Bus Travel Speed Prediction using Attention Network of Heterogeneous Correlation Features*

Yidan Sun, Guiyuan Jiang, Siew-Kei Lam, Shicheng Chen, Peilan He †

Abstract

Accurate bus travel speed prediction can lead to improved urban mobility by enabling passengers to reliably plan their trips in advance and traffic administrators to manage the bus operations more effectively. However, the increasing complexity of public transportation networks pose a significant challenge to existing prediction methods as the bus operations are affected by numerous factors such as varying traffic conditions, tight bus operation schedules, wide-ranging travel demands, frequent accelerations/decelerations at bus stops, delays at intersections, etc. This paper aims to achieve accurate bus speed prediction by identifying important intrinsic and extrinsic features that impact the bus speed, and their significance in specific situations. We propose to jointly incorporate multiple feature components that provide discriminating information to train the prediction model by exploring the spatial correlation, temporal correlation, as well as contextual information (e.g. road characteristics and weather conditions). In particular, we introduce an attribute-driven attention network model to integrate the feature components, which considers the heterogeneous influence of different feature components on bus speed and dynamically assigns weights to the learned latent features based on specific traffic situations. Extensive experiments using real bus travel data involving 42 bus services show that our proposed method outperforms six well-known methods.

1 Introduction

Accurate prediction of the bus travel speed are essential for improving the performance of Intelligent Transportation System, especially in Advance Traffic Management System and Advanced Traveler Information Systems [1]. Passengers can rely on the prediction to reliably plan their trips, allowing them to reach their destinations on time. Traffic administrators can utilize the predictions to manage the public transportation system by scheduling the buses to meet travel demands, hence mitigating crowdedness in buses and at bus stops. This reduces the passenger waiting time at transfer points which contributes to improve the passengers' experience and increase ridership in public transports.

Although traffic speed prediction has been extensively studied [2], achieving accurate travel speed prediction is still a challenging problem due to many complex issues. For example, features of different traffic conditions are usually influenced by numerous factors such as spatial dependencies among different road segments, temporal correlation with historical observations, and external factors (e.g. weather condition, road characteristics). Moreover, there have been limited studies on predicting the travel speed of buses [4], which behaves significantly different from that of general vehicles (e.g. private cars, taxis, etc.). This is because bus travel speed, particularly in urban public transportation networks, is affected by numerous factors such as varying traffic conditions (i.e. traffic flow, congestions), tight bus operation schedules, wide-ranging travel demands, frequent accelerations and decelerations at bus stops, delays at intersections, etc. As such, bus speed prediction remains a challenging and unresolved problem.

To achieve an accurate prediction, it is necessary to explore and incorporate sufficient factors to train the prediction model. In addition, different factors have varying degree of impact on bus travel speed under different situations. For example, weather conditions impact the travel speed more significantly on snowy and rainy days, while road network characteristics (e.g. road type, number of lanes), instead of congestion indicators, affect the travel speed on road segments that are less likely to be congested. Therefore, it is critical to determine the factors that contribute to the bus travel speed and their relative importance in various traffic situations. Based on this hypothesis, we propose an efficient model for bus speed prediction in road segments by 1) identifying sufficient and meaningful intrinsic and extrinsic features that impact the bus travel speed and 2) introducing an attribute-driven attention network model to integrate the feature components, which con-

*This research project is funded 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.

†School of Computer Science and Engineering, Nanyang Technological University, Singapore. {ysun014, phe002}@e.ntu.edu.sg, {gyjiang, assklam, coder.chenshicheng}@ntu.edu.sg. Guiyuan Jiang is the corresponding author.

siders the heterogeneous influence of different feature components on bus travel speed and dynamically assign weights to the learned latent features based on the specific traffic situations. The main contributions of this paper can be summarized as follows:

1) We explore spatial, temporal correlations, and contextual information to extract sufficient features to train our machine learning model in order to achieve accurate bus speed prediction. To aggregate the local neighbourhood information, we utilize a structure-to-vector embedding technique based on the Pearson Correlation of bus speeds between road segments. We modified the embedding technique by introducing separate parameters for each neighbouring road to take into account the heterogeneity of neighbourhood influence on bus speed. We employ the global-spatial correlation as the criteria to group similar road segments (similar in traffic patterns but could be spatially distant) into clusters and train a separate predictor for each cluster. In this way, each cluster can have sufficient training examples which is important for training deep learning based predictors. This strategy can easily scale to large road networks by choosing a suitable number of clusters.

2) In addition to relying on LSTM (Long Short-Term Memory) network to capture the temporal dependency, we extract temporal correlation features that include both recent historical information (short-term pattern) and periodicity information (long-term pattern). In addition to spatiotemporal correlation, we also incorporate *contextual information*, which includes road network characteristics (road length, road type, number of lanes, number of bus stops and traffic lights on the road), and other extrinsic factors such as weather condition information, holiday events, and day-of-week.

3) We propose an attribute-driven attention network model to integrate the multiple feature components, which characterize the spatial, temporal correlations, and contextual information. The attention network considers the heterogeneous influence of different features and can dynamically assign weights to the learned latent features based on specific situations.

4) Finally, we conduct extensive experiments to evaluate the effectiveness of the proposed method using real bus travel data involving 42 bus services, bus line data, road networks of Singapore, and weather condition data. The results clearly show that our method significantly outperforms existing methods.

2 Preliminaries

DEFINITION 2.1. *(Road Network):* We model a road network as an undirected graph $G = (R, E)$, where each node in R indicates a road segment (splited by junctions) and each edge in E indicates a connection between two road segments such that there exists a link (i, j) if road segment r_i is connected to r_j via a road intersection. There are $N = |R|$ road segments in total.

DEFINITION 2.2. *(h-hop Neighbour):* A road segment r_i is called a h-hop neighbour of r_j if the two road segments are connected via h intersections in the shortest path between them. Direct neighbours are called 1-hop neighbors.

