Vehicular Crowdsensing Inference and Prediction With Multitask Pretraining Graph Transformer Networks

Jie Huo, Luhan Wang, Zhaoming Lu, and Xiangming Wen, *Senior Member, IEEE*

***Abstract*—Vehicular crowdsensing has emerged as a prominent sensing paradigm in the Internet of Things (IoT), and its powerful sensing and computing capabilities can provide sufficient data for various applications. To reduce the cost while ensuring the sensing quality, sparse mobile crowdsensing has been proposed, which only requires data from some sensing areas and utilizes spatiotemporal correlation to infer data for other unsensed areas. In real vehicular crowdsensing scenarios, not only the current period sensing data is required to be inferred but also the prediction of the future whole sensing map is of great significance. In this article, we propose multitask pretraining graph transformer networks (MT-PTGTN) that incorporate graph neural networks (GNNs) and transformer to support both data inference and prediction for vehicular crowdsensing. Comprehensively considering the surface and underlying patterns among the sensing grids, MT-PTGTN utilizes pretraining topological mining GNNs and graph attention networks to model two patterns, respectively, which contributes to improving inference accuracy. The transformer with the multilayer attention mechanism is incorporated to capture the temporal correlation of the sensing data and predict the future complete sensing map. Furthermore, to prevent the error propagation from the inference task to the subsequent prediction task, we propose a dynamic multitask learning framework that dynamically adjusts the weights of tasks during training. The experimental evaluation on the real-world dataset demonstrates the superiority of MT-PTGTN in data inference and prediction.**

***Index Terms*—Data inference and prediction, graph attention networks (GATs), transformer networks, vehicular crowdsensing.**

I. Introduction

IN RECENT years, the landscape of technological advancements has witnessed the ubiquitous presence of smart mobile devices and the rapid evolution of the Internet of Things (IoT). A novel paradigm known as mobile crowdsensing (MCS) has surfaced [1]. MCS utilizes widespread smart mobile devices to collect sensing data through conscious or unconscious collaboration among groups and complete large-scale and complex social sensing tasks. Concomitant with the ever-progressing realm of vehicular intelligence and networking, a surge in the integration of high-performance processing units and exquisitely precise sensors within vehicles has become palpable. Cameras, lidar, and millimeter-wave radar are now commonly embedded in intelligent vehicles. As a result, vehicles are steadily transforming into platforms endowed with mobile perception, computing, and storage capabilities [2], [3]. Owing to their wide distribution and fast mobility, vehicles can effectively support large-scale, efficient perception via the vehicle–edge–cloud three-tier architecture. The vehicular crowdsensing system offers several advantages, including the ease of deployment of sensing nodes, fast and extensive coverage of terminals, and robust scalability, thereby making it an ideal solution in MCS for a range of applications, such as traffic monitoring [4], [5], environmental monitoring [6], [7], intelligent transportation [8], [9], and smart cities [10].

To obtain fine-grained, high-quality sensing maps, traditional MCS approaches typically require a large number of mobile participants, such as vehicles, to cover the entirety of the sensing area. However, this approach can result in high costs. Moreover, not all regions can be effectively covered by vehicles with sensing capabilities. To address these issues, sparse MCS has emerged as a promising alternative, garnering significant attention [11], [12]. Within sparse MCS, the sensing area is partitioned into discrete subregions or grids. Sensing tasks are performed only within a subset of these grids, while the correlation between data obtained from these grids is utilized to infer the unsensed grids, resulting in the generation of a complete sensing map. As such, one of the key objectives of sparse MCS is to leverage interdata relationships to infer missing data.

Several studies have been conducted to address the issue at hand. He and Shin [13] and Liu et al. [14] employed compressed sensing techniques and their variants for data reconstruction. Considering spatiotemporal constraints, a matrix completion approach was proposed for inferring missing data [15]. Additionally, generative adversarial networks were utilized to establish data completion models in [16] and [17]. Crucially, the spatiotemporal correlation among sensing data is fundamental for building effective inference models. Since sensing grids in real scenes are often interrelated, some

Manuscript received 14 May 2023; revised 16 July 2023; accepted 29 July 2023. Date of publication 17 August 2023; date of current version 25 December 2023. This work was supported in part by the National Key Research and Development Program of China under Grant 2022YFB2503202; in part by the Beijing Nova Program from Beijing Municipal Science and Technology Commission under Grant Z201100006820123; and in part by the BUPT Excellent Ph.D. Students Foundation under Grant CX2022205. *(Corresponding author: Zhaoming Lu.)*

The authors are with the Beijing Laboratory of Advanced Information Networks and the Beijing Key Laboratory of Network System Architecture and Convergence, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: huojie@bupt.edu.cn; wluhan@bupt.edu.cn; lzy0372@bupt.edu.cn; xiangmw@bupt.edu.cn).

Digital Object Identifier 10.1109/JIOT.2023.3305006

2327-4662 © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Wuhan University. Downloaded on August 27,2024 at 01:31:37 UTC from IEEE Xplore. Restrictions apply.

researchers modeled them as a graph structure and applied graph neural networks (GNNs) to extract complex spatiotemporal correlations among sensing grids [18], [19]. These investigations have considerably improved inference accuracy and reduced system cost while maintaining crowdsensing quality.

However, in sparse MCS, it is crucial to predict a complete map of future cycles in addition to inferring missing data for the current cycle [20], [21], [22]. Especially for applications that require future states, such as traffic congestion monitoring and autonomous driving navigation, where foreknowledge of the forthcoming cycle helps to make decisions in advance and improve service quality. Additionally, predicting the future complete map of data can enable the anticipation of areas that are comparatively more crucial or valuable, thereby allowing for pre-recruitment of vehicles to undertake the perception task. To this end, Wang et al. [21] leveraged deep learning to predict sparse MCS. Wang et al. [18] utilized matrix completion to infer absent data and graph convolutional networks (GCNs) to forecast the future complete map. However, the concurrent consideration of data inference and prediction task presents persistent challenges.

1) Most current methods for data inference and prediction entail a two-step approach, where the inference task is first completed followed by the prediction task. However, the two-step approaches train the inference and prediction models independently, the inference errors will propagate to the subsequent prediction model. Hence, it is necessary to jointly account for the inference task's errors during the prediction task's training.
2) The intricate spatiotemporal correlation between sensing grids provides crucial evidence for data inference and prediction. This correlation is not just a surface pattern, such as path connection and grids adjacency. While most research focuses on the surface pattern in grid modeling, a thorough exploration of the underlying pattern is insufficient.

Given the remarkable modeling ability of GNN for spatial topological features and the success of the transformer in handling time-series problems, we propose a novel model, namely, multitask pretraining graph transformer networks (MT-PTGTN), which combines GNN and transformer to address the aforementioned challenges. The proposed model is developed based on the cloud–edge–vehicle three-layer vehicular crowdsensing architecture, as illustrated in Fig. 1. Specifically, we first employ a pre-trained variable adjacent matrix GCN to capture the underlying patterns among grids, and then integrate graph attention network (GAT) to infer the missing data. Then, the transformer model is introduced to predict the complete future map. Moreover, to avoid error diffusion of the inference task, we jointly train the inference and prediction tasks. The objective of this work is to enhance the accuracy of data inference and prediction, which in turn improves the service quality of vehicular crowdsensing. The main contributions of this article are as follows.

