A Novel Fusion Framework for Sequential Data using Pre-trained Model

Tao Ruan, Canghong Jin, Lei Xu, Jianchao Ding, Shengyu Ying, Minghui Wu, Huanqiang Li

Abstract— **Sequential data is widely used in several fields, such as power payload prediction, traffic flow prediction, and stock trend prediction. Driven by the urgent needs, sequential forecasting based on deep learning methods has received lots of attention in recent years. However, the potential of deep learning methods in sequential data forecasting has not yet fully been exploited in terms of model architecture. In this study, a pre-trained nodal model with multiple fusion layers architecture (Sequence Prediction via Node Fusion, SPNF) was proposed, the model considered both connections of nodes and the temporal components of nodes to predict value of next node. Multiple fusion layers were adopted to capture spatial features and temporal dependencies from historical data. The proposed model also conpensated missing data via a masking mechanism. To validate the proposed model, experiments were also carried out using field-captured traffic data, the performance of proposed model were compared with classical and state-of-the-art models, such as, ARIMA, SVR, LSTM, TGC-LSTM neural networks. The results showed that proposed model yields higher accuracy and robustness than others, especially in the case of large sequence changes occurred.**

Index Terms—**Data Fusion, Sequential Data Prediction, Attention Neural Network, Pre-trained Model**

I. Introduction

SEQUENTIAL data forecasting is one of the most challenging tasks in recent years and the technique could be used in several fields, such as, prediction of power payload in the grid in near future, traffic flow status in Intelligent Transportation Systems (ITS) based on current data and the trends.

The sequential prediction is based on the trends of historical data and takes account of the impacts of pertinent factors and changes in data trends on a specific node. Although sequence prediction task like traffic prediction has a long history dating back to decades, several prediction methods based on statistical models or featured based regression model, such as, auto regressive integrated moving average (ARIMA) and support vector regression (SVR) [4] [13] [19], were explored, these methods have difficulties in capturing the high-dimensional and nonlinear characteristics of traffic flow due to lack of computing power or data volume.

The statistical model has existed for a long time in sequence prediction. For example, ARIMA [16]and its large schools [10][11][12] have played a key role by virtue of their validity and interpretability. However, statistical methods rely on a series of restrictive assumptions. They may fail when dealing with complex and highly nonlinear data.

Deep learning method to solve prediction problem by using a deep belief network (DBN) and the top layer uses a (multitasking) regression layer[9]. Moreover, graphical convolution in space-time missions has driven some deep learning studies. For example, [17] introduced the Convolutional Recursive Network (GCRN) to identify the spatial structure and dynamics of structured data sequences. [5] proposed a traffic graph convolution long-term memory neural network (TGC-LSTM).

In recent years, deep-learning based models achieved great success in several fields like image recognition and natural language processing. Therefore, sequence predicting via deep learning model is a novel and promising tool in this era. For example, long short-term memory neural network (LSTM) is implemented in traffic speed prediction [6]. An evolving fuzzy neural network (EFNN) with cluster method and a Gaussian fuzzy membership function are proposed to forecast travel speed for multi-step ahead [18]. Traffic Graph Convolutional Long Short-Term Memory Neural Network (TGC-LSTM) create traffic graph convolution based on the physical network topology and combine with LSTM to improve the performance of prediction [5].

Despite a variety of available deep learning solutions in sequence prediction problem, we found that existing methods still have some issues in sequential data prediction. Some methods predicted values based on historical data series, which only focus on the segment itself but ignore the characteristic of the adjacent segments and the impacts of neighbor nodes on it. EFNN and TGC-LSTM considered context around target segment and integrate context information during the traffic predict process. Both of EFNN and TGC-LSTM confined in the graph according to the real structure of the traffic network. That means the traffic states of two locations far apart from each other in the traffic network should not be influenced by each other in a short time period. Though the spectral graph convolution models [7] can capture features from K-chosen neighbors of a vertex in the graph, is there other features could be learned besides direct connected nodes is still a question to be answered.

