A Joint Traffic Flow Estimation and Prediction Approach for Urban Networks

1st Ruiyuan Jiang
School of Advanced Technology
Xi'an Jiaotong-Liverpool University
Suzhou, China
Ruiyuan.Jiang20@student.xjtlu.edu.cn

2nd Shangbo Wang
School of Advanced Technology
Xi'an Jiaotong-Liverpool University
Suzhou, China
Shangbo.Wang@xjtlu.edu.cn

3rd Yuli Zhang
School of Advanced Technology
Xi'an Jiaotong-Liverpool University
Suzhou, China
Yuli.Zhang20@student.xjtlu.edu.cn

4th Valerio Selis
Department of Electrical Engineering and Electronics
University of Liverpool
Liverpool, United Kingdom
V.Selis@liverpool.ac.uk

***Abstract*—Classical methods of traffic flow prediction with missing data are generally implemented in two sequential stages: a) imputing the missing data by certain imputation methods such as *k*NN, PPCA based methods etc.; b) using parametric or non-parametric methods to predict the future traffic flow with the completed data. However, the errors generated in missing data imputation stage will be accumulated into prediction stage, and thus will negatively influence the prediction performance when missing rate becomes large. To solve this problem, this paper proposes a Joint Traffic Flow Estimation and Prediction (JTFEP) approach, which considers the missing data as additional unknown network parameters during a deep learning model training process. By updating missing data and the other network parameters via backward propagation, the model training error can generally be evenly distributed across the missing data and future data, thus reducing the error propagation. We conduct extensive experiments for two missing patterns i.e. Completely Missing at Random (CMAR) and Not Missing at Random (NMAR) with various missing rates. The experimental results demonstrate the superiority of JTFEP over existing methods.**

***Index Terms*—JTFEP, traffic flow prediction, missing data, CMAR, NMAR.**

I. INTRODUCTION

Traffic flow prediction plays a vital role in Intelligent Transportation Systems (ITS) and high prediction accuracy is of great importance. However, missing data is inevitable due to equipment failure or data loss during transmission. Most existing traffic prediction models do not consider the missing data problem, except [1]–[4]. Nassiri *et al.* in [1] proposed a model which interpolated the missing points in the data set with the historical data of adjacent detectors. Zhang *et al.* in [2] proposed an improved historical trend method (IHTM) which used the data of neighbor monitors in the same period to interpolate data in time and space. Tensor completion methods such as Tucker Decomposition-based Completion (TDI) and CP Decomposition-based Completion (CP-WOPT) have been shown to have advantages in missing data imputation [3]. Also, Irawati *et al.* put forward a method using enhanced OMP for traffic flow reconstruction by using matrix factorization [4]. The aforementioned traffic prediction methods with missing data are two-stage methods which are generally sequentially implemented via two steps: a) imputing the missing data by certain imputation methods including Local Least Squares (LLS) [5], Probabilistic Principal Component Analysis (PPCA) [6], historical trend method etc.; b) using parametric or non-parametric methods to predict the future traffic flow with the intact data. The two-stage methods have the limitation that when missing rate becomes large, the errors generated in the imputation stage will be accumulated into prediction models, and thus degrading the prediction performance.

To tackle the problems, this paper proposes a Joint Traffic Flow Estimation and Prediction (JTFEP) approach, which considers missing data in the data set as additional model parameters during a deep learning model training process. Hence, missing data can be updated together with the other model parameters like weights and bias in each training epoch in JTFEP. Specifically. The main contributions of this paper are as follows:

(1) We propose a JTFEP method which considers missing data in the original data set as additional model parameters together with weights and bias, in order to update missing data via backward propagation at each iteration. Because of this, the model training error can be generally distributed across the missing data and future data, and thus the error propagation can be reduced.

(2) Extensive experiments have been conducted by using real traffic data provided by Kaohsiung, Taiwan. The results show that JTFEP can demonstrate superiority under various missing rates for two missing patterns: Completely Missing at Random (CMAR) and Not Missing at Random (NMAR).

