Online Metro Origin-Destination Prediction via Heterogeneous Information Aggregation

Lingbo Liu, Yuying Zhu, Guanbin Li, Ziyi Wu, Lei Bai, and Liang Lin, *Senior Member, IEEE*

Abstract—Metro origin-destination prediction is a crucial yet challenging time-series analysis task in intelligent transportation systems, which aims to accurately forecast two specific types of cross-station ridership, i.e., Origin-Destination (OD) one and Destination-Origin (DO) one. However, complete OD matrices of previous time intervals can not be obtained immediately in online metro systems, and conventional methods only used limited information to forecast the future OD and DO ridership separately. In this work, we proposed a novel neural network module termed Heterogeneous Information Aggregation Machine (HIAM), which fully exploits heterogeneous information of historical data (e.g., incomplete OD matrices, unfinished order vectors, and DO matrices) to jointly learn the evolutionary patterns of OD and DO ridership. Specifically, an OD modeling branch estimates the potential destinations of unfinished orders explicitly to complement the information of incomplete OD matrices, while a DO modeling branch takes DO matrices as input to capture the spatial-temporal distribution of DO ridership. Moreover, a Dual Information Transformer is introduced to propagate the mutual information among OD features and DO features for modeling the OD-DO causality and correlation. Based on the proposed HIAM, we develop a unified Seq2Seq network to forecast the future OD and DO ridership simultaneously. Extensive experiments conducted on two large-scale benchmarks demonstrate the effectiveness of our method for online metro origin-destination prediction.

Index Terms—Online Metro System, Time Series Prediction, Origin-Destination Ridership, Heterogeneous Information

1 Introduction

Time series prediction [1], [2] is one of the most active research topics in artificial intelligence. In this work, we pay attention to its practical application in transportation management, e.g., improving the operation efficiency of urban metro systems, since metro has become a popular travel mode for residents. It was reported that over 10 million metro travel transactions are made per day in some metropolises (e.g., Beijing and Shanghai) [3], [4]. Such a huge ridership poses great challenges for metro operation. In this case, accurately forecasting the future ridership is crucial for metro scheduling and route planning.

Due to its significant applications, metro ridership prediction has recently attracted extensive attention in both academic and industrial communities [5], [6], [7], [8], [9]. However, most conventional works were merely proposed for station-level prediction, i.e., forecasting the inflow and outflow of each metro station, as shown in Fig. 1-(b,c). Such information about inflow/outflow ridership is too coarse to reflect the mobility of passengers. To explore more valuable information for metro optimization, we focus on a more challenging task, i.e., metro origin-destination prediction, whose goal is to forecast the ridership between any two stations over the next several time intervals. Specifically, two special types of cross-station ridership are taken into consideration in our work:

- **Origin-Destination (OD) Ridership:** For each station, we aim to forecast the number of passengers entering at time interval t and the stations they will go to. For example, the OD ridership at time interval t can be represented as a matrix[1] $OD_t \in \mathbb{R}^{N\times N}$, where N is the total number of stations. More specifically, $OD_t(i,j)$ denotes the number of passengers that entered station i at time interval t and would head for station j, as shown in Fig. 1-(d).
- **Destination-Origin (DO) Ridership:** We also aim to predict the future outgoing ridership of each station and where these passengers come from. Similarly, the DO ridership at time interval t can be represented as a matrix[1] $DO_t \in \mathbb{R}^{N\times N}$, as shown in Fig. 1-(e). Specifically, $DO_t(i,j)$ is the number of passengers that entered station j at earlier moments and exit from station i at time interval t.

Intuitively, we should utilize the historical OD/DO ridership to forecast the future OD/DO ridership. Unfortunately, in online metro systems, the complete historical OD matrices can not be constructed in real time. One example is illustrated in Fig. 2. Suppose 228 passengers entered station i in the past 15 minutes and 136 people have arrived at their destinations up to now. However, the destinations of the remaining ongoing passengers are unknowable, until they arrive at their exited stations. Under this circumstance, we can only construct an incomplete OD matrix based on the finished trip transactions. When most passengers are still

- *L. Liu, Y. Zhu, G. Li, Z. Wu and L. Lin are with the School of Computer Science and Engineering, Sun Yat-Sen University, China, 510000 (e-mail: liulingb@mail2.sysu.edu.cn; zhuyy76@mail2.sysu.edu.cn; liguanbin@mail.sysu.edu.cn; wuzy39@mail2.sysu.edu.cn; linliang@ieee.org).*
- *L. Lin is also with Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, and with Engineering Research Center for Advanced Computing Engineering Software of Ministry of Education, China.*
- *L. Bai is with the School of Electrical and Information Engineering, the University of Sydney, Australia 2000 (e-mail: lei.bai@sydney.edu.au).*

1. OD matrix and DO matrix may be sparse since the ridership between some stations is usually small or even zero. In this work, we would compress these matrices by merging the small cross-station ridership. More details can be referred to Section 3.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3178184, IEEE Transactions on Pattern Analysis and Machine Intelligence

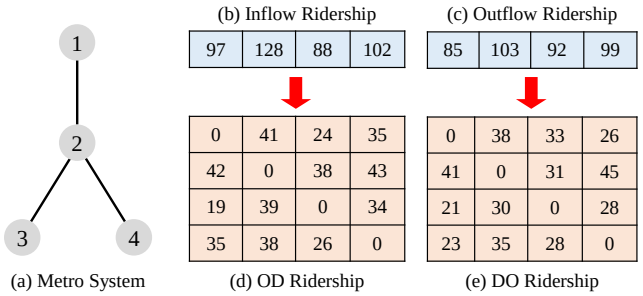


Fig. 1. Illustration of the difference between station-level ridership and origin-destination ridership. (a) is a metro system with four stations. (b) and (c) are the inflow/outflow of each station, respectively. (d) is an Origin-Destination (OD) matrix that represents the destination distribution of incoming passengers. (e) is a Destination-Origin (DO) matrix that represents the origin distribution of outgoing passengers.

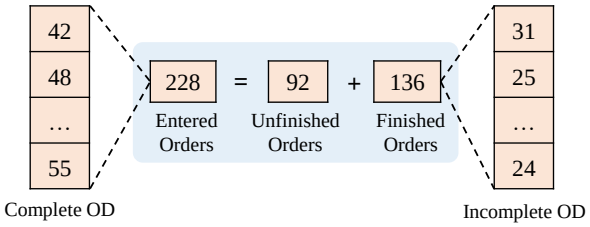


Fig. 2. Illustration of the incomplete OD matrix in online metro systems. Suppose there were 228 passengers that entered station s in the past 15 minutes and 136 people have arrived at their destinations up to now. Unfortunately, the destinations of the remaining people are unknowable. In this case, we can only construct an incomplete OD matrix from finished trip transactions.

on their way to destinations, such an incomplete matrix is very sparse and uninformative, which greatly increases the difficulties of metro origin-destination distribution modeling.

In literature, there are very few methods [10], [11], [12], [13], [14] proposed for online metro origin-destination ridership prediction. Conventional works either take the incomplete historical OD matrices as input or directly utilize DO matrices to forecast the future OD matrices. Despite certain progress, these methods suffer from the following limitations. **First**, these works don't explore the information about unfinished/ongoing trips. Intuitively, human mobilities are usually periodic [15], [16] and we can estimate the potential destinations of those ongoing trips to some extent. Therefore, more information is available for metro OD prediction. **Second**, it is sub-optimal to directly use the historical DO information to toughly forecast the future OD matrices [13], since the former isn't the essential factor affecting the evolution of the latter. **Third**, all above-mentioned methods are unaware of the mutual information between OD and DO ridership, i.e., forecasting the future OD and DO matrices separately. In essence, the previous OD ridership would greatly influence the future DO ridership, which is called OD-to-DO causality in our work. Moreover, there also exists a relationship between the previous DO ridership and the future OD ridership. For example, the DO and OD ridership of tide stations in residential and office areas are usually negatively correlated [17]. Such a relationship is called DO-to-OD correlation. In summary, previous methods only exploit limited information and cannot effectively model the metro ridership distribution.

To tackle the aforementioned problems, we propose a unified neural network module termed Heterogeneous Information Aggregation Machine (HIAM), which fully aggregates heterogeneous information of historical ridership to learn the evolutionary trend of future origin-destination ridership. In particular, our HIAM consists of **i)** two parallel branches respectively for OD and DO modeling, and **ii)** a Dual Information Transformer for OD-DO interaction modeling. Unlike previous works [10], [12] that neglected unfinished transactions, we exploit the information of these transactions explicitly to complement the incomplete OD matrices. Specifically, our OD branch explores the long short-term historical destination distribution to estimate two potential destination matrices of ongoing passengers, which are incorporated with the incomplete OD matrix and fed into graph convolutional gated recurrent units (GCGRU) to generate a compact OD hidden state. Meanwhile, the DO branch feeds the corresponding DO matrix into a GCGRU for DO hidden state generation. To model the internal interaction among OD and DO ridership, our Dual Information Transformer enhances the OD state and DO state mutually by propagating their complementary information in a dual manner. These refined hidden states are respectively fed into the following GCGRU for high-order spatial-temporal representation learning. Based on the tailor-designed HIAM, we develop a unified online metro origin-destination prediction framework with a Seq2Seq architecture [18], which jointly forecasts the OD ridership and DO ridership of the next several time intervals. Finally, we conduct extensive experiments on two large-scale benchmarks (i.e., Shanghai Metro and Hangzhou Metro), and evaluation results show that our approach outperforms existing state-of-the-art methods for both OD prediction and DO prediction.

In summary, the contributions of this work are four-fold:

- We propose a novel Heterogeneous Information Aggregation Machine to facilitate the online metro origin-destination prediction. To the best of our knowledge, our HIAM is the first deep learning approach that fully aggregates heterogeneous information of incomplete OD matrices, unfinished order vectors, and DO matrices to forecast the future cross-station ridership.
- To fully exploit the information of unfinished transactions, our HIAM explores the long short-term historical distribution to estimate the potential destinations of ongoing passengers, which are further utilized to complement the information of incomplete OD matrices.
- A Dual Information Transformer is introduced to propagate the mutual information among OD features and DO features, thus better modeling the internal interaction between OD and DO ridership. To the best of our best knowledge, our work is the first attempt to employ the heterogeneous transformer to address time series forecasting.
- Extensive experiments conducted on two large-scale datasets show the effectiveness of the proposed method for both OD ridership prediction and DO ridership prediction of online metro systems.

The rest of this paper is organized as follows. First, we review some related works of traffic state prediction

and origin-destination prediction in Section 2. We then provide some preliminaries in Section 3 and introduce the proposed approach for online metro prediction in Section 4. Extensive comparison and ablation analysis are conducted in Section 5. Finally, we conclude this paper in Section 6.

