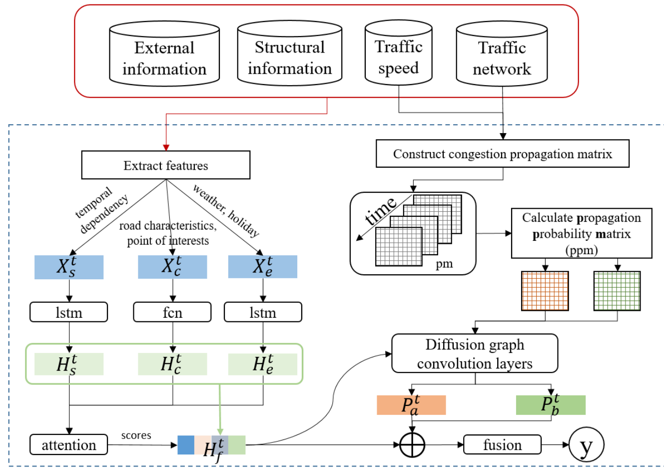

Fig. 1. Proposed framework.

5) Detect congestion and construct propagation matrices.
6) Calculate propagation probability matrices.
7) Extract propagation trends through diffusion graph convolution layers and incorporate them into the final representation
8) Feed final representation into a fusion layer to predict traffic speed.

In the following sub-sections, we will describe these modules in detail.

A. Problem Definition

We denote the traffic network as a graph $G = \langle V, E \rangle$, where $V = \{r_i\}$ is a set of nodes, and each node r_i represents a road segment, $E = \{(i,j)\}$ is a set of edges, (i,j) indicate that r_i and r_j are spatially connected (i.e. share an intersection). $S \in \mathbb{R}^{N \times T}$ is speed matrix (km/h) over the entire period of study, T is the total number of time intervals, and S_i^t is speed value of r_i at time step t. Our model also relies on structural information, including road characteristics, POIs, and stochastic factors such as weather and holidays, to capture traffic situations. Given a graph G and the aforementioned traffic speed information for the past P time steps, our problem is to learn a model Θ which can predict the traffic speed matrix for the entire network in the next H time steps, $Y \in \mathbb{R}^{N \times H}$.

B. Temporal Dependencies

Temporal dependencies in traffic speed have been utilized in many existing works [11]. Our method utilizes three temporal dependencies between the historical and future speed observations for each road segment: recent, daily, and weekly dependencies. As shown in Fig. 1 (left), for each time step t, we calculate matrix $X_S^t \in \mathbb{R}^{N \times d_s}$ from traffic speed. 'Recent' feature vector of r_i at time step t is calculated as $\mathbf{sr}_i^t = \langle S_i^{t-1}, ..., S_i^{t-P} \rangle$. The daily average value is calculated as $sd_i^t = \frac{1}{nd} \sum_{l=1}^{nd} S_i^{t-Td*l}$, where nd is the number of previous days we used to calculate the average speed for corresponding t, and Td is the total number of time intervals in one day ($Td = 1440/5 = 288$ if interval size is 5 mins). Similarly, we calculate the weekly average speed as $sw_i^t = \frac{1}{nw}\sum_{l=1}^{nw} S_i^{t-Tw*l}$, where nw is the number of weeks we considered and Tw is the total number of time intervals in one week ($Td = 1440 * 7/5 = 2016$). Finally, temporal dependency vector for r_i at time step t is represented as $X_{s,i}^t = \mathbf{sr}_i^t \oplus \langle sd_i^t \rangle \oplus \langle sw_i^t \rangle$, where $\oplus$ is concatenate operation. $X_{s,i}^t$ is also the $i-th$ row in matrix $X_s^t \in \mathbb{R}^{N\times d_s}$ ($d_s = P+2$). To learn the temporal dependency, for each time step, we feed X_s^t into a 3-layer LSTM to generate hidden output $H_s^t \in \mathbb{R}^{N\times d_h}$, The calculations are listed as follows:

$$\begin{aligned}
\mathbf{i}_t &= \sigma(\mathbf{W}_{ii}\mathbf{x}_t + \mathbf{b}_{ii} + \mathbf{W}_{hi}\mathbf{h}_{t-1} + \mathbf{b}_{hi}),\\
\mathbf{f}_t &= \sigma(\mathbf{W}_{if}\mathbf{x}_t + \mathbf{b}_{if} + \mathbf{W}_{hf}\mathbf{h}_{t-1} + \mathbf{b}_{hf}),\\
\mathbf{g}_t &= \sigma(\mathbf{W}_{ig}\mathbf{x}_t + \mathbf{b}_{ig} + \mathbf{W}_{hg}\mathbf{h}_{t-1} + \mathbf{b}_{hg}),\\
\mathbf{o}_t &= \sigma(\mathbf{W}_{io}\mathbf{x}_t + \mathbf{b}_{io} + \mathbf{W}_{ho}\mathbf{h}_{t-1} + \mathbf{b}_{ho}),\\
\mathbf{C}_t &= \mathbf{f}_t \circ \mathbf{C}_{t-1} + \mathbf{i}_t \circ \mathbf{g}_t,\ \mathbf{h}_t = \mathbf{o}_t \circ \tanh(\mathbf{C}_t).
\end{aligned} \tag{1}$$

We omit the superscript for general case, where t denotes the t-th time step, $\mathbf{h}_t$, $\mathbf{x}_t$, $\mathbf{h}_{t-1}$ and $\mathbf{C}_t$, are the hidden state at time t, the input vector at time t, the hidden state at time $t-1$ or initial hidden state at time 0, and the cell state at time t, respectively. $\mathbf{i}_t$, $\mathbf{f}_t$, $\mathbf{g}_t$, $\mathbf{o}_t$ refer to the input, forget, cell, and output gates, respectively. σ is a sigmoid function, $\circ$ is Hadamard product, d_h is output dimension of LSTM, and the hidden dimension of LSTM layers is 64. Finally, the output hidden states in our LSTM is H_s^t.

C. Structural Information

Our prediction model takes into consideration road characteristics and surrounding POIs [18]. Road characteristics include road type (e.g., highway, primary way, etc. (one-hot encoding is applied)), length, number of lanes, number of bus stops, and number of traffic signals. The POIs are categorized into 14 types: religion, finance and insurance, business building, hotel, residence, education and culture, food, government, scenic spot, medical care, subway station, sports, entertainment, and living service. For each r_i, we maintain a vector of length 14. We set its midpoint as the center of a circle with diameter $l = 200$ meters. If a POI is located within the circle, the corresponding category will add 1. Finally, the structural feature vector is a concatenation of the aforementioned vectors, which is represented as $X_c^t \in \mathbb{R}^{N\times D_c}$. X_c^t is fed into a 2-layer FCN, which generate $H_c^t \in \mathbb{R}^{N\times d_h}$. The FCN has a hidden dimension of 32 and sigmoid function as activation function.

D. External Factors

Traffic is affected by many external factors such as weather conditions and holidays. Adverse weathers affect travel decisions such as trip mode and departure time. Holidays can also lead to traffic congestion around specific areas (like temples or churches). Incorporating these factors can enhance the capability of our model to distinguish such traffic situations [19]. We construct $X_e^t \in \mathbb{R}^{N\times d_e}$ to encode external factors, which include one-hot encoded weather conditions, concatenate it with a binary variable to indicate whether the day is a holiday or not. Similarly, external factors exhibit temporal dependencies, so we also feed them into a 3-layer lstm as shown in Fig. 1 to obtain latent matrix $H_e^t \in \mathbb{R}^{N\times d_h}$.

E. Attention Fusion Layer

From previous steps, we obtain H_s^t, H_c^t and $H_e^t \in \mathbb{R}^{N\times d_h}$. To enable the model to learn powerful representation for each road, we utilize an attention layer to learn the dynamic importance of each component as follows:

$$\alpha_j^t = FCN(H_j^t),\ j \in \{s, c, e\} \tag{2}$$

$$a_j^t = \frac{\exp \alpha_j}{\sum_{j\in\{s,c,e\}} \alpha_j},\ H_f^t = \sum_{j\in\{s,c,e\}} a_j^t \cdot H_j^t. \tag{3}$$

where FCN is a two-layer fully connected network as discussed in Section III-C, with output dimension $N \times 1$. H_j^t indicates latent representation in $\{H_s^t, H_c^t, H_e^t\}$. $\alpha_j^t \in \mathbb{R}^{N\times 1}$, is a value for each r_i for the corresponding j-th latent representation at time t. a_j^t is calculated through a softmax function to obtain attention scores for all N roads at current time step t for corresponding j component ($\{H_s^t, H_c^t, H_e^t\}$). Finally, each latent representation is multiplied with the scores and undergo a weighted summation to get a fusion representation for all roads $H_f^t \in \mathbb{R}^{N\times d_h}$ as shown in Figure 1 (bottom left). So far, we have incorporated several important feature components into our fused latent representation for all roads.

