Spatio-Temporal Attention based Conditional Traffic Flow Prediction in Urban Areas

No Author Given

No Institute Given

Abstract. Conditional traffic flow prediction is an important step for urban planners to deploy new change to the land or building use, which may cause a significant impact on traffic congestion by the drastic increase of travel demands. However, it is not trivial to model a comprehensive method which can consider multi-modal conditions to predict traffic flow because of their different spatial and temporal complexity. In this paper, we propose a novel conditional traffic flow prediction model STGCAN which can effectively utilize the syntactic and semantic roads information with given daily conditions such as subway ridership demand and weekday-holiday conditions. The experiment results show that our model outperforms the baseline models, and the semantic road features (e.g. POIs) help to improve the performance of our model. Our work provides insight to construct a model that integrates the spatial and temporal information of the data, and we expect our research to be of great help in research of urban planning and smart city fields.

Keywords: spatio-temporal attention · traffic flow prediction · graph convolution network

1 Introduction

Conditional traffic flow prediction[1,2] is an important step for urban planners before deploying new change to the land or building use, which may cause a significant impact on traffic congestion by the drastic increase of travel demands. For example, when SBC Park in San Francisco was planned for construction, extensive transportation management considerations had been taken to carry more than 40,000 visitors while alleviating the traffic congestion[3]. For predicting urban traffic flow, previous studies show that one of key factors to consider is road traffic which is affected by various conditions such as road networks [16,17,18], semantics of a region [6,7], daily travel demand [8,9], or weekday-holidays. However, it is not trivial to model a comprehensive method which can consider all of these multi-modal conditions mentioned above to predict traffic flow because of their different spatial and temporal complexity [4].

With the advent of sophisticated spatio-temporal neural networks, there have been a handful of approaches that leverage the ridership data to conduct conditional traffic prediction. [5] proposes a spatio-temporal deep learning framework leveraging fully convolutional neural network which can forecast the inflow and

outflow of ridership and traffic flow using their historical data on the conditions of holidays and weather. In this research, traffic flow prediction is conducted in units of coarse grid cells of city, which can cause degraded accuracy when several roads overlap in one cell. In other words, the proximity of euclidian distance between two roads does not guarantee that their traffic pattern will be similar if the roads are not connected as discussed in [16]. [8,9] propose models based on conditional generative adversarial networks (GAN) with given ridership demand estimation to predict traffic status. These models also has a similar problem to the [5] as they use grid-cell unit traffic data. Also, these GAN models are not stable for training [19] and are not scalable to handle multi-modal conditions. Meanwhile, many studies suggest that the urban semantics of the road such as land use, POI information, and the purpose of visitors in the region are correlated with traffic flow and ridership demand[10,11,12,14,6]. Since the way people exploit a region can provide more explainability for urban traffic flow prediction, it is important to consider urban semantics of the road for more accurate traffic flow prediction.

In this paper, we propose a novel conditional traffic prediction model, **S**patio-**T**emporal **G**raph **C**onvolution and **A**ttention **N**etwork (STGCAN) which can effectively utilize syntactic and semantic roads information such as road connectivity, POIs, and land use information, under given subway ridership demand and weekday-holiday conditions. STGCAN is composed of a encoder and a decoder spatio-temporal blocks, where the encoder extends the given sparse subway ridership features to be projected to road traffic flow producing an latent embedding space and the decoder transforms it into traffic flow prediction of each road. After the encoder block process the subway station unit ridership data into road unit latent embedding, the decoder block utilizes the spatio-temporal embedding that contains road semantics as well as other syntactic information and apply spatial attention. Both encoder and decoder block also includes temporal attention which can capture temporal relationship of the input features. We compare our model with the baseline models such as CNN, LSTM, BiLSTM, and temporal attention, and show that our model outperforms the others. We summarize our contributions as follows:

- We propose STGCAN which comprises graph convolution and spatio-temporal attention mechanisms which utilize the semantic road features which has not been tried in the previous research.
- Our model can predict traffic flows in multi-modal conditions using subway ridership data which has high practical value for application in urban planning.
- We construct our novel dataset to experiment with our model including traffic speed, subway ridership, POI data, and residential data. We publish our dataset and program code to contribute to research community.

2 Related Work

Conditional Traffic Flow Prediction Previous research on conditional traffic flow prediction propose various approaches to capture spatio temporal features of different types of data. [4] gives overview of spatio-temporal data including dynamic conditions such as POIs, taxi trajectories, business location, or route planning using machine learning. [5] propose a spatio-temporal deep learning framework which can forecast the inflow and outflow of ridership and traffic flow using their historical data on the conditions of holidays and weather. [6] use kernel ridge regression to describe the non-linear non-additive relationships of impacting factors such as points-of-interest (POIs), geo-tagged tweets, weather, vehicle collisions. [7] use GCN and GRU models combined to consider both static and dynamic factors such as distribution of roadside POIs and weather for forecasting traffic states. [8,9] propose models based on conditional generative adversarial networks with given ridership demand estimation to predict traffic status. Such models predictions are based on traffic dependencies on diverse conditions including spatial-temporal features. However, they lack the analysis of urban semantics of adjacent locations of road.

Traffic Flow and Ridership with Urban Semantics There have been diverse approaches to find the relationship between urban semantics and traffic flow. [10] analyzes the correlation between (POI) and the real-time traffic in Beijing, China and the main congestion areas using cluster analysis and linear regression. [11] found that gentrification has a positive correlation with collision patterns in Los Angles County using multivariate regression method. [12] proposes that population and area of the metropolitan area, the vitality of the regional economy (median housing costs) has positive correlation with ridership demands through linear regression. [14] uses taxi trajectory from GPS data and subway demands in Wuxi, China by constructing two directed graphs each to evaluate the statistical results before and after the opening of a new subway. However, these researches could not overcome the limitation of linear regression, and lack the insight of explicit traffic congestion prediction that can be used real-time.

