Learning Heterogeneous Spatial-Temporal Representation for Bike-Sharing Demand Prediction

Youru Li,[†,‡] Zhenfeng Zhu,[†,‡] Deqiang Kong,[≡] Meixiang Xu,[†,‡] Yao Zhao[†,‡]
[†]Institute of Information Science, Beijing Jiaotong University, Beijing, China
[‡]Beijing Key Laboratory of Advanced Information Science and Network Technology, Beijing, China
[≡]Microsoft Multimedia, Beijing, China
[†,‡]{liyouru,zhfzhu,xumx0721,yzhao}@bjtu.edu.cn, [≡]kodeqian@microsoft.com

Abstract

Bike-sharing systems, aiming at meeting the public's need for "last mile" transportation, are becoming popular in recent years. With an accurate demand prediction model, shared bikes, though with a limited amount, can be effectively utilized whenever and wherever there are travel demands. Despite that some deep learning methods, especially long short-term memory neural networks (LSTMs), can improve the performance of traditional demand prediction methods only based on temporal representation, such improvement is limited due to a lack of mining complex spatial-temporal relations. To address this issue, we proposed a novel model named STG2Vec to learn the representation from heterogeneous spatial-temporal graph. Specifically, we developed an event-flow serializing method to encode the evolution of dynamic heterogeneous graph into a special language pattern such as word sequence in a corpus. Furthermore, a dynamic attention-based graph embedding model is introduced to obtain an importance-awareness vectorized representation of the event flow. Additionally, together with other multi-source information such as geographical position, historical transition patterns and weather, *e.g.*, the representation learned by STG2Vec can be fed into the LSTMs for temporal modeling. Experimental results from Citi-Bike electronic usage records dataset in New York City have illustrated that the proposed model can achieve competitive prediction performance compared with its variants and other baseline models.

Introduction

Bike-Sharing systems have been widely used in urban public transportation due to their convenience and environmental friendliness in recent years. As a representative product of the sharing economy, it is often hailed as a good helper to solve the "last mile" in citizen transportation. Its users can check out a bike where they depart and return it to a station close to their destination. However, due to the high frequency and randomness of using, the system has come to be unbalanced in bike distribution. This will result in short supply of bikes in some places and oversupply in others, thus reducing user satisfaction. In general, to solve this unbalanced bike-sharing distribution problem, it is vital to propose an accurate demand prediction model.

Bike-sharing demand prediction can usually be defined as a time series prediction problem from multi-source and heterogeneous data. Traditionally, time series prediction can be considered as building a suitable predictive model (Yule 1927) for a series of data points indexed in time order so as to make good use of the complex sequence dependencies. As a representative of the statistical regression methods, the auto-regressive moving average model (ARMA) and the auto-regressive integrated moving average (ARIMA) model (Box and Pierce 1968) are both well-known models for time series prediction. As machine learning methods grow popular gradually, more researchers focused on the studies to establish nonlinear prediction model based on a large scale of historical data. Typical models such as the support vector regression (SVR)(Drucker et al. 1996) based on kernel methods and the artificial neural networks (ANN) (Davoian and Lippe 2007) with strong nonlinear function approximation ability and the k-Nearest Neighbor (K-NN) regression (Wang and Chaib-draa 2013) based on distance metric in feature space and some tree-based ensemble learning methods, for instance, the random forests (RF) regression (Johansson et al. 2014) and the gradient boosting regression tree (GBRT) (Li and Bai 2016).

With the rise of deep learning methods, the recurrent neural network (RNN) (Rumelhart, Hinton, and Williams 1986) gradually becomes the state-of-the-art method for temporal modeling. However, with longer driving sequence, some problems such as vanishing gradient limit the prediction accuracy of this model. To address these issue, the long short-term memory units (LSTM) (Hochreiter and Schmidhuber 1997) and its variants the gated recurrent unit (GRU) (Cho et al. 2014a) were proposed based on the original RNN which balances memorizing and forgetting by adding multiple threshold gates. Learning from cognitive neuroscience and inspired by some successful applications in natural language processing (Cho et al. 2014b), some researchers (Liang et al. 2018) introduce attention mechanisms to the encoding-decoding framework based on LSTMs to better select from input series and encode information in long-term memory for time series prediction.

Although the LSTM-based models can achieve satisfactory effect in temporal modeling, their ability to model complex non-linear spatial-temporal relations is clearly insufficient (Yao et al. 2018). Particularly, bike-sharing demand

is greatly affected by external conditions. It is necessary to make full use of multi-source heterogeneous information in historical data. As a station-level prediction problem, it is vital to utilize the complex heterogeneous spatio-temporal graph which describes bicycle riding relationships. Inspired by some significant applications in unstructured data embedding (Mikolov et al. 2013) and structured data embedding (Zhu et al. 2013; Guo and Berkhahn 2016; Dai, Dai, and Song 2016) through deep learning methods, we proposed a novel model named STG2Vec to learn the representation of heterogeneous spatio-temporal graph. Specifically, we proposed an event-flow serializing method to represent the evolution process of interaction between the current site and its neighbors from a heterogeneous graph structure within a time-step as a series. Furthermore, a dynamic attention-based graph embedding model is proposed to obtain an importance-awareness vectorized representation of the event-flow.