F. Construct Congestion Propagation Matrix

In the following, we introduce the formulation of congestion propagation. Firstly, we determine that a road segment r_i is congested at time t if S_i^t is lower than a threshold speed th_i, where th_i is specific to each r_i. Specifically, in r_i's historical data-set, there exist 75% historical traffic speed observations which are higher than th_i. Based on speed matrix $S \in \mathbb{R}^{N\times T}$ and corresponding thresholds, a congestion matrix with same dimension is constructed, $C \in \{0,1\}^{N\times T}$, where $C_i^t = 1$ if $S_i^t < th_i$, otherwise, $C_i^t = 0$. Then, we extract congestion propagation as paths in the following way: at time t, we detect a set of newly emerging congested roads $r_i \in V'$ where $C_i^t = 1$ while $C_i^{t-1} = 0$. For each $r_i \in V'$, a path $pa = \langle r_j, .., r_i \rangle$ satisfying the following three conditions is detected as a congestion propagation path: **(1)** each two adjacent nodes in pa are spatially connected (physically reachable); **(2)** the initial node of path $C_j^{t-1} = 1$ (r_j could be a source of propagation); and **(3)** except for r_j, all the remaining nodes in pa belong to V' (all the remaining roads propagate congestion within the same time step).

We present congestion propagation at each time interval as paths rather than road pairs. We consider the situations within the same time interval; sometimes mild congestion will propagate to a very local surrounding area; while severe congestion could propagate farther. Based on obtained paths $\{pa\}$, at t, we construct congestion **P**ropagation **M**atrix $PM^t \in \{0,1\}^{N\times N}$. For any pa, every two adjacent road segments $r_i, r_j \in pa$, we have $PM^t(i,j) = 1$, where $PM^t(i,j)$ represents the element in i-th row and the j-th column, otherwise, $PM^t(i,j) = 0$.

G. Calculate Congestion Propagation Probability Matrix

Congestion propagation behavior exhibits different patterns on workdays and weekends. For instance, on workdays, heavy traffic demands occur in residual areas to business areas during morning peak hours, while on weekends, more people may congregate at entertainment areas in the afternoon. Thus, we try to incorporate these knowledge into our model by calculating congestion **P**ropagation **P**robability **M**atrix (PPM). After constructing congestion propagation matrices over the entire studied period, we obtain a 3-way tensor $PM \in \{0,1\}^{T\times N\times N}$, from which we calculate PPM. We extracted PM^t for two months (August-September, 2018) from historical data (18 weekends and 43 workdays). For each t, we obtained two PPMs: PPM_{wo}^{tt} and PPM_{we}^{tt}, where $tt \in \{1,2,...,Td\}$, Td is the number of time intervals in one day, PPM_{wo}^{tt} is average matrix of PMs at corresponding tt for all 43 workdays. Similarly, PPM_{we}^{tt} is the average matrix for 16 weekends.

H. Diffusion Graph Convolution Layer

We utilize diffusion graph convolution layer [20] to incorporate congestion propagation behaviors into our traffic prediction model based on the obtained probability matrices $\{PPM_{wo}^t, PPM_{we}^t\}$. [20] has demonstrated the benefits of capturing signal diffusion process on graph structure. In general, they utilize adjacent matrix $A \in \mathbb{R}^{N\times N}$ as follows:

$$Z = \sum_{k=0}^{K} A^k X W_k \tag{4}$$

where $X \in \mathbb{R}^{N\times f}$ is the feature matrix for the entire road network, $W_k \in \mathbb{R}^{f\times m}$ is the weight matrix of GCN layer to be learned, and k is the power of the matrix. However, this layer is not suitable for our problem because: 1) congestion propagation cannot be presented as a fixed matrix A due to the dynamic patterns; 2) congestion propagations are associated with directions, e.g., the probabilities of a road segment for being a source and target of propagations are different. To incorporate these two properties into our model, we designed a new diffusion convolution layer as follows:

$$P_a^t = \sum_{k=0}^{K} {PPM_q}^k X^t W_{k,p},\ P_b^t = \sum_{k=0}^{K} {(PPM_q)'}^k X^t W_{k,q}, \quad P_f^t = P_a^t \oplus P_b^t. \tag{5}$$

$$PPM_q = \begin{cases} ppm_{wo}^t, & if\ t\ is\ on\ workdays. \\ ppm_{we}^t, & if\ t\ is\ on\ weekends. \end{cases} \tag{6}$$

where X^t represents H_f^t, $W_{k,p}$ and $W_{k,q}$ are weight matrices, and we changed A to propagation probability matrices, i.e., PPM_q^t is calculated in Section III-G as Eq. 6, whose value depends on the day of week and time of day of the current time step. In addition, because the i-th row represents the probability that r_i propagate congestion, while i-th column indicates that r_i is the target of congestion propagation, we used PPM_q to get $P_a^t \in \mathbb{R}^{N\times d_{h2}}$, and the transportation of PPM_q, i.e. $(PPM_q)'$, to obtain $P_b^t \in \mathbb{R}^{N\times d_{h2}}$. We concatenate these two terms together and obtain $P_f^t \in \mathbb{R}^{N\times(2d_{h2})}$ to distinguish directed and dynamic congestion propagation patterns.

I. Fusion Layer and Loss Function

In the final step, we take $H_f^t \in \mathbb{R}^{N\times d_h}$ and $P_f^t \in \mathbb{R}^{N\times 2d_{h2}}$ as input matrices, concatenate them into a prediction layer, i.e., a 3-layer FCN to predict the speed matrix $Y \in \mathbb{R}^{N\times H}$. The loss function is as follows:

$$Loss = \sum_{t=P}^{T} (\|Y^t - \hat{Y}^t\|^2 + \lambda \|\frac{Y^t - \hat{Y}^t}{\hat{Y}^t}\|^2) \tag{7}$$

where t represents all the time steps from P (we used previous P steps to predict) to T, and $\hat{Y}^t$ is the ground-truth speed matrix for time step t, λ is a hyperparameter to control the weight of the second loss term, which we set as 0.1.

IV. EXPERIMENT AND RESULTS

A. Datasets

Road Network: The traffic network in our experiments is obtained from OpenStreetMap[1]. A rectangle area in Downtown area of Singapore (Southwest: 1.2718, 103.8002; Northeast: 1.3323, 103.8653) is selected, and 500 road segments in this area are tested.

Traffic Speed Data: The traffic speed matrix S is calculated based on historical bus trajectories derived from bus arrival data[2]. In this paper, we apply time interval for: 5 minutes, 10 minutes and 15 minutes, to test the model's performance for diverse time granularity. Bus traffic data are from Aug. 01 to Nov. 30, 2018. We used 100 days for training set and remaining 22 days as testing set. The horizon of prediction $H = 1$.

Weather data: Hourly-grained weather data are collected during the same time period of the bus speed data[3].

B. Baselines and Evaluation Metrics

The following baselines are chosen for comparison. 1) `HisAvg` [21] uses historical traffic speed to calculate average speed time series as prediction results. 2) `LSTM` [22], a RNN with long-term and short-term memory which has been extensively used in time series prediction problems. 3) `GCNLSTM` [8] combines graph convolution with LSTM to capture both spatial and temporal information for prediction. 4) `DKFN` [9], a novel deep Kalman Filtering Network which captures self and neighbor dependencies as well as bias and noises among traffic data. 5) `BTSP` [4] identifies important intrinsic and extrinsic features for bus travel speed and predicts speed using an attribute-driven attention network. 6) `GraphWave` [2] proposed a graph WaveNet with self-adjacency matrix to capture spatial and temporal dependencies for traffic prediction. 7) `DCRNN-Path` [23] proposed a path-based deep convolution RNN to predict traffic speed. 8) `CPTP-woCP` is an ablation study baseline

[1] https://www.openstreetmap.org/export

[2] https://www.mytransport.sg/content/mytransport/home/dataMall.html

[3] https://www.timeanddate.com/weather/singapore/singapore

by removing the congestion propagation aware component. We use the following performance metrics: Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Root Mean Square Error (RMSE).