2 Related Works

2.1 Traffic Time Series Prediction

Accurate prediction of future traffic states is a crucial task of time series analysis and it has widespread applications in intelligent transportation systems. In literature, a large number of methods [19], [20], [21], [22], [23], [24], [25] have been proposed to address this task. Early works usually applied time series models for prediction, but could not well model the traffic patterns of complex and unconstrained scenarios [26], [27], [28]. Recently, deep neural networks have become the mainstream approach in this field. For instance, Wang et al. [29] developed an end-to-end convolutional neural network to automatically discover the supply-demand patterns from car-hailing service data. Zhang *et al.* [15] utilized three residual networks [30] to learn the closeness, period and trend properties for citywide traffic flow prediction. Yao *et al.* [31] proposed a Deep Multi-View Spatial-Temporal Network for taxi demand prediction, which learned spatial relations with a deep CNN and modeled temporal correlations with a Long Short-Term Memory (LSTM) cell [32]. Liu *et al.* [33], [34] incorporated a hierarchical convolutional LSTM network with an attention mechanism to learn the spatial-temporal representations dynamically.

Subsequently, graph convolutional networks (GCN [35], [36], [37], [38]) were widely adopted to model various complex systems with non-Euclidean structures. For instance, Li *et al.* [39] modeled the traffic flow as a diffusion process on a directed graph and captured the spatial dependency with bidirectional random walks. Guo *et al.* [40] introduced attention mechanisms into spatial-temporal graph networks for dynamical traffic prediction. Bai *et al.* [41] utilized a hierarchical graph convolutional structure to capture both the spatial and temporal correlations for multi-step passenger demand prediction. Sun *et al.* [42] developed a multi-view graph convolutional network for crowd flow prediction, where different views captured various interactions and spatial correlations between different irregular regions. Cao *et al.* [43] proposed a Spectral Temporal Graph Neural Network, which incorporated a Graph Fourier Transform and a Discrete Fourier Transform to jointly capture inter-series correlations and temporal dependencies in the spectral domain. Recently, GCN has also been employed for metro ridership prediction. Han *et al.* [44] transformed the city metro network into a graph and made predictions using graph convolutional neural networks. Liu *et al.* [8] modeled a metro system as graphs with physical/virtual topologies and proposed a unified Physical-Virtual Collaboration Graph Network, which fully learned the complex ridership patterns from those tailor-designed graphs. Nevertheless, most previous methods merely forecast the traffic states of every region or station, which can only provide limited information for urban traffic management. Thus we focus on the more meaningful origin-destination prediction in this work.

2.2 Traffic Origin-Destination Prediction

Traffic origin-destination prediction is a challenging task that aims to forecast the traffic flow or demand between any two positions. Recently, this task has attracted increasing attention in both academic and industrial communities. For instance, Liu *et al.* [45] incorporated local spatial context, temporal evolution context, and global correlation context to forecast the future taxi OD demand. Shi *et al.* [46] extracted temporal features for each OD pair with LSTM units and learned the spatial dependency of origins and destinations respectively. Wang *et al.* [47] applied graph convolutions among geographical and semantic neighbors to model the taxi OD transferring patterns. Ke *et al.* [48] characterized the OD pair-wise relationships with multiple OD graphs and developed a spatial-temporal encoder-decoder residual framework to model both the spatial dependencies across different OD pairs and the temporal dependencies of the OD pairs themselves.

All methods mentioned above were proposed for ride-hailing applications, where both the origin and destination of a passenger are known once a taxi request is generated. However, in online metro systems, the destination of a passenger is unknowable until he/she reaches the destination station, thus the complete OD matrices cannot be obtained immediately. Thus online metro origin-destination prediction essentially belongs to the category of incomplete time series analysis [49], [50]. To address this problem, Gong *et al.* [12] used some indication matrices to mask and neglect those unfinished metro orders and applied a non-negative matrix factorization strategy to learn the latent properties of entered and exited stations from incomplete OD matrices. Both Zhang *et al.* [11] and Cheng *et al.* [14] used the boarding (entering) demand to replace the unavailable OD matrices. Specifically, Zhang *et al.* [11] developed a channel-wise attentive split–convolutional neural network to assign different values for OD features, while Cheng *et al.* [14] developed a high-order weighted dynamic mode decomposition to learn time-evolving features of a metro system. Recently, Noursalehi *et al.* [13] toughly used the historical DO matrices to forecast the future OD matrices with a multi-resolution spatial-temporal neural network model. However, all previous works have never explicitly exploited the information of unfinished transactions. Moreover, they either only predicted OD matrices or forecasted OD/DO matrices separately, completely neglecting their intrinsic correlation. In contrast, our method fully aggregates various information to jointly forecast the future OD ridership and DO ridership.

2.3 Transformer Architecture

Transformer [51] is an advanced neural network block that aggregates information from the entire input sequence with an attention mechanism [52]. Specifically, it consists of a multi-head self-attention layer, a point-wise feed-forward layer, and a normalization layer. The global computation and perfect memory mechanism make it suitable for long sequence modeling. Recently, transformer has been widely applied to various tasks of artificial intelligence, inducing natural language processing [53], [54], computer vision [55], [56], [57], and time series analysis [58], [59]. For instance,

Zhou *et al.* [60] proposed an efficient transformer-based model, which incorporated a ProbSparse Self-attention mechanism and distilling operation to capture time-series long-range dependencies between outputs and inputs efficiently. Inspired by the success of these works, we apply a transformer to learn the global information interaction between all metro stations. However, unlike most previous works that only transferred homogeneous information, our method develops a Dual Information Transformer, which propagates OD information and DO information mutually by generating heterogeneous queries, keys, and values. To the best of our knowledge, our work is the first attempt to employ the heterogeneous transformer to address time series prediction.

3 PRELIMINARIES

In this section, we briefly introduce some notations and the definition for online metro origin-destination prediction. For brevity, the frequently used notations in this paper are summarized in Table 1.

1) Transaction Data: In an online metro system, a large number of trip transactions are made over time. For each finished transaction, we can know both its entry station and exit station, as well as their corresponding time-stamps. However, for each ongoing transaction, only the entry station and entry time-stamp are knowable.

2) Matrix Compression Preprocessing: It's worth noting that the ridership between some stations is usually small or even zero. Inspired by the previous work [8], we would take into consideration the origin-destination pairs that have high ridership. Specifically, for each origin station, we measure the destination distribution of all its entered passengers and then select the top $K-1$ stations with the highest destination probability. Thus, we can generate an OD mapping matrix $M_{od} \in \mathbb{R}^{N \times K}$, where N is the number of metro stations. More specifically, for the origin station i, $M_{od}(i,j)$ $(j = 1, ..., K-1)$ is the index of its j-th destination station with high ridership. Moreover, $M_{od}(i,K)$ is set to -1, which is utilized to indicate the indexes of the remaining destination stations. In the same way, we can also generate a DO mapping matrix $M_{do} \in \mathbb{R}^{N \times K}$, where $M_{do}(i,j)$ is the index of the j-th most related origin station for the destination station i. With a value of -1, $M_{do}(i,K)$ denotes the indexes of remaining origin stations. In this work, these mapping matrices are utilized to guide the compression of metro OD and DO matrices.

3) Compact OD/DO Matrix Generation: At each time interval t, we measure the numbers of various types of transactions to generate the following vector and matrices.

i) Incomplete OD Matrix $IOD_t \in \mathbb{R}^{N \times K}$: Specifically, $IOD_t(i,j)$ is the number of passengers that entered station i at time interval t and have exited from station $M_{od}(i,j)$, where $j = 1, ..., K-1$. Moreover, $IOD_t(i,K)$ is the total number of passengers which entered station i and have exited from the remaining stations.

ii) Unfinished Order Vector $U_t \in \mathbb{R}^N$: This vector is utilized to record the information of ongoing transactions. Specifically, $U_t(i,j)$ denotes the number of passengers that entered at station i at time interval t but have not reached their destinations. Such information has never been explored in previous works [10], [12], [13].

TABLE 1
Some notations for online metro origin-destination ridership prediction.

Notations	Description
t	the current time interval
n	the length of input historical sequence
m	the length of output future sequence
N	the number of metro stations
K	the number of considered OD/DO pairs for each station
$M_{od} \in \mathbb{R}^{N \times K}$	the index of considered OD pairs
$M_{do} \in \mathbb{R}^{N \times K}$	the index of considered DO pairs
$U_{t-n+i} \in \mathbb{R}^N$	the unfinished order vector $(i = 1, ..., n)$
$IOD_{t-n+i} \in \mathbb{R}^{N \times K}$	the incomplete OD matrix $(i = 1, ..., n)$
$OD_{t+i} \in \mathbb{R}^{N \times K}$	the complete OD matrix $(i = 1, ..., m)$
$DO_{t+i} \in \mathbb{R}^{N \times K}$	the complete DO matrix $(i = -n+1, ..., m)$

iii) Complete OD Matrix $OD_t \in \mathbb{R}^{N \times K}$: Different to IOD_t, this matrix records the complete OD ridership, and it is usually served as the predicted target in our work. Specifically, $OD_t(i,j)$ is the number of passengers that entered station i at time interval t and would exit from station $M_{od}(i,j)$, where $j = 1, ..., K-1$. The number of passengers that would head for the remaining stations is stored as $OD_t(i,K)$.

iv) DO Matrix $DO_t \in \mathbb{R}^{N \times K}$: This matrix denotes the complete DO ridership at time interval t. More specifically, $DO_t(i,j)$ is the number of passengers that entered station $M_{do}(i,j)$ at earlier moments and exit from station i at time interval t. Similarly, $DO_t(i,K)$ is the total number of exited passengers at the remaining stations.

Definition 1. Online Metro Origin-Destination Prediction Assuming that the current time interval is t and we aim to utilize the historical ridership of previous n time intervals to forecast the complete OD and DO ridership of future m time intervals:

$$\{IOD, U, DO\}_{t-n+i} \;\Rightarrow\; \{OD, DO\}_{t+j} \tag{1}$$

where $i = 1, 2, ..., n$ and $j = 1, 2, ..., m$.