In general, main contributions in this paper are:

- We developed an event-flow serializing method to represent the evolution process of the dynamic heterogeneous graph as a series.
- A novel dynamic attention-based graph embedding model named STG2Vec is proposed to learn an importance-awareness vectorized representation from heterogeneous spatial-temporal graph.
- To better utilize the multi-source information, we introduced the CE-LSTM to combine the embedded multi-source information with the output of proposed STG2Vec for collaborative temporal modeling.

Related Work

With the wide application of bike-sharing in urban transportation, progress have been made in related researches accordingly. Studies including data analysis and visualization (Yan et al. 2018), have employed data of bike-sharing trajectories to solve specific problems in urban management (Bao et al. 2017; He et al. 2018) and other areas. Demand prediction, as the most classic problem, has received the widest attention in this field. Based on predictive granularity, there are three groups of prediction models in existing researches: city-level, cluster-level, and station-level. For the city-level and cluster-level groups, traditional methods (Chen et al. 2016) usually predict the bike demand for a whole city or design a clustering algorithm to cluster bike stations into groups as prediction units. Although city-level or cluster-level does simplify the problem, it's not as good as station-level prediction for bike-sharing managers to help when scheduling. However, the station-level prediction is difficult because the bike-sharing demand pattern of a station is highly dynamic and context-dependent.

Although station-level prediction is being challenged, it has attracted the attention of many researchers. In station-level hourly demand prediction tasks, some researchers furtherly explore the external context data such as time factors and weather information together with feature engineering and model lightweight processing (Hulot, Aloise, and Jena 2018) for industrial applications. Others have made contribution to temporal modeling by introducing recurrent neural network (Chen et al. 2017) for bike-sharing demand prediction. Meanwhile, other researchers are interested in utilizing the underlying correlations between stations to predict the hourly demand at station-level with deep learning techniques such as graph convolutional neural network (GCN) together with the classic support vector regression model (Lin et al. 2017). Although above studies have improved the accuracy and efficiency of station-level demand prediction models in different ways, there is no way to align temporal modeling with complex non-linear spatial-temporal relations mining.

In bike-sharing demand prediction, any stations are not isolated. The station establishes complex connections through riding relationships to each other which can be expressed by graph structures. In order to capture complex non-linear spatial-temporal relations among stations, it is necessary to establish a length-fixed representation from graph structures. Embedding technology can usually obtain low-dimensional and length-fixed numeric vectors representation from non-linear structures. Typically, the DeepWalk (Perozzi, Al-Rfou, and Skiena 2014) and node2vec (Grover and Leskovec 2016) are widely used in social network mining. These method were proposed inspired by some deep learning embedding methods which had led to significant progress in natural language processing (Mikolov et al. 2013). These make embedding methods used for representation learning in data mining available.

In the specific problem, what is established by the riding relationship among stations is a structurally unstable dynamic heterogeneous graph structure. Although the above methods can learn the embedded representation of the graph structure, they are insufficient when extending to the dynamic graph modeling. Naively applying existing embedding algorithms to each snapshot of dynamic graphs independently usually leads to unsatisfactory performance in terms of stability, flexibility and efficiency. The study on dynamic graphs embedding proposed a DynGEM model based on deep autoencoders (Goyal et al. 2018) which inspires a new idea for dynamic graph representation learning, but it still has room to improve when capturing dependencies between each snapshots of dynamic graphs.

Preliminaries

In this section, we will introduce some notations and the definition of bike-sharing demand prediction. Bike-sharing demand prediction, as a time series prediction problem, can be defined as a station-level check-out/in prediction issue. Given a set of historical trips records: $T_H = (T_{r_1}, T_{r_2}, \ldots, T_{r_H})$, where each trip $T_r = (L_o, L_d, \tau_o, \tau_d)$. Specifically, where L_o denotes the start station, consists of latitude $L_o.lat$ and longitude $L_o.lon$; where L_d denotes the destination station, consists of latitude $L_d.lat$ and longitude $L_d.lon$; τ_o and τ_d are the time corresponding to check-out and check-in in each bike-sharing historical trip. Furthermore, to better describe the problem, we summarize some notations used in our task definition in Table1.