To overcome aforementioned issues, a novel sequential prediction model with characteristic of station was proposed. The proposed method is capable to take both spatial and temporal information into account. The motivation was demonstrated in Figure 1. Different from other aforementioned models, the proposed model extracted the characteristics of different nodes in data series, as well as the potential correlations among them. Then, a heterogeneous graph among a set of nodes based on proximity and time series was generated. graph2vec models(such as [7] [2] [14]) was adopted to encode every node in the graph, which contains

Manuscript received December 20, 2019; revised April 07, 2020. This work was supported in part by Zhejiang Communication Investment Group Co. LTD. (Project No.201902) and Zhejiang Department of Transportation (Project No.2020007).

T. Ruan (Corresponding author: taoruan@outlook.com), L. Xu and H. Li are with Zhejiang Institute of Transportation Co. Ltd., Hangzhou, China, 310028; C. Jin and M. Wu are with Zhejiang University City College; D. Ding is with Zhejiang Provincial Public Security Department Expressway Traffic Police Corps and S. Ying is with Zhejiang University;

both geographical and data trend features in the vector. A framework named Sequence Prediction via Node fusion (SPNF) was utilized to integrate time series values with node features vectors, and optimal vectors in multiple transformer layers. By associating the pre-trained node vector, the future value could be predicted data value via recurrent neural network.

The methodology could be summarized as follows:

- Two different selection strategies were designed to build connections among all targeted nodes, including location-based selection and sequence similar selection. Then, we propose a graph neural network model to present node features by nearby nodes, and then fuse node vectors into time series values.
- All the nodes are pre-trained in multiple layers with mask mechanism to generate the best vectors. Our model refers to the structure of BERT and implements additional processes to capture sequence data. All the nodes based on above strategies are pre-trained and fused during the predicting process to improve the prediction accuracy.
- We perform extensive experiments on a field-captured sequence data set: Seattle Loop Dataset. The results showed that SPNF outperforms other statistical methods and Deep learning methods.

According to above previous studies, we design our model(SFDF), which integrate nearby segments and similar segments into the time series values.

In the following of the paper, firstly, the problem were described and theories were presented in Section 2. Then, the architecture of SPNF was given and the two different selection strategies were depicted in Section 3. Next, experiments were carried out by using data captured in real traffic flow, and the comparison between other methods was done in Section 4. At the end, the work of this study was summarized.

II. Preliminary

A. Sequential forecast problem

Sequential predictions for a special node typically use a series of data values with historical time steps as input, which can be represented by vectors $X_t = [x_{t-n}, x_{t-(n-1)}, ..., x_{t-2}, x_{t-1}]$. However, in some cases, such as traffic flow prediction, the speed value at one location p may be influenced by the speeds of nearby locations or even further locations, especially when traffic jam propagates through the traffic network. While, different locations have specific speed change patterns due to their different geographical locations such as tunnels, bridges, and bottlenecks in the road. Therefore, we define our prediction problem is to predict value of target node via graphical network and the node characteristic.

B. Heterogeneous graph embedding

In order to take station information of the road into account, we need to present them by vectors, which could be integrated with speed vectors. In this way, the traffic network can be seen as a graph and stations are nodes in the graph. In this study, there are two different kinds of relationship categories: the physical and the latent relationship, by which, we generate a heterogeneous graph for all the targeted stations.

Generally, heterogeneous graph is consisted of nodes and the relationships among them, both of which play an important role in AI research and applications. However, heterogeneous graphs are symbolic and logical, where numerical machine learning methods could hardly be applied. This disadvantage is one of the most important challenges for the usage of graph. To provide a general paradigm to support computing on knowledge graph, various knowledge graph embedding methods have been proposed, such as TransE [3], TransH [20] and TransR [15].

III. Forecast Model

In this section, the components and the architecture of the proposed SPNF was described in this section. The illustrations of the models in following sub-sections all take the traffic speed prediction as example. There are four key components, including station graph construct strategy, vector representation model, transform model in pre-trained process with station infusion and spatio-temporal prediction.