II. METHODOLOGY

A. Problem Description

In this research, the goal is to predict the traffic flow in a certain period of time based on historical traffic information with missing points. We assume a spatio-temporal setting for traffic flow data throughout this paper. In general, modern spatio-temporal data sets collected from detector networks can be organized as matrix time series. Thus, we define traffic flow data as a two-dimensional matrix X ($N \times T$), where N represents the number of detectors and T represents the sampling time instants.

To demonstrate the superiority of our method in predicting traffic flow based on incomplete historical data, we intentionally remove certain amount of data in original data set as the missing points. From [7], we can know that CMAR and NMAR missing patterns show different distributions of missing points in original data set X. Considering the missing points in the historical data, original data set can be shown as:

$$X = \begin{bmatrix} \mathrm{X}_{1,1} & \mathrm{NA} & \cdots & \mathrm{X}_{1,T-1} & \mathrm{X}_{1,T} \\ \mathrm{X}_{2,1} & \mathrm{X}_{2,2} & \cdots & \mathrm{NA} & \mathrm{X}_{2,T} \\ \mathrm{NA} & \mathrm{X}_{3,2} & \cdots & \mathrm{X}_{3,T-1} & \mathrm{X}_{3,T} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathrm{X}_{N-1,1} & \mathrm{NA} & \cdots & \mathrm{NA} & \mathrm{X}_{N-1,T} \\ \mathrm{X}_{N,1} & \mathrm{X}_{N,2} & \cdots & \mathrm{X}_{N,T-1} & \mathrm{NA} \end{bmatrix} \tag{1}$$

where, *NA* represents the missing points in the historical data.

B. Missing Data Separation and Initialization

JTFEP needs to consider the missing data as additional model parameters during the model training process. Therefore, we need to extract missing data from the original data set before prediction process. Based on the structure of the incomplete measured historical data (as shown in (1)), we separate the original data set into known data set K and missing data set M. The procedure of missing data separation is shown as follows:

1) We preset two data sets K and M which are the same as original data set X.

2) In known data set K, we keep all the known points as they are and all the missing points are replaced by 0. On the contrary, in missing data set M, all the known points are replaced by 0, and we set unknown parameters a_i with random initial values ($i = 1, 2, \cdots, n$) for all the missing points in M, where n represents the number of missing points in original data set. As shown in (2) and (3):

$$K = \begin{bmatrix} \mathrm{X}_{1,1} & 0 & \cdots & \mathrm{X}_{1,T-1} & \mathrm{X}_{1,T} \\ \mathrm{X}_{2,1} & \mathrm{X}_{2,2} & \cdots & 0 & \mathrm{X}_{2,T} \\ 0 & \mathrm{X}_{3,2} & \cdots & \mathrm{X}_{3,T-1} & \mathrm{X}_{3,T} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathrm{X}_{N-1,1} & 0 & \cdots & 0 & \mathrm{X}_{N-1,T} \\ \mathrm{X}_{N,1} & \mathrm{X}_{N,2} & \cdots & \mathrm{X}_{N,T-1} & 0 \end{bmatrix} \tag{2}$$

$$M = \begin{bmatrix} 0 & a_2 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & a_{n-2} & 0 \\ a_1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & a_3 & \cdots & a_{n-1} & 0 \\ 0 & 0 & \cdots & 0 & a_n \end{bmatrix} \tag{3}$$

In JTFEP, we consider the missing data as additional model parameters such as weights and bias in model training. Hence, before training the model, we need to initialize the unknown parameter a_i in the missing data set M. As mentioned, the procedure of missing data separation and initialization can be shown in Algorithm 1.