Spatio-Temporal Traffic Prediction Models The survey[15] gives overall perspective of spatio-temporal traffic prediction using graph neural networks such as graph convolution or graph attention networks. One of the earlier trials to capture spatio-temporal relations in traffic prediction is DCRNN [16] which captures the spatial dependency using bidirectional random walks on the graph, and the temporal dependency using the encoder-decoder architecture with scheduled sampling. STGCN[17] utilizes stacks of spatial and temporal convolutional graph network to capture spatio-temporal features of the traffic data. GMAN[18] uses STAttention block which is multi-attention blocks to run spatial, temporal embedding with gated fusion and puts STAttention block in encoders and decoders which is following transform mechanism.

3 Preliminary Definitions

3.1 Notations and Definitions

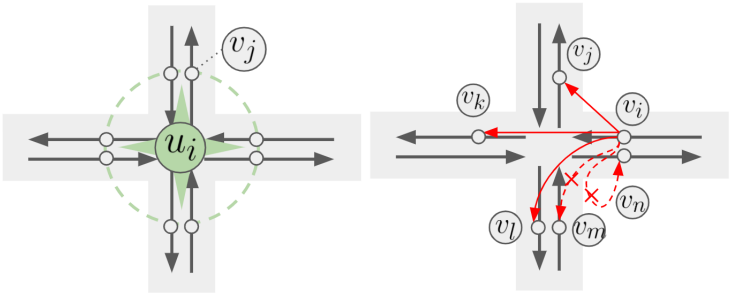
We list the preliminaries to be used throughout the paper in Table 1.

Table 1: Preliminary Notations and Definitions

Notations	Descriptions
$N_t \in \mathbb{N}$	Number of time slots within a day.
$N_c \in \mathbb{N}$	Number of subway stations.
$N_s \in \mathbb{N}$	Number of target estimation roads.
$N_f \in \mathbb{N}$	Number of semantic road features.
$U = \{u_1, ..., u_{N_c}\}$	A set of subway stations.
$V = \{v_1, ..., v_{N_s}\}$	A set of roads.
$G_c = (U, V, E_c)$	Bipartite subway-road network.
$G_s^{(u)} = (V, E_s^{(u)})$	Undirected road network of passengers.
$G_s^{(d)} = (V, E_s^{(d)})$	Directed road network of vehicles.
$\mathcal{X}^\tau \in \mathbb{R}^{N_s \times N_t}$	Traffic speed on day τ at each road.
$\mathcal{D}^\tau \in \mathbb{R}^{N_c \times N_t \times 2}$	Ridership (on/off) on day τ at each subway station.
$\mathcal{M}^\tau \in \mathbb{R}^{N_s \times N_f}$	Semantic features of each road on day τ.
$P^\tau \in \{0, 1\}^8$	Weekday and holiday (7+1) features on day τ.

Definition 1 (Subway-Road Network). We denote a bipartite subway-road network as $G_c = (U, V, E_c)$, where $U = \{u_1, ..., u_{N_c}\}$ is a set of subway stations, $V = \{v_1, ..., v_{N_s}\}$ is a set of roads, and edges $E_c = \{(u_i, v_j)\}$, where $u_i \in U$ and $v_j \in V$. In general, subway stations in Seoul are located at intersections to let passengers to easily move to their destination through an entrance and to allow people to cross the road by using an underpass. In Fig. 1, a subway station u_i is connected to eight roads including v_j as the station is located on the the crossroad.

Definition 2 (Road Network). We denote two different road networks by the way to establish road connectivity: an undirected road network, $G_s^{(u)}$, in terms of passengers, and a directed road network, $G_s^{(d)}$, in terms of vehicles. First, we denote an undirected road network as $G_s^{(u)} = (V, E_s^{(u)})$, where V is a set of roads and $E_s^{(u)} = \{(v_i, v_j)\}$ is a set of connectivity between the roads. We assume that passengers can cross the road and move to their intended destination through crosswalks or underpasses. Thus, in Fig. 1, v_i is connected to each road $\{v_j, v_k, v_l, v_m, v_n\}$ by the crossroad regardless of the traffic directions. Second, we denote a directed road network as $G_s^{(d)} = (V, E_s^{(d)})$, where $V = \{v_1, ..., v_{N_s}\}$ is a set of roads and $E_s^{(d)} = \{(v_i \rightarrow v_j)\}$ is a set of directed connectivity between



(a) Subway-road connectivity (b) Road connectivity

Fig. 1: Subway and road connectivity

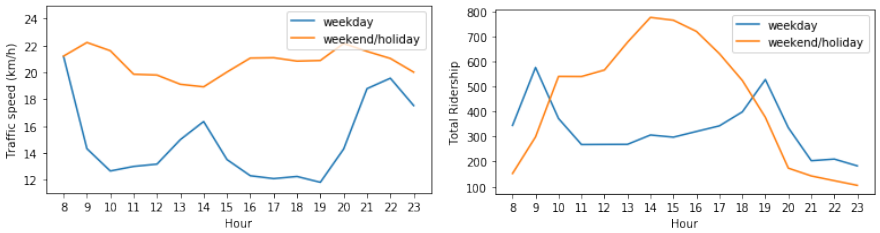
the roads. Since roads in our dataset have starting and ending points at the crossroads in many cases, we consider different scenarios where two roads can be connected. In Fig. 1, a road v_i is connected to v_j, v_k, v_l since the vehicle at v_i can drive to as they are on the same traffic direction. However, we do not consider the connections $(v_i \rightarrow v_m)$ and $(v_i \rightarrow v_n)$ are established since their traffic direction is opposite or they are in a rare U-turn case.