1) We propose a novel pretraining topological mining GNN (TMGNN), which integrates both surface and underlying patterns to efficiently extract topological features. GAT and variable adjacent matrix GCN are employed to model the two patterns, respectively, and their outputs are aggregated to generate the final grid features.
2) We incorporate transformer networks into graph-based data prediction. Leveraging the advantages of its multilayer attention mechanism in time-series forecasting, the temporal correlation of sensing data is captured, enabling efficient and accurate prediction of future complete maps.
3) We propose a dynamic multitask learning framework that quantifies the difficulty of each task by utilizing key performance indicators (KPIs). The weights of inference and prediction tasks are dynamically adjusted during training, resulting in the acquisition of optimal solutions that mitigate error accumulation.
4) We conduct extensive simulations on two typical urban sensing datasets (traffic monitoring and environmental monitoring), and the results demonstrate the superiority, robustness, and applicability of the proposed MT-PTGTN scheme.

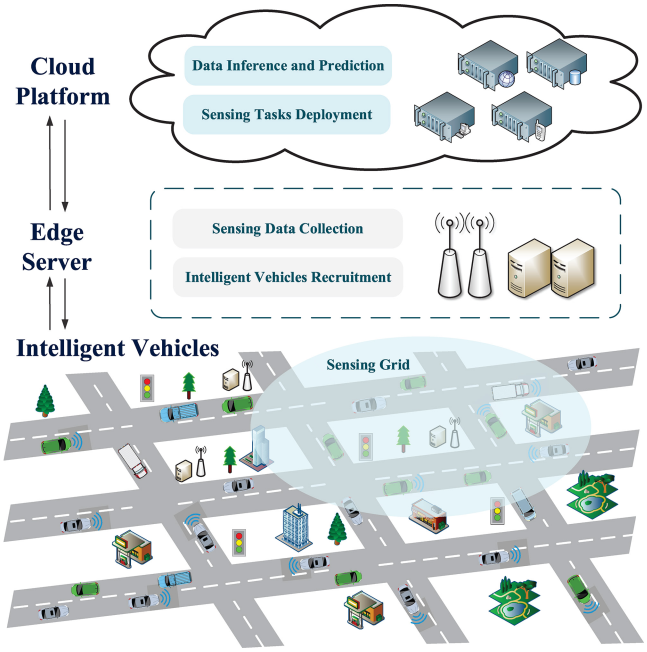

Fig. 1. Architecture of the vehicular crowdsensing system.

The remainder of this article is organized as follows. Section II reviews related works and highlights the key distinctions and improvements of our work. Section III elaborates on the system model and problem formulation. In Section IV, the MT-PTGTNs are designed, and their architecture is explained. Section V reports and discusses the experiment's results. Finally, Section VI concludes this article.

II. Related Work

In this section, we discuss related work on data inference and prediction for MCS.

A. Data Inference for MCS

Sparse MCS aims to reduce the overhead by reducing the number of sensing tasks [12], [23], resulting in a more cost-effective and practical solution for real-world scenarios.

Authorized licensed use limited to: Wuhan University. Downloaded on August 27,2024 at 01:31:37 UTC from IEEE Xplore. Restrictions apply.

For the uncovered areas, their corresponding values can be deduced from the collected sensing values. Therefore, missing data inference is a key component of Sparse MCS [18], [24], [25], [26]. Traditional data inference methods have predominantly focused on linear relationships between grids. For instance, Pan and Li [27] and Marchang and Tripathi [28] employed K-nearest neighbor (KNN) and its variants to infer missing data in sparse MCS. Additionally, He and Shin [13] and Xie et al. [29] mapped the sensing data into a matrix and applied the method of matrix completion to recover the missing data. Recent developments in deep learning have led to a proliferation of methods for restoring missing data in sparse MCS. Awan et al. [16] proposed a conditional generative adversarial network (CGAN) that leverages class-specific distributions to impute missing data, thereby generating optimal estimates for missing values. Zhang et al. [17] designed an end-to-end GAN model that can impute missing values in multivariate time series. Lin et al. [30] compared missing value imputation for two supervised deep neural networks, a multilayer perceptron (MLP), and a deep belief network (DBN). Furthermore, Zhi-Xuan et al. [31] introduced a factorized inference method for multimodal deep Markov models that can be applied to multimodal data completion.

For large-scale urban sensing, numerous research works have proposed data inference solutions [18], [26], [32]. One such work for air quality assessment is the surrounding environment perception system UbiAir, proposed by Wu et al. [26], which supports fine-grained urban air quality monitoring through MCS on the shared bicycle system. It employs Bayesian compressive sensing as an inference model to reconstruct an accurate air quality map covering the entire monitoring space. Wang et al. [18] combined traditional deep matrix factorization and GAN to generate spatiotemporal crack data and propose an algorithm called generative high-fidelity matrix completion. Marchang et al. [32] explored methods, such as linear regression, elastic network, ridge, and decision tree for regression on missing data inference in sparse MCS, and proposed a divide-and-conquer polynomial-time algorithm to reduce sensing tasks. However, while these works have made notable strides in improving the accuracy of data inference in large-scale urban sensing, most of them only focus on the surface relationship between sensing grids. The underlying correlations between the grids require further mining.

B. Data Prediction for MCS

In some large-scale MCS applications, users not only seek to infer the data of other unsensed subregions in the current sensing period but also wish to predict the full-image data of the future sensing period [21], [22]. Such predictions provide more information for users' decision-making, which is very important for improving the service quality of MCS. Chen et al. [33] introduced PAS, a prediction-based driving system that predicts potential near-term vehicle routes and the probability of ride requests across a city. Chan et al. [34] proposed a pheromone-based neural network traffic prediction and rerouting system and designed a weighted missing data imputation (WEMDI) method to deal with missing data. Recently, some works consider both missing data inference and future prediction in MCS [18], [21], [22], [35]. Wang et al. [21] proposed a deep learning industrial perception and forecasting scheme based on sparse MCS, which consists of two parts: matrix completion and future prediction and is implemented by the deep matrix decomposition method and nonlinear autoregressive neural network, respectively. By proposing Bi-GAN, Gupta and Beheshti [35] used a bidirectional recurrent network in a generative adversarial environment for the combined task of imputing and predicting values for irregular observations and time-series data of varying lengths. Liu et al. [22] proposed a sparse MCS urban forecasting scheme consisting of matrix completion and near-term forecasting. Here, a bipartite graph-based matrix completion algorithm with spatiotemporal constraints is proposed to recover unsensed data accurately, and then a neural network-based continuous conditional random field is proposed to learn to predict future data. Wang et al. [18] proposed a framework for urban inference and prediction in sparse MCS. The framework uses a matrix completion algorithm for bipartite graphs to recover the current complete map, and based on this, a GCN with spatiotemporal attention is proposed to predict future maps.