Algorithm 1 Missing Data Separation and Initialization

Input: Original data set X;
Output: Known data set K; Missing data set M;

Set missing data in X;
Set K=M=X
for i in K, M **do**
  for j in K_i, M_i **do**
    if K_{ij}, M_{ij} is a missing point **then**
      $K_{ij} \leftarrow 0$
      $M_{ij} \leftarrow$unknown parameters a_i
    else
      $K_{ij} \leftarrow K_{ij}$
      $M_{ij} \leftarrow 0$
    end if
  end for
end for
Initialize a_i in M;
return K, M;

C. Methodology

In this study, we use the sum of known data set and missing data set as the input of LSTM and consider the missing data set as additional model parameters together with weights and bias in the model training process. Hence, missing data can be updated in each training epoch so that the error propagation can be reduced and the prediction accuracy will be improved. The structure of JTFEP is shown in Figure 1.

At time t ($t = 1, 2, \cdots, T$), we use the sum of K_t and M_t as the input data of the current time, where K_t represents the *t-th* column of the known data set K (as shown in (2)) and M_t represents the *t-th* column of the missing data set M (as shown in (3)), the expressions of K_t and M_t are shown as below.

$$K_t = [X_{1,t}, 0, X_{3,t}, \cdots, X_{N-2,t}, 0, X_{N,t}] \tag{4}$$

$$M_t = [0, a_i, 0, \cdots, 0, a_{i+1}, 0] \tag{5}$$

We combine the data in the current time and the data in the memory cell as the input data of JTFEP at time *t*, which can be rewritten as X_{ct}, as shown in (6).

$$X_{ct} = [X_t, h_{t-1}] = [K_t + M_t, h_{t-1}] \tag{6}$$

where, h_{t-1} is a $w \times 1$ vector which represents the data in the memory cell of previous time, where *w* represents the number of nodes in the output layer.

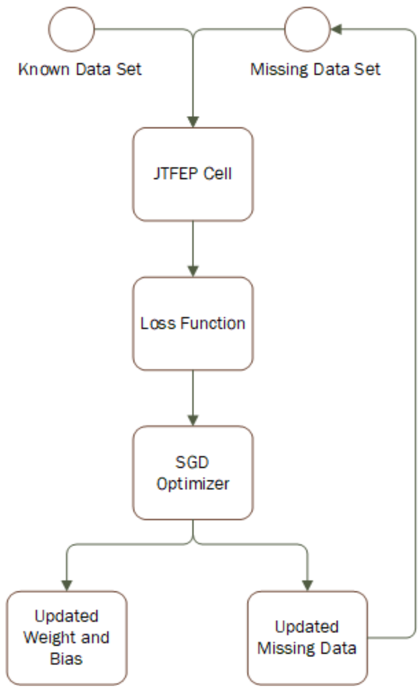

Fig. 1. Structure of JTFEP

Referring to the structure of LSTM, there are four components in the framework of our method which are forget gate, input gate, cell gate and output gate. The specific architecture of a JTFEP unit is shown in Figure 2. h_{t-1} denotes the output at time *t-1*, i_t, f_t, c_t and o_t are input gate, forget gate, cell gate and output gate at time *t*, and h_t denotes the output at time *t*.

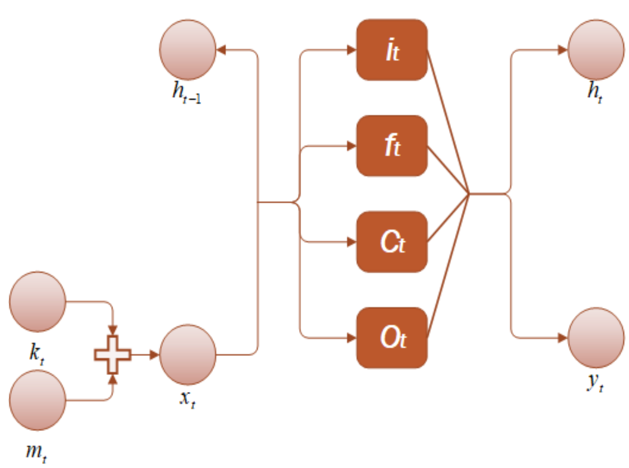

Fig. 2. Specific architecture of a JTFEP unit

Based on the architecture of JTFEP, the specific calculation process of forward propagation is shown below.
In forget gate:

$$f_t = s(W_f \cdot X_{ct} + b_f) \tag{7}$$

where, $s(x)$ is the *Sigmoid* function, f_t is the output of forget gate, W_f is the weight allocated for forget gate, b_f is the bias allocated for forget gate.
In input gate:

$$i_t = s(W_i \cdot X_{ct} + b_i) \tag{8}$$

where, i_t is the output of input gate, W_i is the weight allocated for input gate, b_i is the bias allocated for cell gate.
In cell gate:

$$g_t = tanh(W_g \cdot X_{ct} + b_g) \tag{9}$$

$$s_t = g_t \cdot i_t + s_{t-1} \cdot f_t \tag{10}$$

where, $tanh(x)$ is the *tanh* function, g_t is the output of cell gate, W_g is the weight allocated for cell gate, b_g is the bias allocated for cell gate.
In output gate:

$$o_t = s(W_o \cdot X_{ct} + b_o) \tag{11}$$

$$h_t = s_t \cdot o_t \tag{12}$$

where, o_t is the output of output gate, W_o is the weight allocated for output gate, b_o is the bias allocated for output gate, h_t are predicted data.

After achieving the output value h_t, we set a loss function L to calculate the loss between prediction results and real traffic flow. The expression of loss function is shown in (13).

$$L = \frac{1}{n} \cdot \sum_{i=1}^{n} (h_t(i) - y_p(i))^2 \tag{13}$$

where y_p are real traffic flow data.

Based on the loss value achieving from the forward propagation, we can update missing data set M_t, weights and bias through gradient descent (SGD) optimizer in the backward propagation of JTFEP. The procedure of updating the model parameters is shown below.

1) Partial derivatives of JTFEP model parameters: Based on the loss value achieving from (13), we can calculate the partial derivative of missing data set M_t. Because we regard the input data of JTFEP as the sum of known data set K_t and missing data set M_t, so that the two data sets are linear superposition relationship. That is, the partial derivatives of the two data sets are equal which keep the same as that of X_{ct}. According to the forward propagation of JTFEP, the partial derivative of X_{ct} can be the sum of partial derivatives of prediction loss over X_{ct} in the four gates. Hence, the partial derivative of missing data set M_t can be calculated by (14) and (15).

$$\frac{\partial L}{\partial M_t} = \frac{\partial L}{\partial K_t} = \frac{\partial L}{\partial X_{ct}} \tag{14}$$

$$\frac{\partial L}{\partial X_{ct}} = \frac{\partial L}{\partial f_t} \cdot \frac{\partial f_t}{\partial X_{ct}} + \frac{\partial L}{\partial i_t} \cdot \frac{\partial i_t}{\partial X_{ct}} + \frac{\partial L}{\partial g_t} \cdot \frac{\partial g_t}{\partial X_{ct}} + \frac{\partial L}{\partial o_t} \cdot \frac{\partial o_t}{\partial X_{ct}} \tag{15}$$

The partial derivatives of weights allocated for forget gate, input gate, cell gate and output gate can be calculated through the chain rule of derivative, as shown below:

$$\frac{\partial L}{\partial W_f} = \frac{\partial L}{\partial f_t} \cdot \frac{\partial f_t}{\partial W_f \cdot (K_t + M_t)} \cdot \frac{\partial W_f \cdot (K_t + M_t)}{\partial W_f} \tag{16}$$

$$\frac{\partial L}{\partial W_i} = \frac{\partial L}{\partial i_t} \cdot \frac{\partial i_t}{\partial W_i \cdot (K_t + M_t)} \cdot \frac{\partial W_i \cdot (K_t + M_t)}{\partial W_i} \tag{17}$$

$$\frac{\partial L}{\partial W_g} = \frac{\partial L}{\partial g_t} \cdot \frac{\partial g_t}{\partial W_g \cdot (K_t + M_t)} \cdot \frac{\partial W_g \cdot (K_t + M_t)}{\partial W_g} \tag{18}$$

$$\frac{\partial L}{\partial W_o} = \frac{\partial L}{\partial o_t} \cdot \frac{\partial o_t}{\partial W_o \cdot (K_t + M_t)} \cdot \frac{\partial W_o \cdot (K_t + M_t)}{\partial W_o} \tag{19}$$