DEFINITION 2.3. *(Bus Speed):* Suppose there are K time intervals in the time span of the historical dataset, i.e., $\mathbf{T} \in \mathbb{R}^K$. The matrix of the historical bus speeds (e.g., 30 km/hour) is denoted as $\hat{\mathbf{y}} \in \mathbb{R}^{N \times K}$, and $\hat{y}_{n,k}$ is the bus travel speed on road segment r_n at time interval t_k. 15 minutes is set as the length of the time interval (i.e if t_k is 8:00am-8:15am, then t_{k+1} is 8:15am-8:30am). We use the vector $\hat{\mathbf{y}}_n \in \mathbb{R}^K$ to denote the sequence of bus speeds of road segment r_n over the entire time span. In addition, we use $\mathbf{y} \in \mathbb{R}^N$ to denote the target vector to be predicted.

Bus Speed Prediction: This problem aims to predict the bus travel speed at time interval $t + h$ for any road segment, i.e. given the historical bus travel speed data until time interval t, we want to predict the bus travel speed for h time intervals after t. We set $h = 1$ in our experiments. In addition to historical bus speed records, we also incorporate temporal correlation, spatial dependency across different road segments, contextual features such as weather conditions (refer to Section 5.1 for more details).

Overview of the Proposed Method: The proposed solution consists of two general steps. The *first step* is feature extraction, as shown in Figure 1, which models and captures spatial (local and global), temporal (short and long term) correlations, and contextual information (e.g. road characteristics and weather condition). Each group of features is represented by a feature vector. The *second step* performs model training and prediction using deep learning. The input to the deep learning based model is a sequence of feature vectors generated from a historical record. The trained model is then used to make predictions for a given road segment and time interval.

3 Feature Extraction

3.1 Spatial Correlation Feature

3.1.1 Local-Spatial Correlation Intuitively, road segments sharing similar traffic conditions (e.g. traffic flow, congestions, etc.) are likely to have similar bus travel speed patterns. For example, segments belonging to the same street usually have similar speed

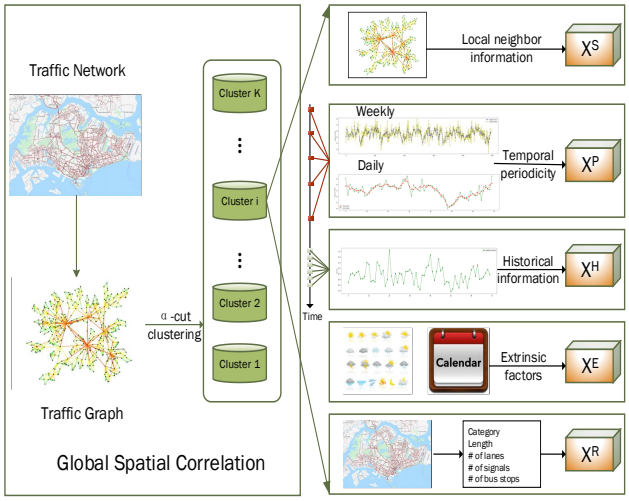

Figure 1: Overview of correlation feature extraction.

patterns especially during peak hours, where a large number of people commute to work or home. Motivated by the First Law of Geography [29] - "near things are more related than distant things", existing works propose to utilize CNN (Convolutional Neural Network [30]) and graph embedding (called `structure2vec` [14]) to capture the neighbourhood information of spatially near regions. However, the CNN methods operate on grid-structure network and employ the same convolution kernels to all roads for aggregating local neighborhood information, while the existing graph embedding techniques typically treat all the neighbours with equal importance. It is evident that neighbours have varying levels of correlations with the target road segment, and existing works also report that incorporating regions with weak correlations to predict a target region actually hurts the performance [34].

To address this issue, we develop a graph embedding technique to extract neighbourhood information as a feature vector, which will be fed to the prediction model, based on the `structure2vec` method. For road segment r_i at time interval t, a feature vector of length H is constructed as $\mathbf{X}^S = (\mu_1^{i,t}, \cdots, \mu_H^{i,t})$, where

(3.1)
$$\mu_h^{i,t} \leftarrow \theta_1 \cdot \hat{y}_{i,t} + (1-\theta_1) \sum_{r_u \in N(r_i)} \theta_{i,u} \mu_{h-1}^{u,t}, 2 \leq h \leq H$$

where $N(r_i)$ is the set of 1-hop neighbours of r_i, $\hat{y}_{i,t}$ is the bus speed of r_i at time t, θ_1 and $\theta_{i,u}$ are hyper parameters to control the weights of different terms, and $\theta_{i,u}$ will be discussed later. Initially, $\mu_1^{i,t} = \hat{y}_{i,t}$.

The difference between our model and the `structure2vec` method is that the latter assume that all neighbours have equal impact to r_i, thus there are no parameters $\theta_{i,u}$ ($r_u \in N(r_i)$) in their model. However, it has been demonstrated that different neighbours have different degrees of influence [6]. In our graph embedding model, the heterogeneity of neighbourhood influence is taken into consideration by introducing a separate parameter $\theta_{i,u}$ for each neighbouring road segment. The parameters $\theta_{i,u}$ ($r_u \in N(r_i)$) are estimated by calculating the Pearson Correlation between road segment r_i and its 1-hop neighbors r_u, i.e. $N(r_i)$.