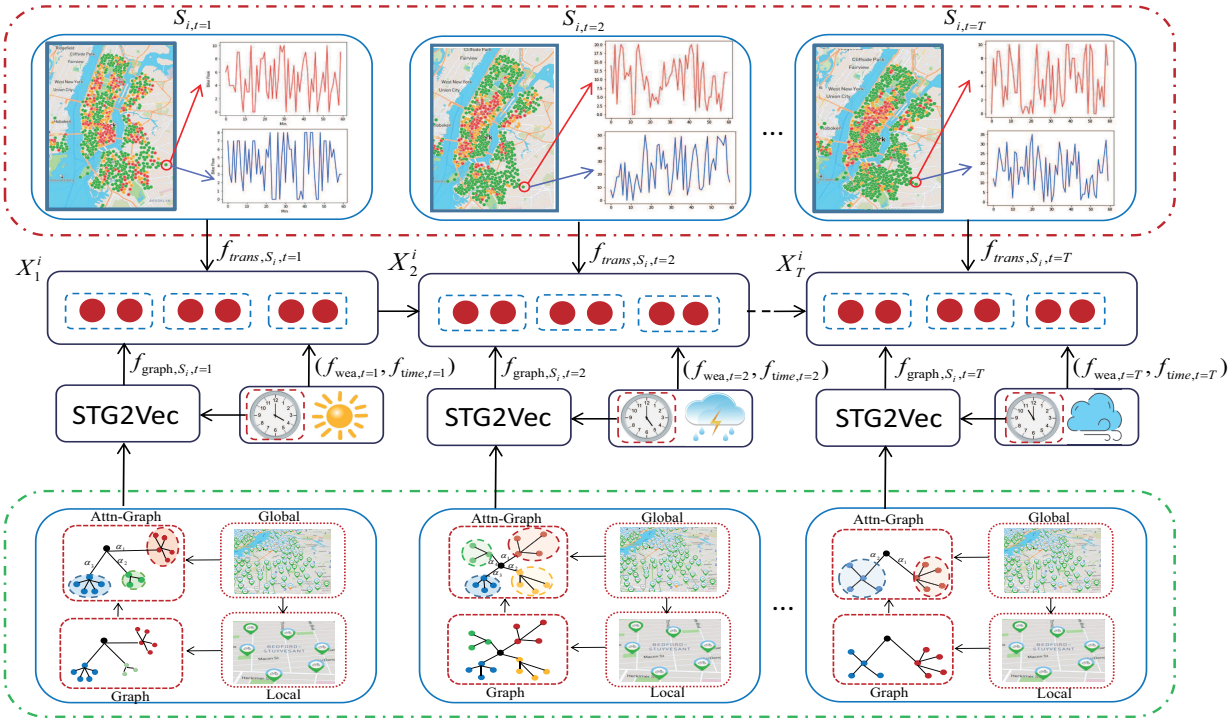

Figure 1: Graphical illustration of learning collaborative representation from nulti-source and heterogeneous spatial-temporal data. This figure is composed of three parts. The top part displays the process of learning historical transition patterns. The bottom part shows the Attn-Graph structure can be produced at the global level from the original graph which responds to the riding relationship between the central site and its neighbors at the current time interval extracted at the local level. Meanwhile, in the middle part of this figure, together with multi-source information from weather, time, geographical position and historical transition patterns, the representation learned by STG2Vec can be fed into the LSTMs for temporal modeling.

Table 1: Notations and Description In Task-level

Notation	Description
S_i	The i^{th} station
$O_{S_i,t}$	Check out of station S_i in time t
$I_{S_i,t}$	Check in of station S_i in time t
$f_{trans,S_i,t}$	Feature of transition in ST-index
$f_{wea,t}$	Meteorology feature in time t
$f_{time,t}$	Feature of time in time t
$f_{graph,S_i,t}$	Embedding of graph in ST-index

Given a set of historical trips which contains geographic and temporal information, we can predict the $O_{S_i,T+1}$ and $I_{S_i,T+1}$ of each station S_i in next time interval by extracting feature of transition, meteorology, time and embedding of graph from each historical time intervals. Typically, time series prediction usually uses a historical sequence of values as the input data. Given jointly feature X_t^i in time t and station S_i which can be concatenated by $f_{trans,S_i,t}$, $F_{wea,t}$, $f_{time,t}$ and $f_{graph,S_i,t}$, the context features at historical time intervals and stations can be defined as $X^i = (X_1^i, X_2^i, ..., X_T^i)$, where $X_t^i \in X^i$ and $X_t^i = (f_{trans,S_i,t}, F_{wea,t}, f_{time,t}, f_{graph,S_i,t})$. Meanwhile, historical values of check-out and check-in in each bike-sharing station $y^i = (y_1^i, y_2^i, ..., y_T^i)$ are also given. Generally, we learn a nonlinear mapping function by using the historical context features X^i and its corresponding target value y^i to obtain the predicted value $\tilde{y^i}_{T+1}$ for check-out/in respectively with the following formulation:

$$\tilde{y_{T+1}^i} = \mathscr{F}(X^i, y^i) \tag{1}$$

where mapping $\mathscr{F}(\cdot)$ is the nonliner mapping function we take for making prediction.

Methodology

In this section, the STG2Vec model for heterogeneous spatial-temporal graph embedding and the CE-LSTM for collaborative temporal modeling we proposed will be introduced in details. In general, together with multi-source information from weather, time, geographical position and historical transition patterns, the representation learned by STG2Vec can be fed into the LSTMs for temporal modeling by the CE-LSTM. Specifically, we will provide details for our proposed core model: STG2Vec in components of event-flow serialize method and dynamic attention-based graph embedding model respectively.