Despite numerous studies proposing solutions for data inference and prediction in MCS, most of these studies perform these tasks separately. The error of the inference task will be propagated to the subsequent prediction, thereby reducing prediction accuracy. In contrast to the existing methods, the MT-PTGTN proposed in this article adopts a multitask framework for joint training, which helps reduce the impact of error propagation. Furthermore, the model leverages a pre-trained GNN to mine the hidden relationships among grids, thereby facilitating data inference and prediction. The primary objective of this article is to enhance the accuracy of vehicular MCS data inference and prediction and elevate the service quality of large-scale urban sensing.

III. System Model and Problem Formulation

In this section, we first introduce the system architecture of vehicular crowdsensing. Afterward, we analyze and formulate the data inference and prediction problem.

The vehicular crowdsensing system, illustrated in Fig. 1, is mainly consists of three parts.

1) *Cloud Platform:* The vehicular MCS cloud platform serves as the brain of the entire system, boasting impressive computing and storage capabilities, which enables the inference and prediction of sensor data across the entire city and provides support for the management and deployment of the sensing tasks.
2) *Edge Server:* The edge server is responsible for recruiting vehicles to complete MCS tasks. Upon recruitment, the edge server then facilitates the collection of sensing data from the vehicles within a particular area. These data are then processed to extract sensing feature values within their coverage areas, which are then reported back to the cloud platform.

Authorized licensed use limited to: Wuhan University. Downloaded on August 27,2024 at 01:31:37 UTC from IEEE Xplore. Restrictions apply.

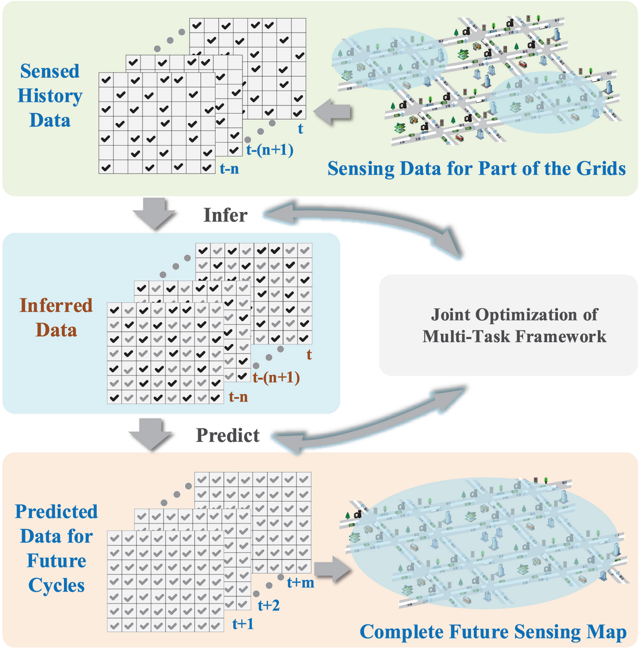
Sensed History Data
t
t-(n+1)
t-n
Sensing Data for Part of the Grids
Infer
Inferred Data
t
t-(n+1)
t-n
Joint Optimization of Multi-Task Framework
Predict
Predicted Data for Future Cycles
t+m
t+2
t+1
Complete Future Sensing Map

Fig. 2. Data inference and prediction process in vehicular crowdsensing system.

3) *Intelligent Vehicles:* Intelligent vehicles are equipped with various sensors, such as lidar, and the wireless communication module, which are responsible for the physical execution of sensing tasks and the uploading of sensing data.

The process of data inference and prediction is shown in Fig. 2. In the large-scale urban crowdsensing, the entire sensing map is usually divided into several sensing grids (or subregions). There are complex correlations between sensing grids. This set of associated points can be regarded as a kind of non-Euclidean data, which is constructed as a graph structure. To be specific, the urban sensing network is regarded as a graph $\mathcal{G} = (\mathbf{V}, \mathbf{E})$. The vertex set $\mathbf{V}$ consists of sensing grids. If grids V_i and V_j are adjacent (or directly connected by a road), the two grids are considered to be neighbors and connected by edge e_{ij}. That is, if V_i and V_j are neighbors, the value of e_{ij} is 1, otherwise it is 0. The edges between adjacent grids form the set $\mathbf{E}$. For the historical cycle, it can be divided into sensed grids and unsensed grids according to whether they are perceived or not. The input of our model is the sensed grid measurement value of several historical cycles, that is, $X_{\text{his}} \in R^{N_{\text{his}} \times N_{\text{sen}}}$. Here, N_{his} is the number of historical cycles, and N_{sen} is the number of sensed grids in each cycle. Therefore, the missing data inference process can be expressed as

$$I(\mathbf{X}_{\text{his}}) = \widehat{\mathbf{Y}}_{\text{inf}}. \tag{1}$$

The error in the inference phase is measured by the mean absolute value error (MAE), which can be calculated as

$$\varepsilon_I = \frac{1}{N_{\text{his}} N_{\text{uns}}} \left(\sum_{h=1}^{N_{\text{his}}} \sum_{i=1}^{N_{\text{uns}}} \left| y_h[i] - \hat{y}_h[i] \right| \right) \tag{2}$$

where N_{uns} is the number of unsensed grids. After completing the missing data, we conduct predictions for the complete sensing map for the next few cycles. The data prediction stage is based on the completed historical map. Similarly, its process and error can be expressed as follows:

$$P(I(\mathbf{X}_{\text{his}})) = P\left(\widehat{\mathbf{Y}}_{\text{inf}}\right) = \widehat{\mathbf{Y}}_{\text{pre}} \tag{3}$$

$$\varepsilon_P = \frac{1}{N_{\text{pre}} N_{\text{grids}}} \left(\sum_{t=1}^{N_{\text{pre}}} \sum_{i=1}^{N_{\text{grids}}} \left| y_t[i] - \hat{y}_t[i] \right| \right) \tag{4}$$

where $\hat{Y}_{\text{pre}} \in R^{N_{\text{pre}} \times N_{\text{grids}}}$ is the output of the prediction stage, and $\hat{y}_t[i]$ is the grid V_i of the tth cycle. And N_{pre}, N_{grids} are, respectively, defined as the number of predicted historical periods and the total number of sensing grids in the area.

This article aims to improve the accuracy of data inference and prediction in urban sensing. Therefore, given the sparse sensing data of several historical periods, the goal of this article is to infer the missing data and predict the value of future periods. Considering that the data inference task will cause error propagation to the prediction task, we employ a multitask learning framework to dynamically adjust the weights of the two tasks. The optimization goal of the MT-PTGTN is to minimize the weighted error of the two tasks

$$\min \quad w_I \varepsilon_I + w_P \varepsilon_P \tag{5}$$

$$\text{s.t.} \quad P(I(\mathbf{X}_{\text{his}})) = \widehat{\mathbf{Y}}_{\text{pre}} \tag{6}$$

where w_I and w_P are the weights of the inference task and the prediction task, respectively. The multitask learning framework and weight adjustment method will be introduced in detail in the next section.