Definition 3 (Traffic Speed). Among the different metrics to measure the traffic flow, we choose traffic speed as it best indicates the traffic congestion empirically. The traffic speed data on each road $v_i \in V$ is measured for N_t time slots on a day. We denote the traffic speed value on the day τ as $\mathcal{X}^\tau = \{X_1^\tau, ..., X_{N_s}^\tau\} \in \mathbb{R}^{N_s \times N_t}$, where $X_s^\tau \in \mathbb{R}^{N_t}$ is a traffic speed matrix at road s with N_t time slots on the day τ.

Definition 4 (Subway Ridership). The ridership of each subway station $u_j \in U$ is measured for N_t time slots on a day. We denote the subway ridership as $\mathcal{D}^\tau = \{D_1^\tau, ..., D_{N_c}^\tau\} \in \mathbb{R}^{N_c \times N_t \times 2}$, where $D_c^\tau \in \mathbb{R}^{N_t \times 2}$ is a number of passengers at station c with N_t time slots who ride on/off on the day τ.

Definition 5 (Semantic Road Features). In order to give semantic features of road, we extract N_f types of daily road features from POI, residential area, and road property which are described in the Dataset section. We denote the semantic road features on the day τ as $\mathcal{M}^\tau = \{M_1^\tau, ..., M_{N_f}^\tau\} \in \mathbb{R}^{N_s \times N_f}$.

Definition 6 (Weekend and Holiday Features). Since the traffic patterns are highly correlated to the holidays or the days of the week as shown in Fig.2, we denote weekend and holiday features as $P^\tau \in \{0, 1\}^8$, where the first seven features are the one-hot encoding of the day of the week and the last one feature is indication of the national holiday.



(a) Sample traffic speed (b) Sample subway ridership

Fig. 2: Hourly pattern of sample traffic speed and ridership demand on weekday and weekend/holiday.

3.2 Problem Definition

Given the condition on a day τ which are the subway ridership $\hat{\mathcal{D}}^{\tau}$, the semantic road feature $\hat{\mathcal{M}}^{\tau}$, and the weekday-holiday information $\hat{P}^{\tau}$, we aim to estimate the traffic speed of roads on that day, that is $\hat{\mathcal{X}}^{\tau}$.

4 Methodology

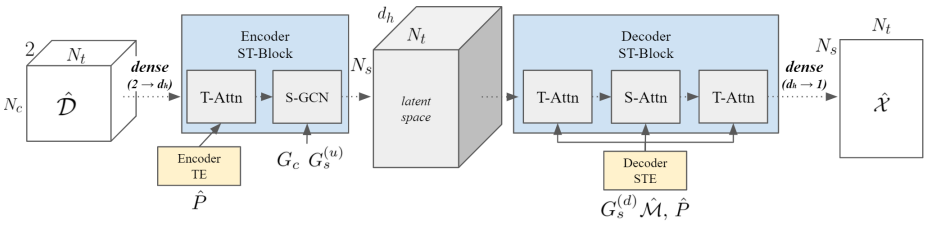


Fig. 3: The overview of our proposed STGCAN model.

The overview of our proposed STGCAN model architecture is described in Fig.3. Our model consists of the encoder spatio-temporal (ST) block and the decoder ST block, with two dense layers where one at the front which expands the last feature dimension from 2 to d_h, and the other one at the last which shrinks it from d_h to 1 and produce the predicted traffic speed value, $\hat{\mathcal{X}}$. In the following subsections, we first explain the spatial graph convolution network (S-GCN) which is used in encoder ST-block. Next, we explain how to make encoder STE and decoder STE which are fed to spatial and temporal attention in encoder and decoder ST-blocks. Then, we explain how to apply spatial attention (S-Attn)

and temporal attention (T-Attn). Finally, we explain how to combine S-GCN, S-Attn, S-Attn to construct the encoder and the decoder ST-blocks.

4.1 Spatio Graph Convolution Network

In order to calculate the impact on nearby roads through passengers using subways, we propose spatial graph convolution network (S-GCN) in the encoder ST-block. Basically, graph convolution network (GCN)[23] is a method of convolution using the embedding of the connected nodes to construct the target node embedding. We slightly modify standard GCN to fit our subway-road network which transforms the input of N_c subway ridership into N_s of road unit features. The key idea of our S-GCN is to train the weighted adjacency matrix of the k-hop neighborhood roads to each subway station for the graph convolution. Before leveraging S-GCN, we first extract each adjacency matrix from G_c and $G_s^{(u)}$, respectively. We construct a bipartite adjacency matrix from G_c as $A_c \in \{0,1\}^{N_s \times N_c}$, where $A_{c(i,j)} = 1$, if $(v_i, u_j) \in E_c$, and $A_{c(i,j)} = 0$, otherwise. Also, we construct an adjacency matrix from $G_s^{(u)}$ as $A_s \in \{0,1\}^{N_s \times N_s}$, where $A_{s(i,j)} = 1$, if $(v_i, v_j) \in E_s^{(u)}$, and $A_{s(i,j)} = 0$, otherwise. Then, we establish a k_c-hop connectivity adjacency matrix A_{sc} as following equation.

$$A_{sc} = \left(\bigvee_{i=0}^{k_c} A_s^i A_c \right) \in \{0,1\}^{N_s \times N_c} \tag{1}$$