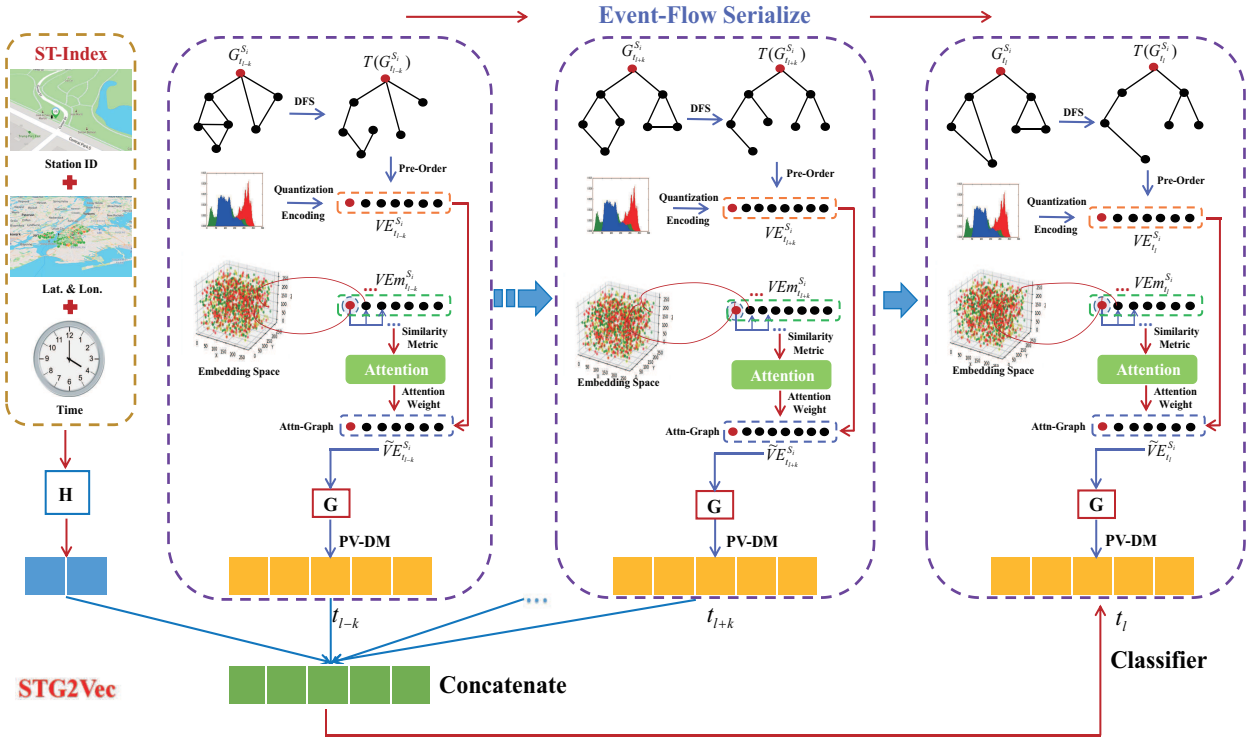

Figure 2: Graphical illustration of learning representation of attention-based heterogeneous spatial-temporal graph. The figure is composed of two parts. The top displays the process of event-flow serialize, and the bottom structure of STG2Vec. Event-flow serialize is the process of encoding the dynamic heterogeneous graph into a special language pattern such as sequence of words in a corpus. The STG2Vec takes dynamic attention-based graph embedding by event-flow series with the spatial-temporal index.

Mutil-Source Information Representation

Bike-sharing demand in station-level is affected by multiple complex factors, such as its geographical position and historical transition patterns, meteorology, time and correlation between its neighbors.

Firstly, we extracted the feature of transition $f_{trans,S_i,t}$ in ST-index with statistical indicators such as total, mean, variance, median, mode, minimum, and maximum in each time interval. Secondly, we defined time features for each time in station S_i as $f_{time,t}$: rush or normal time of the day, day of the week, week of the month, month of the season. Meanwhile, as a kind of transportation, bike-sharing demand is affected by meteorology significantly. Thirdly, we define the meteorology feature in time t as $f_{wea,t}$, which contains air temperature, dew point temperature, relative humidity, wind speed, wind direction, visibility and weather condition type. In addition, the bike-sharing traffic of nearby stations can affect each other. Finally, to utilize this correlation between stations, we take embedding of the graph in ST-index by the STG2Vec we proposed.

Specifically, most attributes of meteorology and time features are categorical variables with sparse one-hot encoding. In order to achieve a dense representation of joint information, we transform the time-weather attributes with time-index into a low-dimensional vector by neural networks similar to sentence embedding (Le and Mikolov 2014).

Event-Flow Serialize

The event of bike-sharing checking may occur at any time, which makes the graph structure changing dynamically. The goal of language modeling is to estimate the likelihood of a specific sequence of words appearing in a corpus (Perozzi, Al-Rfou, and Skiena 2014). Event-flow serialize is the process to encode the evolution of dynamic heterogeneous graph into a special language pattern such as word sequence in a corpus. Firstly, we divided a time interval (1h) by minutes. For instance, time interval $t = (t_1, t_2, ..., t_l)$, where $l \in [1, L]$. Secondly, we traversed each unicom undirected sub-graph in Depth-First Search (DFS) and corresponding Depth-First Spanning Tree (DFST) can be obtained reseparately. For instance, in t and S_i, the DFST: $T(G_{tl}^{S_i})$ generated by G_t^i can be traversed in Pre-order into a node set $VE_{t_l}^{S_i}$. It should be noted that each node can be represented as a vector which consists of four parts: real-time net inflow, outflow, longitude and latitude. For instance, the node vector can be symbolized as $n_{t_l}^{S_i} = (I_{S_i,t_l}, O_{S_i,t_l}, lon.S_i, lat.S_i)$.Meanwhile, we took hierarchical quantization encoding with inflow and outflow in each node to build a corpus with a reasonable frequency distribution. Then, the event-flow series in t and S_i can be represented as $VE_t^{S_i} = (VE_{t_1}^{S_i}, ..., VE_{t_l}^{S_i}, ..., VE_{t_L}^{S_i})$. The offline corpus was established with $VE_t^{S_i}$ in different ST-Index and

a new set of nodes can be find in the corpus by a hierarchical search which consists of length, location and quantitative level matching. Finally, the propose of event-flow serialize is to finish the information extraction of the evolution process of dynamic graph that is latent and spatially sensitive.