C. Experimental Setup

Our experiments are implemented in PyTorch framework on Intel(R) Xeon(R) CPU E5-1650 v2 @ 3.50GHz with 32G RAM. All the hyperparameters are tuned by grid search. We set dimension d_h as 64, and set diffusion step of GCN as $K = 3$, the dimension of P_b^t and P_a^t (i.e.d_{h2}) are all 32. We train our model using Adam optimizer with a learning rate of 0.0001. The dropout of diffusion GCN is 0.5. We normalized features into range $[0, 1]$ and exclude all missing values. For baselines, all the parameters are tuned to fit the dataset.

D. Results and Analysis

This section shows the overall performance of all methods for various time intervals (5min/10min/15min). As shown in Table I, `CPTP` outperforms all the baselines on all three metrics, regardless of the time interval size. `CPTP` outperforms the traditional historical average method by a large margin as `HisAvg` only relies on historical speed. `LSTM` and `GCLSTM` use RNN to incorporate short-and long-term dependencies inherently within speed time series, `GCLSTM` improves the prediction by using GCN to capture spatial correlations. They perform better than `HisAvg`, but still inferior to `CPTP`. This is because `CPTP` explicitly incorporates diverse temporal dependencies in the features and takes into consideration surrounding information. `DKFN`, `GraphWave`, `BTSP` and `DCRNN-Path` have comparable performance and none of them can consistently outperform others. It can be observed that they still under-perform compared to `CPTP`. `GraphWave` utilizes graph and dilated casual convolution techniques to uncover unseen spatial and temporal dependencies on graph structure. It achieves good performance in highway networks that exhibit stable traffic patterns; however, it failed in our dataset because the downtown area has more complex traffic patterns that frequently fluctuate due to noise. `DKFN` performs slightly better than `GraphWave` as it uses the dependency observations as noise measurement rather than treating them as fully reliable. However, it does not incorporate other features such as structural and stochastic factors. `BTSP` considers many features but employs `structure2vec` and clustering techniques to incorporate spatial correlations, which are pre-calculated rather than learned. As such, it shows lower prediction performance as it is unable to capture evolving spatial correlations. `DCRNN-Path` uses path-distance weighted adjacency matrix to incorporate propagation patterns. Path distance is stationary, but propagation changes over time. It also neglects several impact factors that are considered in our model. `CPTP-woCP` is an ablation study that demonstrates the effectiveness of congestion propagation aware diffusion GCN. In the absence of this component, the performance declines notably. This confirms that congestion propagation behavior plays a vital role in traffic prediction.

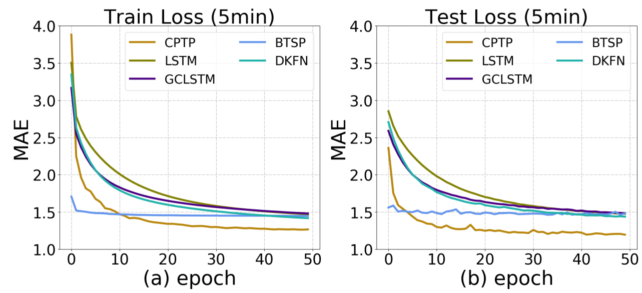

Fig. 2. Train and test loss.

Figure 2 shows the train and test loss of our method and selected baselines (which have comparable training/testing time with ours). It can be observed that our method does not experience over-fitting, and the model loss stabilizes after nearly 20 epochs. Figure 3 are case studies for two specific road segments, where `rs I` locate close to a scenic area. Therefore it shows a declining speed trend around 10:00 am on Sunday in (c) due to visitors, while on Monday it exhibits free to flow around 18:00 in (a), as the scenic area is rarely visited during this time. `rs II` is a segment of the busy Orchard Road. It can be observed that the speed fluctuates strongly compared to `rs I`, and its periodic patterns are not as evident in (b) and (d). This is because there are always many visitors, i.e., more local residents on Sunday and more overseas tourists on Monday. In both situations, our method achieves good prediction performance, as shown in the figure. The training efficiency of our model is also evaluated for a dataset with 13160 samples. The training time is 36.78s for each epoch, and the testing time is 12.56s for each epoch.

V. CONCLUSIONS

We propose a traffic speed prediction model that incorporates congestion propagation behavior while jointly considering correlations among other commonly used impact factors. A deep learning-based module is developed to infer traffic states based on heterogeneous feature components extracted from seasonal, structural, and stochastic factors. A novel diffusion graph convolution network method is employed to incorporate congestion propagation behavior while simultaneously learning spatial correlations. The experiment results on Singapore roadway and traffic datasets demonstrated the superiority of our model and the significance of using congestion propagation behavior for traffic prediction.