4 METHODOLOGY

In this work, we propose a unified neural network module termed Heterogeneous Information Aggregation Machine (HIAM), which fully captures various information (i.e., incomplete OD ridership, unfinished orders, DO ridership) to effectively model the metro origin-destination distribution. As shown in Fig. 3, our HIAM consists of an OD modeling branch, a DO modeling branch, and a Dual Information Transformer for OD-DO interaction modeling. Specifically, at each iteration, the OD branch feeds an incomplete OD matrix generated from finished orders and two estimated destination matrices for those unfinished orders into graph convolutional gated recurrent units to learn a compact OD hidden state for memorizing the OD evolution patterns. Meanwhile, the DO branch takes the corresponding DO matrix as input to generate a DO hidden state that memorizes the DO evolution patterns. The Dual Information Transformer is then applied to enhance the OD state and DO state by exploring their mutual information. Based on

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2022.3178184, IEEE Transactions on Pattern Analysis and Machine Intelligence

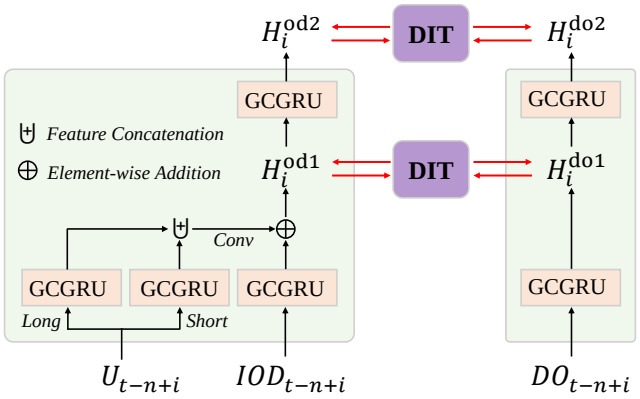


Fig. 3. The architecture of the proposed Heterogeneous Information Aggregation Machine. This module is composed of two parallel branches respectively for OD and DO modeling, and a Dual Information Transformer for OD-DO interaction modeling.

the proposed HIAM, we develop an online metro origin-destination prediction framework to jointly forecast the complete OD and DO ridership of the next m time intervals.

4.1 Metro Topology Modeling

Metro is essentially a complex traffic system with a non-Euclidean topology. Recently, GCN has been proven to be effective for non-Euclidean data embedding [37], [61], [62]. Inspired by these works, we also model a metro system as a directed graph and incorporate it into graph convolution gated recurrent units (GCGRU) to learn spatial-temporal representation for metro ridership.

In this work, we directly utilize the physical topology of the studied metro system to guide the construction of the graph network. By definition, a graph is composed of nodes, edges as well as the weights of edges, i.e., $G = (V, E, W)$. Specifically, V is the set of N nodes and each node represents a metro station in the real world. $E \in \mathbb{R}^{N\times N}$ is an edge connection matrix. $E(i,j)$ is 1 if the stations i and j are directly connected in the metro system, otherwise, it's set to 0. We then obtain the edge weight matrix $W \in \mathbb{R}^{N\times N}$ by applying a linear normalization on each row of W:

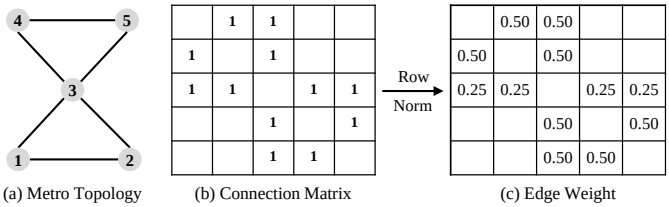
$$W(i,j) = \frac{E(i,j)}{\sum_{k=1}^{N} E(i,k)}. \tag{2}$$

One example of graph construction for a metro system with five stations is illustrated in Fig. 4.

Graph convolution and gated recurrent units are then integrated for representation learning. Let us assume that the input data of nodes is $I_t = \{I_t^1, I_t^2, ..., I_t^N\}$, where I_t^i can be the ridership data $IOD_t(i)$, $U_t(i)$, $DO_t(i)$ or their features. Instead of using spectral convolution [35], [63], here we utilize spatial convolution [64] to aggregate information from neighbor nodes. Specifically, the convolutional feature $f(I_t^i) \in \mathbb{R}^d$ of the node i is computed by:

$$f(I_t^i) = \Theta_l I_t^i + \sum_{j\in\mathcal{N}(i)} W(i,j) \odot \Theta_n I_t^j, \tag{3}$$

where $\odot$ is the Hadamard product and $\mathcal{N}_p(i)$ is the neighbor set of node i in the constructed graph. Θ_l denotes the self-loop parameters and Θ_n represents the neighbor parameters. d is the dimension of features.



(a) Metro Topology (b) Connection Matrix (c) Edge Weight

Fig. 4. Illustration of the graph construction for non-Euclidean metro systems. (a) is the physical topology of a metro system with five stations. Matrix (b) recodes the connectivity of edges in the constructed graph, while matrix (c) is the normalized weights of edges.

We then embed graph convolutions into gated recurrent units for temporal modeling. Notice that all original regular convolutions in GRU are replaced with our graph convolutions, and we apply the modified formulation to compute the reset gate, update gate, and new information. For convenience, the computation of the output hidden state $H_t = \{H_t^1, H_t^2, ..., H_t^N\}$ is denoted as:

$$H_t = \text{GCGRU}(I_t, H_{t-1}), \tag{4}$$

where H_t^i is the hidden state of the i-th node and its feature dimension is also set to d. Thanks to this graph convolutional gated recurrent unit, we can learn spatial-temporal features effectively from the ridership data of non-Euclidean metro systems.

4.2 OD Modeling Branch

In HIAM, an OD modeling branch is specially developed to learn the OD distribution of ridership by jointly exploiting the information of incomplete OD matrices and unfinished order vectors with multiple series-parallel GCGRU. The overall architecture of our OD branch is shown in the left sub-graph of Fig. 3.

In this work, we fully explore the potential information of unfinished order vectors rather than directly feed them into GCGRU. In particular, considering the periodicity of resident mobilities, we estimate two potential destination matrices for unfinished orders based on their long short-term historical distribution. Let's assume that the current time interval is t and we take the destination estimation for U_{t-n+i} as an example to explain our working mechanism. As shown in Fig. 5, two destination distributions of historical unfinished orders are measured:

- **Short-term Destination Distribution** $DD^s_{t-n+i} \in \mathbb{R}^{N\times K}$: The unfinished order distribution at the same time interval of yesterday is utilized to estimate the destinations of ongoing passengers at the recent time interval $t-n+i$. Specifically, $DD^s_{t-n+i}(j,k)$ is the percentage of ongoing passengers that entered station j at time interval $t-n+i$ of yesterday but have not reached their destination station $M_{od}(j,k)$ until the time interval t of yesterday.
- **Long-term Destination Distribution** $DD^l_{t-n+i} \in \mathbb{R}^{N\times K}$: The unfinished order distribution at the same time interval of the same weekday/weekend is utilized to estimate the destinations of ongoing passengers. Let's assume that today is Monday. $DD^s_{t-n+i}(j,k)$ is the overall percentage of passengers that entered station j at time interval $t-n+i$ of all Monday in

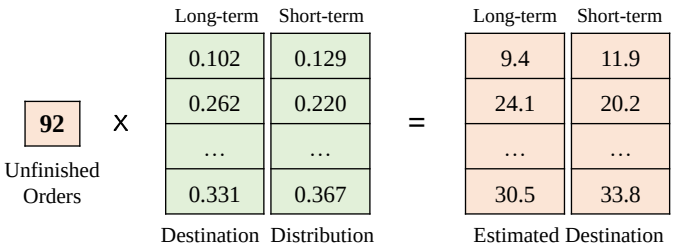


Fig. 5. Illustration of the potential destination matrices estimated for unfinished orders. Specifically, we first measure two long short-term destination distributions of historical unfinished orders and then estimate the potential destinations of ongoing passengers.

the training set but have not reached their destination station $M_{od}(j,k)$ after $n-i$ time intervals.

Notice that DD^s_{t-n+i} may be unstable on Monday and Saturday, since the ridership distribution is different between weekdays and weekends. Besides, DD^l_{t-n+i} may be smooth and cannot well reflect the recent ridership distribution. Therefore, both the long short-term distributions are used for destination estimation. Specifically, we generate two potential destination matrices UOD^l_{t-n+i} and $UOD^s_{t-n+i} \in \mathbb{R}^{N\times K}$ with following formulations:

$$\begin{aligned} UOD^l_{t-n+i}(j,k) &= U_{t-n+i}(j) * DD^l_{t-n+i}(j,k), \\ UOD^s_{t-n+i}(j,k) &= U_{t-n+i}(j) * DD^s_{t-n+i}(j,k). \end{aligned} \tag{5}$$

These estimated matrices and the corresponding incomplete OD matrix IOD_{t-n+i} are fed into three individual GCGRU for hidden state generation:

$$\begin{aligned} H^l_i &= GCGRU(UOD^l_{t-n+i},\ H^l_{i-1}), \\ H^s_i &= GCGRU(UOD^s_{t-n+i},\ H^s_{i-1}), \\ H^{iod}_i &= GCGRU(\ IOD_{t-n+i},\ H^{iod}_{i-1}). \end{aligned} \tag{6}$$

The hidden states H^l_i and H^s_i are then fused on each node with a 1*1 convolutional layer, whose output is further utilized to complement the hidden state H^{iod}_i of IOD_{t-n+i} and generate a compact OD hidden state $H^{od1}_i \in \mathbb{R}^{N\times d}$. This process can be formulated as:

$$H^{od1}_i = H^{iod}_i + Conv(H^l_i \uplus H^s_i, W_{1*1}), \tag{7}$$

where $\uplus$ denotes an operator of feature concatenation and W_{1*1} is the parameters of the convolutional layer.

As shown in Fig. 3, we then enhance the compact OD state H^{od1}_i with a Dual Information Transformer (DIT) described in Section 4.4. The enhanced state $\hat{H}^{od1}_i$ is further fed into the following GCGRU and DIT to learn high-order spatial-temporal features. This process can be formulated as:

$$\hat{H}^{od1}_i, \hat{H}^{do1}_i = DIT(H^{od1}_i,\ H^{do1}_i), \tag{8}$$

$$H^{od2}_i = GCGRU(\hat{H}^{od1}_i,\ \hat{H}^{od2}_{i-1}), \tag{9}$$

$$\hat{H}^{od2}_i, \hat{H}^{do2}_i = DIT(H^{od2}_i,\ H^{do2}_i), \tag{10}$$

where H^{do1}_i and H^{do2}_i are the hidden states of DO_{t-n+i}, and their generation is descried in Section 4.3.

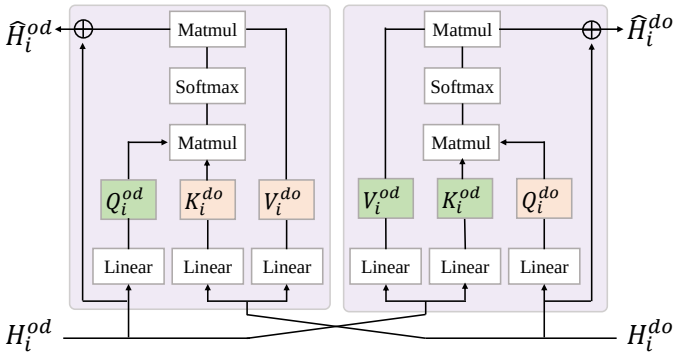


Fig. 6. The architecture of the proposed Dual Information Transformer. In this module, we perform information prorogation among OD branch and DO branch to jointly model the OD and DO distribution. The enhanced OD and DO features are more informative for ridership prediction.