IV. MT-PTGTN: Multitask Pretraining Graph Transformer Networks

In this section, we describe the design details of the MT-PTGTN. Fig. 3 illustrates the detailed framework of our model. First, we propose a TMGNN to infer missing data through extracting surface and underlying patterns between grids (Section IV-A). Second, we introduce the transformer to capture the temporal correlation between sensed data and predict a complete map of future cycles (Section IV-B). Finally, we design a dynamic multitask learning framework to jointly train the inference and prediction tasks (Section IV-C).

A. Topological Mining GNN

The accurate inference of missing data in grid-based MCS systems relies heavily on the mining of the topological correlations between the grids. These correlations can be classified into two categories, namely, surface and underlying patterns. A surface pattern between grids is characterized by a direct connection. For instance, in urban sensing scenarios, a surface pattern can be defined as the adjacent position or road link between two sensing grids. Obviously, the surface pattern describes an explicit relationship that can be directly obtained and counted. However, feature relevance between grids may not only depend on their surface pattern. Some nonadjacent or roadless grids also have significant correlations, so we characterize the feature as the underlying pattern.

Authorized licensed use limited to: Wuhan University. Downloaded on August 27,2024 at 01:31:37 UTC from IEEE Xplore. Restrictions apply.

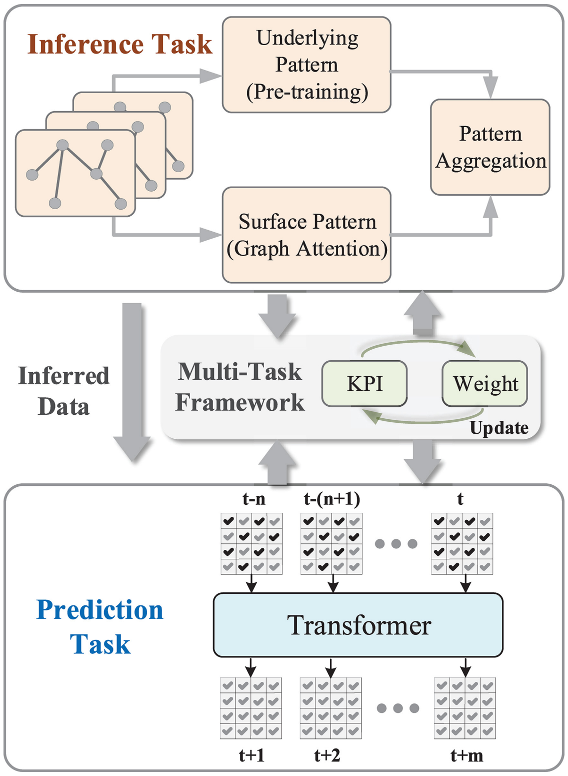

Fig. 3. Framework of MT-PTGTN.

To explore and capture the above surface and underlying patterns, this article proposes a TMGNN based on a pretraining framework. Specifically, we adopt a variable adjacency matrix GCN to mine the surface and potential associations between grids. And utilize GAT to mine the weighted detail surface pattern with the attention mechanism. The proposed MT-PTGTN takes advantage of GCNs in modeling graph structures, which can effectively capture the relationship between the nodes and their neighbors, and further extends it to identify the underlying patterns in the pretraining process. The model is capable of capturing both local and global dependencies between nodes, allowing it to discover subtle yet important relationships in the data. The variable adjacent matrix GCN can be expressed as

$$H^{(l+1)} = \sigma\left(A_{\mathrm{und}}H^{(l)}W^{(l)}\right) \tag{7}$$

where A_{und} is the trainable correlation adjacent matrix, initialized with $\hat{D}_{\mathrm{sur}}^{-(1/2)}\hat{A}\hat{D}_{\mathrm{sur}}^{-(1/2)}$, and $\hat{A}_{\mathrm{und}} = A_{\mathrm{und}} + I$. A_{und} is shared among different adjacent matrix variable GCN layers. And W is a trainable weight matrix in the pretraining process, the adjacent matrix is set as a variable. Its optimization result $\hat{A}_{\mathrm{und}}$ represents the underlying pattern of grids.

The surface pattern indicates the direct connections between nodes. This surface form plays an intuitional and diverse role in graph data inference. Therefore, for each grid, it is necessary to pay different attention to their surface from neighbors. To this end, GAT is utilized to mine the importance weights of neighbor nodes. Specifically, GAT employs a self-attention mechanism to conduct a weighted combination for feature aggregation in forward propagation. And these aggregation weights assign different importance within the neighborhood of surface form.

The attention coefficient a_{ij} indicates the importance of node j relative to node i, which can be expressed as

$$a_{ij} = \mathrm{attention}\left[Wv_i, Wv_j\right] \tag{8}$$

where a_{ij} is realized by the self-attention function

$$\mathrm{attention}\left[Wv_i, Wv_j\right] = \mathrm{LeakyReLU}\left(a^T\left[Wv_i \| Wv_k\right]\right) \tag{9}$$

where $\|$ represents the concatenation operation between two matrices. a^T is the transposition of a, which is denoted as the learnable parameters. A shared weights variable is denoted as W. And the LeakyReLU activation function is utilized to encapsulate this process.

By convention, we introduce the softmax function to normalize the attention coefficients so that they are comparable across various nodes

$$\alpha_{ij} = \mathrm{softmax}\left(a_{ij}\right) = \frac{\exp\left(a_{ij}\right)}{\sum_{k\in N_i}\exp\left(a_{ik}\right)} \tag{10}$$

where $j \in N_i$, and N_{i} is the neighborhood of node i, including node i itself. Therefore, the attention score can be calculated totally as follows:

$$\alpha_{ij} = \frac{\exp\left(\mathrm{LeakyReLU}\left(a^T\left[Wv_i \| Wv_j\right]\right)\right)}{\sum_{k\in N_i}\exp\left(\mathrm{LeakyReLU}\left(a^T\left[Wv_i \| Wv_k\right]\right)\right)}. \tag{11}$$

To merge the two pattern representations, we conduct a weighted combination of the attention coefficient matrix and the underlying matrix as the final topological pattern representation. Based on this representation, the TMGNN both shares the same input features $H^{(l)}$ and layer-specific trainable weight matrix $W^{(l)}$ as GAT, and the aggregation of adjacent matrices for the two patterns can be expressed mathematically as

$$H^{(l+1)} = \sigma\left(\left(\beta\bar{A}_{\mathrm{cor}} + (1-\beta)A_{\mathrm{att}}\right)H^{(l)}W^{(l)}\right) \tag{12}$$

where $\hat{A}_{\mathrm{und}}$ is the pre-trained underlying correlation matrix, and A_{att} represents the attention matrix.

By integrating both surface and underlying patterns, our proposed TMGNN provides a more comprehensive approach to modeling the global topological feature of grids, enabling the accurate inference of missing data.