We repeat A_{sc} for N_t times and stack to construct $A_{sct} = [A_{sc}]_{t=1}^{t=N_t} \in \mathbb{R}^{N_t \times N_s \times N_c}$. Then, we conduct element-wise matrix multiplication ($\circ$) of a weight matrix $\mathbf{W}_A$ with ReLU activation:

$$\mathbf{A} = \mathrm{ReLU}(A_{sct} \circ \mathbf{W}_A) \in \mathbb{R}^{N_t \times N_s \times N_c} \tag{2}$$

where $\mathbf{W}_A \in \mathbb{R}^{N_t \times N_s \times N_c}$ is a learnable parameter. Using the weighted connectivity of the roads of each time slot $\mathbf{A}_t$, we produce output $H_t^{(l)}$ of this hidden layer as

$$H_t^{(l)} = \mathrm{ReLU}(\mathbf{A}_t H_t^{(l-1)} \mathbf{W}_{g,t} + \mathbf{b}_{g,t}) \tag{3}$$

where $H_t^{(l-1)} \in \mathbb{R}^{N_c \times d_h}$, $H_t^{(l)} \in \mathbb{R}^{N_s \times d_h}$, $\mathbf{W}_H \in \mathbb{R}^{N_t \times d_h \times d_h}$, $\mathbf{b}_H \in \mathbb{R}^{N_t \times N_s \times d_h}$, and $\mathbf{W}_H$, $\mathbf{b}_H$ are learnable parameters. Note that the dimension of input and output of the S-GCN has been changed from $H^{(l-1)} \in \mathbb{R}^{N_c \times N_t \times d_h}$ to $H^{(l)} \in \mathbb{R}^{N_s \times N_t \times d_h}$.

4.2 Spatio-Temporal Embedding for Attention

We leverage spatio-temporal embedding module to feed attention modules in each encoder and decoder ST-block. In general, all the size of the embeddings to be used throughout our STGCAN is unified to d_h for simplicity. Our spatio-temporal embedding simply consists of half of spatio embedding and the other

half of temporal embedding. In the case of temporal embedding, one half includes time slot feature while the other half contains the weekday-holiday features. In the case of spatio embedding, syntatic and semantic features takes each half. We recommend to set d_h to be dividable by 4, as each spatio and temporal embedding consists of two different types of features.

Encoder Temporal Embedding To leverage the temporal attention in the encoder ST-Block, we need $N_c \times N_t$ embeddings corresponding to combinations of each station and time step of the ridership data. To enable this, we first create one-hot encoding of the size $\mathbb{R}^{N_c}$ corresponding to each station, and process through two dense layers to generate station embedding, $e^C \in \mathbb{R}^{d_h/2}$. We also create one-hot encoding of size $\mathbb{R}^{N_t}$ containing information for each time step, and process through two dense layers to generate the time-step embedding, $e^T \in \mathbb{R}^{d_h/4}$. Finally, we process $P^\tau \in \{0,1\}^8$ through two dense layers to generate the weekday-holiday embedding, $e^P \in \mathbb{R}^{d_h/4}$. Finally, we concatenate each embedding to generate an encoder temporal embedding of size $e^E = \{e^C \| e^T \| e^P\} \in \mathbb{R}^{d_h}$, and $\{e^E_{c,t}\} \in \mathbb{R}^{N_c \times N_t \times d_h}$ totally. This embedding provides information so that temporal attention can learn characteristics of different time steps and day-holidays information on each station.

Decoder Spatio-Temporal Embedding To utilize the spatial attention and the temporal attention in Decoder ST-Block, we need $N_s \times N_t$ embeddings for combinations of each road and time step to utilize $N_s \times N_t \times d_h$ dimensional latent space embedding. For temporal embedding, we utilize the same $e^T \in \mathbb{R}^{d_h/4}$ and $e^P \in \mathbb{R}^{d_h/4}$ created from the encoder temporal embedding in the previous subsection. In case of spatio embedding, it consists of a synthetic embedding and a semantic embedding. For syntactic embedding, we extract node2vec[26] features from $G_s^{(d)}$ to leverage a vector of $\mathbb{R}^{d_v}$ corresponding to each road, where d_v is the embedding size. Subsequently, we change the feature size from d_v to $d_h/4$ via one dense layer to create an embedding of the dimension $e^{Sy} \in \mathbb{R}^{d_h/4}$. For semantic embedding, we process $\mathcal{M}_s^\tau \in \mathbb{R}^{N_f}$ which is a semantic road feature of a road s, through one dense layer to construct $e^{Sm} \in \mathbb{R}^{d_h/4}$ for each road. Finally, we concatenate each embedding to generate an encoder temporal embedding of $e^D = \{e^{Sy} \| e^{Sm} \| e^T \| e^P\} \in \mathbb{R}^{d_h}$, and $\{e^D_{s,t}\} \in \mathbb{R}^{N_s \times N_t \times d_h}$ totally.

4.3 Spatial and Temporal Attention

The non-linear transformation function f used in this paper is defined in Eq.4, where $\mathbf{W}$ and $\mathbf{b}$ are learnable parameters, and ReLU[27] is an activation function.

$$f(x) = \mathrm{ReLU}(x\mathbf{W} + \mathbf{b}) \tag{4}$$

Each spatial and temporal attention module receives the input $H^{(l-1)}$ and produces the output $H^{(l)}$. The input shape of each encoder ST-block is $\mathbb{R}^{N_c \times N_t \times d_h}$, and the input shape of each decoder ST-block is $\mathbb{R}^{N_s \times N_t \times d_h}$. Each attention

module in encoder and decoder consumes the corresponding encoder or decoder embedding as we explained before, and we denote them as e_{v_i,t_j} or $v_i \in \mathcal{V}$ for convenience, where $\mathcal{V} = U$ in encoder ST-block and $\mathcal{V} = V$ in decoder ST-block and $1 \leq j \leq N_t$. In addition, encoder ST-block utilizes $\{e^E_{c,t}\} \in \mathbb{R}^{N_c \times N_t \times d_h}$, while decoder ST-block utilizes $\{e^D_{s,t}\} \in \mathbb{R}^{N_s \times N_t \times d_h}$. Note that spatial attention is only used in the decoder block as described in Fig.3.