4.3 DO Modeling Branch

In HIAM, a DO modeling branch is also developed to learn the DO distribution of metro ridership. As shown in the right sub-graph of Fig. 3, our DO branch is composed of two stacked GCGRU, each of which is followed by a Dual Information Transformer. Specifically, at iteration i, the first GCGRU takes the DO matrix DO_{t-n+i} as input to generate an initial DO hidden state $H^{do1}_i \in \mathbb{R}^{N\times d}$:

$$H^{do1}_i = GCGRU(DO_{t-n+i},\ H^{do1}_{i-1}). \tag{11}$$

We then employ Eq. (8) to enhance this DO state by absorbing the OD information of H^{od1}_i. The enhance state $\hat{H}^{do1}_i$ is fed into the second GCGRU and the output hidden state is computed by:

$$H^{do2}_i = GCGRU(\hat{H}^{do1}_i, \hat{H}^{do2}_{i-1}). \tag{12}$$

Similar to the OD branch, $H^{do2}_i \in \mathbb{R}^{N\times d}$ is further refined by another Dual Information Transformer, which is formulated as Eq. (10). Thanks to the tailor-designed OD-DO interactive mechanism, our method can effectively learn the DO evolutionary trend with the aid of previous OD information.

4.4 Dual Information Transformer

In previous works [10], [12], OD and DO ridership are modeled separately. However, we notice that they usually have strong causality and correlation. For OD-to-DO causality, the historical OD ridership essentially affects the future DO ridership since the latter is the spatial-temporal evolution result of the former. Moreover, we can also infer the future OD ridership based on the DO-to-OD correlation. For example, the DO and OD ridership of some tide stations are usually negatively correlated [17], e.g., their future OD ridership would increase/decrease when the recent DO ridership decreases/increases. Taking the causality and correlation into consideration, we propose a novel Dual Information Transformer (DIT) to model the OD and DO distribution jointly by propagating their mutual information in a dual manner. After the interaction, our OD and DO features become more informative for ridership prediction.

As shown in Fig. 6, our DIT is implemented with two cross-branch transformers [51], where the right one propagates information from OD branch to DO branch, and the

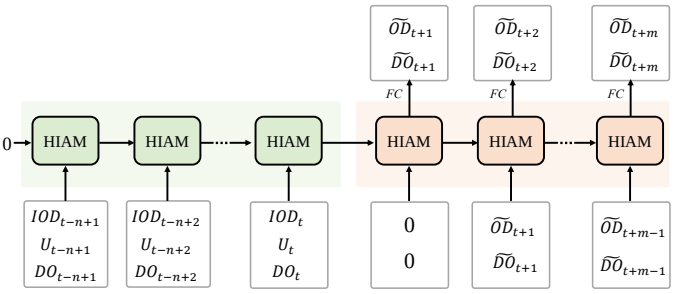


Fig. 7. The architecture of our online metro origin-destination prediction framework. The framework is developed with a Seq2Seq architecture, whose encoder and decoder are based on the proposed module, i.e., Heterogeneous Information Aggregation Machine (HIAM).

left one transfers information in the opposite direction. Here we take as an example the interaction between the OD feature H_i^{od} and DO feature H_i^{do} to illustrate the working mechanism of our DIT. Specifically, H_i^{od} and H_i^{do} are first respectively fed into three linear layers for query, key, and value embedding:

$$\begin{aligned} Q_i^{od} &= Conv(H_i^{od}, W_{1*1}^{qod}), \;\; Q_i^{do} = Conv(H_i^{do}, W_{1*1}^{qdo}), \\ K_i^{od} &= Conv(H_i^{od}, W_{1*1}^{kod}), \;\; K_i^{do} = Conv(H_i^{do}, W_{1*1}^{kdo}), \\ V_i^{od} &= Conv(H_i^{od}, W_{1*1}^{vod}), \;\; V_i^{do} = Conv(H_i^{do}, W_{1*1}^{vdo}), \end{aligned} \tag{13}$$

where all linear layers are implemented by 1*1 convolutional layers with individual parameters. Same as H_i^{od} and H_i^{do}, these query/key/value features also have a dimension of $N \times d$. Based on an attention mechanism, we compute two propagation coefficients $C_i^{o2d} \in \mathbb{R}^{N \times N}$ and $C_i^{d2o} \in \mathbb{R}^{N \times N}$, which dynamically determines the amount of information propagated among OD feature and DO feature:

$$\begin{aligned} C_i^{o2d} &= softmax(Q_i^{do}(K_i^{od})^T), \\ C_i^{d2o} &= softmax(Q_i^{od}(K_i^{do})^T), \end{aligned} \tag{14}$$

where T denotes a operator of matrix transposition and the softmax function is applied on each column. More specifically, $C_i^{o2d}(j,k)$ is the weight of information transferred from $H_i^{od}(j)$ to $H_i^{do}(k)$, while $C_i^{d2o}(j,k)$ denotes the weight of information transferred from $H_i^{do}(j)$ to $H_i^{od}(k)$. Finally, the cross-branch information prorogation is performed with the following formulations:

$$\begin{aligned} \hat{H}_i^{od} &= H_i^{od} + C_i^{d2o} V_i^{do}, \\ \hat{H}_i^{do} &= H_i^{do} + C_i^{o2d} V_i^{od}. \end{aligned} \tag{15}$$

After the information interaction, the OD state and DO state not only are enhanced mutually, but also better capture the inherent causality/correlation among OD and DO ridership. For convenience, Eq. 13-15 are simplified as:

$$\hat{H}_i^{od}, \hat{H}_i^{do} = DIT(H_i^{od}, \; H_i^{do}). \tag{16}$$

As stated in Eq. (8) and (10), the proposed DIT is applied after each level of GCGRU in HIAM for transferring cross-branch information hierarchically.

4.5 Online Origin-Destination Prediction Framework

Finally, we apply the proposed HIAM to develop a unified online metro origin-destination prediction framework, which forecasts the OD and DO ridership simultaneously for the next several time intervals. Following previous works [65], [66], [67], our framework is implemented with a Seq2Seq architecture [18], as shown in Fig. 7.

Specifically, our framework is composed of an encoder and a decoder, both of which are based on a multi-step HIAM. It is worth noting that the HIAM in the decoder is simplified by removing two GCGRU whose inputs are the estimated destination matrices, because there don't exist unfinished orders at the forecasting stage. Here we introduce the details of our framework. **In the encoder**, at iteration i ($i = 1, ..., n$), the heterogeneous data $\{IOD_{t-n+i}, U_{t-n+i}, DO_{t-n+i}\}$ are fed into our HIAM for learning origin-destination evolution information. Notice that the initial hidden states of all GCGRU are set to zero and the final hidden states are used to initialize the hidden states of the decoder. **In the decoder**, the input data of the first iteration is set to zero and the output hidden state of HIAM is respectively fed into a fully connected layers to forecast the OD matrix $\tilde{OD}_{t+1} \in \mathbb{R}^{N \times K}$ and DO matrix $\tilde{DO}_{t+1} \in \mathbb{R}^{N \times K}$. At iteration j ($j \geq 2$), our HIAM takes as input the predicted ridership matrices of the previous iteration to forecast $\tilde{OD}_{t+j}$ and $\tilde{DO}_j$. After m iterations, we can obtain a sequence of future OD ridership $\{\tilde{OD}_{t+1}, ..., \tilde{OD}_{t+m}\}$ and a sequence of future OD ridership $\{\tilde{DO}_{t+1}, ..., \tilde{DO}_{t+m}\}$.

TABLE 2
Details of SHMOD and HZMOD datasets. "# Stations" denotes the number of metro stations. "# OD/DO Pairs" refers to the number of considered OD/DO pairs for matrix compression.

Dataset	SHMOD	HZMOD
City	Shanghai, China	Hangzhou, China
# Stations	288	80
# OD/DO Pairs	76	26
Daily Ridership	8.82 M	2.35 M
Time Interval	15 minutes	15 minutes
Training Set	7/01/2016 - 8/31/2016	1/01/2019 - 1/18/2019
Validation Set	9/01/2016 - 9/09/2016	1/19/2019 - 1/20/2019
Testing Set	9/10/2016 - 9/30/2016	1/21/2019 - 1/25/2019

5 Experiments

In this section, we perform extensive experiments to verify the effectiveness of our model. First, we would introduce the experimental settings (e.g., dataset construction, implementation details, and evaluation strategy). We then compare the proposed method with ten basic and advanced approaches. Finally, we conduct ablation studies to explore the influence of each component in our method.

5.1 Experiments Settings

5.1.1 Dataset Construction

In this section, we introduce two large-scale benchmarks for online metro origin-destination prediction, which are constructed with billions of transactional records collected from the metro systems of Shanghai and Hangzhou, China. For brevity, these datasets are termed as SHMOD and HZMOD respectively, and their details are summarized in Table 2.

Specifically, SHMOD contains the transactional data of 288 stations, whose period ranges from Jul. 1st, 2016 to

Sept. 30th, 2016. HZMOD is built based on the transactional data of 80 stations collected in January 2019. On these datasets, the time interval is set to 15 minutes uniformly. As mentioned above, OD/DO matrices are usually sparse, thus for each station, we mainly consider **1)** the number of passengers heading for/coming from its most relevant $K-1$ stations, and **2)** the number of passengers heading for/coming from the remaining stations. In this work, K is set to 76 on SHMOD and 26 on HZMOD, because we observe that the ridership of such a small number of OD pairs accounts for a major proportion (i.e., 70%) of the total ridership. We then employ the data processing method described in Section 3 to generate incomplete OD matrices, unfinished order vectors, and OD matrices for each inferring time interval t. Finally, these datasets are officially divided into a training set, a validation set, and a testing set.

5.1.2 Implementation Details

In this work, the proposed method is implemented with the PyTorch framework [68]. The length n of input sequences is 4, and so is the length m of output sequences. The batch size is set to 8 for SHMOD and 32 for HZMOD, while the feature dimension d is set to 96 uniformly. Xavier uniform [69] is utilized to initialize the weights of filters and PReLU [70] is used as the activation function. In our transformer, multi-head attention is adopted and the head number is 4. During the training phase, the input data and their ground-truths are normalized with Z-score Normalization[2]. The learning rate is initialized to 0.001 for 60 epochs and decays by 0.2/0.5 for SHMOD/HZMOD every 20 epochs. Adam optimizer [71] is applied to minimize the mean absolute error between the predicted results and the corresponding ground-truths for 300 epochs. All interim models are evaluated on the validation set and the one with the best performance is chosen as our final model, which would be formally evaluated on the testing set for fair comparisons.