B. Transformer for Sensing Data Prediction

In the stage of predicting sensing data, we leverage the transformer network's robustness in handling lengthy sequential data, allowing us to capture the temporal dependencies present within the sensing data and anticipate future data. The transformer architecture comprises two essential components: 1) an encoder and 2) a decoder [36]. First, to make use of the sequence order information, the position embedding is added to the bottoms of the encoder and decoder stacks. Then, the encoder and decoder both incorporate stacked self-attention and point-wise fully connected layers. The stacked self-attention mechanism empowers the model to focus on relevant elements within the input data, without being limited

Authorized licensed use limited to: Wuhan University. Downloaded on August 27,2024 at 01:31:37 UTC from IEEE Xplore. Restrictions apply.

by any fixed temporal order. Moreover, stacked with a multi-head self-attention mechanism, the position-wise feed-forward network is another sublayer of the encoder and decoder. The key components of the transformer will be introduced in more detail below.

1) Position Embedding: In the transformer, positional embedding is introduced to inject the order information of the sequence. The position embedding can be calculated as

$$\vec{P}_j = \left(PE_{(j,1)}, PE_{(j,1)}, \ldots, PE_{(j,d_{\text{model}})}\right) \tag{13}$$

$$PE_{j,2i} = \sin\left(\frac{j}{10000^{2i/d_{\text{model}}}}\right) \tag{14}$$

$$PE_{j,2i+1} = \cos\left(\frac{j}{10000^{2i/d_{\text{model}}}}\right) \tag{15}$$

where j is the position index and i is the embedding dimension.

2) Multi-Head Attention: The attention layer is a core component of the transformer. The particular attention in this model is Dot-Product Attention, which consists of three parts: 1) query Q with dimension d_k; 2) key k with dimension d_k; and 3) value V with dimension d_v, as follows:

$$\text{Attention}(Q, K, V) = \text{Softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V. \tag{16}$$

Multi-head attention is carried out in the transformer to enable the model to participate in various representation subspaces. The multi-head attention does not rely on a single attention function that maps queries, keys, and values to d_{model} dimensional vectors, but involves linear projections of queries, keys, and values N_h times. Each projection is different and projects the query, key, and value to the d_k, d_k, and d_v dimensions, respectively. After the projection is complete, the attention function is executed in parallel with the projected version. Therefore, each attention function produces a d_v dimensional output. These outputs are then concatenated and projected again to produce the final output, which can be expressed as

$$\text{MultiHead}(Q, K, V) = \text{Concat}\left(\text{head}_1, \ldots, \text{head}_{N_h}\right)W^O \tag{17}$$

$$\text{head}_i = \text{Attention}\left(QW_i^Q, KW_i^K, VW_i^V\right) \tag{18}$$

where $W_i^Q \in R^{d_{\text{model}} \times d_k}$, $W_i^K \in R^{d_{\text{model}} \times d_k}$, $W_i^V \in R^{d_{\text{model}} \times d_k}$ and $W^O \in R^{hd_v \times d_{\text{model}}}$ are the projection matrix to be learned. The primary advantage of multi-head attention is to focus on information from different representation subspaces at each location. In other words, multi-head attention enables the model to jointly concentrate on multiple aspects of the input sequence, effectively capturing complex correlations in the temporal dimension of sensing data.

3) Position-Wise Feed-Forward Networks: Each layer in the encoder and decoder also contains a fully connected feed-forward neural network that is applied to each location separately and identically. It can be expressed as

$$\text{FFN}(\vec{x}) = \max\left(\vec{0}, \vec{x}W_1 + \vec{b}_1\right)W_2 + \vec{b}_2. \tag{19}$$

Based on graph-based data inference, the temporal correlation of sensing data can be captured through the transformer, resulting in generating the future complete sensing map.

C. Dynamic Multitask Framework

In most cases, the data prediction for future cycles relies on the inferred full sensing map. However, if the training processes of the inference and prediction model are independent, the risk of overfitting and error propagation in the inference model may impact the following prediction results. To address this issue, this article proposes a multitask learning framework, which dynamically adjusts the weights assigned to both the inference and prediction tasks throughout the training process. The primary objective of this framework is to mitigate error accumulation stemming from the inference task, thereby bolstering the accuracy of the prediction task.

Inspired by dynamic task prioritization [37], this article chooses to prioritize learning "difficult" tasks by assigning them higher task-specific weights, thus facilitating the learning process. A KPI is employed to quantify the difficulty of each task. It is worth noting that the selection of KPIs is crucial and should have an intuitive meaning. In the prediction task, we adopt the MAE as the KPI. The inference and prediction errors are thresholded to obtain a KPI between 0 and 1. The KPI κ of task i can be expressed as

$$\kappa_i(t) = \begin{cases} 1, & \text{if } \text{error}_i(t) \geq 1 \\ \text{error}_i(t), & \text{otherwise} \end{cases}$$

where t represents the tth training round, and error_i is the MAE loss for task i. Then, the moving average model is utilized for $\kappa_i(t)$

$$\bar{\kappa_i}(t) = \varphi\kappa_i(t) + (1 - \varphi)\bar{\kappa_i}(t-1) \tag{20}$$

where φ is denoted as the decay factor, $\varphi \in [0, 1]$. φ determines the update speed of the model, and the larger the φ, the more stable it becomes. Therefore, the weight $w_i(t)$ of task i is set as

$$w_i(t) = -(1 - \bar{\kappa_i}(t))^{\gamma_i} \log \bar{\kappa_i}(t) \tag{21}$$

where γ_i is a task-level focusing parameter that adjusts the weight of easy or difficult tasks. As the KPI value κ_i increases, the weight of task i is decreased. In other words, the greater the error of the task, the lower its KPI, so its weight is higher and its priority is to be learned. Finally, to ensure the stability of training process, the weight of task i is normalized as

$$\bar{\omega}_i(t) = \frac{\omega_i(t)}{\sum_j \omega_j(t)}. \tag{22}$$

Therefore, given task-specific weights w_i and task-specific loss functions L_i, the multiobjective optimization problem in this article can be formulated as

$$\mathcal{L}^*(t) = \sum\nolimits_i \omega_i(t) \cdot \mathcal{L}_i(t) \tag{23}$$

The dynamic multitask learning framework process is shown in Algorithm 1. First, a new adjacency matrix is generated to represent the underlying pattern between grids through a pretraining process. Then, we perform pattern aggregation to combine the surface pattern with the underlying pattern and infer missing data, where the surface pattern is realized by the GAT networks. Then, based on the completed data, the transformer model is used for data prediction. During the training