3.1.2 Global-Spatial Correlation We developed a method to take the global correlation into consideration to improve the accuracy of the bus speed predictions. We partition all roads into k clusters such that roads of similar traffic patterns (in terms of bus speeds) are grouped into the same cluster. We also use Pearson Correlation score of bus speeds to estimate the similarity between road segments, $w_{i,j}$ ($r_i, r_j \in R$). A given road network G weighted by the Pearson Correlation coefficients is partitioned into k disjoint clusters, i.e. $\{G_1, G_2, \ldots, G_k\}$. We define function $w(G_i, G_j)$ as the sum of similarity of all links having one endpoint in G_i and the other endpoint in G_j,

$$w(G_i, G_j) = \sum_{r_p \in G_i, r_q \in G_j} w_{p,q} \tag{3.2}$$

For a given partition $G = \{G_1, G_2, \ldots, G_k\}$, the *cut* of a partition G_i is defined as the summation of similarity values associated with all links having one endpoint in G_i and the other endpoint in any partition other than G_i, i.e. $w(G_i, \overline{G_i})$. Similarly, the *association* of a partition G_i is defined as the summation of similarity values associated with all links having both of their endpoints in G_i, i.e. $w(G_i, G_i)$. We employ the k-way α-Cut algorithm [7] to partition the given road graph into k clusters, by minimizing the following objective function.

$$\alpha - Cut(G) = \sum_{i=1}^{k} (\alpha \times \frac{w(G_i, \overline{G_i})}{|G_i|} - (1-\alpha) \times \frac{w(G_i, G_i)}{|G_i|}) \tag{3.3}$$

The objective function $\alpha - Cut(G)$ combines two terms, where the first term minimizes the *cut* representing the inter-partition similarity while the second term maximizes the *association* representing the intra-partition similarity. The parameter $\alpha \in [0, 1]$ controls the weights of the two terms.

We consider the global correlation as the criteria to group similar road segments into clusters and train a separate predictor for each cluster, instead of training a single predictor for all road segments or one predictor for each road segment. This strategy is more robust (compared to training one predictor for all roads) as

only road segments of similar patterns share the same predictor. In addition, this also increases the training examples which is important for training deep learning based predictors. Moreover, this strategy is more efficient as less predictors are needed (compared to training a predictor for each road) and is able to scale to large road networks by controlling the number of clusters.

3.2 Temporal Correlation Feature

3.2.1 Short-term Historical Information We consider two types of historical information: *recent observations* and *historical trend*.

For a road segment r_i, the recent observations consist of the bus speeds of previous lr time slots, i.e. $\mathbf{ro} =< \hat{y}_{i,t-lr}, \hat{y}_{i,t-lr+1}, \ldots, \hat{y}_{i,t-1} >$, where lr is the number of recent observations utilized. Considering that bus travel speeds typically suffer from severe fluctuations due to traffic congestions and delays at intersections, we do not directly use **ro** as features for the prediction model. Instead, the feature vector $\mathbf{x}^{lr}_{i,t} =< s_{i,t-lr}, s_{i,t-lr+1}, \ldots, s_{i,t-1} >$ is used for prediction model training, where $s_{i,j}$ is calculated as the average of W speed observations, i.e. $\frac{1}{W}\sum_{k=0}^{W-1} \hat{y}_{i,j-k}$, where parameter W is set to 2 in our implementation.

For a road segment r_i, we calculate the average bus speed of all days at each time interval t of 15 mins (there are 72 intervals in total from 6:00 to 24:00), which produces a time series of 72 average speeds. The obtained time series is then decomposed into three components namely: the trend, season and remainder using STL decomposition [12]. The obtained trend sequence, **dt**, will be used for training predictors. Similarly, we obtain an average speed sequence with a period of one week, where each value in the sequence is calculated as the average speed at the time interval of the week (there are 7×72 intervals in a week). A weekly trend sequence **wt** is obtained by decomposing the weekly average speed sequence. For a road segment r_i at time t, we build a feature vector $\mathbf{x}^{tr}_{i,t} =< dt_{i,t}, wt_{i,t} >$, where $dt_{i,t}$ is the value in r_i's daily trend at time t and $wt_{i,t}$ is the value in r_i's weekly trend at time t, respectively.

Then we integrate the two feature vectors $\mathbf{x}^{lr}_{i,t}$ and $\mathbf{x}^{tr}_{i,t}$ and concatenate them to form the feature vector $\mathbf{X}^{H}_{i,t}$ to represent the historical information.

3.2.2 Long-term Temporal Periodicity Traffic speed typically repeats periodically [5], meaning that the traffic speed at a certain period is similar to the same time period of the previous day or previous week. Thus we incorporate the periodicity information of a road segment at the time interval to improve prediction accuracy. Also due to the severe fluctuation in bus travel speed, we use historical average speeds which is more reliable. As such, for a road segment r_i, the periodicity information is constructed as a feature vector $\mathbf{X}^{P}_{i,t} =< y^0_{i,t}, y^1_{i,t}, \ldots, y^7_{i,t} >$, where $y^0_{i,t}$ is the average speed of road r_i at time interval t of all days, and $y^j_{i,t}$ $(1 \le j \le 7)$ is the average speed of r_i at time interval t of all the j-th day of a week.

3.3 Contextual Features

3.3.1 Road Network Characteristics A feature vector $\mathbf{X}^R$ is constructed to capture the road characteristics of a road segment. It includes the length of the road segment, road type (e.g. primary road, second primary road, highway), number of lanes, number of bus stops on the road segment, number of traffic lights (e.g. 0, 1, or 2).

3.3.2 Extrinsic Factors Traffic speed can be affected by many complex extrinsic factors, such as weather condition and activity event. In order to provide more opportunity for identifying specific traffic situations, we augment each training sample with additional extrinsic features. Let $\mathbf{X}^{E}_{i,t}$ be the feature vector that represents these extrinsic factors at predicted time interval t. In this work, we mainly incorporate *weather condition* (one-hot encoding), *holidays* (yes: 1, no: 0), *time interval* t and the *day-of-week*. This is because, 1) vehicles typically travel at a slower speed during heavy rains, 2) holiday events significantly affect the traffic flow which indirectly affects bus travel speed, 3) the congestion level changes with the time of the day, and hence bus travel speed is correlated with the time interval t, and 4) week day and weekend exhibit significant differences in their traffic patterns.