5.1.3 Evaluation Strategy

As mentioned above, those normalized ground-truths are used as supervision during training. Therefore, when evaluating, we first convert the output OD/DO matrices to the original scale with an inverted Z-score Normalization. Network-wide Mean Absolute Percentage Error (MAPE) is then adopted to evaluate the performance of different methods. In particular, we take into consideration the percentage error of the whole metro network, rather than compute the error of each OD/DO pair separately which is sensitive to a small denominator. Specifically, the MAPE for OD prediction is computed as:

$$MAPE = \frac{\sum_{i=1}^{N}\sum_{j=1}^{K}|\tilde{OD}(i,j) - OD(i,j)|}{\sum_{i=1}^{N}\sum_{j=1}^{K}|OD(i,j)|}. \quad (17)$$

The MAPE for DO prediction is computed with the same formulation. Notice that our method forecasts the ridership of the next m time intervals, and we would measure the MAPE for each time interval in the following sections.

2. https://en.wikipedia.org/wiki/Standard_score

5.2 Comparison with State-of-the-Art Methods

In this work, we compare the proposed method with the following ten basic and advanced approaches. Notice that there are not any public source codes of existing methods for metro OD/DO prediction [11], [12], [13], thus these works are not involved in our comparison. To facilitate future research on this task, we would release our source codes and benchmarks once accepted.

- **Historical Average (HA):** HA is a periodic baseline that uses historical average ridership to forecast future ridership. More specifically, the OD/DO ridership at 9:00-9:15 am on a specific Monday is predicted as the average of historical observations from the corresponding timestamp of all Mondays in the training set.
- **Random Forest (RF):** As a traditional machine learning method, RF is reimplemented to forecast the future metro ridership with some decision trees. Specifically, the number of trees is set to 10 in our work. These trees are expanded automatically until all leaves contain one sample or are pure.
- **Long Short-Term Memory (LSTM [72]):** This model employs two fully-connected LSTM layers to forecast the future metro ridership in a Seq2Seq manner [18]. The dimension of hidden states is set to 256.
- **Gated Recurrent Unit (GRU [73]):** The architecture of this model is similar to that of the previous model. The main difference lies in that this model adopts GRU layers rather than LSTM layers. The dimension of hidden states is also set to 256.
- **GraphWaveNet [74]:** In this network, an adaptive dependency matrix is learned to discover graph structure automatically, while a stacked dilated 1D convolution component is developed to model long-range temporal sequences. The official code[3] is used to reimplement this method on our SHMOD and HZMOD benchmarks.
- **Diffusion Convolutional Recurrent Neural Network (DCRNN [39]):** DCRNN is a representative model for traffic forecasting, in which spatial dependencies are captured through bidirectional random walks on graphs and temporal dependencies are modeled with an encoder-decoder architecture. This model is easily reimplemented to forecast the metro OD/DO ridership with its official code[4].
- **Spatial-Temporal Graph to Sequence Network (STG2Seq [41]):** In this model, spatial and temporal relationships of traffic flow are captured simultaneously with a hierarchical graph convolutional structure. Based on the official code[5], this model is reimplemented for metro OD/DO ridership prediction.
- **Physicla-Virtual Collaborative Graph Network (PVCGN [8]):** PVCGN is a recent method designed for metro ridership prediction. In this model, a physical graph, a similarity graph, and a correlation graph are incorporated to learn the complex patterns of metro ridership. The official code[6] is adopted to reimplement this model for metro OD/DO prediction.

3. https://github.com/nnzhan/Graph-WaveNet
4. https://github.com/liyaguang/DCRNN
5. https://github.com/LeiBAI/STG2Seq
6. https://github.com/HCPLab-SYSU/PVCGN

TABLE 3
Performance of OD prediction and DO prediction on the SHMOD Dataset.

Ridership	Time Interval	HA	RF	LSTM	GRU	GraphWaveNet	DCRNN	STG2Seq	PVCGN	DGSL	Informer	Ours
OD	*15 min*	46.28%	75.05%	43.80%	44.96%	40.49%	41.14%	42.56%	38.69%	40.05%	40.06%	37.81%
	30 min	46.21%	72.19%	43.08%	42.22%	40.20%	41.13%	41.58%	38.69%	40.00%	40.28%	37.79%
	45 min	46.12%	71.96%	44.55%	42.85%	40.46%	41.55%	41.47%	38.92%	40.32%	40.20%	37.99%
	60 min	46.05%	78.69%	46.95%	43.99%	41.54%	42.23%	41.77%	39.34%	40.76%	40.48%	38.36%
DO	*15 min*	47.18%	65.47%	43.99%	42.32%	42.29%	40.53%	40.77%	38.72%	40.27%	40.79%	38.65%
	30 min	47.22%	65.72%	41.91%	41.32%	42.04%	40.59%	40.03%	39.01%	40.54%	41.33%	38.56%
	45 min	47.23%	66.15%	42.75%	42.16%	42.04%	41.10%	40.22%	39.42%	41.09%	41.71%	38.80%
	60 min	47.19%	66.75%	44.00%	43.06%	42.18%	41.89%	40.95%	39.96%	41.69%	42.10%	39.18%

TABLE 4
Performance of OD prediction and DO prediction on the HZMOD Dataset.

Ridership	Time Interval	HA	RF	LSTM	GRU	GraphWaveNet	DCRNN	STG2Seq	PVCGN	DGSL	Informer	Ours
OD	*15 min*	33.00%	57.33%	32.19%	32.85%	32.29%	31.42%	30.54%	29.83%	30.68%	29.66%	27.86%
	30 min	32.98%	55.27%	32.46%	32.82%	32.84%	31.61%	30.97%	30.42%	30.87%	29.79%	27.90%
	45 min	32.95%	55.88%	33.83%	34.13%	33.52%	32.12%	31.37%	30.84%	31.07%	30.06%	28.04%
	60 min	32.91%	63.14%	35.46%	35.90%	34.79%	32.73%	31.46%	30.62%	31.30%	30.60%	28.22%
DO	*15 min*	34.19%	52.97%	31.23%	32.55%	33.64%	30.89%	30.12%	30.06%	30.58%	30.51%	28.57%
	30 min	34.26%	53.11%	31.25%	31.92%	33.76%	31.07%	29.79%	30.49%	30.63%	30.49%	28.64%
	45 min	34.31%	53.39%	32.32%	33.14%	34.63%	31.66%	30.40%	30.99%	31.02%	30.90%	28.83%
	60 min	34.32%	53.76%	34.02%	34.75%	35.72%	32.46%	31.52%	31.68%	31.57%	31.56%	29.09%

- **Discrete Graph Structure Learning (DGSL [75]:** This method proposes a probabilistic graph model to learn the graph structure of time series data by optimizing the mean performance over the graph distribution and sampling the discrete graph differentiably. Based on its official code[7], this method is reimplemented on the SHMOD and HZMOD datasets.
- **Informer [60]:** This is an efficient transformer-based model for long sequence time-series forecasting. This model incorporates three efficient and effective modules to capture long-range dependencies between input and output. Based on its official code[8], we reimplement Informer to forecast metro OD/DO ridership. Notice that the hyper-parameters input sequence length of the encoder, start token length and prediction sequence length of the decoder are set to 4, 2, and 4, respectively.

The performance of all compared methods are summarized in Table 3 for SHMOD and Table 4 for HZMOD. We can observe that the traditional method RF obtains unacceptable MAPE at all time intervals and even performs worse than the simple baseline HA, since RF has a limited capacity to capture the complex spatial-temporal distribution of OD/DO ridership. Compared with HA, these RNN-based models LSTM and GRU have certain performance improvements, especially for the first two-step predictions (i.e., 15 and 30 minutes), when explicitly learning the temporal representations from input data. By modeling the spatial/temporal distribution with graph convolutions, those graph-based methods (e.g., GraphWaveNet, STG2Seq, DCRNN, DGSL, and PVCGN) outperform those traditional baselines and common recurrent neural networks. We find that the multi-graph model PVCGN is better than single-graph models in most cases. Moreover, we obverse that the transformer-based model Informer obtains comparable results on the HZMOD dataset, but its performance is not satisfactory on the SHMOD dataset. The reason is that Informer mainly captures temporal dependencies, without learning spatial dependencies explicitly. To our knowledge, the spatial complexity of the Shanghai metro system is much greater than that of the Hangzhou metro system.

Despite progress in model architecture and performance, all the above methods only take finished orders as input data for OD prediction, while ignoring the mutual information between OD and DO distributions. By contrast, our method introduces unfinished orders creatively to enhance the incomplete OD matrices and fully explore the OD-DO information interaction, thereby achieving state-of-the-art performance for both the OD and DO prediction. For instance, on the SHMOD dataset, our HIAM obtains the lowest MAPE 38.36% and 39.18% respectively for OD ridership and DO ridership in terms of 60-minute prediction. On the HZMOD dataset, our method also outperforms all previous approaches consistently with large margins during each time interval. These comparisons greatly demonstrate the effectiveness of the proposed HIAM for online metro origin-destination prediction.

5.3 Ablation Studies

In this subsection, we perform extensive analyses to verify the effectiveness of each component of the proposed HIAM.

5.3.1 *Effectiveness of Unfinished Order Information*

As mentioned above, to facilitate the OD prediction, our method takes unfinished transactions into consideration and estimates the potential destinations of ongoing passengers based on long short-term historical distributions. To show the effectiveness of our unfinished order usage strategy, we implement five variants of our method for OD modeling. Note that these variants don't involve the DO prediction.

- **IOD-Net:** This variant directly utilizes these previous incomplete OD matrices $\{IOD_{t-n+i}|i = 1, ..., n\}$ to forecast the future complete OD ridership.
- **IOD+U-Net:** This variant takes these incomplete OD matrices and those corresponding unfinished order vectors $\{U_{t-n+i}|i = 1, ..., n\}$ for OD prediction. Note that these unfinished order vectors are directly fed into GCGRU without potential destination estimation.

7. https://github.com/chaoshangcs/GTS
8. https://github.com/zhouhaoyi/Informer2020

TABLE 5
Performance of different input information for OD prediction.

Dataset	Time Interval	IOD	IOD+U	IOD+U(short)	IOD+U(long)	IOD+U(short+long)
SHMOD	*15 min*	41.04%	39.58%	39.58%	39.38%	38.69%
	30 min	41.01%	39.61%	39.56%	39.34%	38.43%
	45 min	41.54%	40.17%	40.08%	39.75%	38.63%
	60 min	42.43%	41.06%	40.94%	40.43%	39.18%
HZMOD	*15 min*	31.47%	29.89%	29.50%	29.48%	28.75%
	30 min	32.25%	30.42%	29.88%	29.86%	28.87%
	45 min	33.60%	31.37%	30.75%	30.55%	29.31%
	60 min	35.11%	32.60%	31.91%	31.55%	30.05%

TABLE 6
The influence of OD-to-DO causality on DO prediction performance. Here we explore the OD-to-DO causality to facilitate the metro DO prediction. '*Input* → *Output*' denotes that the historical input data is used to forecast the future output data.