Authorized licensed use limited to: Wuhan University. Downloaded on August 27,2024 at 01:31:37 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1 Dynamic Multitask Learning Framework Process

```
Input: The sensed data $\mathbf{X}_{his}$ for several periods of history.
Output: The inferred data $\hat{\mathbf{Y}}_{inf}$ and the predicted data $\hat{\mathbf{Y}}_{pre}$;
Pretraining: Training $A_{und}$ initialized with $\hat{D}_{sur}^{-\frac{1}{2}}\hat{A}\hat{D}_{sur}^{-\frac{1}{2}}$, and
    obtain the fixed adjacent matrix $\hat{A}_{und}$;
for $t$ in multitask training rounds do
    Carry out historical missing data inference process based on surface pattern and underlying pattern, and obtain the inferred value $\hat{\mathbf{Y}}_{inf}$;
    Based on the completed historical data, use the transformer to predict the complete map of the future cycle and obtain $\hat{\mathbf{Y}}_{pre}$;
    Update KPIs and weights for inference and prediction tasks according to Eq. (20) and (21);
    Optimize multitask framework jointly according to Eq. (23);
end for
return: $\hat{\mathbf{Y}}_{inf}$ and $\hat{\mathbf{Y}}_{pre}$
```

process, we employ a dynamic multitask learning framework to continuously update the KPIs and weights of the inference and prediction tasks to facilitate joint training and prevent error propagation underlying, thus enabling adaptive learning between tasks.

V. Performance Evaluation

In this section, we first introduce the adopted dataset and experimental settings. Then, we present the results of data inference and prediction and compare our proposed scheme with other baseline methods.

A. Dataset and Settings

To ensure a more realistic evaluation of our proposed scheme, we select two urban sensing datasets that represent two typical scenarios of traffic monitoring and environmental monitoring. The details of the two datasets are as follows.

1) *TaxiSpeed* [38] dataset is real traffic speed data collected from GPS trajectories of taxis in Beijing, China. We use 40 adjacent intersections as a benchmark to divide the grid, and the entire traffic network is divided into 1578 sensing grids. The average vehicle speed of each grid is regarded as sensing data. For each grid, the average vehicle speed is calculated as its sensing data.
2) *U-Air* [39] is an air quality dataset that consists of important air quality data collected from monitoring stations deployed in Beijing, China. The dataset covers a total of 36 areas, measuring environment variables, such as PM2.5 and SO_2. The most representative PM2.5 indicator is selected for performance evaluation.

Table I presents the statistical details of the two urban sensing datasets. Both datasets' sensing data can be gathered by intelligent vehicles, which are appropriate in vehicular MCS scenarios. As the PM2.5 data exhibit significant fluctuations, logarithmic processing is performed on it. The parameter setting details of our proposed MT-PTGTN scheme are shown in Table II.

TABLE I
Statistics of Two Sensing Datasets

Statistics Type	TaxiSpeed	U-Air
City	Beijing	Beijing
Data Type	Urban Traffic	Urban Environment
Data	Traffic Speed	PM2.5
Cycle	1 hour	10 minutes
Duration Time	3 days	49 days
Grid Size	40 adjacent intersections	$1000 * 1000m^2$
Grid Number	36	1578

TABLE II
Parameter Settings in MT-PTGTN

Parameter	Value
Variable Adjacenct Matrix GCN Layer	2
GAT Layer	2
Learning Rate	0.005
Dropout Rate	0.1
GNN Activation Function	LeakyReLu
Attention Heads of the Encoder (Transformer)	3
Attention Heads of the Decoder (Transformer)	3
Encoder Layer (Transformer)	2
Decoder Layer (Transformer)	2
Optimizer	Adam

To evaluate the performance of the scheme on data inference, a subset of grids in the dataset is randomly selected and its corresponding sensing data is used as input to the inference model. The data inference model is then employed to complete the remaining grids with missing data. This article mainly compares the following inference methods.

1) *KNN:* This method assigns the mean value of the KNNs in the feature space as the inferred value for a missing data point [27].
2) *K-Means-Based Gaussian Mixture Process (K-GMP):* The grids are first divided into k clusters using K-means clustering, which then models each cluster as a Gaussian mixture distribution [40]. The EM algorithm is then used to estimate the parameters of each distribution, and the mean value is calculated as the estimated value for each cluster.
3) *GCN:* This method employs GCN [41] to impute missing data, where the adjacent matrix is nontrainable.
4) *GAT:* This method utilizes GAT [42] to infer missing data.

In the data prediction phase, we considered classic time series prediction models, such as recurrent neural networks (RNNs) and long short-term memory (LSTM). Since data prediction relies on the results of missing data completion, several data inference and prediction methods are combined for a comprehensive comparison, as follows.

1) *GCN-Transformer* utilizes GCN to infer missing data and applies transformer models to predict the complete map of future cycles.

Authorized licensed use limited to: Wuhan University. Downloaded on August 27,2024 at 01:31:37 UTC from IEEE Xplore. Restrictions apply.

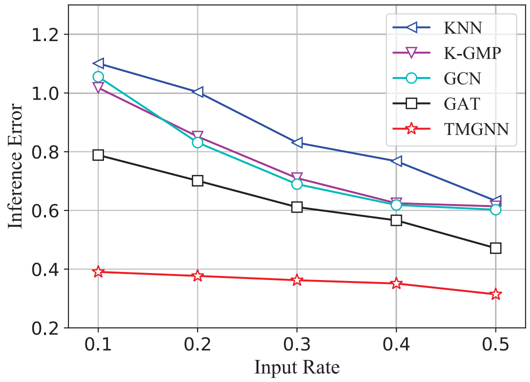

Fig. 4. Inference accuracy under different input rates in the TaxiSpeed dataset.

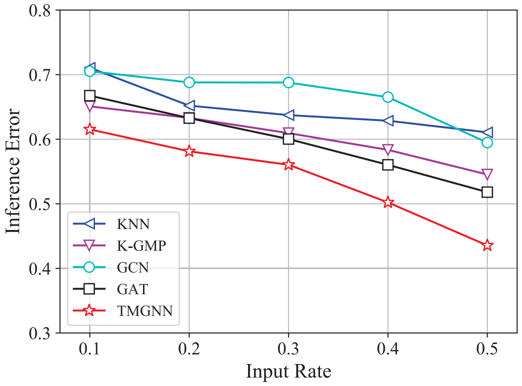

Fig. 5. Inference accuracy under different input rates in the U-Air dataset.

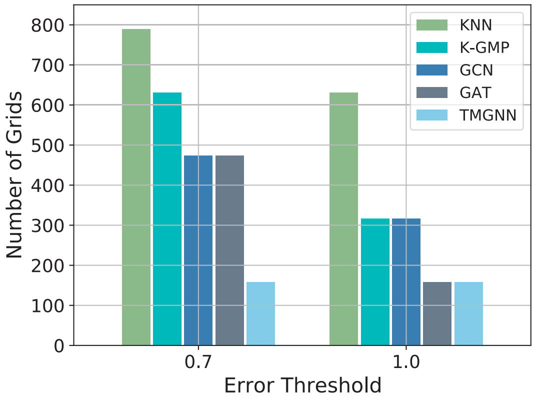

Fig. 6. Number of sensing grids required to achieve different inference error thresholds in the TaxiSpeed dataset.