Spatio Attention Since traffic flow in road network affects each connected road by different ways, we apply an attention mechanism is to capture these features. The main idea is to infer the road traffic feature from different features of the road using their connectivity at each time step.

$$h^{(l)}_{v_i,t_j} = \sum\nolimits_{v_k \in \mathcal{V}} \alpha_{v_i,v_k} \cdot h^{(l-1)}_{v,t_j} \tag{5}$$

As shown in the formula above, v_i calculates the attention for v_k as α_{v_i,v_k}, where $\sum_{v_k \in \mathcal{V}} \alpha_{v_i,v_k} = 1$. This value is calculated by considering only the correlation between roads regardless of t_j.

$$s_{v_i,v} = \frac{\langle f_{s,1}(h^{(l-1)}_{v_i,t_j} \| e_{v_i,t_j}), f_{s,2}(h^{(l-1)}_{v,t_j} \| e_{v,t_j}) \rangle}{\sqrt{2d_h}} \tag{6}$$

$f^{(k)}_{s,1}$, $f^{(k)}_{s,2}$ and $f^{(k)}_{s,3}$ denotes three different linear layers as mentioned above. Then, we normalize $s_{v_i,v}$ with softmax:

$$\alpha_{v_i,v} = \frac{\exp(s_{v_i,v})}{\sum_{v_i \in V} \exp(s_{v_i,v})} \tag{7}$$

In addition, multi-headed can be considered to stabilize learning. At this time, the attention of K heads is calculated as follows:

$$s^{(k)}_{v_i,v} = \frac{\langle f^{(k)}_{s,1}(h^{(l-1)}_{v_i,t_j} \| e_{v_i,t_j}), f^{(k)}_{s,2}(h^{(l-1)}_{v,t_j} \| e_{v,t_j}) \rangle}{\sqrt{2d_h}} \tag{8}$$

$$\alpha^{(k)}_{v_i,v} = \frac{\exp(s^{(k)}_{v_i,v})}{\sum_{v_i \in V} \exp(s^{(k)}_{v_i,v})} \tag{9}$$

$$h^{(l)}_{v_i,t_j} = \|^K_{k=1} \{ \sum\nolimits_{v \in V} \alpha^{(k)}_{v_i,v} \cdot f^{(k)}_{s,3}(h^{(l-1)}_{v,t_j}) \} \tag{10}$$

Finally, by adding Add and Normalization layer, we reduce the dimension from $d_h \times K$ to d_h for the output used as input in the next layer.

Temporal Attention Traffic conventions on roads affect different ways over time. To model these features, we utilize temporal attention. The key idea is to learn different features for different time zones on each road for the time axis.

$$h^{(l)}_{v_i,t_j} = \sum\nolimits_{r=1 \ldots N_t} \beta_{t_j,t_r} \cdot h^{(l-1)}_{v_i,t_j} \tag{11}$$

As shown in the formula above, basically v_i calculates the attention for v_k as β_{v_i,v_k}, which is $\sum_{v_k \in \mathcal{V}} \beta_{v_i,v_k} = 1$. Regardless of this road, the value is calculated by considering the correlation between different time steps. In order to apply multi-head similar to the above spatial attention, the following formula is used.

$$u_{t_j,t}^{(k)} = \frac{\langle f_{t,1}^{(k)}(h_{v_i,t_j}^{(l-1)} \| e_{v_i,t_j}), f_{t,2}^{(k)}(h_{v_i,t}^{(l-1)} \| e_{v_i,t}) \rangle}{\sqrt{2d_h}} \tag{12}$$

$$\beta_{t_j,t}^{(k)} = \frac{\exp(u_{t_j,t}^{(k)})}{\sum_{r=1\ldots N_t} \exp(u_{t_j,t_r}^{(k)})} \tag{13}$$

$$ht_{v_i,t_j}^{(l)} = \|_{k=1}^{K} \{ \sum\nolimits_{r=1\ldots N_t} \beta_{t_j,t_r}^{(k)} \cdot f_{t_r,3}^{(k)}(h_{v,t}^{(l-1)}) \} \tag{14}$$

The output dimension of temporal attention is also ultimately reduced by adding Add and Normalization layer, from $d_h \times K$ to d_h for the output to be used in the next module.

4.4 Encoder/Decoder ST-Block

Our STGCAN consists of two main ST-blocks, that are encoder ST-block and decoder ST-block. First, the encoder ST-block consists of a T-Attn and a S-GCN module. When T-Attn module in encoder block captures the temporal feature using an attention mechanism, it utilizes the encoder temporal embedding (TE) which contains the features from the $\hat{P}$. Then, S-GCN module converts station-wise data into road-wise data while capturing the relationship between the subway and road, using G_c and $G_s^{(u)}$, which extends the spatial-dimension of the input from N_c to N_s, and parse to latent space. Second, the decoder ST-block consists of two T-Attn blocks at the front and the back, and one S-Attn block between them in a stacked structure. Each temporal and spatial attention network in the decoder ST-block makes use of decoder spatio-temporal embedding (STE) which contains the features from $G_s^{(d)}$, $\hat{\mathcal{M}}$, and $\hat{P}$.