Input → *Output*	SHMOD				HZMOD			
	15 min	*30 min*	*45 min*	*60 min*	*15 min*	*30 min*	*45 min*	*60 min*
IOD → DO	41.33%	41.26%	41.66%	42.38%	31.81%	31.81%	32.75%	34.12%
IOD+U → DO	40.99%	40.81%	41.18%	41.87%	31.57%	31.64%	32.11%	33.05%
IOD+U(short+long) → DO	40.17%	39.89%	40.05%	40.54%	30.45%	30.31%	30.49%	31.12%
DO → DO	40.41%	40.57%	41.20%	42.05%	30.61%	31.23%	32.44%	33.99%
IOD+U(short+long), DO → DO	38.65%	38.56%	38.80%	39.18%	28.57%	28.64%	28.83%	29.09%

TABLE 7
The influence of DO-to-OD correlation on OD prediction performance. Here the DO-to-OD correlation is incorporated to promote the metro OD prediction. '*Input* → *Output*' denotes that the historical input data is used to forecast the future output data.

Input → *Output*	SHMOD				HZMOD			
	15 min	*30 min*	*45 min*	*60 min*	*15 min*	*30 min*	*45 min*	*60 min*
DO → OD	42.31%	41.95%	42.17%	42.87%	33.36%	33.56%	34.28%	35.50%
IOD+U(short+long) → OD	38.69%	38.43%	38.63%	39.18%	28.75%	28.87%	29.31%	30.05%
IOD+U(short+long), DO → OD	37.81%	37.79%	37.99%	38.36%	27.86%	27.90%	28.04%	28.22%

- **IOD+U(short)-Net:** The architecture of this variant is similar to that of IOD+U-Net. In this network, the potential destinations of each U_{t-n+i} are first estimated on the basis of the short-term destination distribution of historical unfinished orders.
- **IOD+U(long)-Net:** Different from the previous variant, this network employs the long-term destination distribution of historical unfinished orders to estimate the potential destinations of those ongoing passengers.
- **IOD+U(short+long)-Net:** This network incorporates both long-term and short-term historical distributions to estimate the passengers' potential destinations for enhancing the incomplete OD information.

The performance of all variants is summarized in Table 5. We can observe that IOD-Net obtains poor MAPE at all time intervals on both datasets, since this model uses very limited information for ridership prediction. By directly introducing unfinished data, IOD+U-Net can decrease the MAPE from 42.43% to 41.06% on SHMOD and from 35.11% to 32.60% on HZMOD for the 60-minute prediction. This phenomenon well demonstrates the significance of the information aggregation from unfinished orders. Moreover, the variants IOD+U(short)-Net and IOD+U(long)-Net perform better than IOD+U-Net, when estimating the potential destinations of unfinished orders explicitly. For instance, at the fourth time interval, IOD+U(short)-Net obtains a MAPE 31.91% on HZMOD, while IOD+U(long)-Net obtains a MAPE 40.43% on SHMOD. Finally, IOD+U(short+long)-Net achieves obvious performance improvement by incorporating the long short-term estimated destination information. Specifically, compared with IOD-Net, this model reduces the MAPE, on average, by 2.77% on the SHMOD dataset and by 3.86% on the HZMOD dataset. Therefore, we conclude that the usage of unfinished orders with long short-term destination estimation can greatly promote online OD prediction.

5.3.2 *Influence of OD-to-DO Causality*

In this subsection, we explore the influence of OD-to-DO causality on metro DO prediction, i.e., utilizing the historical OD ridership information to forecast the future DO ridership. Here we implement the following variants that do not use historical DO information but purely exploit OD-to-DO causality for DO prediction.

- **IOD→DO:** This variant only takes incomplete OD matrices $\{IOD_{t-n+i}|i=1,...,n\}$ to forecast the future DO ridership $\{DO_{t+j}|i=1,...,m\}$.
- **IOD+U→DO:** This variant forecasts the future DO ridership from historical $\{IOD_{t-n+i}, U_{t-n+i}|i=1,...,n\}$. Notice that the potential destinations of U_{t-n+i} are not estimated in this variant.
- **IOD+U(short+long)→DO:** In this variant, historical incomplete OD matrices and the long short-term estimated destinations of unfinished orders are incorporated to predict the future DO ridership.

The performance of the above variants is summarized in Table 6. Specifically, the IOD→DO model performs poorly, since incomplete OD matrices only record the number of passengers who have completed their trips. We can observe that the performance of those OD-to-DO models can be further improved when incorporating

TABLE 8
Performance of different OD-DO interaction methods on the SHMOD and HZMOD datasets.

Dataset	Ridership	Time Interval	**W/O**	**SS**	DIT
SHMOD	OD	*15 min*	38.69%	38.66%	37.81%
		30 min	38.43%	38.57%	37.79%
		45 min	38.63%	38.84%	37.99%
		60 min	39.18%	39.24%	38.36%
	DO	*15 min*	40.41%	39.30%	38.65%
		30 min	40.57%	39.22%	38.56%
		45 min	41.20%	39.51%	38.80%
		60 min	42.05%	39.94%	39.18%
HZMOD	OD	*15 min*	28.75%	28.49%	27.86%
		30 min	28.87%	28.51%	27.90%
		45 min	29.31%	28.74%	28.04%
		60 min	30.05%	29.13%	28.22%
	DO	*15 min*	30.61%	29.17%	28.57%
		30 min	31.23%	29.14%	28.64%
		45 min	32.44%	29.35%	28.83%
		60 min	33.99%	29.77%	29.09%

the information about unfinished orders. In particular, the IOD+U(short+long)→DO model achieves very competitive performance, even better than the DO→DO model that uses the historical DO ridership to forecast the future DO ridership. Finally, as shown in the last row of Table 6, the IOD+U(short+long),DO→DO model achieves the best performance on both datasets, when learning the OD-to-DO causality and DO-to-DO mapping simultaneously. These experiments demonstrate that OD-to-DO causality can well facilitate the metro DO ridership prediction.

5.3.3 *Influence of DO-to-OD Correlation*

In [13], only the historical DO ridership was used to forecast the future OD ridership, due to the delayed availability of complete OD matrices. In this subsection, we delve into the impact of DO-to-OD correlation on OD prediction. Here we implement a variant of our model termed DO→OD, which utilizes $\{DO_{t-n+i}|i = 1, ..., n\}$ to forecast $\{OD_{t+j}|j = 1, ..., m\}$. As shown in Table 7, the DO→OD variant obtains unsatisfactory MAPE on both datasets, performing much worse than the IOD+U(short+long)→OD variant that exploits incomplete OD matrices and unfinished orders for future OD prediction. However, we find that the performance of OD prediction can be further improved, when the DO-to-OD correlation and IOD+U(short+long)-to-OD mapping are learned simultaneously. Specifically, as shown in the last row of Table 7, the IOD+U(short+long),DO→OD variant reduces the MAPE, on average, by 0.75% on the SHMOD dataset and by 1.24% on the HZMOD dataset, compared with the IOD+U(short+long)→OD variant. Therefore, we can draw the following conclusions. **i)** It is inappropriate to perform OD prediction only using the historical DO ridership. **ii)** The DO-to-OD correlation can facilitate the metro OD ridership prediction to a certain extent.

5.3.4 *Exploration of OD-DO Interaction Operation*

In this work, we introduce a novel Dual Information Transformer (DIT) to capture the mutual information among OD distribution and DO distribution. Here we explore the influences of different OD-DO interaction operations:

- **W/O Interaction**: This method doesn't perform OD-DO interaction, which means that the future OD ridership and DO ridership are forecasted separately.
- **Single-Station (SS) Interaction**: This method propagates the OD and DO information on the same station. That is to say, the OD information of station i is only used to enhance the DO information of station i, and vice versa. More specifically, the OD hidden state and DO hidden state of station i are concatenated and fed into a 1*1 convolutional layer to generate the enhanced OD hidden state and DO hidden state.
- **DIT Interaction**: The method propagates information between all stations with the proposed DIT, in which the OD/DO information of station i can be transferred to enhance the DO/OD information of station j.

The performance of all OD-DO interaction methods are summarized in Table 8. We can observe that the model with Single-Station Interaction surpasses the model without interaction significantly for DO prediction, and can also improve the performance of OD prediction to a certain extent, since the causality of OD-to-DO is easily captured but the correlation of DO-to-OD is challenging for Single-Station Interaction. When adopting cross-station interaction, our DIT can fully exploit the OD-DO mutual information, thereby achieving the best performance for both OD and DO prediction. For instance, compared with "W/O Interaction", our DIT reduces the MAPE, on average, by 0.75% for OD prediction and 2.26% for DO prediction on the SHMOD dataset. Similar performance improvement can be obtained on the HZMOD dataset. These experiments verify the superiority of the proposed DIT for OD-DO information interaction.

TABLE 9
Performance of different lengths of the input sequence on the SHMOD dataset.

Ridership	Time Interval	Input Sequence Length				
		1	**2**	**3**	**4**	**5**
OD	*15 min*	38.99%	38.20%	37.90%	37.81%	37.79%
	30 min	39.00%	38.24%	37.88%	37.79%	37.72%
	45 min	39.19%	38.53%	38.12%	37.99%	37.93%
	60 min	39.46%	38.94%	38.48%	38.36%	38.39%
DO	*15 min*	39.24%	38.82%	38.71%	38.65%	38.55%
	30 min	39.06%	38.80%	38.63%	38.56%	38.46%
	45 min	39.35%	39.10%	38.91%	38.80%	38.68%
	60 min	39.74%	39.54%	39.31%	39.18%	39.04%

TABLE 10
Performance of different lengths of the input sequence on the HZMOD dataset.

Ridership	Time Interval	Input Sequence Length				
		1	**2**	**3**	**4**	**5**
OD	*15 min*	28.44%	28.02%	27.96%	27.86%	27.82%
	30 min	28.45%	28.12%	28.00%	27.90%	27.84%
	45 min	28.58%	28.29%	28.19%	28.04%	28.02%
	60 min	28.72%	28.52%	28.41%	28.22%	28.26%
DO	*15 min*	29.15%	28.81%	28.58%	28.57%	28.46%
	30 min	29.16%	28.99%	28.66%	28.64%	28.47%
	45 min	29.33%	29.24%	28.88%	28.83%	28.64%
	60 min	29.67%	29.55%	29.18%	29.09%	28.95%

5.3.5 *Impact of Different Input Sequence Length*

As described in Section 3, we utilize the data of previous n time intervals to forecast the ridership of future m time intervals. Following [8], we fix the output sequence length m to 4 and explore the effect of input sequence length n for online origin-destination prediction. As shown in Table 9 and Table 10, the MAPE gradually decreases as the variate n increase from 1 to 4, and longer sequence no longer results

TABLE 11
Performance of compressed OD/DO matrices and original OD/DO matrices on the SHMOD dataset.