Dynamic Attention-based Graph Embedding

Graphical illustration of STG2Vec is given in Figure 2. In STG2Vec, outlined in Algoithm 1, the position information in t is mapped to a unique vector as the spatial-temporal index (ST-Index), represented by a column in matrix H and each set of nodes in corresponding graph within event-flow series can be seen as a special word is also mapped to a unique vector, represented by a column in matrix G. The ST-Index and the contextual special words are concatenated to predict the next word in fixed-length surroundings sampled from a sliding window over a event-flow series. Specifically, given a general spatial-temporal graph contextual series $VE_{t_1}^{S_i}, ..., VE_{t_l}^{S_i}, ..., VE_{t_L}^{S_i}$, the objective of STG2Vec is to maximize the average log probability

$$\frac{1}{L}\sum_{t=k}^{L-k} \log p(VE_{t_l}^{S_i} | VE_{t_{l-k}}^{S_i}, ..., VE_{t_{l+k}}^{S_i}) \tag{2}$$

The prediction task is typically done via a multiclass classifier, such as softmax. There, we have

$$p(VE_{t_l}^{S_i} | VE_{t_{l-k}}^{S_i}, ..., VE_{t_{l+k}}^{S_i}) = \frac{e^{y_{VE_{t-k}^{S_i}}}}{\sum_j e^{y_j}} \tag{3}$$

Each of y_j is un-normalized log-probability for each output intermediate graph j, computed as

$$y = b + Uh(VE_{t_{l-k}}^{S_i}, ..., VE_{t_{l+k}}^{S_i}; G) \tag{4}$$

where U, b are the softmax parameters. h is constructed by a concatenation of intermediate graph vectors extracted from G and the ST-Index vector extracted from H. In addition, we take stochastic gradient decent (SGD) to train the STG2Vec and the gradient obtained by backpropagation can be used to update parameters in our model.

Furthermore, to obtain an importance-awareness vectorized representation of the event-flow, we use the attention mechanism to take importance-based sampling for the sequence of nodes encoded by the event-flow serialize and train STG2Vec again with the new input of sampled nodes series. In addition, after training of STG2vec, the fixed-length vector representation formed by each node corresponding to different ST-Index can constitute an embedding space. Then we use the ST-Index as the key, and the corresponding vector represents the value to construct a hash map. Specifically, for an instance, each node in the set of node $VE_{t_l}^{S_i}$ can obtained the corresponding fixed-length vector representation by the hash map and these representations can form a set of embedded representation $VEm_{t_l}^{S_i}$. Furthermore, we can get normalized attention weights by measuring the similarity of the length-fixed vector corresponding to each node one by one. Finally, the attention-based graph $\hat{VE}_{t_l}^{S_i}$ can be produced by importance-based sampling from $VE_{t_l}^{S_i}$.

Algorithm 1 STG2Vec (G, H, ω, d, τ, L)

Require:
G: Event-flow series matrix, H: ST-Index matrix, ω: window size, d: embedding size, τ: training epochs, L: event-flow series length
Ensure:
$f_{graph,S_i,t} \in \mathbb{R}^d$: Embedding of graph in ST-index
while $iter = 1 < \tau$ **do**
  Initialization: Sample Θ and Φ from G, H
  for $VE_t^{S_i} \in \Theta$ **do**
    $VE_t^{S_i} \leftarrow (VE_{t_1}^{S_i}, ..., VE_{t_l}^{S_i}, ..., VE_{t_L}^{S_i})$
    for $VE_{t_l}^{S_i} \in VE_t^{S_i}$ **do**
      $\alpha_{t_l}^{S_i} \leftarrow CalAttnWeights(VE_{t_l}^{S_i})$
      $VE_{t_l}^{S_i} \leftarrow \hat{VE}_{t_l}^{S_i} \leftarrow \alpha_{t_l}^{S_i} \cdot VE_{t_l}^{S_i}$
    end for
  end for
  PV-DM(Θ, Φ, ω, d) (Le and Mikolov 2014)
end while

Collaborative Temporal Modeling

To better utilize the external data and capture with complex non-linear spatial-temporal relations, we proposed the CE-LSTM. Together with multi-source information from weather, time, geographical position and historical transition patterns, the representation learned by STG2Vec can be fed into the LSTMs for collaborative temporal modeling.