A. Station graph construct

In this section, we propose two strategies to build the connections among stations in the road. First one is real adjacent station in the traffic network and the second one is traffic similarity station.

For two adjacent stations on a road, it is simple to connect them directly, which means they have a strong influence. Besides physical relationships, in this article, we want to find latent connections between various stations. Since each station has a unique traffic flow variation characteristic, we utilize this feature to find similar stations. We choose *dynamic time warping* (DTW) method to evaluate similarity of two stations. DTW is an algorithm for measuring similarity between two temporal sequences which may vary in time or speed. Given two series $X = (x_1, x_2, ..., x_N)$, and $Y = (y_1, y_2, ..., y_M)$, represented by the sequences of values (or curves represented by the sequences of vertices) $DTW(X,Y) = c_p * (X,Y) = min\{c_p(X,Y), p \in P^{N\times M}\}$

We give the detail procedure of constructing heterogeneous graph of stations in Algorithm 1

B. Station Embedding

We propose a graph embedding model to present nodes and edges in heterogeneous graph, which has been illustrated in KG graph[21]. All the translation-based approaches obey the same principle $h + r \approx t$ and function

$$f_r(h,t) = ||\boldsymbol{h} + \boldsymbol{r} - \boldsymbol{t}||_{\boldsymbol{2}}^{\boldsymbol{2}} = (|h + r - t|)^\top W_r(|h + r - t|) \quad (1)$$

where $|\boldsymbol{h} + \boldsymbol{r} - \boldsymbol{t}| = (|h_1 + r_1 - t_1|, |h_2 + r_2 - t_2|, ..., |h_n + r_n - t_n|)$ and W_r is a relation-specific weight matrix.

In our work, the heterogeneous knowledge graph contains two different relations between two entities: *physical connected* and *traffic similarity*. For simplicity, we could use a translation distance model TransE [3] to exploit the relations as vectors in the same space R^d. Given two embedded entities e_i and e_j connected by r, the $e_i + r \approx e_j$ when (h, r, t)

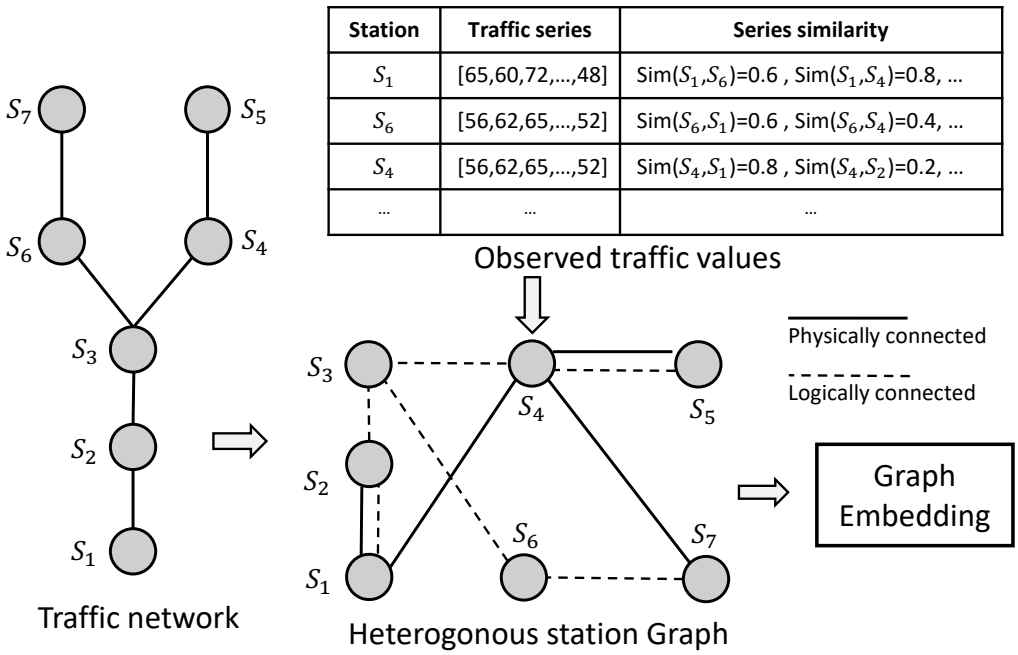


Fig. 1: The relationships among nodes by considering both physical location and sequential characteristics.