4 Prediction Model

4.1 Overview of the Network Structure Figure 2 shows the structure of our deep learning based prediction model, which consists of four major components, i.e. input layer, hybrid layer, attention layer, prediction layer. The *input-layer* component consists of all the features discussed in Section 3, i.e. $\mathbf{X}^{V}_{i,t} \cup \mathbf{X}^{S}_{i,t} \cup \mathbf{X}^{H}_{i,t} \cup \mathbf{X}^{P}_{i,t} \cup \mathbf{X}^{E}_{i,t} \cup \mathbf{X}^{R}_{i,t}$, where $\mathbf{X}^{V}_{i,t}$ contains the speed of r_i at time t, i.e. $\hat{y}_{i,t}$. In the *hybrid-layer* component, $\mathbf{X}^{R}_{i,t}$ is fed into a fully connected neural network (FNN) because it contains only static features, while other terms ($\mathbf{X}^{V}_{i,t}$, $\mathbf{X}^{S}_{i,t}$, $\mathbf{X}^{H}_{i,t}$, $\mathbf{X}^{P}_{i,t}$ and $\mathbf{X}^{E}_{i,t}$) share the same network structure with a Long Short-Term Memory (LSTM) network [19]. The output of hybrid-layer component, i.e. the learned

latent representations of the extracted feature vectors, are fed into the *attention-layer* component to calculate the importance (weights) of different features based on specific situation. Then the weights as well as the latent feature representations serve as inputs to the *prediction-layer* component to make predictions.

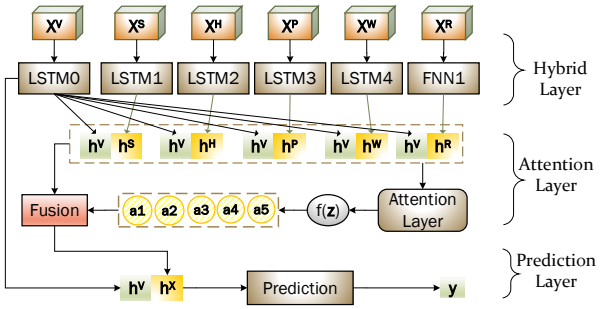

Figure 2: Structure of the prediction model.

4.2 LSTM Layers We use the LSTM network to learning the hidden temporal dependency among the features. The architecture of the LSTM cell can be described with the following equations:

$$
\begin{aligned}
\mathbf{i}_{tt} &= \sigma(\mathbf{W}_{ix}\mathbf{x}_{tt} + \mathbf{W}_{ih}\mathbf{h}_{tt-1} + \mathbf{b}_i) \\
\mathbf{f}_{tt} &= \sigma(\mathbf{W}_{fx}\mathbf{x}_{tt} + \mathbf{W}_{fh}\mathbf{h}_{tt-1} + \mathbf{b}_f) \\
\mathbf{o}_{tt} &= \sigma(\mathbf{W}_{ox}\mathbf{x}_{tt} + \mathbf{W}_{oh}\mathbf{h}_{tt-1} + \mathbf{b}_o) \\
\widetilde{\mathbf{C}_{tt}} &= \tanh(\mathbf{W}_{Cx}\mathbf{x}_{tt} + \mathbf{W}_{Ch}\mathbf{h}_{tt-1} + \mathbf{b}_C) \\
\mathbf{C}_{tt} &= \mathbf{i}_{tt} * \widetilde{\mathbf{C}_{tt}} + \mathbf{f}_{tt} * \mathbf{C}_{tt-1} \\
\mathbf{h}_{tt} &= \mathbf{o}_{tt} * \tanh(\mathbf{C}_{tt})
\end{aligned} \tag{4.4}
$$

where tt stands for the tt-th time interval, $\mathbf{i}_{tt}$, $\mathbf{f}_{tt}$, $\mathbf{o}_{tt}$ refer to the output of the input gate, forget gate and output gate respectively. $\mathbf{x}_{tt}$, $\mathbf{c}_{tt}$, $\mathbf{h}_{tt}$ are the input vector, state vector and hidden vector respectively, and $\mathbf{h}_{tt-1}$ is the former output of $\mathbf{h}_{tt}$. $\widetilde{\mathbf{C}_{tt}}$ and $\mathbf{C}_{tt}$ are the input state and output state of the memory cell, and $\mathbf{C}_{tt-1}$ is the former state of $\mathbf{C}_{tt}$. σ is a sigmoid function. $\mathbf{W}_{ix}, \mathbf{W}_{fx}, \mathbf{W}_{ox}, \mathbf{W}_{Cx}$ are the weight matrices connecting $\mathbf{x}_{tt}$ to the three gates and the cell input, $\mathbf{W}_{ih}, \mathbf{W}_{fh}, \mathbf{W}_{oh}, \mathbf{W}_{Ch}$ are the weight matrices connecting $\mathbf{x}_{tt-1}$ to the three gates and the cell input, $\mathbf{b}_i$, $\mathbf{b}_f$, $\mathbf{b}_o$, $\mathbf{b}_C$ are the bias terms of the three gates and the cell gate.

4.3 FNN Layers We use fully-connected neural network (FNN) layers in both hybrid-layer and prediction-layer component.

In **Hybrid-Layer**: The features of road characteristics $\mathbf{X}^R_{i,t}$, which are static, are fed into FNN layers to learn high-order interactions, as follows.

$$
\begin{aligned}
\mathbf{z}^1_R &= relu(\mathbf{W}^1_R(\mathbf{X}^R_{i,t}) + \mathbf{b}^1_R) \\
\mathbf{z}^j_R &= relu(\mathbf{W}^j_R(\mathbf{z}^{j-1}_R) + \mathbf{b}^j_R) \quad 1 < j \leq L
\end{aligned} \tag{4.5}
$$

where $\mathbf{z}^j_R$ is the output of the j-th layer and L is the number of FNN layers, $\mathbf{W}^j_R$ ($\mathbf{W}^1_R$) is the weight matrix connecting neurons in j-th (1-st) layer and $\mathbf{z}^{j-1}_R$ ($\mathbf{X}^R_{i,t}$), $\mathbf{b}^j_R$ is the bias terms for the j-th FNN layer.