Firstly, we define the jointly representation by concatenating in each time interval and station as

$$X_t^i = (f_{trans,S_i,t}, F_{wea,t}, f_{time,t}, f_{graph,S_i,t}) \tag{5}$$

Then, $X^i = (X_1^i, X_2^i, ..., X_T^i)$ is fed into LSTM networks. Furthermore, we can learn the nonliner mapping function by these formulation (Hochreiter and Schmidhuber 1997) of the calculating process in LSTM cells as follows:

$$i^t = \sigma(W_{xi}X_t^i + W_{hi}h^{t-1} + W_{ci}c^{t-1} + b_i) \tag{6}$$

$$f^t = \sigma(W_{xf}X_t^i + W_{hf}h^{t-1} + W_{cf}c^{t-1} + b_f) \tag{7}$$

$$c^t = f^t c^{t-1} + i^t \tanh(W_{xc}X_t^i + W_{hc}h^{t-1} + b_c) \tag{8}$$

$$o^t = \sigma(W_{xo}X_t^i + W_{ho}h^{t-1} + W_{co}c^{t-1} + b_o) \tag{9}$$

$$h^t = o^t \tanh(c^t) \tag{10}$$

where $\sigma(\cdot)$ represents the activation function of sigmoid and W matrices with double subscript the connection weights between the two cells. In addition, i^t represents input gate state, f^t forget gate state, c^t cell state, o^t output gate and h^t the hidden layer output in current time-step. Finally, we can take the last element of output vector h^{t-1} as the predicted value. It can be represented as:

$$\tilde{y}^{i,t} = h^{t-1} \tag{11}$$

the final output value can be contacted to a vector:

$$\tilde{y_{T+1}^i} = (\tilde{y}^{i,1}, \tilde{y}^{i,2}, ..., \tilde{y}^{i,T+1}) \tag{12}$$

Experiments

In this section, we will make a data description firstly. Then the baseline methods for comparison, evaluation metric and parameter settings will be introduced as well. Furthermore, to evaluate the performance of the proposed model, we conducted experiments on a realworld dataset, compared with several baseline models.

Data Description and Settings

To evaluate our model, we collected two datasets, *i.e.*, Citi-Bike Dataset and MesoWest Dataset from NYC and the details of them are shown in Table 2. For bike data, the stations with number of trip records less than 1,000 in our time span are filtered out. This is a common practice used in similar works (Yao et al. 2018). Because in the real-world applications, it's not very meaningful to predict such a low-demand station. In our experiment, we set an hour as the length of the time interval and split datasets in station-level. In addition, there are 12,281 hours is available and 9,825 samples selected randomly are used for training and the remaining 2,456 samples are used for testing. Furthermore, when testing the prediction result, we use the previous 12 time intervals (*i.e.*, 12 hours) to predict the bike-sharing demand in the next time interval for each station.

- **Citi-Bike Dataset**[1]**:** We collect the trip data of Citi-Bike system in NYC, from 2017/1/1-2018/5/31 (UTC) as our dataset. The data includes: origin station (station ID, station name, station latitude and longitude), destination station (station ID, station name, station latitude and longitude), start time (when a bike is checked out), stop time (when a bike is checked in).
- **MesoWest Dataset**[2]**:** MesoWest is an ongoing cooperative project to provide access to current and archive weather observations across the United States. The data are recorded by a station located near to Central Park containing air temperature, dew point temperature, relative humidity, wind speed, wind direction, visibility and weather condition type.

Table 2: Details of Bike-sharing and Meteorology Datasets

Time Span (UTC)		2017/1/1-2018/5/31
Data Sources	Category	Attribute
Citi-Bike	Stations(In)	44
	Stations(Out)	47
	Bikes	1,345
	Records	402,340
Meteorology	TEMP / °F	[5.00,93.92]
	DEWP / °F	[-14.08,77.00]
	HR	[11.91%,100%]
	WSP / mph	[0.00,26.46]
	WD	[0°,360°]
	VISIB / miles	[0,10]
	Weather	Sunny, etc.

[1]https://www.citibikenyc.com
[2]https://mesowest.utah.edu

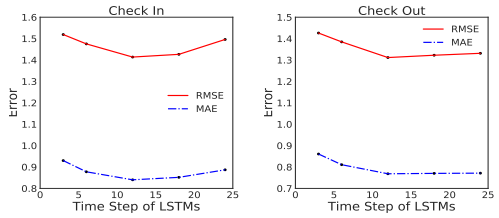

Figure 3: Parameter Sensitivity of Time Steps: S Over Tasks.

Evaluation Metric

Two commonly used metrics: the Root Mean Squared Errors (RMSE) and Mean Absolute Errors (MAE) are adopted to evaluate the performance of all compared models as follows:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\tilde{y}_t^i - y_t^i)^2} \tag{13}$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\tilde{y}_t^i - y_t^i| \tag{14}$$

where $\tilde{y}_t^i$ is prediction, y_t^i is real value and N is the number of testing samples.

Comparing Methods

For fairness, we use the same contex features and loss funtion for all models. We carefully tuned each model respectively and tested for five times to reduce random errors and the final averaged results are showed in Table 3. The baseline models compared with our proposed method are as follows.