Similarly, the partial derivatives of loss over bias allocated for the four gates are shown as follows.

$$\frac{\partial L}{\partial b_f} = \frac{\partial L}{\partial f_t} \cdot \frac{\partial f_t}{\partial b_f} \tag{20}$$

$$\frac{\partial L}{\partial b_i} = \frac{\partial L}{\partial i_t} \cdot \frac{\partial i_t}{\partial b_i} \tag{21}$$

$$\frac{\partial L}{\partial b_g} = \frac{\partial L}{\partial g_t} \cdot \frac{\partial g_t}{\partial b_g} \tag{22}$$

$$\frac{\partial L}{\partial b_o} = \frac{\partial L}{\partial o_t} \cdot \frac{\partial o_t}{\partial b_o} \tag{23}$$

2) Update of model parameters: Based on the partial derivatives of missing data set, weights and bias, we update these model parameters in each training epoch. In this study, we use two different learning rates to update model parameters because missing data set and weights have different orders of magnitude. That is, we allocate one learning rate $l1$ for the update of missing data set and another learning rate $l2$ for the update of weights and bias.

We update M_t through gradient descent method, as shown below:

$$M_{t,i+1} = M_{t,i} - l1 \cdot \frac{\partial L}{\partial M_{t,i}} = M_{t,i} - l1 \cdot \frac{\partial L}{\partial X_{ct,i}} \tag{24}$$

where, $M_{t,i+1}$ represents the updated M_t using in the *(i+1)-th* epoch, M_{ti} is the M_t using in the *i-th* epoch, $X_{ct,i}$ represents the input X_{ct} in the *i-th* epoch.

Also, we update the weights and bias by the following equations:

$$W_{i+1} = W_i - l2 \cdot \frac{\partial L}{\partial W_i} \tag{25}$$

$$b_{i+1} = b_i - l2 \cdot \frac{\partial L}{\partial b_i} \tag{26}$$

where, W_{i+1} and b_{i+1} are weights and bias allocated for each gates in the *(i+1)-th* epoch and W_i and b_i using in the *i-th* epoch.

And then we use the updated model parameters achieved from model training to predict the traffic flow in the future time series. We summarize the procedure of JTFEP in Alogrithm 2.

Algorithm 2 Procedure of JTFEP

Input: Original data set X which contains missing data, sampling from T time instants;
Output: Predicted data, P;

Separate X to known data set K, missing data set M via (2) and (3).
Initialize missing data set M.
Set *epochs* = m.
Set $l1$ = 0.1, $l2$ = 0.01.
Initialize weights and bias allocated for four gates.
for $i = 0$ to m **do**
  for $t = 1$ to T **do**
    X_t = K_t + M_t.
    Get Y_p from forward propagation of JTFEP via (6) to (12).
    Get loss based on Y_p and X_t via (13).
    Calculate partial derivatives of model parameters via (14) to (23).
    Update M_t through via (24), then using the updated M_t in the next training epoch.
  end for
  Update weights and bias via (25) and (26).
end for
Get predicted data P based on updated model parameters.
return P;

III. Experiments

A. Data Description

In this section, we evaluate the prediction performance of the JTFEP model on a real-world data set: Kaohsiung data set. This data set consists of traffic flow data from 33 road detectors in Kaohsiung Taiwan. We select the 33 detectors from six main roads which are Zhong Zheng Road, Kai xuan Road, Ming Quan Road, Ming Zu Road, San Duo Road and Wu Fu Road, The locations of detectors are shown in Figure 3.