4.5 Optimization

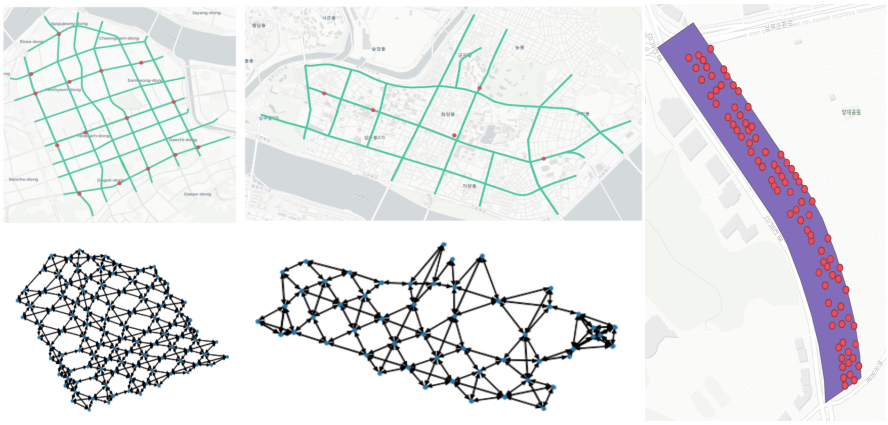
For training of our model, we set the objective loss function for back-propagation as mean absolute error (MAE) between predicted value and ground truths.

$$L(\Theta) = \frac{1}{N_s N_t} \sum\nolimits_{s=1}^{N_s} \sum\nolimits_{t=1}^{N_t} |\mathcal{X}_{s,t} - \hat{\mathcal{X}_{s,t}}| \tag{15}$$

5 Evaluation

5.1 Dataset

To train STGCAN model, there are four different types of dataset required: traffic speed, subway ridership, road semantic features, and national holiday.



(a) Gangnam ($N_s = 212$, $N_c = 15$) (b) Sungsu ($N_s = 106$, $N_c = 6$) (c) POI extraction

Fig. 4: Data processing of each region and semantic road feature.

First, for the traffic speed data, we construct a daily and hourly traffic speed and a road network dataset of Seoul, which is publicly accessible at TOPIS[1](Seoul Transport Operation and Information Service). The traffic speed in this dataset is measured by the average speed of taxis passing each road at each time step. Second, the subway ridership data and station location (latitude, longitude) of subway line 1 to 8 in Seoul is provided in Open Data Plaza[2] for each day and hour. Third, to extract daily road semantic feature which slightly changes everyday as business open and close, we utilize the density of POI types in each road and residential area information. The POI dataset is provided in [3] which contains the location, business types, and the opening and closing date. We explain more detail in Data Processing section how we create $\mathcal{M}$ from the list of road semantic features above. Finally, for the national holiday, we utilize official holiday in Korea for each year. All types of our dataset is extensively constructed from Jan 1st, 2015 to Dec 31st, 2019.

5.2 Data Processing

For target dataset, we select Gangnam and Seongsu-dong as their urban environments are dynamically changing, and their road network is simpler to construct than other areas. Using the geographical information system (GIS) tools, we filter out the road network, and subway stations within those regions as in Fig.4a,4b. To extract semantic road features, we extract POI data and residential area data

[1] https://topis.seoul.go.kr/
[2] http://data.seoul.go.kr/dataList/OA-12921/F/1/datasetView.do
[3] https://www.localdata.go.kr/

within 50 meters from each road on the right side considering the right-hand traffic, by moving the line of the road to the right and create a new polygon to query the features as in Fig.4c. For POI data, we use 10 most frequent categories: mail order business, general restaurant, general health functional food sales business, convenient restaurant, medical device sales (rental) business, clinic, beauty business, publishing company, distribution and sales business, and convenient stores. Each POI feature of a road is processed by counting the type of POIs and divide by the length of the road. For residential area feature, we extract the indication of existence of the apartment information in Seoul as they affect the nearby traffic flow stronger than low-rise residential area. After extract all the road semantic features, we concatenate 10 POI features, 1 lane size feature, 1 residential area indicator, 1 road length feature, 1 maximum speed limit feature to create $N_f = 14$ features on each day. Note that this semantic features slightly changes everyday as old business close and new business open, other features are static which does not change easily.

5.3 Experimental Settings

Metrics We compare using Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE), which are the most commonly used performance comparisons.

Hyperparameters In our data, we use $N_s = 212/102$ (Gangnam/Sungsu) for the number of roads, $N_t = 16$ (8h to 23h) for the number of time slots in a day in hourly unit, $d_h = 64$ for the dimension of hidden features, $d_v = 64$ for node2vec dimension, $K = 2$ for the number of attention heads, and $k = 3$ for the number of hops in S-GCN.

Baselines We compare models such as Historical Average (HA) model using historical means of each road, Historical Average (HA-P) model reflecting weekend and holiday information of each road, Fully Connected Nueral network with CNN (FCN-CNN), FCN-LSTM, FCN-BiLSTM, FCN with temporal attention (FCN-Attn), and our STGCAN with/without $\mathcal{M}$ features in spatio-temporal embedding. The FCN module takes the role to change spatial dimension from N_c to N_s. The FCN-based models concatenates the input with the spatio-temporal embedding without $\mathcal{M}$ feature to give minimum embedding for each road and time step. In STGCAN without $\mathcal{M}$ feature, we give spatio-temporal embedding as $e^D = \{e^{Sy} \| e^T \| e^P\} \in \mathbb{R}^{d_h}$, where we process node2vec feature on dense layer of $d_v \to d_h/2$ to produce $e^{Sy} \in \mathbb{R}^{d_h/2}$.