- **Temporal:** We only take the context feature of transition $f_{trans,S_i,t}$ in ST-index with statistical indicators and conduct temporal modeling with LSTMs.
- **Weather:** It utilizes the joint information of time-weather represented in low-dimensional embedded vector.
- **Graph:** This considers the correlation between stations learned by the STG2Vec without attention mechanism.
- **Attn-Graph:** This variant contains the importance-awareness vectorized representation learned by STG2Vec.
- **CE-LSTM:** Together with temporal and weather information, the importance-awareness vectorized representation learned by STG2Vec is also employed for temporal modeling with LSTMs.

Parameters Setting

There are some parameters in STG2Vec, *i.e.*, embedding dimension d, sampling window size ω and epochs τ. Taking into account efficiency and performance, the setting is: $d = 3, \omega = 10, \tau = 50$. In addition, we transform time-weather attributes into three-dimensional vector by sentence embedding with default setting in Gensim (3.4.0). Furthermore, we take a single-layered LSTM with size of hidden units: $h = 64$, batchsize $b = 256$ and time steps $S = 12$ which confirmed by grid search and showed in Figure3 partially. The first 80% of training samples were selected for training the remaining for parameters tuning.

Table 3: Comparison with Different Variants and Baseline Methods

Model	Overall Performance of Bike-sharing Demand Prediction			
	Check In		Check Out	
	RMSE	MAE	RMSE	MAE
HA	1.7325	0.9105	1.6400	0.8561
Lasso	1.6488	1.0054	1.5685	0.9549
KNN	1.4803	0.8435	1.3765	0.7707
RF	1.5660	0.9433	1.4780	0.8742
GBRT	1.4978	0.8984	1.4035	0.8277
RNN	1.4345	0.8571	1.3267	0.7790
GRU	1.4239	0.8427	1.3313	0.7713
Temporal	1.4960	0.8869	1.3731	0.8156
Temporal + Weather	1.4884	0.8800	1.3723	0.8090
Temporal + Graph	1.4265	0.8517	1.3223	0.7702
Temporal + Attn-Graph	1.4195	0.8431	1.3173	0.7705
Temporal + Weather + Graph	1.4189	0.8428	1.3204	0.7739
Spectral Embedding	1.4449	0.8601	1.3393	0.7807
DeepWalk	1.4337	0.8444	1.3356	0.7876
DNGR	1.4327	0.8511	1.3327	0.7802
CE-LSTM	**1.4138**	**0.8402**	**1.3115**	**0.7684**

Performance Comparison

Table 3 shows the average performance of the proposed method compared to other baseline competitors. As we can see, the HA perform poorly because only values of previous demands in the the same time of the day are used. Generally, with the collaborative representation learned from multi-source and heterogeneous spatial-temporal data, even the Lasso with l_1-norm regularization can improves the accuracy of the prediction. Usually, the collaborative representations corresponding to similar predicted target values are similar, it makes the K-NN regression training based on distance metrics can achieve considerable performance. Besides, GBRT and RF are tree-based methods widely used in time series prediction. The GBRT model, with the characteristics of low deviation and high variance with respect to RF, performs better in specific experiments. For deep learning methods which can capture the dependency of correlated relationship within time steps, the CE-LSTM achieved state-of-the-art performance comparing to RNN, GRU and others. We can see that the LSTMs is more conducive to take temporal modeling with collaborative representations learned from multi-source and heterogeneous spatio-temporal data.

Furthermore, we verified the performance of different variants and showed results in Table 3. We can see that LSTMs has a poor performance by only taking the context feature of transition. Meanwhile, the introduction of spatio-temporal graph into temporal representation can make a greater contribution to the improvement of predictive performance than weather information. In addition, the experimental results prove that the dynamic attention-based graph embedding can outperform graph embedding without attention mechanism significantly. Ultimately, the CE-LSTM, modeling by temporal representation, attention-based dynamic graph embedding and weather information, achieved the best performance. We can draw the conclusion that learning importance-aware vectorization representation by STG2Vec makes it possible to successfully mine the correlations between stations and to further enrich the connotation of collaborative representation in demand prediction tasks.

Additionally, we use Spectral Embedding, DeepWalk (Perozzi, Al-Rfou, and Skiena 2014) and DNGR (Cao, Lu, and Xu 2016) to learn statical graph embedding for each time step and the representation of nodes can be obtained with the same dimensions d respectively. We only replace the dynamic graph node representation learned by STG2Vec with the node vector obtained by the above statical graph embedding methods in CE-LSTM for performance comparison. Experimental results showed that considering only statical spatial dependence makes a limited improvement if there is a lack of mining the information contained in the evolution of the dynamic graph.

Conclusion

This paper proposed a novel model named STG2Vec to learn the representation from heterogeneous spatial-temporal graph. Specifically, an event-flow serialize method and a dynamic attention-based graph embedding model are proposed for obtaining an importance-awareness vectorized representation from heterogeneous spatial-temporal graph. Additionally, together with multi-source information from weather, time, geographical position and historical transition patterns, the representation learned by STG2Vec can be fed into the LSTMs for temporal modeling for bike-sharing demand predcition. The experimental results show that the proposed method achieved competitive performance comparing to baseline models. For future work, we will further optimize the connection method between representations for adapting to more external information introduction.