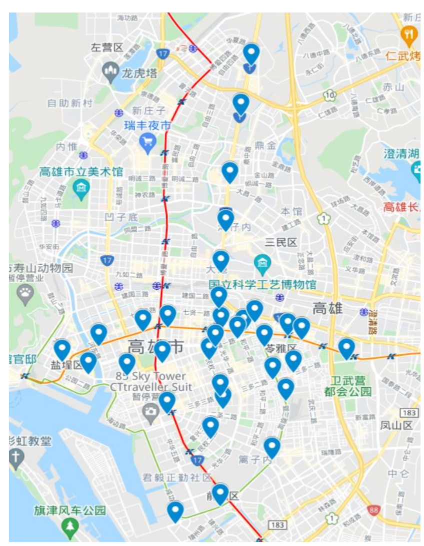

Fig. 3. Locations of selected detectors

Each detector provides the traffic flow data from 10:00-22:00 with a 5-minute resolution (12 time intervals per hour). We organize the raw data set into a time series matrix and set the missing points according to the missing rates and missing patterns. In experiment steps, 80 percent of the data is used as the training set and the remaining 20 percent is used as the test set.

B. Evaluation Metrics

We use two metrics to evaluate the prediction performance of the JTFEP model:

(1) Root Mean Square Error (RMSE):

$$RMSE = \sqrt{\frac{1}{n} \cdot \sum_{i=1}^{n} (y_r(i) - y_p(i))^2} \tag{27}$$

(2) Mean Absolute Error (MAE)

$$MAE = \frac{1}{n} \cdot \sum_{i=1}^{n} |y_r(i) - y_p(i)| \tag{28}$$

where n represents the length of prediction results, y_p are prediction results and y_r are real data.

C. Baseline Methods

We compare the performance of the JTFEP model with the following baseline methods:

(1) K-Nearest Neighbor (*k*NN) [8] with LSTM, which firstly uses *k*NN model to impute the missing points in the original data set, then predicts traffic flow in the future time series with LSTM based on the intact data.

(2) Multiple Imputation by Chained Equations (MICE) Imputation [9] with LSTM. This method is divided into two steps. The first step is iteratively imputing each incomplete variable by regressing on the rest of other covariates. The second step is using LSTM to predict the future traffic flow based on the data after imputation.

(3) Historical Average (HA) [10] with LSTM, which firstly uses the average traffic flow in the historical periods as the interpolated data of missing points in the original data set, then uses LSTM to predict traffic flow based on the data after imputation.

(4) K-Nearest Neighbor (*k*NN) with Support Vector Regression (SVR) [11]. SVR is a model which uses historical data to train the model and obtains the relationship between the input and output, and then predicts the future traffic data by the trained model. This baseline method firstly uses *k*NN to impute the missing data in the original data set, then predicts the future traffic flow according to the interpolated data.

(5) K-Nearest Neighbor (*k*NN) with Autoregressive Integrated Moving Average model (ARIMA) [12], which firstly uses *k*NN model to impute the missing points in the original data set, then uses ARIMA to realize traffic flow prediction based on the intact data.

D. Experimental Results

To evaluate the performance of JTFEP, missing data are intentionally generated with different missing rates that ranges from 0 to 0.4 at every 0.05 increment as usual. Also, we test the prediction accuracy for two missing patterns which are Completely Missing at Random (CMAR) and Not Missing at Random (NMAR) respectively.

(1) *CMAR*

CMAR may occur due to a prolonged physical damage, malfunction of the communication device or temporal detector deployment. For CMAR missing pattern, all the missing data are independently and uniformly distributed over the spatio-temporal domain. In this part, we simply remove a certain amount of observed entries randomly from the observed data set.

We conduct extensive experiments and apply the mean value of RMSE and MAE to compare the prediction performance of these prediction methods, as shown below.