In **Prediction-Layer**: similar to (5), the FNN makes prediction based on the output of the attention layer, i.e. $\mathbf{h}^V_{tt} \oplus \mathbf{h}^x_{tt}$,

$$
\begin{aligned}
\mathbf{z}^1_P &= relu(\mathbf{W}^1_P(\mathbf{h}^V_{tt} \oplus \mathbf{h}^x_{tt}) + \mathbf{b}^1_P) \\
\mathbf{z}^j_P &= \sigma(\mathbf{W}^j_P(\mathbf{z}^{j-1}_P) + \mathbf{b}^j_P) \qquad 1 < j \leq L'
\end{aligned} \tag{4.6}
$$

where $\oplus$ is the concentration operation, L' is the number of FNN layers, σ is sigmoid function.

4.4 Attention Layer The feature vectors discussed in Section 3 do not contribute equally to the bus speed prediction especially under specific traffic situations. For example, road characteristics (e.g. road type) should be given more attention to road segments that are located in light or non-congested areas, instead of the features that are used to characterize traffic congestions. In our model, the attention mechanism [32] is employed to discriminate the importance of different feature components automatically. The key idea is to assign weights to different feature components, where the weights a^j_{tt} are parameters learned by the model. Formally, the final fused feature is calculated as

$$
\mathbf{h}^x_{tt} = \sum_j a^j_{tt} \cdot (\mathbf{h}^V_{tt} \oplus \mathbf{h}^j_{tt}) \tag{4.7}
$$

where $\mathbf{h}^j_{tt}$ is the output of LSTM/FNN (i.e. the learned latent representation) for term $\mathbf{X}^j$ ($j \in \{S, H, P, R, E\}$), tt is the time-step of training procedure, a^j_{tt} is the weight for $(\mathbf{h}^V_{tt} \oplus \mathbf{h}^j_{tt})$, and $\sum a^j_{tt} = 1$. The weight parameter a^j_{tt} is learned through the attention layer,

$$
z^j_{tt} = \mathbf{V}^T_{aj} relu(\mathbf{W}_{aj}(\mathbf{h}^V_{tt} \oplus \mathbf{h}^j_{tt}) + \mathbf{b}_{aj}) \tag{4.8}
$$

$$
a^j_{tt} = \frac{exp(z^j_{tt})}{\sum_j exp(z^j_{tt})} \tag{4.9}
$$

where $\mathbf{W}_{aj}$ is weight matrices connecting neurons in attention layer and the input $\mathbf{h}^V_{tt} \oplus \mathbf{h}^j_{tt}$, $\mathbf{V}^T_{aj}$ connect neurons in attention layer with z^j_{tt}, $\mathbf{b}_{aj}$ is the bias terms.

Loss Function. The following loss function contains two terms: *mean square error* and *square of mean absolute percentage error*. The former pays more attention

to predictions of large values while minimizing the latter can prevent the training from being dominated by large value samples. λ is a hyperparameter to control the weights of different terms.

$$L_{loss} = \sum_{i=1}^{F} (\| \hat{y}_i - y_i \|^2 + \lambda \| \frac{\hat{y}_i - y_i}{\hat{y}_i} \|^2) \tag{4.10}$$

The algorithm Adam is utilized for optimization. The training process repeats for 50 epochs. To prevent overfitting, the dropout mechanism is applied to each hidden layer, where the rate of dropout is set to 0.5.

5 Results and Analysis

5.1 Experimental Settings Datasets. (1) *Road network data:* We use a subset of Singapore road network covered by a rectangle area (Southwest: 1.3346, 103.6757; Northeast: 1.3572, 103.7092), comprised of 703 road segments. It is used to derive topological attributes of the road segments, including the information of types of road segments (e.g. primary, secondary, residential), number of lanes, whether the ends of the road segment are associated with traffic signals, and if there is a bus stop along the road segment (1 if yes).
(2) *Bus line data.* The bus route information includes the ID and the GPS location of each bus stop in sequential order. We map the bus routes to the road network to find out the road segments covered by the bus lines. The results are verified by comparing with Google Map via visualization. Based on the map-matched bus line routes, the number of intersections and the number of traffic signals on any road segment can be calculated. 42 bus services are used in the experiment.
(3) *Bus speed data:* The bus speed data is calculated based on historical bus trajectories, obtained from Land Transport Authority, Singapore. The granularity of the trajectories is one point per minute, each point contains the GPS location of the bus and the corresponding timestamp. A bus speed dataset is calculated with the time span from May 06 to July 07, 2017. The data of last two weeks is used for testing while the remaining data is used for training. In addition to the speed values, the time related features including the time-of-day and day-of-week of the speed record are also extracted.
(4) *Weather data:* Weather condition influences the bus travel speed by affecting the bus stopping time at bus stops as well as the moving speed of vehicles. Hourly-grained weather data are collected during the same time period of the bus speed data.