Algorithm 1 Station Graph Constructor

Input: a station set $\mathcal{P}$, a traffic network $\mathcal{N}$,related historical speed set $\{\mathcal{V}\}_{i=1}^{N}$
Output: Heterogeneous Graph $G = (V, E)$ Initialize: empty graph $\mathcal{G}$ Connection Type: α, β

for $p \in \mathcal{P}$ **do**
  $G \leftarrow the\ station\ p \in \mathcal{P}$
  for $q \in \mathcal{N}$ **do**
    if q is directly connected with p in $\mathcal{N}$ **then**
      Add $E_{p,q,\alpha} \rightarrow G$
      Add $q \rightarrow G$
    end if
  end for
end for
for $p, q \in \mathcal{P}$ **do**
  average speed vectors every 5 min as $\mathcal{V}_p, \mathcal{V}_q$
  sim = DTW($\mathcal{V}_p, \mathcal{V}_q$)
  if sim θ **then**
    Add $E_{p,q,\beta} \rightarrow G$
    Add $q \rightarrow G$
  end if
end for
return: G

holds. Other more complex methods could be implemented in graph as well, such like TransH [22], TransR [15] or KG2E [8].

C. Attention based pre-trained Model

Since the Masked Language Model(MLM) is strictly more powerful than bidirectional model and single directional model in natural language tasks, we refer similar idea in our study. Similar with MLM in pre-train model, here, we select masked speed with their related stations. We describe attention based model in Figure 2, which contains two significant components: station embedding model and traffic pre-trained model.

Station embedding method is mentioned in above section and here we focus on the attention based trained model. An attention function can be described as mapping a query and a set of key-value pairs to an output, where the query, keys, values, and output are all vectors. In our model, we select k length part of series for station m as input vector T_m. Therefore, a matrix $[T_1, T_2, ..., T_m]$ illustrates the traffic values of station 1 to m in a real networks. In addition, we integrate related station vector S_m by concatenate operation. For each iteration, we mask parts of input vectors, and predict values by rest parts of data. After the training process, we could get related station vectors as output.

In all of our experiments, we set a parameter δ to decide whether the station need to mask. For each time sequences, we replace the i-th token with [MASK] token if the generate random value is larger than δ. Here, refer to ERINE [23] giving a aligning sequence $\{e_1, ..., e_m\}$ to the token sequence $\{w_1, ..., w_n\}$, we select related stations and integrate vectors into traffic speed series. In order to reflect the correlation between mathematical concepts and duplicated entities in math questions, we mask these entities by [MASK] at the same time and predict these masked words by corresponded hidden vectors.

In our training procedure, the strategy of token replacement is the same as BERT model. We replace the chosen station by follow rules: (1) the token [MASK] 80% of the time (2) a random entity 10% of the time. Then, transformer function is used to predict the original speed or station with cross entropy loss.