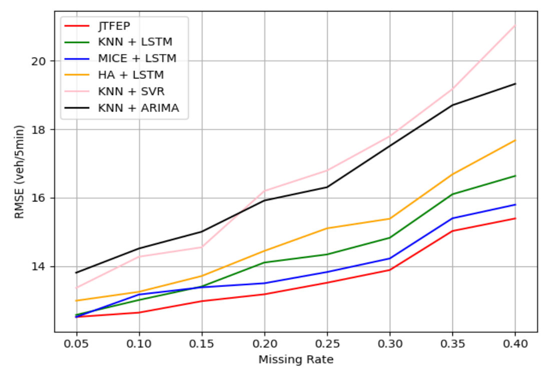

Fig. 4. RMSE of different methods for CMAR missing pattern

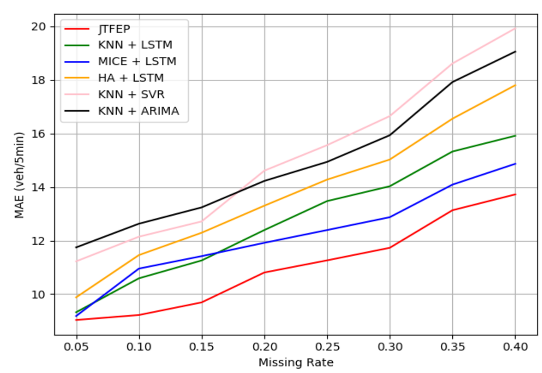

Fig. 5. MAE of different methods for CMAR missing pattern

Figure 5 and Figure 6 show the prediction performance of JTFEP and baseline methods respectively for CMAR missing pattern. As can be seen, the proposed JTFEP clearly outperforms the other methods under all selected missing rates from 0 to 0.4. The results also reveal that both RMSE and MAE increase for all methods with the increase of missing rate while the change amplitude of JTFEP is the smallest, and the superiority of the JTFEP becomes more visible when the missing rate gets larger. Meanwhile, our results suggest that for CMAR missing pattern, JTFEP method inherits the advantages compared with the other baseline methods.

(2) *NMAR*

NMAR is often caused by a long time malfunction of loop detectors [6]. For NMAR missing pattern, the occurrence of missing data is scattered and simultaneous over different roads. That is, in the two-dimensional matrix of traffic flow data, the distribution of missing data is fixed and continuous in both time series and temporal series.

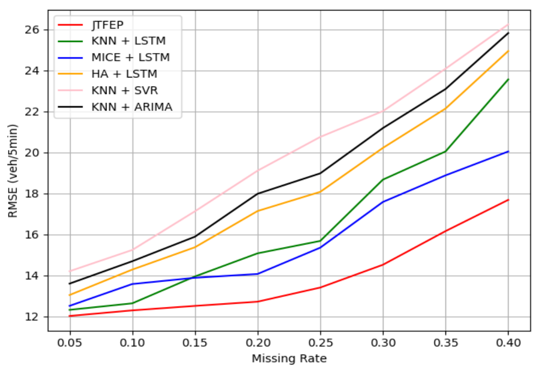

Fig. 6. RMSE of different methods for NMAR pattern

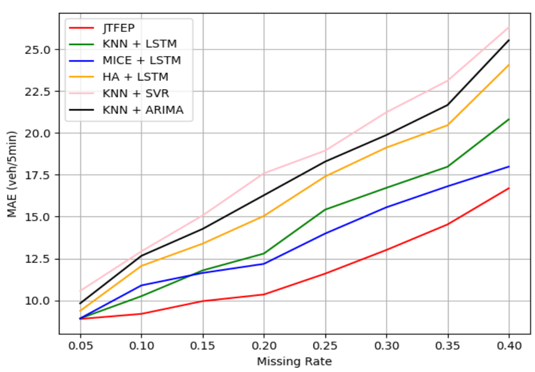

Fig. 7. MAE of different methods for NMAR pattern

From Figure 8 and Figure 9, it can be seen that JTFEP has advantages over the baseline methods basically under all missing rates ranging from 0 to 0.4. Both RMSE and MAE of JTFEP increase slowly with the missing rate from 0.05 to 0.2, but the increase rates of the two metrics become sharper when the missing rate is up to 0.2, and the increase reaches the maximum when the missing rate exceeds 0.35. Basically, JTFEP gets the best prediction performance compared with baseline methods for NMAR missing pattern.