5.4 Result

Table 2 shows the overall experimental results. First of all, we set HA and HA-P as baseline approaches that predict traffic by the historical information. We see that traffic prediction highly depends on weekday-holiday information P.

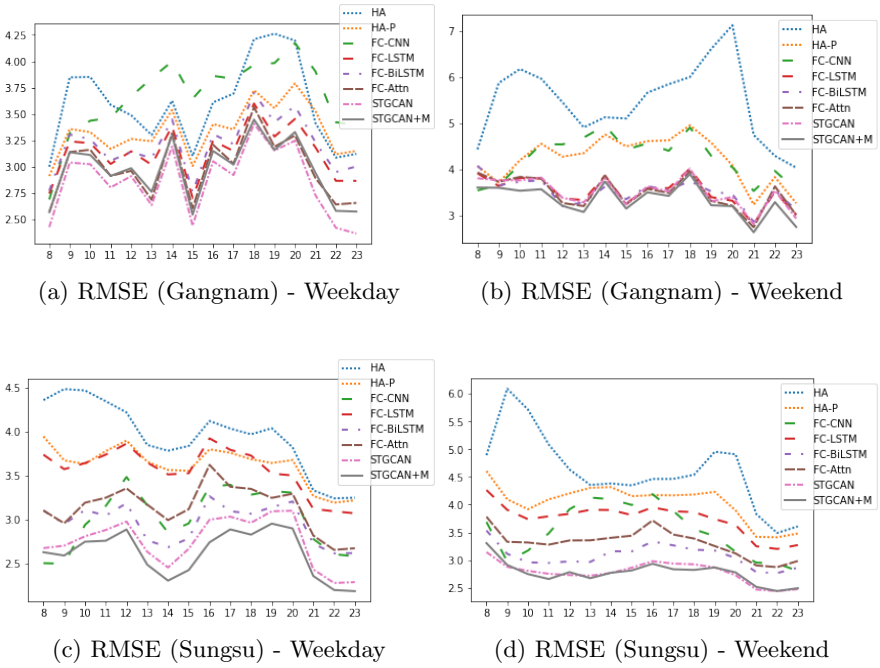
Table 2: Result comparison of STGCAN with other baseline models

	Gangnam				Sungsu			
	MSE	RMSE	MAE	MAPE	MSE	RMSE	MAE	MAPE
HA	18.65	4.32	3.22	17.68	17.64	4.20	3.02	16.39
HA-P	19.71	4.44	2.92	15.32	18.01	4.24	2.79	14.81
FCN-CNN	10.54	3.25	2.32	12.43	10.29	3.21	2.26	11.83
FCN-LSTM	10.71	3.27	2.34	12.37	13.2	3.63	2.46	13.43
FCN-BiLSTM	10.20	3.19	2.27	12.16	9.04	3.01	2.06	11.08
FCN-Attn	10.19	3.19	2.22	11.85	9.81	3.13	2.02	10.94
STGCAN	9.75	3.12	2.16	**11.31**	7.68	2.77	1.90	10.20
STGCAN-M	**9.29**	**3.05**	**2.08**	11.34	**6.36**	**2.52**	**1.74**	**9.44**

While the deep learning based models also utilize the P information, they show better results than HA-P since they use more spatial and temporal features from the input. Note that FCN-Attn performs better than FCN-CNN, FCN-LSTM, and FCN-BiLSTM. This implies that using different attention on different time step is more effective method than using sequential information for traffic flow prediction.

Finally, STGCAN and STGCAN-M outperform the other deep learning based models as they utilize spatial information derived from road network and semantic information. The performance of the STGCAN-M model which takes road semantic $\mathcal{M}$ into consideration is better than STGCAN. This implies that integrating semantic road information also improves the accuracy of traffic preference. When we compare STGCAN and STGCAN-M in different regions, the gap of performance improvement is greater in Seongsu-dong than in Gangnam. Seongsu-dong is one of the fast commercializing regions in Seoul [22]. This results in more POI information in Seongsu-dong and the performance gap of the STGCAN when using $\mathcal{M}$ feature in Sungsu-dong is more effective than Gangnam.

Figure 5 shows that the RMSE plot of each model at different time steps on weekday-holiday condition in Gangnam and Sungsu-dong. In general, every model shows better performance in weekend/holiday than in weekday and have less deviation. On weekday, we see that the results in Sungsu is less erroneous than results in Gangnam at the rush hour (8h-11h, 17h-19h). This is because there exists less congestion during the rush hour in Sungsu-dong, which makes it easier to predict the traffic flow since Seongsu-dong has less office buildings than Gangnam. On the other hand, on the weekend, the results of STGCAN in Sungsu-dong are much better than other models in Gangnam. This is because the semantic features have more information on the Sungsu-dong and they help better prediction especially on the weekend when travelers visits the region a lot.



(a) RMSE (Gangnam) - Weekday (b) RMSE (Gangnam) - Weekend

(c) RMSE (Sungsu) - Weekday (d) RMSE (Sungsu) - Weekend

Fig. 5: RMSE of each region and time step of weekday and weekend.

6 Discussion

Experiment results show that our model performs well regardless of a region. There still exist some limitations in the proposed model. First of all, we find that a standard way to apply graph convolution using column normalized adjacency matrix through several layers instead of k-hop adjacency matrix as our S-GCN is less effectively trained. This could be because the standard GCN model do not train the weights of various edges as STGCAN does, or the input adjacency matrix has insufficient information in terms of strength of connectivity.

During the training of the STGCAN model, we have difficulty in finding the condition where the model converges to learn. Unlike the other baseline neural network models that show better performance by setting the early stopping with validation dataset, our attention-based STGCAN model seems to train more deep features over a longer period of time. Empirically, we find that the model shows good performance when the training epochs is performed about 3,000 times with learning rate of 0.002 on the Adam optimizer[28]. However, we need more extensive research to find best conditions that our model can converge fast and shows the most optimal performance.