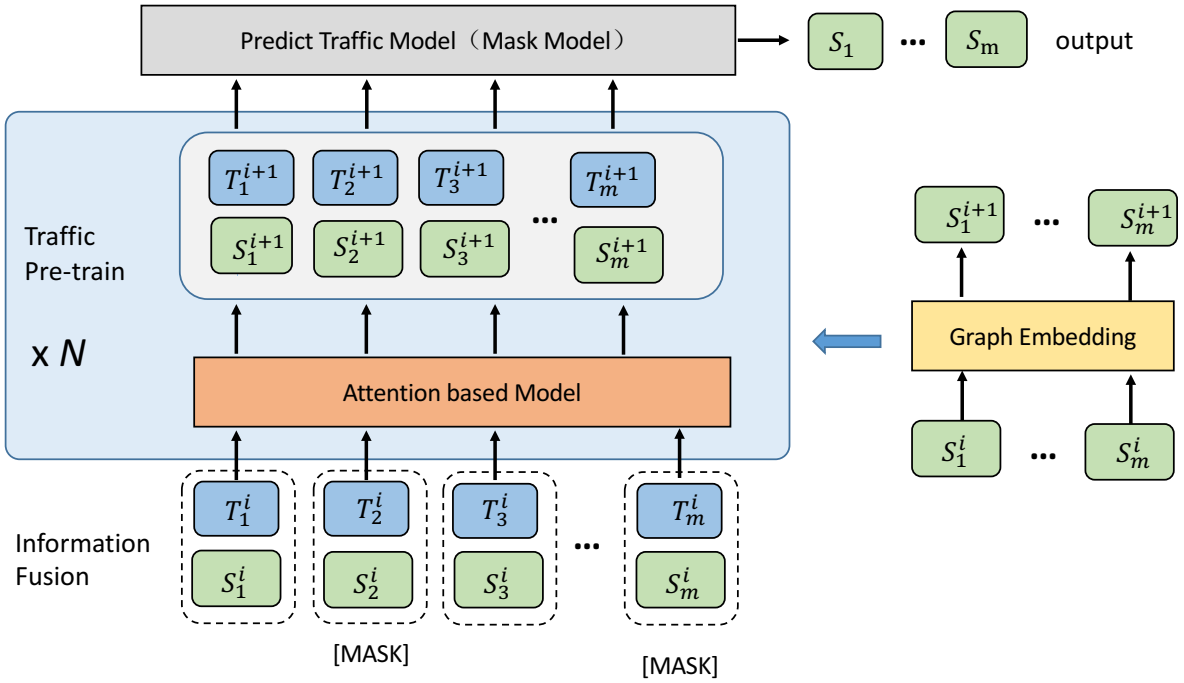


Fig. 2: Attention based Pre-training process

D. Spatio-temporal prediction

After based on pre-trained station vectors, here, we choose RNN to to deal with sequence data. The architecture of RNNs, which also is the basic structure of GRU. We turn to calculate the speed of target station. For a given station, all speed value selected records from historical log are arranged in chronological order for recurrent neural network learning.

$$a_t = b + Wh_{t-1} + Ux_t \tag{2}$$

$$h_t = tanh(a_t) \tag{3}$$

where h_{t-1} is last output of RNN unit, multiple matrix W which are parameters between h_{t-1} and h_t, plus the product of vector U and the input x_t in time t, b are bias vectors for different parts, add them all up and get the a_t. h_t is the output result.

IV. Experiments

The data for our model should satisfy two requirements: the observed points have relationship with others and each point has a relative long time series data. In this article, we use a traffic speed data set to evaluate our model. Of course, other data sets that meet above criteria can be used as well.

A. Dataset Description

Seattle Loop Dataset [1] is collected by the inductive loop detectors deployed on freeways in Seattle area. The freeways contains I-5, I-405, I-90, and SR-520, shown in the above picture. This dataset contains spatio-temporal speed information of the freeway system. In the picture, each blue icon demonstrates loop detectors at a milepost. The speed information at a milepost is averaged from multiple loop detectors on the main lanes in a same direction at the specific milepost. The time interval of the dataset is 5-minute.

The dataset is collected during the whole year in 2015, where the network-wide traffic is characterized by the 323 station speed values and the spatial dimension of the input data is set as, $P = 323$. Since, the unit of a time step in loop detector data is 5 minutes, the dataset has 105120 time steps in total.

In Figure 3, we demonstrate the speed distribution of all traffic network and find that most of traffic values are from 40 to 80 miles. The heatmap represents correlation between stations on the same path are relative higher, but there are still some distant stations that have high correlation.