IV. Conclusion

This paper proposes JTFEP for traffic flow prediction with missing data which considers the missing points in the original data as additional model parameters during a deep learning model training process, so that missing data can be updated in each training epoch together with weights and bias. We compare the prediction accuracy of JTFEP for CMAR and NMAR missing patterns with baseline methods: *k*NN with LSTM, MICE Imputation with LSTM, HA with LSTM, *k*NN with SVR and *k*NN with ARIMA. Through extensive experiments, we find that JTFEP has superiority over baseline methods for the two missing patterns, especially when the missing rate is high. Basically, JTFEP performs better than the LSTM based two-stage method or model driven model such as ARIMA under the missing rate ranging from 0 to 0.4 for CMAR and NMAR missing patterns.

Future works can extend the framework to other deep learning models such as Graph Convolution Network (GCN) model and Gated Recurrent Unit (GRU) model, thus the prediction accuracy can be improved when the missing data rate is high.

V. Acknowledgments

Shangbo Wang is the corresponding author of this paper. The work was under support of XJTLU Research Development Funding RDF-21-02-105.

References

[1] V. Nassiri, G. Molenberghs, G. Verbeke, and J. Barbosa–Breda, "Iterative multiple imputation: a framework to determine the number of imputed datasets*," *American Statistician*, pp. 1–25, 2018.

[2] X. Zhang, L. Zhang, Y. Tao, L. Tao, and C. Wang, "Research on interpolation method of missing traffic flow data based on improved historical trend method," in *2020 Eighth International Conference on Advanced Cloud and Big Data (CBD)*, 2020.

[3] Q. Song, H. Ge, J. Caverlee, and X. Hu, "Tensor completion algorithms in big data analytics," *ACM transactions on knowledge discovery from data*, vol. 13, no. 1, pp. 6.1–6.48, 2019.

[4] I. D. Irawati, A. B. Suksmono, and I. Edward, "Enhanced omp for missing traffic reconstruction based on sparse svd," in *2019 26th International Conference on Telecommunications (ICT)*, 2019.

[5] Chang, Gang, Zhang, Yi, Yao, and Danya, "Missing data imputation for traffic flow based on improved local least squares," *Tsinghua Science and Technology*, 2012.

[6] L. Qu, L. Li, Y. Zhang, and J. Hu, "Ppca-based missing data imputation for traffic flow volume: A systematical approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 3, pp. 512–522, 2009.

[7] "Missing data estimation for traffic volume by searching an optimum closed cut in urban networks," 2019.

[8] P. Meesad and K. Hengpraprohm, "Combination of knn-based feature selection and knnbased missing-value imputation of microarray data," in *2008 3rd International Conference on Innovative Computing Information and Control*, 2008, pp. 341–341.

[9] P. Sun, "Mice-da: A mice method with data augmentation for missing data imputation in ieee ichi 2019 dacmi challenge," in *2019 IEEE International Conference on Healthcare Informatics (ICHI)*, 2019, pp. 1–3.

[10] H. Gao, J. Y. Zhao, and L. Jia, "Summary of short-time traffic flow forecasting methods," *Journal of University of Jinan(Science and Technology)*, 2008.

[11] J. Zou, C. Li, Q. Yang, and Q. Li, "Fault prediction method based on svr of improved pso," in *The 27th Chinese Control and Decision Conference (2015 CCDC)*, 2015, pp. 1671–1675.

[12] B. M. Williams and L. A. Hoel, "Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results," *Journal of Transportation Engineering*, vol. 129, no. 6, pp. 664–672, 2003.