REFERENCES

[1] G. Jiang, S.-K. Lam, F. Ning, P. He, and J. Xie, "Peak-hour vehicle routing for first-mile transportation: Problem formulation and algorithms," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 8, pp. 3308–3321, 2019.

[2] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph wavenet for deep spatial-temporal graph modeling," *Inter. Joint Con. on Artif. Intell.-Pac. Rime*, 2020.

TABLE I

COMPARISON OF OVERALL PERFORMANCE OF ALL METHODS, IN TERMS OF MAE, MAPE AND RMSE.

	5 min			10 min			15 min		
	MAE	MAPE	RMSE	MAE	MAPE	RMSE	MAE	MAPE	RMSE
HisAvg	3.408	22.534	4.948	2.702	15.713	4.137	2.375	13.045	3.776
LSTM	1.474	11.797	2.25	1.464	11.302	2.13	1.35	10.306	1.983
GCNLSTM	1.482	11.871	2.258	1.42	10.905	2.078	1.305	9.955	1.935
DKFN	1.433	11.396	2.177	1.395	10.742	2.044	1.294	9.880	1.913
BTSP	1.479	10.261	2.235	1.561	10.58	2.066	1.236	8.801	2.34
GraphWave	2.245	11.614	4.255	1.495	10.494	2.523	1.331	8.748	2.577
DCRNN-Path	2.912	15.341	3.766	2.527	13.822	3.29	2.011	12.87	3.019
CPTP-woCP(ours)	1.233	8.75	1.829	1.195	**7.774**	1.728	1.095	7.016	1.651
CPTP(ours)	**1.194**	**8.463**	**1.715**	**1.16**	7.821	**1.697**	**1.062**	**6.888**	**1.621**

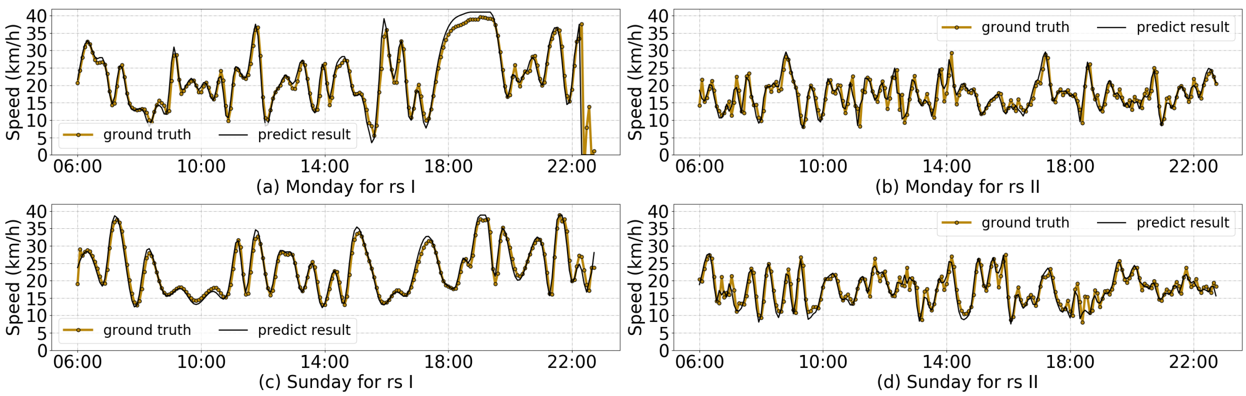

Fig. 3. Case study for two selected roads: `rs I` is a segment of Holland Road, which is close to a scenic area (i.e., Botanic Garden); `rs II` is a segment of Orchard Road, which is the most visited street in Singapore for shopping, time interval is 5 minutes.

[3] Z. S. Jian, L. Shu, L. Y. Long, X. J. Gang, and W. Y. Bing, "Urban traffic flow forecasting based on memory time-series network," in *IEEE Inter. Conf. on Intell. Transp. Syst. (ITSC)*, 2020, pp. 1–6.

[4] Y. Sun, G. Jiang, S.-K. Lam, S. Chen, and P. He, "Bus travel speed prediction using attention network of heterogeneous correlation features," in *Proc. 2019 SIAM Int. Conf. Data Min.* SIAM, 2019, pp. 73–81.