5.2 Baseline Methods We compare our method with several baseline methods that can be adapted to prediction problem with graph data structures, including: (1) **ARIMA** (Auto Regression Integrated Moving Average) [9]: It makes predictions solely based on historical data. (2) **LR** (Linear Regression): LR is utilized to model the relationship between bus speed and all the impact factors/features. (3) **SVR** (Support Vector Regression) [3]: It is a variant of Support Vector Machine that is used for classification. (4) **LSTM** (Long Short-Term Memory) [26]: A 3-layer LSTM network with $relu$ activation is used to estimate the traffic speed for each cluster. The size of hidden layers is fixed as 128. (5) **FMSTA** [8]: FMSTA is a low-rank tensor decomposition based method which incorporates various properties in spatiotemporal data for forecasting real-world problems. (6) **DL-STF** [17]: DL-STF is a RNN based method which solves the wind speed prediction problem using spatiotemporal information. It models spatiotemporal information with graph structure and forecasts the wind speed of all nodes (stations) at the same time.

In our implementation, the number of FNN layers L and L' (in hybrid-layer and prediction-layer component) are set to 3 where each layer has 12 neurons. Each of the LSTM layers in the hybrid-layer component contains 3 layers where each layer has 12 neurons. In this way, the total number of parameters in our model is almost the same as the baseline LSTM. In the graph embedding to extract the local-correlation features, H is set to be 5, θ_1 are empirically selected to maximize the performance for each cluster. In α-cut algorithm, α is set to 0.5 and there are 15 clusters in total. In the experiments, a separate model is trained for each road segment for ARIMA and SVR, as this achieves better results than training one predictor for each cluster. For other baselines, one model is trained for each cluster.

Table 1: Comparison of results on MAE, MAPE and RMSE.

Methods	MAE (km/h)	MAPE (%)	RMSE (km/h)
ARIMA	3.308	21.2	4.333
LR	3.352	22.2	4.808
SVR	2.789	18.9	3.830
LSTM	3.219	21.1	4.216
FMSTA	4.801	23.3	6.533
DL-STF	3.198	20.8	4.152
Proposed	**2.061**	**14.8**	**3.380**

5.3 Results The performance measures used are the Mean Absolute Error (MAE), the Mean Absolute Percentage Error (MAPE), and Root Mean Square Error (RMSE). Table 1 shows that the proposed method achieves the best performance. This is because the proposed method not only extracted sufficient features that can capture the spatial, temporal correlations and contextual information, but it also takes into account the

heterogeneous influence of the multiple feature vectors using the attention network. The major reason that the baseline methods fail to obtain good predictions is that they are sensitive to the severe fluctuations in bus speed values due to the frequent stopping of buses during congestions, and at bus stops or traffic signals. For example, ARIMA could not produce promising results because it is sensitive to anomalies and the predictions will become unreliable if there is a huge difference in speed values between two consecutive time steps. In the problem considered in this paper, the fluctuations of bus speeds behave similarly to anomalies thus leading to poor prediction results. Although the baselines FMSTA and DL-STF also consider spatiotemporal information, they do not sufficiently capture the local/global spatial correlation and short/long-term temporal correlation, which is important for bus speed prediction due to the severe fluctuations. They also do not take into account the heterogeneous influence of different feature components. Even considered temporal correlation and contextual information including road network characteristics and weather condition, the baseline LSTM also does not obtain promising results, since it does not consider sufficient spatial correlation and the dynamic importance of different feature components. Among all the baselines, SVR shows the best performance, but it requires a separate predictor for each road segment which is impractical. In addition, it takes the longest training time for all the road segments.

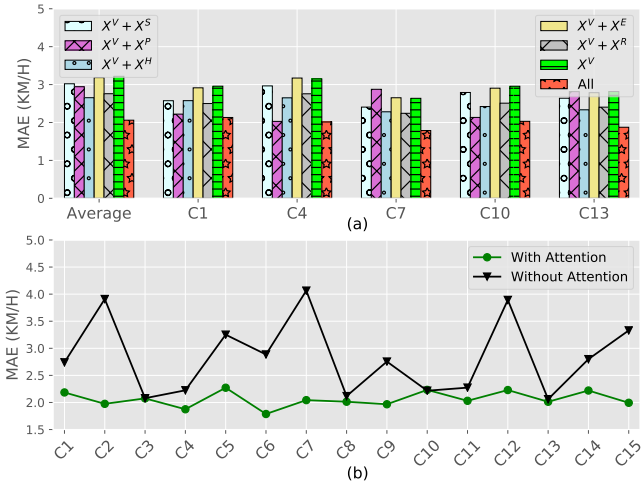

Figure 3: (a) Effect of the extracted features, (b) Effect of the attention mechanism.

Effect of Extracted Features. Figure 3 (a) shows the effectiveness of different feature components. It shows the average results over all clusters and the results on cluster 1, 4, 7, 10 and 13 where all the clusters are ordered based on cluster size. It can be observed that the worst performance is obtained when using only speed value, i.e. $\mathbf{X}^V$. On the other hand, when all feature vectors are taken into account, i.e. feature combination `All`, the best performance is achieved. Moreover, each feature component has different influence strength over different clusters. For example, $\mathbf{X}^P$ shows a strong impact on the prediction accuracy for clusters 1, 4 and 10 compared to other clusters (i.e. 7 and 13), as lower prediction errors are obtained by the feature combination of $\mathbf{X}^V + \mathbf{X}^P$. $\mathbf{X}^R$ typically is less important than other features for cluster 4 and 10, but has a higher impact on clusters 7 and 13. This clearly reinforces our hypothesis to improve prediction accuracy with an adaptive attention mechanism that dynamically assigns weights to different feature components.

Effect of Attention Mechanism. Figure 3 (b) demonstrates the effect of attention mechanism which could capture dynamic importance of each feature components, by comparing the results using and without using attention mechanism. It can be observed that the attention mechanism obtains large improvements over most clusters but almost no improvement over cluster 3, 10 and 13. The range of improvements also varies significantly across clusters. This is reasonable as different clusters are associated with different situations based on specific roadway characteristics and extrinsic factors.