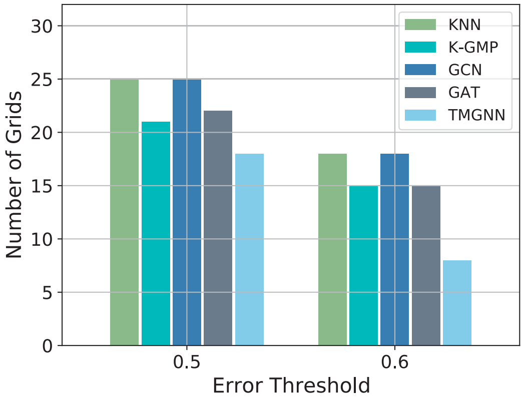

Fig. 7. Number of sensing grids required to achieve different inference error thresholds in the U-Air dataset.

2) *GAT-Transformer* employs GAT and transformer to complete data inference and prediction, respectively.
3) *TMGNN-RNN* is based on our proposed TMGNN for data inference. And relying on the inference results, RNN is used for complete map prediction of future cycles.
4) *TMGNN-LSTM* employs TMGNN and LSTM for data inference and prediction, respectively.
5) *TMGNN-Transformer* uses TMGNN and transformer for data inference and prediction, respectively, without multitask joint training.

B. Results of Data Inference

In terms of data inference, we first compare the performance of several inference algorithms given various input rates. In this article, input rate refers to the ratio of sensed grids to the total number of grids. The input rate is varied between 0.1 and 0.5, meaning that only 10% to 50% of the grids are sensed during each perception cycle. Moreover, the accuracy of inference results is evaluated using the mean absolute error (MAE). Figs. 4 and 5 exhibit the inference accuracy of different methods on the TaxiSpeed and U-Air datasets, respectively. As the input rate progressively increases, the corresponding inference error for each model exhibits a decline. This is because a larger input rate means more real sensing information has been obtained. With these data, richer intergrid correlations can be learned, consequently culminating in the attainment of more precise inference outcomes. From the figures, it is evident that our proposed MT-PTGTN algorithm attains the lowest MAE under various input rates. This outcome is attributed to the proposed underlying pattern and surface pattern relationship mining, which captures intergrid associations more comprehensively, thus effectively enhancing the inference accuracy. Our model performs well on inference accuracy, particularly when the input rate is 0.1, demonstrating the suitability of our method in sparse crowdsensing scenarios with a limited number of sensing grids.

To assess system performance in terms of cost, we conducted a comparative analysis of the number of sensing grids required by various methods to achieve the same error threshold, as depicted in Figs. 6 and 7. The abscissa is the error (MAE) threshold, and the ordinate is the number of grids required to be sensed below the error threshold. It is noteworthy that, to streamline the analysis process, we assume that the edge server can recruit a sufficient number of intelligent vehicles to complete the urban sensing task. This implies that all sensing tasks can be accomplished, and the number of grids can be deemed as the system cost. The number of grids increases when the error threshold is lowered because achieving lower errors requires more accurate inferences from the model. Given that our proposed MT-PTGTN exhibits superior inference accuracy, when reaching the same error threshold, MT-PTGTN requires the least number of grids compared to other methods. The results from Figs. 6 and 7 demonstrate

Authorized licensed use limited to: Wuhan University. Downloaded on August 27,2024 at 01:31:37 UTC from IEEE Xplore. Restrictions apply.

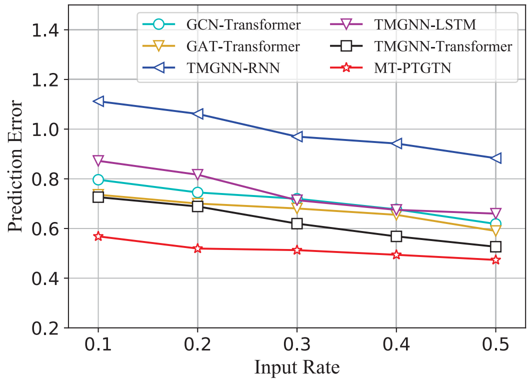

Fig. 8. Prediction accuracy under different input rates in the TaxiSpeed dataset, where the number of input historical cycles is 2 and the number of predicted future cycles is 4.

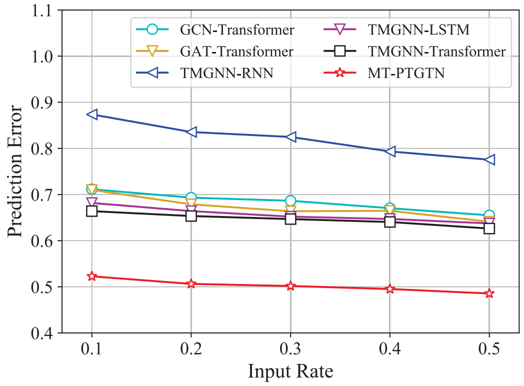

Fig. 9. Prediction accuracy under different input rates in the U-Air dataset.

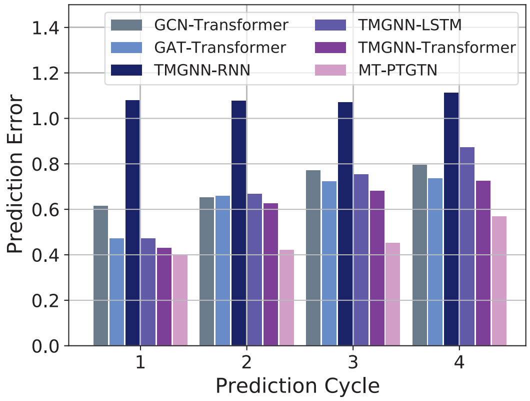

Fig. 10. Prediction accuracy for different numbers of prediction cycles in the TaxiSpeed dataset.

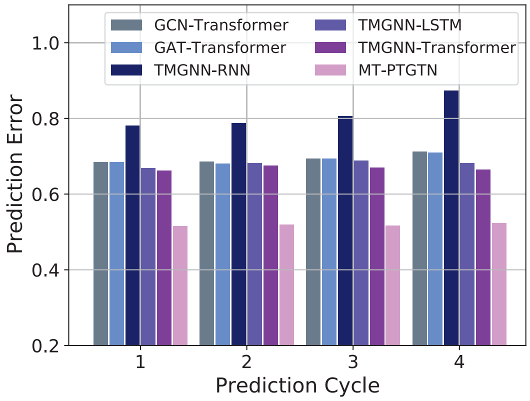

Fig. 11. Prediction accuracy for different numbers of prediction cycles in the U-Air dataset.

that MT-PTGTN performs better on both datasets, improving inference quality while reducing system cost.