[5] N. Hoang, L. Wei, and C. Fang, "Discovering congestion propagation patterns in spatio-temporal traffic data," *IEEE Trans. Big Data*, vol. 3, no. 2, pp. 169–180, 2016.

[6] P. Z. Yi, L. Y. Xuan, W. W. Feng, Y. Yong, Z. Yu, and Z. J. Bo, "Urban traffic prediction from spatio-temporal data using deep meta learning," in *ACM Inter. Conf. on Knowl. Discov. & Data Min. (SIGKDD)*, 2019, pp. 1720–1730.

[7] Y. Bing, Y. H. Teng, and Z. Z. Xing, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," *Inter. Joint Con. on Artif. Intell.-Pac. Rime (IJCAI)*, 2018.

[8] C. Z. Yong, H. Kristian, K. R. Min, and W. Y. Hai, "Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 11, pp. 4883–4894, 2019.

[9] C. F., Z. Q. Chen, B. Subhodip, L. Shuo, R. Naren, and L. Chang-Tien, "Graph convolutional networks with kalman filtering for traffic prediction," in *Int. Conf. on Adv. in Geogr. Inf. Syst. (IJCAI)*, 2020, pp. 135–138.

[10] S. Q. Yu, M. R. Bo, H. J. Ming, N. H. Yi, and G. M. Yang, "Graph attention convolutional network: Spatiotemporal modeling for urban traffic prediction," in *IEEE Inter. Conf. on Intell. Transp. Syst. (ITSC)*, 2020, pp. 1–6.

[11] G. S. Nan, L. Y. Fang, F. Ning, S. Chao, and W. H. Yu, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," in *AAAI Conf. on Artif. Intell. (AAAI)*, vol. 33, 2019, pp. 922–929.

[12] S.-S. Kim, M. Chung, and Y.-K. Kim, "Urban traffic prediction using congestion diffusion model," in *IEEE Inter. Conf. on Consum. Electron.-Asia (ICCE-Asia)*. IEEE, 2020, pp. 1–4.

[13] M. C. Xi, Z. J. Biao, X. X. C. Daniel, and X. Jin, "Evolution regularity mining and gating control method of urban recurrent traffic congestion: a literature review," *J. of Adv. Transp.*, vol. 2020, 2020.

[14] X. H. Yi, V. Amin, Z. Xun, L. Y. Hua, and L. Jun, "Predicting traffic congestion propagation patterns: A propagation graph approach," in *Int. Conf. Adv. Geogr. Inf. Syst. (SIGSPATIAL)*, 2018, pp. 60–69.

[15] I. Ryo, M. Akihisa, and S. Masatoshi, "Mining spatio-temporal patterns of congested traffic in urban areas from traffic sensor data," in *IEEE Inter. Conf. on Intell. Transp. Syst. (ITSC)*, 2016, pp. 731–736.

[16] N. A. M and S. Vilmos, "A novel congestion propagation modeling algorithm for smart cities," *Pervasive and Mob. Comput.*, p. 101387, 2021.

[17] H. Bushra, A. K. Abir, R. M. Fazle, and A. M. Eunus, "Modelling road traffic congestion from trajectories," in *Inter. Conf. on Data Sci. and Infor. Tech.*, 2020, pp. 117–122.

[18] P. He, G. Jiang, S.-K. Lam, and Y. Sun, "Learning heterogeneous traffic patterns for travel time prediction of bus journeys," *Inf. Sci.*, vol. 512, pp. 1394–1406, 2020.

[19] G. Jiang, S.-K. Lam, P. He, C. Ou, and D. Ai, "A multi-scale attributes attention model for transport mode identification," *IEEE Trans. Intell. Transp. Syst.*, 2020.

[20] L. Y., Y. Rose, S. Cyrus, and L. Yan, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," *Int. Conf. on Learn. Representations (ICLR)*, 2018.

[21] L. Jing and G. Wei, "A summary of traffic flow forecasting methods [j]," *J. of Highway and Transp. Res. and Dev.*, vol. 3, pp. 82–85, 2004.

[22] H. Sepp and S. Jürgen, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[23] D. X. Lei, X. Yu, Z. Chao, D. Yang, Z. Q. Pei, and R. W. Xiong, "Traffic congestion prediction by spatiotemporal propagation patterns," in *IEEE Inter. Conf. on Mob. Data Manage. (MDM)*. IEEE, 2019, pp. 298–303.