B. Experimental Settings

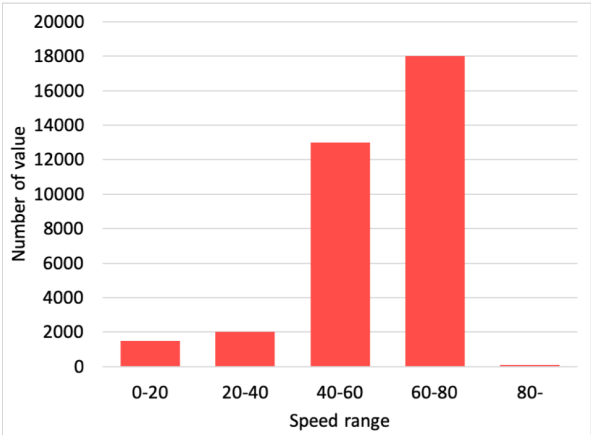
1) Platform: All the experiments are conducted on two environments. First one is a Cloudera platform with 24 physical machines, which is used to do pre-process and generate dataset. The other platform is a Dell server 64-bit system (16 core CPU, each with 2.6GHz, GPU GTX 1080ti, 32G main memory). The algorithms and models in our paper were implemented by Python 3.

2) Comparison Models: To evaluate the accuracy of our model, we compared the proposed model with several most updated methods:

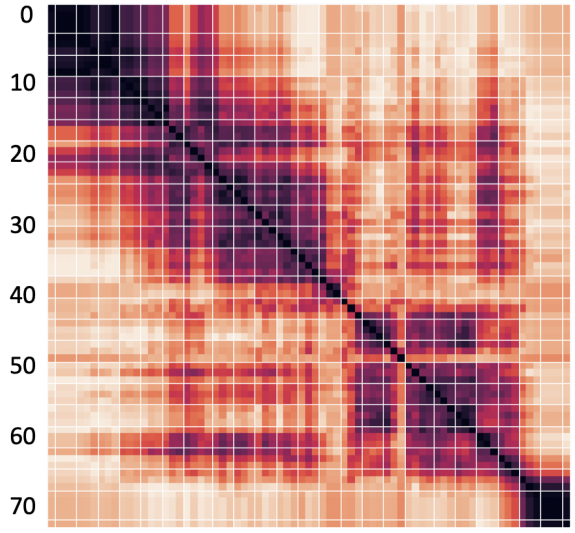
ARIMA is fitted to time series data either to better understand the data and predicts future points in the series (forecasting). Here, we implement by *statsmodels* open source project and *SARIMAX* method.

SVR uses the same principles as the SVM for classification, with only a few minor differences because output is a real number it becomes very difficult to predict the information at hand, which has infinite possibilities. Here, we implement SVR with scikit-learn package.

LSTM a specific type of recurrent neural network, is the most widely-used model in recent years for traffic prediction.



(a) Speed value distribution in dataset



(b) Station similarity correlation value

Fig. 3: Statistic features of dataset

SBU-LSTM is a deep stacked bidirectional and unidirectional LSTM (SBU- LSTM) neural network architecture is proposed, which considers both forward and backward dependencies in time series data, to predict traffic speed. Implement code is from article [6].

TGC-LSTM treats the traffic network as a graph and propose a Traffic Graph Convolutional Long Short-Term Memory Neural Network to learn the interactions between roadways in the traffic network and forecast the network-wide traffic state. We implement with code in article [5]

3) Evaluation Metric: To measure the effectiveness of different traffic speed prediction methods, the Mean Absolute Errors (MAE) and Mean Absolute Percentage Errors (MAPE) are computed using the following equations:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - \hat{x}_i| \tag{4}$$

$$MAPE = \frac{100}{n}\sum_{i=1}^{n}|\frac{x_i - \hat{x}_i}{x_i}| \tag{5}$$

where x_i is the observed traffic speed, and $\hat{x}_i$ is the predicted speed. The forecasting correctness of spatial distribution is also an important index for this comparison as we perform prediction on multiple locations, thus we define an average correlation value (ACV) to measure the ability of spatial distribution forecasting:

$$ACV = \frac{1}{n_t}\sum_{t=1}^{n} Corr(z_{:t}, N_{:t}) \tag{6}$$

where $z_{:t}$ = predicted traffic flow vector at time point t; $N_{:t}$ = actual traffic flow vector; n_t = number of prediction steps.