Ackowledgments

This work was jointly sponsored by the National Key Research and Development of China (No.2016YFB0800404) and the National Natural Science Foundation of China (No.61572068, No.61532005) and the Fundamental Research Funds for the Central Universities of China (No.2018YJS032).

References

Bao, J.; He, T.; Ruan, S.; Li, Y.; and Zheng, Y. 2017. Planning bike lanes based on sharing-bikes' trajectories. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017*, 1377–1386.

Box, G. E. P., and Pierce, D. 1968. Distribution of residual autocorrelations in autoregressive-integrated moving average time series models. *Publications of the American Statistical Association* 65(332):1509–1526.

Cao, S.; Lu, W.; and Xu, Q. 2016. Deep neural networks for learning graph representations. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, 2016*, 1145–1152.

Chen, L.; Zhang, D.; Wang, L.; Yang, D.; Ma, X.; Li, S.; Wu, Z.; Pan, G.; Nguyen, T. M. T.; and Jakubowicz, J. 2016. Dynamic cluster-based over-demand prediction in bike sharing systems. In *UbiComp 2016*, 841–852.

Chen, P.; Hsieh, H.; Sigalingging, X. K.; Chen, Y.; and Leu, J. 2017. Prediction of station level demand in a bike sharing system using recurrent neural networks. In *VTC 2017*, 1–5.

Cho, K.; van Merrienboer, B.; Bahdanau, D.; and Bengio, Y. 2014a. On the properties of neural machine translation: Encoder-decoder approaches. In *EMNLP*, 103–111.

Cho, K.; van Merrienboer, B.; Gülçehre, Ç.; Bahdanau, D.; Bougares, F.; Schwenk, H.; and Bengio, Y. 2014b. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *EMNLP*, 1724–1734.

Dai, H.; Dai, B.; and Song, L. 2016. Discriminative embeddings of latent variable models for structured data. In *ICML 2016*, 2702–2711.

Davoian, K., and Lippe, W. 2007. Time series prediction with parallel evolutionary artificial neural networks. In *ICDM 2007*, 10–15.

Drucker, H.; Burges, C. J. C.; Kaufman, L.; Smola, A. J.; and Vapnik, V. 1996. Support vector regression machines. In *NIPS, 1996*, 155–161.

Goyal, P.; Kamra, N.; He, X.; and Liu, Y. 2018. Dyngem: Deep embedding method for dynamic graphs. *arXiv: 1805.11273*.

Grover, A., and Leskovec, J. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016*, 855–864.

Guo, C., and Berkhahn, F. 2016. Entity embeddings of categorical variables. *arXiv: 1604.06737*.

He, T.; Bao, J.; Li, R.; Ruan, S.; Li, Y.; Tian, C.; and Zheng, Y. 2018. Detecting vehicle illegal parking events using sharing bikes' trajectories. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018*, 340–349.

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

Hulot, P.; Aloise, D.; and Jena, S. D. 2018. Towards station-level demand prediction for effective rebalancing in bike-sharing systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018*, 378–386.

Johansson, U.; Boström, H.; Löfström, T.; and Linusson, H. 2014. Regression conformal prediction with random forests. *Machine Learning* 97(1-2):155–176.

Le, Q. V., and Mikolov, T. 2014. Distributed representations of sentences and documents. In *ICML 2014*, 1188–1196.

Li, X., and Bai, R. 2016. Freight vehicle travel time prediction using gradient boosting regression tree. In *ICMLA 2016*, 1010–1015.

Liang, Y.; Ke, S.; Zhang, J.; Yi, X.; and Zheng, Y. 2018. Geoman: Multi-level attention networks for geo-sensory time series prediction. In *IJCAI 2018*, 3428–3434.

Lin, L.; He, Z.; Peeta, S.; and Wen, X. 2017. Predicting station-level hourly demands in a large-scale bike-sharing network: A graph convolutional neural network approach. *arXiv: 1712.04997*.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS 2013*, 3111–3119.

Perozzi, B.; Al-Rfou, R.; and Skiena, S. 2014. Deepwalk: online learning of social representations. In *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2014*, 701–710.

Rumelhart, D. E.; Hinton, G. E.; and Williams, R. J. 1986. Learning representations by back-propagating errors. *Nature* 323(6088):533–536.

Wang, Y., and Chaib-draa, B. 2013. A KNN based kalman filter gaussian process regression. In *IJCAI 2013*, 1771–1777.

Yan, Y.; Tao, Y.; Xu, J.; Ren, S.; and Lin, H. 2018. Visual analytics of bike-sharing data based on tensor factorization. *J. Visualization* 21(3):495–509.

Yao, H.; Wu, F.; Ke, J.; Tang, X.; Jia, Y.; Lu, S.; Gong, P.; Ye, J.; and Li, Z. 2018. Deep multi-view spatial-temporal network for taxi demand prediction. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, 2018*, 2588–2595.

Yule, G. U. 1927. On a method of investigating periodicities in disturbed series, with special reference to wolfer's sunspot numbers. *Philosophical Transactions of the Royal Society of London* 226(226):267–298.

Zhu, Z.; Xin, P.; Wei, S.; and Zhao, Y. 2013. Orthogonal graph-regularized matrix factorization and its application for recommendation. In *ICME 2013*, 1–6.