For the future research, using extra social features such as social network data real-time transportation that shows more explicit spatio-temporal urban

semantics would expand our research. Testing and generalizing our model to show valid performance in regions other than Seongsu-dong and Gangnam would also contribute to our work.

7 Conclusion

In this work, we propose a STGCAN model that learns the spatial-temporary feature to perform conditional traffic preference. This model performs conditional traffic flow prediction using diverse methods from subway ridership to encoder and decoder. Each encoder and decoder performs dynamic graph convolution and spatial, temporal attention to train syntactic, semantic features of spatial road networks. Our model outperforms compared to the historical average and other baseline neural network models such as CNN, LSTM, BiLSTM, and temporal attention. Our work provides insight to construct a model that performs conditional traffic with a combination of spatial and temporal information, and we expect our research to be of great help in research of urban planning and smart city fields.

References

1. Xie, Peng, et al. "Urban flow prediction from spatiotemporal data using machine learning: A survey." Information Fusion 59 (2020): 1-12.
2. Kalamaras, Ilias, et al. "An interactive visual analytics platform for smart intelligent transportation systems management." IEEE Transactions on Intelligent Transportation Systems 19.2 (2017): 487-496.
3. Luten, Kevin. Mitigating traffic congestion: The role of demand-side strategies. The Association, 2004.
4. Yuan, Haitao, and Guoliang Li. "A Survey of Traffic Prediction: from Spatio-Temporal Data to Intelligent Transportation." Data Science and Engineering 6.1 (2021): 63-85.
5. Zhang, Junbo, et al. "Flow prediction in spatio-temporal networks based on multitask deep learning." IEEE Transactions on Knowledge and Data Engineering 32.3 (2019): 468-478.
6. Wu, Fei, Hongjian Wang, and Zhenhui Li. "Interpreting traffic dynamics using ubiquitous urban data." Proceedings of the 24th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems. 2016.
7. Zhu, Jiawei, et al. "AST-GCN: Attribute-Augmented Spatiotemporal Graph Convolutional Network for Traffic Forecasting." IEEE Access (2021).
8. Yingxue Zhang, Yanhua Li, Xun Zhou, Xiangnan Kong, and Jun Luo. 2019. Traffic-GAN: Off-Deployment Traffic Estimation with Traffic Generative Adversarial Networks. In ICDM
9. Zhang, Y., Li, Y., Zhou, X., Kong, X., Luo, J. (2020). Curb-GAN: Conditional Urban Traffic Estimation through Spatio-Temporal Generative Adversarial Networks. Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2, 842–852. https://doi.org/10.1145/3394486.3403127
10. Zhang, Tianqi, et al. "Impact analysis of land use on traffic congestion using real-time traffic and POI." Journal of Advanced Transportation 2017 (2017).

11. González, S. R., Loukaitou-Sideris, A., Chapple, K. (2019). Transit neighborhoods, commercial gentrification, and traffic crashes: Exploring the linkages in Los Angeles and the Bay Area. Journal of Transport Geography, 77(June 2018), 79–89. https://doi.org/10.1016/j.jtrangeo.2019.04.010
12. Taylor, Brian D., et al. "Analyzing the determinants of transit ridership using a two-stage least squares regression on a national sample of urbanized areas." (2003).
13. Jun, Myung-Jin, et al. "Land use characteristics of subway catchment areas and their influence on subway ridership in Seoul." Journal of Transport Geography 48 (2015): 30-40.
14. Li, Miaoyi, et al. "Examining the interaction of taxi and subway ridership for sustainable urbanization." Sustainability 9.2 (2017): 242.
15. Jiang, Weiwei, and Jiayun Luo. "Graph Neural Network for Traffic Forecasting: A Survey." arXiv preprint arXiv:2101.11174 (2021).
16. Li, Yaguang, et al. "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting." arXiv preprint arXiv:1707.01926 (2017).
17. Yu, Bing, Haoteng Yin, and Zhanxing Zhu. "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting." arXiv preprint arXiv:1709.04875 (2017).
18. Zheng, Chuanpan, et al. "Gman: A graph multi-attention network for traffic prediction." Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 34. No. 01. 2020.
19. Shen, Haocheng, et al. "Counterfeit Anomaly Using Generative Adversarial Network for Anomaly Detection." IEEE Access 8 (2020): 133051-133062.
20. Vaswani, Ashish, et al. "Attention is all you need." arXiv preprint arXiv:1706.03762 (2017).
21. Transit-oriented development in a high-density city: Identifying its association with transit ridership in Seoul, Korea
22. Seoul Metropolitan Government.Comprehensive Measures for Gentrification in Seoul. Government Document, 2015
23. Kipf, Thomas N., and Max Welling. "Semi-supervised classification with graph convolutional networks." arXiv preprint arXiv:1609.02907 (2016).
24. Liang, Yuxuan, et al. "Revisiting convolutional neural networks for citywide crowd flow analytics." Proceedings of ECML-PKDD. Vol. 2020. 2020.
25. Hu, Wangsu, et al. "Discovering urban travel demands through dynamic zone correlation in location-based social networks." Joint European Conference on Machine Learning and Knowledge Discovery in Databases. Springer, Cham, 2018.
26. Grover, Aditya, and Jure Leskovec. "node2vec: Scalable feature learning for networks." Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining. 2016.
27. Nair, Vinod, and Geoffrey E. Hinton. "Rectified linear units improve restricted boltzmann machines." Icml. 2010.
28. Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." arXiv preprint arXiv:1412.6980 (2014).