Ridership	Time Interval	With Compression		Without Compression	
		Top $K-1$ Stations	Remaining Stations (Merged)	Top $K-1$ Stations	Remaining Stations (Non-merged)
OD	*15 min*	47.56%	14.14%	48.36%	84.40%
	30 min	47.30%	14.54%	48.04%	84.15%
	45 min	47.40%	14.96%	48.15%	84.10%
	60 min	47.75%	15.36%	48.55%	84.15%
DO	*15 min*	49.09%	14.04%	50.05%	82.63%
	30 min	48.91%	14.25%	49.82%	82.56%
	45 min	49.11%	14.62%	50.00%	82.64%
	60 min	49.52%	14.95%	50.41%	82.78%

TABLE 12
Performance of compressed OD/DO matrices and original OD/DO matrices on the HZMOD dataset.

Dataset	Time Interval	With Compression		Without Compression	
		Top $K-1$ Stations	Remaining Stations (Merged)	Top $K-1$ Stations	Remaining Stations (Non-merged)
OD	*15 min*	33.56%	13.31%	34.01%	62.42%
	30 min	33.57%	13.44%	33.94%	61.89%
	45 min	33.71%	13.60%	34.22%	61.89%
	60 min	33.94%	13.65%	34.60%	62.21%
DO	*15 min*	34.88%	13.32%	35.40%	61.14%
	30 min	34.86%	13.59%	35.28%	60.66%
	45 min	35.02%	13.85%	35.48%	60.90%
	60 min	35.29%	14.05%	35.92%	61.45%

in significant improvement. To explain this phenomenon, we measure the commuting time of metro passengers. As shown in Fig. 8, we can see that most passengers (i.e., 87.10% for ShangHai and 96.54% for HangZhou) complete their metro journey within one hour, i.e., four time intervals. It is reasonable to use the OD/DO ridership of the previous hour to predict the OD/DO ridership of the next hour. Therefore, the length m of the input sequence is set to 4 consistently in this work.

5.3.6 *Are compressed OD/DO matrices useful?*

As mentioned in Section 3, we compress those original OD/DO matrices with a dimension $N \times N$ to form compact matrices with a dimension $N \times K$, i.e., forecasting the OD/DO ridership of top $K-1$ most relevant stations and the total ridership of remaining stations. In this subsection, we explore the influence of matrix compression for online metro origin-destination prediction. To this end, we implement a variant of our model that utilizes the historical $N \times N$ ridership matrices to forecast the future $N \times N$ ridership matrices. Here we measure the prediction performance of the top $K-1$ stations and the remaining stations separately.

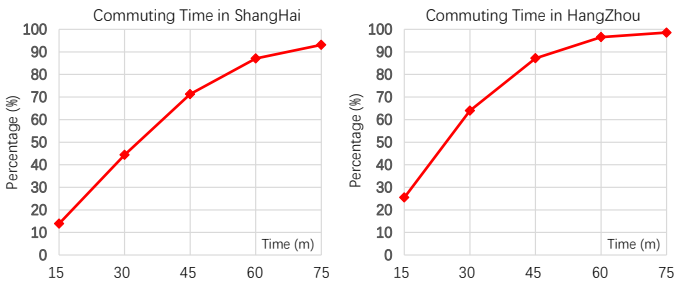


Fig. 8. The Cumulative Density Function of the commuting time of metro passengers in Shanghai and Hangzhou. We can observe that the commuting time of most passengers is within one hour.

As shown in Table 11, the MAPE of small cross-station ridership is more than 80% on the SHMOD dataset when we use original sparse matrices, since the ridership between weakly-relevant stations usually lacks regularity. Such poor performance is unacceptable and meaningless for practical application. By contrast, when merging these weakly relevant stations, our method obtains a small MAPE of about 14% for OD and DO prediction, since the evolution patterns of the total ridership of these stations are easily captured. Moreover, we find that introducing the non-merged ridership between weakly relevant stations would degrade the prediction performance between highly relevant stations. Specifically, the MAPE of the top $K-1$ stations of non-compressed matrices is 49.17% on average, while that of compressed matrices is 48.33% on average. This is because the irregular ridership of weakly relevant stations confuses the prediction model to a certain degree. As shown in Table 12, we can observe that the performance of compressed OD/DO matrices is also better on the HZMOD dataset. In summary, the OD/DO matrix compression is meaningful for online origin-destination prediction.

6 Conclusion

In this work, we focus on a crucial yet challenging task, online metro origin-destination prediction, i.e., forecasting the OD ridership and DO ridership for multiple time intervals in the future. However, conventional methods either directly used the limited information of incomplete OD matrix for inference, or completely neglected the causality and correlation between these two types of cross-station ridership. To facilitate this problem, we introduce a novel Heterogeneous Information Aggregation Machine (HIAM), which fully exploits heterogeneous information of historical data (e.g., incomplete OD matrices, unfinished order vectors, and DO matrices) to jointly learn the spatial-temporal patterns of OD and DO ridership. Based on HIAM, we

develop a Seq2Seq network to forecast the future OD and DO ridership simultaneously. Finally, we conduct extensive experiments on two large-scale benchmarks, and experiment results show that our method achieves state-of-the-art performance for online metro origin-destination prediction.

REFERENCES

[1] X. Chen and L. Sun, "Bayesian temporal factorization for multidimensional time series prediction," *TPAMI*, 2021.

[2] G. Spadondesouza, S. Hong, B. Brandoli, S. Matwin, J. F. Rodrigues, and J. Sun, "Pay attention to evolution: Time series forecasting with deep graph-evolution learning," *TPAMI*, 2021.

[3] "Beijing subway," https://en.wikipedia.org/wiki/Beijing_Subway.

[4] "Shanghai subway," https://en.wikipedia.org/wiki/Shanghai_Metro.

[5] X. Ma, J. Zhang, B. Du, C. Ding, and L. Sun, "Parallel architecture of convolutional bi-directional lstm neural networks for network-wide metro ridership prediction," *TITS*, vol. 20, no. 6, pp. 2278–2288, 2018.

[6] S. Fang, Q. Zhang, G. Meng, S. Xiang, and C. Pan, "Gstnet: Global spatial-temporal network for traffic flow prediction." in *IJCAI*, 2019, pp. 2286–2293.

[7] S. Hao, D.-H. Lee, and D. Zhao, "Sequence to sequence learning with attention mechanism for short-term passenger flow prediction in large-scale metro system," *TRC*, vol. 107, pp. 287–300, 2019.

[8] L. Liu, J. Chen, H. Wu, J. Zhen, G. Li, and L. Lin, "Physical-virtual collaboration modeling for intra-and inter-station metro ridership prediction," *TITS*, 2020.

[9] Z. Li, N. D. Sergin, H. Yan, C. Zhang, and F. Tsung, "Tensor completion for weakly-dependent data on graph for metro passenger flow prediction," in *AAAI*, vol. 34, no. 04, 2020, pp. 4804–4810.

[10] Y. Gong, Z. Li, J. Zhang, W. Liu, Y. Zheng, and C. Kirsch, "Network-wide crowd flow prediction of sydney trains via customized online non-negative matrix factorization," in *CIKM*, 2018, pp. 1243–1252.

[11] J. Zhang, H. Che, F. Chen, W. Mae, and Z. He, "Short-term prediction of urban rail transit origin-destination flow: A channel-wise attentive split-convolutional neural network method," *arXiv preprint arXiv:2008.08036*, 2020.

[12] Y. Gong, Z. Li, J. Zhang, W. Liu, and Y. Zheng, "Online spatio-temporal crowd flow distribution prediction for complex metro system," *TKDE*, 2020.

[13] P. Noursalehi, H. N. Koutsopoulos, and J. Zhao, "Dynamic origin-destination prediction in urban rail systems: A multi-resolution spatio-temporal deep learning approach," *TITS*, 2021.

[14] Z. Cheng, M. Trepanier, and L. Sun, "Real-time forecasting of metro origin-destination matrices with high-order weighted dynamic mode decomposition," *arXiv preprint arXiv:2101.00466*, 2021.

[15] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," in *AAAI*, vol. 31, no. 1, 2017.

[16] H. Yao, X. Tang, H. Wei, G. Zheng, and Z. Li, "Revisiting spatial-temporal similarity: A deep learning framework for traffic prediction," in *AAAI*, vol. 33, no. 01, 2019, pp. 5668–5675.

[17] X. Gong and Y. Lu, "Data mining based research on urban tide traffic problem," in *ICITS*. IEEE, 2008, pp. 122–127.

[18] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *arXiv preprint arXiv:1409.3215*, 2014.

[19] L. Lin, J. Li, F. Chen, J. Ye, and J. Huai, "Road traffic speed prediction: a probabilistic model fusing multi-source data," *TKDE*, vol. 30, no. 7, pp. 1310–1323, 2017.

[20] J. Zhang, Y. Zheng, J. Sun, and D. Qi, "Flow prediction in spatio-temporal networks based on multitask deep learning," *TKDE*, vol. 32, no. 3, pp. 468–478, 2019.

[21] Z. Pan, W. Zhang, Y. Liang, W. Zhang, Y. Yu, J. Zhang, and Y. Zheng, "Spatio-temporal meta learning for urban traffic prediction," *TKDE*, 2020.

[22] C. Zheng, X. Fan, C. Wang, and J. Qi, "Gman: A graph multi-attention network for traffic prediction," in *AAAI*, 2020.

[23] D. A. Tedjopurnomo, Z. Bao, B. Zheng, F. Choudhury, and A. Qin, "A survey on modern deep neural network for traffic prediction: Trends, methods and challenges," *TKDE*, 2020.

[24] X. Yin, G. Wu, J. Wei, Y. Shen, H. Qi, and B. Yin, "A comprehensive survey on traffic prediction," *arXiv preprint arXiv:2004.08555*, 2020.

[25] S. Guo, Y. Lin, H. Wan, X. Li, and G. Cong, "Learning dynamics and heterogeneity of spatial-temporal graph data for traffic forecasting," *TKDE*, 2021.

[26] M. Lippi, M. Bertini, and P. Frasconi, "Short-term traffic flow forecasting: An experimental comparison of time-series analysis and supervised learning," *TITS*, vol. 14, no. 2, pp. 871–882, 2013.

[27] J. Guo, W. Huang, and B. M. Williams, "Adaptive kalman filter approach for stochastic short-term traffic flow rate prediction and uncertainty quantification," *TRC*, vol. 43, pp. 50–64, 2014.

[28] P. Dell'Acqua, F. Bellotti, R. Berta, and A. De Gloria, "Time-aware multivariate nearest neighbor regression methods for traffic flow prediction," *TITS*, vol. 16, no. 6, pp. 3393–3402, 2015.

[29] D. Wang, W. Cao, J. Li, and J. Ye, "Deepsd: Supply-demand prediction for online car-hailing services using deep neural networks," in *ICDE*. IEEE, 2017, pp. 243–254.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.