TABLE I: Results of comparison models on dataset

Methods	Prediction		
	MAE	MAPE	ACV
ARIMA	4.931	9.699%	-0.004
SVR	3.135	8.790%	0.918
LSTM	2.711	6.644%	0.951
SBU-LSTM	2.606	6.528%	0.953
TGC-LSTM	2.595	6.418%	0.955
SPNF	**2.549**	**6.325%**	**0.958**

TABLE II: Results of comparison models in various similarity stations

Methods	low similarity			High similarity		
	MAE	MAPE	ACV	MAE	MAPE	ACV
ARIMA	37.799%	16.636	-0.040	36.969%	16.564	-0.113
SVR	9.388%	3.245	0.925	7.899%	2.838	0.910
LSTM	6.634%	2.569	0.959	6.115%	2.486	0.941
SBU-LSTM	6.457%	2.517	0.960	6.601%	2.594	0.934
TGC-LSTM	6.152%	2.487	0.962	5.845%	2.509	0.943
SPNF	**6.134%**	**2.424**	**0.964**	**5.735%**	**2.489**	**0.967**

C. Experiment Result

Here, we evaluate several traditional and state of the art prediction models in Table I. It is easy to observe that ARIMA has the worst performance in all methods and correlation of predict value series and ground truth value series is unrelated. Deep learning related methods, such as LSTM, SBU-LSTM and TGC-LSTM, have similar performance in MAE and MAPE. Moreover, LSTM with external structure will improve somewhat prediction effectiveness. Our model has superior performances in whole stations.

In terms of effectiveness, we do experiments in two different sub datasets, which describe the similarity of stations by their DWT values. The results show that our model is better than others in both groups, and stations with unique traffic trend features will get more benefit from our attention and pre-trained models.

V. Conclusion

To summarize, a novel fusion framework-SPNF was proposed for traffic flow prediction. In this framework, both temporal and spatial information was taken into account and heterogeneous graph was generated. Also, a pre-trained model with multiple fusion layers architecture was proposed to improve the efficiency. To validate, the field-captured traffic data based experiment was conducted. The results showed that proposed model yields much better accuracy and robustness than commonly used mythologies(ARIMA,SVR,LSTM,etc.), especially in complicated traffic scenario.

Future study will focus on the integration of other related factors into the model, such as road accidents, inclement weather, etc.

REFERENCES

[1] Seattle loop dataset. https://drive.google.com/drive/folders/1XuK0fgI6lmSUzmToyDdHQy8CPunlm5yr?usp=sharing.

[2] Peter W. Battaglia, Jessica B. Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, and Ryan Faulkner. Relational inductive biases, deep learning, and graph networks. 2018.

[3] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, pages 2787–2795, 2013.

[4] Haibo Chen and Susan Grant-Muller. Use of sequential learning for short-term traffic flow forecasting. *Transportation Research Part C: Emerging Technologies*, 9(5):319–336, 2001.

[5] Zhiyong Cui, Kristian Henrickson, Ruimin Ke, and Yinhai Wang. Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting. *arXiv preprint arXiv:1802.07007*, 2018.

[6] Zhiyong Cui, Ruimin Ke, Yinhai Wang, et al. Deep stacked bidirectional and unidirectional lstm recurrent neural network for network-wide traffic speed prediction. In *6th International Workshop on Urban Computing (UrbComp 2017)*, 2016.

[7] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in neural information processing systems*, pages 3844–3852, 2016.

[8] Shizhu He, Kang Liu, Guoliang Ji, and Jun Zhao. Learning to represent knowledge graphs with gaussian embedding. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 623–632. ACM, 2015.

[9] Wenhao Huang, Guojie Song, Haikun Hong, and Kunqing Xie. Deep architecture for traffic flow prediction: Deep belief networks with multitask learning. 2014.