Prediction Time. The experiments are tested on Intel(R) Xeon(R) CPU E5-1650 v2 @ 3.50GHz with 32G RAM. Even though the training time of the prediction model is longer, the prediction can be efficiently achieved. During the testing phase, the prediction time for 2000 records is as follows: ARIMA 0.028s, LR 0.020s, SVR 0.021s, LSTM 0.024s, FMSTA 0.170s, DL-STF 0.024s, and our method 0.027s.

6 Related Works

6.1 Models of Traffic Speed Prediction Existing methodologies on traffic speed prediction can be typically divided into two categories, i.e. parametric approach and non-parametric approach [2]. Parametric methods include the ARIMA [5] and its variations, the Kalman filter (KF) [21], Bayesian network models [16], and hidden Markov model (HMM) [31, 33]. Parametric approaches rely on predetermined models based on certain theoretical assumptions, in which model parameters are calculated and calibrated with recent observation data. These methods typically focus on predicting short or long term future traffic and most of them rely on an assumption that the state of current traffic is available. Parametric approaches are often inefficient for a large scale transportation system or for long-term

vehicle speed prediction [22, 26].

Many recent non-parametric approaches for traffic speed prediction are based on machine learning (ML) technique, such as linear regression (LR) [27], support vector regression (SVR) [3], random forest [18] and neural network (NN) based approaches [11, 25]. A literature review [24] on short-term traffic forecasting showed that researchers have shifted their focus from the classical methods (parametric approach and classical ML methods such as LR) to neural network based approaches (e.g. Look-up convolution recurrent neural network [25]) due to the explosive increase in data accessibility and computing power. Moreover, the above-mentioned methods are often used in a hybrid manner. For example, the KF method is combined with the NN methods in [23], and fuzzy inference systems are often combined with NN (i.e. fuzzy-neural networks [10, 28]).

6.2 Features for Traffic Speed Prediction In addition to incorporating as many extrinsic factors as possible (e.g. weather condition), many studies have focused on capturing spatiotemporal information for the traffic prediction problems. Even though some parametric approaches consider spatiotemporal correlation in their models, e.g. HMM [33], the spatiotemporal correlation is mainly investigated by deep learning (DL) based methods. Traffic speed of a road segment has inherent *temporal patterns* (e.g. Repeatability and Similarity [5]) and this can be naturally taken care by LSTM [26]. A deep stacked bidirectional and unidirectional LSTM network architecture [13] is proposed that considers both forward and backward dependencies in traffic speed series. The work in [11] proposed a convolutional neural network (CNN), named PCNN, to model the intricate natures of temporal features, including periodicity, local coherence, etc., for short-term traffic prediction. Attention mechanism was integrated into deep learning models [36, 38] for time series prediction problem to capture various importance of different spatial neighbors. However, the above-mentioned methods do not take spatial dependency into consideration.

CNN based methods have been applied to deal with grid-based crowd flow prediction [35] and traffic speed prediction for ring road [30], by aggregating correlated neighborhood information (nearby road segments or regions) using CNN technique to improve the predictive accuracy. However, their approach is targeted at grid networks or ring road networks where each road segment has a fixed number of upstream and downstream road segments. Graph embedding [14] is another technique to capture local-spatial correlations by aggregating the neighborhood information of spatially nearby regions, while graph convolution [15] is a spectral approach that ensures strictly localized filter and exhibits low computational complexity. The above-mentioned methods, i.e. CNN, graph convolution and graph embedding, are efficient for capturing local-spatial correlations but do not pay enough attention to the global correlations (e.g. highly correlated road segments that are spatially disconnected).

7 Conclusions

This paper investigates the problem of predicting bus travel speed on urban road networks, which is a challenging task because bus travel speed is affected by numerous factors (such as congestions, delays at intersections and bus stops). We proposed to jointly incorporate multiple correlation features into the prediction model to improve prediction accuracy. We developed methods to extract features for capturing local-spatial correlation, global-spatial correlation, recent and distant historical information, temporal periodicity, as well as contextual information such as road network characteristics and other extrinsic factors. Using a combination of these features, we proposed a DL based approach with attention mechanism to dynamically control the weight parameters (importance) of different feature components. The experimental results demonstrated that the proposed approach significantly outperforms the baseline approaches.

References

[1] E. I. Vlahogianni, M. G. Karlaftis and J. C Golias, *Short-term traffic forecasting: Where we are and where we're going*, Transport. Res. C-Emer., 43 (2014), pp. 3–19.

[2] B. Jiang and Y. Fei, *Vehicle speed prediction by two-level data driven models in vehicular networks*, IEEE Trans. Intell. Transp. Syst., 18(7):1793–1801, 2017.

[3] Chun-Hsin Wu., Jan-Ming Ho. and D. T. Lee, *Travel-Time Prediction With Support Vector Regression*, IEEE Trans. Intell. Transp. Syst., 5(4):276–281, 2004.

[4] N. Julio; R. Giesen and P. Lizana, *Real-time prediction of bus travel speeds using traffic shockwaves and machine learning algorithms*, Res. Transp. Econ., 59(2016), pp. 250–257.

[5] Z. Hou and X. Li, *Repeatability and similarity of freeway traffic flow and long-term prediction under big data*, IEEE Trans. Intell. Transp. Syst., 17(6):1786–1796, 2016.

[6] A. Achar, R. R. S. V. and A. Sivasubramaniam, *Predicting vehicular travel times by modeling heterogeneous influences between arterial roads*, in AAAI, 2018, pp. 2063–2070.

[7] T. Anwar, C. Liu H. L. Vu and C. Leckie, *Spatial*

partitioning of large urban road networks, in EDBT, 2014, pp. 343–354.

[8] M. T. Bahadori, Q. R. Yu and Y Liu, *Fast multivariate spatio-temporal analysis via low rank tensor learning*, in NIPS, 2014, pp. 3491–3499.

[9] G. E. Box and D. A. Pierce, . *Distribution of residual autocorrelations in autoregressive-integrated moving average time series models*, J. Amer. Stat. Assoc., 65(332), pp. 1509–1526, 1970.