[31] H. Yao, F. Wu, J. Ke, X. Tang, Y. Jia, S. Lu, P. Gong, J. Ye, and Z. Li, "Deep multi-view spatial-temporal network for taxi demand prediction," in *AAAI*, 2018.

[32] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in *NIPS*, 2015, pp. 802–810.

[33] L. Liu, R. Zhang, J. Peng, G. Li, B. Du, and L. Lin, "Attentive crowd flow machines," in *ACM Multimedia*. ACM, 2018, pp. 1553–1561.

[34] L. Liu, J. Zeng, G. Li, G. Zhan, Z. He, B. Du, and L. Lin, "Dynamic spatial-temporal representation learning for traffic flow prediction," *TITS*, 2020.

[35] J. Bruna, W. Zaremba, A. Szlam, and Y. Lecun, "Spectral networks and locally connected networks on graphs," in *ICLR*, 2014.

[36] D. Duvenaud, D. Maclaurin, J. Aguileraiparraguirre, R. Gomezbombarelli, T. D. Hirzel, A. Aspuruguzik, and R. P. Adams, "Convolutional networks on graphs for learning molecular fingerprints," in *NIPS*, 2015, pp. 2224–2232.

[37] T. Chen, L. Lin, X. Hui, R. Chen, and H. Wu, "Knowledge-guided multi-label few-shot learning for general image recognition," *TPAMI*, 2020.

[38] T. Chen, T. Pu, H. Wu, Y. Xie, L. Liu, and L. Lin, "Cross-domain facial expression recognition: A unified evaluation benchmark and adversarial graph learning," *TPAMI*, 2021.

[39] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," in *ICLR*, 2018.

[40] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial-temporal graph convolutional networks for traffic flow forecasting," in *AAAI*, vol. 33, 2019, pp. 922–929.

[41] L. Bai, L. Yao, S. Kanhere, X. Wang, Q. Sheng *et al.*, "Stg2seq: Spatial-temporal graph to sequence model for multi-step passenger demand forecasting," in *IJCAI*, 2019.

[42] J. Sun, J. Zhang, Q. Li, X. Yi, Y. Liang, and Y. Zheng, "Predicting citywide crowd flows in irregular regions using multi-view graph convolutional networks," *TKDE*, 2020.

[43] D. Cao, Y. Wang, J. Duan, C. Zhang, X. Zhu, C. Huang, Y. Tong, B. Xu, J. Bai, J. Tong *et al.*, "Spectral temporal graph neural network for multivariate time-series forecasting," *NeurIPS*, vol. 33, pp. 17 766–17 778, 2020.

[44] Y. Han, S. Wang, Y. Ren, C. Wang, P. Gao, and G. Chen, "Predicting station-level short-term passenger flow in a citywide metro network using spatiotemporal graph convolutional neural networks," *ISPRS International Journal of Geo-Information*, vol. 8, no. 6, p. 243, 2019.

[45] L. Liu, Z. Qiu, G. Li, Q. Wang, W. Ouyang, and L. Lin, "Contextualized spatial-temporal network for taxi origin-destination demand prediction," *TITS*, 2019.

[46] H. Shi, Q. Yao, Q. Guo, Y. Li, L. Zhang, J. Ye, Y. Li, and Y. Liu, "Predicting origin-destination flow via multi-perspective graph convolutional network," in *ICDE*. IEEE, 2020, pp. 1818–1821.

[47] Y. Wang, H. Yin, H. Chen, T. Wo, J. Xu, and K. Zheng, "Origin-destination matrix prediction via graph convolution: a new perspective of passenger demand modeling," in *KDD*, 2019, pp. 1227–1235.

[48] J. Ke, X. Qin, H. Yang, Z. Zheng, Z. Zhu, and J. Ye, "Predicting origin-destination ride-sourcing demand with a spatio-temporal encoder-decoder residual multi-graph convolutional network,"

Transportation Research Part C: Emerging Technologies, vol. 122, p. 102858, 2021.
[49] Q. Ma, S. Li, L. Shen, J. Wang, J. Wei, Z. Yu, and G. W. Cottrell, "End-to-end incomplete time-series modeling from linear memory of latent variables," *TCYB*, vol. 50, no. 12, pp. 4908–4920, 2019.
[50] Q. Ma, S. Li, and G. Cottrell, "Adversarial joint-learning recurrent neural network for incomplete time series classification," *TPAMI*, 2020.
[51] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.
[52] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.
[53] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.
[54] Z. Zhang, Y. Wu, J. Zhou, S. Duan, H. Zhao, and R. Wang, "Sg-net: Syntax guided transformer for language representation," *TPAMI*, 2020.
[55] N. Parmar, A. Vaswani, J. Uszkoreit, L. Kaiser, N. Shazeer, A. Ku, and D. Tran, "Image transformer," in *ICML*, 2018, pp. 4055–4064.
[56] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.
[57] L. Liu, M. Liu, G. Li, Z. Wu, and L. Lin, "Road network guided fine-grained urban traffic flow inference," *arXiv preprint arXiv:2109.14251*, 2021.
[58] M. Xu, W. Dai, C. Liu, X. Gao, W. Lin, G.-J. Qi, and H. Xiong, "Spatial-temporal transformer networks for traffic flow forecasting," *arXiv preprint arXiv:2001.02908*, 2020.
[59] N. Wu, B. Green, X. Ben, and S. O'Banion, "Deep transformer models for time series forecasting: The influenza prevalence case," *arXiv preprint arXiv:2001.08317*, 2020.
[60] H. Zhou, S. Zhang, J. Peng, S. Zhang, J. Li, H. Xiong, and W. Zhang, "Informer: Beyond efficient transformer for long sequence time-series forecasting," in *AAAI*, 2021.
[61] L. Bai, L. Cui, Y. Jiao, L. Rossi, and E. Hancock, "Learning backtrackless aligned-spatial graph convolutional networks for graph classification," *TPAMI*, 2020.
[62] Z.-H. Lin, S. Y. Huang, and Y.-C. F. Wang, "Learning of 3d graph convolution networks for point cloud analysis," *TPAMI*, 2021.
[63] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," *arXiv preprint arXiv:1606.09375*, 2016.
[64] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.
[65] B. Liao, J. Zhang, C. Wu, D. McIlwraith, T. Chen, S. Yang, Y. Guo, and F. Wu, "Deep sequence learning with auxiliary information for traffic prediction," in *KDD*, 2018, pp. 537–546.
[66] Z. Pan, Y. Liang, W. Wang, Y. Yu, Y. Zheng, and J. Zhang, "Urban traffic prediction from spatio-temporal data using deep meta learning," in *KDD*, 2019, pp. 1720–1730.
[67] Q. Zhang, J. Chang, G. Meng, S. Xiang, and C. Pan, "Spatio-temporal graph structure learning for traffic forecasting," in *AAAI*, vol. 34, no. 01, 2020, pp. 1177–1185.
[68] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *NeurIPS*, vol. 32, pp. 8026–8037, 2019.
[69] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *International Conference on Artificial Intelligence and Statistics*, 2010, pp. 249–256.
[70] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *ICCV*, 2015, pp. 1026–1034.
[71] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[72] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
[73] K. Cho, B. van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder–decoder for statistical machine translation," in *EMNLP*, 2014, pp. 1724–1734.
[74] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph wavenet for deep spatial-temporal graph modeling," in *AAAI*, 2019, pp. 1907–1913.
[75] C. Shang and J. Chen, "Discrete graph structure learning for forecasting multiple time series," in *ICLR*, 2021.

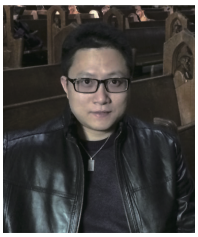
Lingbo Liu received the Ph.D degree from the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China, in 2020. From March 2018 to May 2019, he was a research assistant at the University of Sydney, Australia. His current research interests include machine learning and urban computing. He has authorized and co-authorized on more than 20 papers in top-tier academic journals and conferences. He has been serving as a reviewer for numerous academic journals and conferences such as TPAMI, TKDE, TNNLS, TITS, CVPR, ICCV and IJCAJ.

Yuying Zhu received the B.E. degree from the School of Informatics, Xiamen University, Xiamen, China, in 2020, and she is currently pursuing the Master's degree in computer science in the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China. Her current research interests include deep learning and data mining.

Guanbin Li uanbin Liuanbin LiG(M'15) is currently an associate professor in School of Data and Computer Science, Sun Yat-sen University. He received his PhD degree from the University of Hong Kong in 2016. His current research interests include computer vision, image processing, and deep learning. He is a recipient of ICCV 2019 Best Paper Nomination Award. He has authorized and co-authorized on more than 80 papers in top-tier academic journals and conferences. He serves as an area chair for the conference of VISAPP. He has been serving as a reviewer for numerous academic journals and conferences such as TPAMI, IJCV, TIP, TMM, TCyb, CVPR, ICCV, ECCV and NeurIPS.

Ziyi Wu received the B.E. degree from the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China, in 2020, where he is currently pursuing the Master's degree in computer science. His current research interests include salient object detection and self-supervised learning.

Lei Bai is a postdoctoral reserach fellow at the School of Electrical and Information Engineering, the University of Sydney, Australia. His reserach interests lie in Machine Learning, Spatial-temporal Learning, and their applicaitons (e.g., Intelligent Transportation, IoT Analytics, and Healthcare). Lei has published a set of peer reviewed papers on top conference and journals such as NeurIPS, CVPR, IJCAI, KDD, Ubicomp, and TITS. He is serving or has served as a program committee member or reviewer for TPAMI, NeurIPS, ICLR, CVPR, ICCV, AAAI, IJCAI, ACM Transactions on Sensor Networks, and so on.



Liang Lin (M'09, SM'15) is a Full Professor of computer science at Sun Yat-sen University. He served as the Executive Director and Distinguished Scientist of SenseTime Group from 2016 to 2018, leading the R&D teams for cutting-edge technology transferring. He has authored or co-authored more than 200 papers in leading academic journals and conferences, and his papers have been cited by more than 16,000 times. He is an associate editor of IEEE Trans. Neural Networks and Learning Systems and IEEE Trans. Human-Machine Systems, and served as Area Chairs for numerous conferences such as CVPR, ICCV, SIGKDD and AAAI. He is the recipient of numerous awards and honors including Wu Wen-Jun Artificial Intelligence Award, the First Prize of China Society of Image and Graphics, ICCV Best Paper Nomination in 2019, Annual Best Paper Award by Pattern Recognition (Elsevier) in 2018, Best Paper Dimond Award in IEEE ICME 2017, Google Faculty Award in 2012. His supervised PhD students received ACM China Doctoral Dissertation Award, CCF Best Doctoral Dissertation and CAAI Best Doctoral Dissertation. He is a Fellow of IET.