[10] Yiannis Kamarianakis and Poulicos Prastacos. Forecasting traffic flow conditions in an urban network: Comparison of multivariate and univariate approaches. 2003.

[11] Yiannis Kamarianakis and Poulicos Prastacos. Space-time modeling of traffic flow. 2005.

[12] Yiannis Kamarianakis, Wei Shen, and Laura Wynter. Real-time road traffic forecasting using regime-switching space-time models and adaptive lasso. 2012.

[13] S Vasantha Kumar and Lelitha Vanajakshi. Short-term traffic flow prediction using seasonal arima model with limited input data. *European Transport Research Review*, 7(3):21, 2015.

[14] John Boaz Lee, Ryan A. Rossi, Sungchul Kim, Nesreen K. Ahmed, and Eunyee Koh. Attention models in graphs: A survey. 2018.

[15] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. Learning entity and relation embeddings for knowledge graph completion. In *Twenty-ninth AAAI conference on artificial intelligence*, 2015.

[16] Ahmed Mohammed S and COOK ALLEN R. Analysis of freeway traffic time-series data by using box-jenkins techniques. 1979.

[17] Youngjoo Seo, Michaël Defferrard, Pierre Vandergheynst, and Xavier Bresson. Structured sequence modeling with graph convolutional recurrent networks. 2016.

[18] Jinjun Tang, Fang Liu, Yajie Zou, Weibin Zhang, and Yinhai Wang. An improved fuzzy neural network for traffic speed prediction considering periodic characteristic. *IEEE Transactions on Intelligent Transportation Systems*, 18(9):2340–2350, 2017.

[19] PVV Theja and Lelitha Vanajakshi. Short term prediction of traffic parameters using support vector machines technique. In *2010 3rd International Conference on Emerging Trends in Engineering and Technology*, pages 70–75. IEEE, 2010.

[20] Fei Wang and Jimeng Sun. Survey on distance metric learning and dimensionality reduction in data mining. *Data mining and knowledge discovery*, 29(2):534–564, 2015.

[21] Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, 29(12):2724–2743, 2017.

[22] Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. Knowledge graph embedding by translating on hyperplanes. In *Twenty-Eighth AAAI conference on artificial intelligence*, 2014.

[23] Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. Ernie: Enhanced language representation with informative entities. *arXiv preprint arXiv:1905.07129*, 2019.

T Ruan, member of ASCE and NACE, received his Ph.D degree in Civil Engineering from Clemson University, United States in 2016. He is currently a research scientist in Zhejiang Institute of Transportation Co. Ltd., Hangzhou, China. His research mainly focused on artificial intelligence and data mining techniques for road traffic evaluation and safety analysis.

C. Jin received the Ph.D degree in computer science from Zhejiang University, China, in 2015. From 2016, he was an assistant professor of computer science in Zhejiang University City College. His research interest includes the data mining and social network analysis, especially spatio-temporal data mining.

L. Xu received his Master degree in computer science from Zhejiang University, Hangzhou, China in 2015. His research interest includes system architecture, cloud computing, etc.

J. Ding received his Bachelor's degree in traffic management from Zhejiang Police College, Hangzhou, China in 1999. His main research focus is the traffic safety management and implementation of artificial intelligence into management system.

S. Ying is a graduate student at Zhejiang University. His main research focus is data mining and mobile application system analysis.

M. Wu received the Ph.D. degree in 2011 in computer science from Zhejiang University, China. He is currently a professor at Zhejiang University City College. With several academic articles published in reputable journals and conferences, including AAAI, KDD, WWW, etc, he has been rewarded Google Faculty Award three times. His research interests include Mobile Application and Artificial Intelligence.

H. Li received his Ph.D. degree in Civil Engineering from Zhejiang University, Hangzhou, China in 2008. He is currently a Senior Engineer in Zhejiang Institute of Transportation, Hangzhou, China. His research interests include Geotechnical Information System and digital infrastructure, etc.