[10] W. Chen, J. An, R. Li, L. Fu, G. Xie, M. Z. A. Bhuiyan and K Li, *A novel fuzzy deep-learning approach to traffic flow prediction with uncertain spatial–temporal data features*, Future Gener. Comput. Syst., 89 (2018), pp. 78–88.

[11] M. Chen, X. Yu and Y Liu, *Pcnn: Deep convolutional networks for short-term traffic congestion prediction*, IEEE Trans. Intell. Transp. Syst., (99):1–10, 2018.

[12] R. B. Cleveland, W. S. Cleveland, J. E. McRae and I. Terpenning, *STL: A seasonal-trend decomposition*, J. Off. Stat., 6(1):3–73, 1990.

[13] Z. Cui, R. Ke and Y Wang, 2017. *Deep stacked bidirectional and unidirectional lstm recurrent neural network for network-wide traffic speed prediction* arXiv preprint arXiv:1801.02143, 2017.

[14] H. Dai, B. Dai, and L Song, *Discriminative embeddings of latent variable models for structured data* in ICML, 2016, pp. 2702–2711.

[15] M. Defferrard, X. Bresson and P. Vandergheynst, *Convolutional neural networks on graphs with fast localized spectral filtering* in NIPS, 2016, pp. 3844–3852.

[16] X. Fei, C.-C. Lu and K. Liu, 2011. *A bayesian dynamic linear model approach for real-time short-term freeway travel time prediction* Transport. Res. C-Emer., 19(6):1306–1318, 2011.

[17] A. Ghaderi, B. M. Sanandaji and F. Ghaderi, *Deep forecast: deep learning-based spatio-temporal forecasting* arXiv preprint arXiv:1707.08110, 2017.

[18] B. Hamner, *Predicting travel times with context-dependent random forests by modeling local and aggregate traffic flow*, in ICDM Workshops, pp. 1357–1359, 2010. IEEE.

[19] S. Hochreiter and J. Schmidhuber, *Long short-term memory*, Neural comput., 9(8):1735–1780, 1997.

[20] Y. Jia, J. Wu and Y. Du, *Traffic speed prediction using deep learning method*, in ITSC, pp. 1217–1222, 2016. IEEE.

[21] B. A. Kumar, L. Vanajakshi and S. C. Subramanian, *Bus travel time prediction using a time-space discretization approac*, Transport. Res. C-Emer., 79:308–332, 2017.

[22] S. Lefèvre, C. Sun, R. Bajcsy and C. Laugier, *Comparison of parametric and non-parametric approaches for vehicle speed prediction*, in ACC, 2014, pp. 3494–3499.

[23] H. Liu, H. Van Zuylen, H. Van Lint and M. Salomons, *Predicting urban arterial travel time with state-space neural networks and kalman filters*, Transp. Res. Record, 1968(1):99–108, 2006.

[24] E.I. Vlahogianni, M.G. Karlaftis and J.C. Golias, *Short-term traffic forecasting: Where we are and where were going*, Transport. Res. C-Emer, 43, pp.3-19, 2014.

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

[26] 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*, Transport. Res. C-Emer., 54:187–197, 2015.

[27] G. Ristanoski, W. Liu and J. Bailey, *Time series forecasting using distribution enhanced linear regression*, in PAKDD, 2013, pp. 484–495. Springer.

[28] J. Tang, F. Liu, Y. Zou, W. Zhang and Y. Wang, *An improved fuzzy neural network for traffic speed prediction considering periodic characteristic*, IEEE Trans. Intell. Transp. Syst., 18(9):2340–2350, 2017.

[29] W. R. Tobler, *A computer movie simulating urban growth in the detroit region*, Econ. Geogr. 46(sup1):234–240, 1970.

[30] J. Wang, Q. Gu, J. Wu, G. Liu and Z. Xiong, *Traffic speed prediction and congestion source exploration: A deep learning method*, in ICDM, 2016, 499–508. IEEE.

[31] S. Wang, F. Li, L. Stenneth and S. Y. Philip, *Enhancing traffic congestion estimation with social media by coupled hidden markov model*, in ECML-PKDD, 2016, 247–264. Springer.

[32] D. Wang, J. Zhang, W. Cao, J. Li, and Y. Zheng, *When will you arrive? estimating travel time based on deep neural networks*, in AAAI, 2018, 2500–2507.

[33] B. Yang, C. Guo and C. S. Jensen, *Travel cost inference from sparse, spatio temporally correlated time series using markov models*, Proc. VLDB Endowment, 6(9):769–780, 2013.

[34] H. Yao, F. Wu, J. Ke, X. Tang, Y. Jia, S. Lu, P. Gong and J. Ye, *Deep multi-view spatial-temporal network for taxi demand prediction*, in *AAAI*, 2018, pp. 2588–2595.

[35] J. Zhang, Y. Zheng and D. Qi, 2017. *Deep spatio-temporal residual networks for citywide crowd flows prediction*, in AAAI, 2017, pp. 1655–1661.

[36] Yuxuan Liang, Songyu Ke, Junbo Zhang, Xiuwen Yi and Yu Zheng, *GeoMAN: Multi-level Attention Networks for Geo-sensory Time Series Prediction*, in *IJCAI*, 2018, pp. 3428-3434

[37] Chao Huang, Junbo Zhang, Yu Zheng and Nitesh V Chawla, *DeepCrime: Attentive Hierarchical Recurrent Networks for Crime Prediction*, in *CIKM*, 2018, pp. 1423–1432

[38] Huaxiu Yao, Xianfeng Tang, Hua Wei, Guanjie Zheng and Zhenhui Li, *Revisiting Spatial-Temporal Similarity: A Deep Learning Framework for Traffic Prediction*, in *AAAI*, 2019