C. Results of Data Prediction

In terms of data prediction, Figs. 8 and 9 present a comparative analysis of the prediction errors between MT-PTGTN and other prediction schemes across various input rates. The analysis is conducted using two-cycle historical data and a four-cycle predicted future data configuration. The input rate is also varied from 0.1 to 0.5, similar to the data inference task. It can be seen from the figures that as the input rate increases, the prediction error of each scheme decreases gradually. This can be attributed to the availability of more real data and their richer intercorrelations, which enhances the accuracy of future cycle forecasts. The performance of MT-PTGTN is better than that of the TMGNN-Transformer scheme, which does not utilize the dynamic multitask learning framework. The reason for this result is that the dynamic multitask learning framework of MT-PTGTN continuously updates the KPIs and weights of inference and prediction tasks and obtains tradeoffs during the training process. Therefore, MT-PTGTN avoids the error accumulation from the inference task to the prediction task, thereby reducing the error rate of the prediction task. The results reveal that MT-PTGTN consistently outperforms other prediction schemes across different input rates, exhibiting the lowest prediction error. Notably, the model exhibits substantial improvement when the input rate is set to 0.1, indicating its efficacy in leveraging smaller historical data to achieve precise predictions of future data cycles. These outcomes suggest that the proposed MT-PTGTN scheme is capable of efficiently enhancing prediction accuracy, thus underscoring its superiority over existing prediction schemes.

To evaluate the performance of MT-PTGTN and other schemes in predicting cycles of varying lengths, we analyze prediction errors across cycles 1 to 4. For this experiment, the historical period length is set to 2 and the input rate is 0.1. Figs. 10 and 11 present the experimental results, which demonstrate that MT-PTGTN consistently outperforms other schemes in terms of prediction accuracy, across both datasets, regardless of the number of prediction cycles. The errors of each scheme increase with the length of the prediction cycle due to the higher uncertainties of the sensing map with longer periods. It can be observed that our model's prediction error does not significantly increase as the number of prediction cycles increases. These results can be linked to the transformer's multilayer attention mechanism, which enables it to more deeply understand the temporal correlation between data, aiding its excellent prediction of long-period future data. The transformer's ability to leverage complex temporal dependencies is instrumental in enhancing the model's prediction accuracy and making it robust to variations in cycle lengths.

Authorized licensed use limited to: Wuhan University. Downloaded on August 27,2024 at 01:31:37 UTC from IEEE Xplore. Restrictions apply.

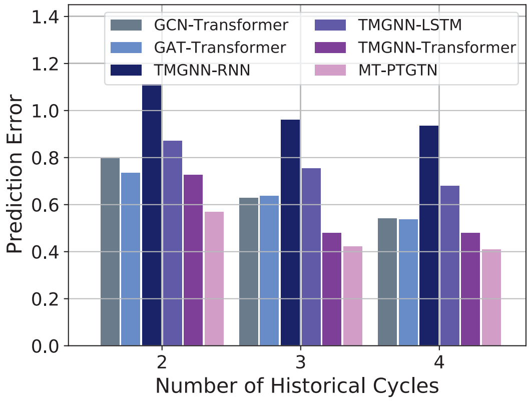

Fig. 12. Prediction accuracy for different numbers of historical cycles in the TaxiSpeed dataset.

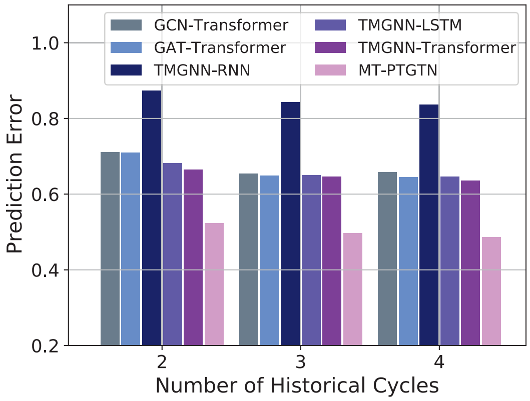

Fig. 13. Prediction accuracy for different numbers of historical cycles in the U-Air dataset.

In a similar vein, this article assesses the errors of several schemes by inputting historical data of different lengths, as illustrated in Figs. 12 and 13. In this experiment, the prediction cycle length is set to 4. The findings demonstrate that MT-PTGTN surpasses other methods in terms of prediction accuracy when the input length ranges from 2 to 4. Furthermore, even with a small number of input cycles, the error of MT-PTGTN remains low, highlighting its ability to generate relatively accurate predictions with minimal data. This outcome can be attributed to several factors: 1) our model mines the surface and underlying relationship between grids; 2) the transformer captures the temporal correlation; 3) the multitask learning framework dynamically adjusts the inference and prediction task weights to avoid error accumulation from the inference task to the prediction task. Overall, the experimental results demonstrate that MT-PTGTN exhibits excellent performance in both data inference and prediction tasks, which effectively improves the quality of sensing data inference and prediction and reduces the cost of vehicular MCS systems.

VI. Conclusion

In this article, we investigate data inference and prediction in vehicular crowdsensing and propose MT-PTGTNs. In terms of data inference, we propose a topological pattern mining GNN model, which considers both surface pattern and underlying pattern to complete missing data. In terms of data prediction, the transformer is introduced to predict the complete future sensing map based on the inferred results. Furthermore, we design a dynamic multitask learning framework to jointly train inference and prediction tasks, and achieve a tradeoff to avoid error propagation. Finally, we conduct extensive experimental evaluations on two typical real-world datasets. The results show that compared with other schemes, the proposed MT-PTGTN can achieve lower MAE for the inference and prediction tasks, which effectively improves the accuracy of vehicular crowdsensing data inference and prediction. In the future, we will further explore sensing data prediction for longer periods. Moreover, the selection of grids and the allocation of sensing tasks play indispensable roles within the domain of vehicle crowdsensing. Based on data inference and prediction, subsequent grid selection and task assignment are also promising directions for future research endeavors.

References

[1] A. Capponi, C. Fiandrino, B. Kantarci, L. Foschini, D. Kliazovich, and P. Bouvry, “A survey on mobile crowdsensing systems: Challenges, solutions, and opportunities,” *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2419–2465, 3rd Quart., 2019.

[2] L. Liu et al., “Evenness-aware data collection for edge-assisted mobile crowdsensing in Internet of Vehicles,” *IEEE Internet Things J.*, vol. 10, no. 1, pp. 1–16, Jan. 2023.

[3] L. Pu, X. Chen, G. Mao, Q. Xie, and J. Xu, “Chimera: An energy-efficient and deadline-aware hybrid edge computing framework for vehicular crowdsensing applications,” *IEEE Internet Things J.*, vol. 6, no. 1, pp. 84–99, Feb. 2018.

[4] X. Wang, J. Zhang, X. Tian, X. Gan, Y. Guan, and X. Wang, “Crowdsensing-based consensus incident report for road traffic acquisition,” *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 8, pp. 2536–2547, Aug. 2018.

[5] A. Ali, M. A. Qureshi, M. Shiraz, and A. Shamim, “Mobile crowd sensing based dynamic traffic efficiency framework for urban traffic congestion control,” *Sustain. Comput.: Inform. Syst.*, vol. 32, Dec. 2021, Art. no. 100608.

[6] Y. Cheng et al., “AirCloud: A cloud-based air-quality monitoring system for everyone,” in *Proc. 12th ACM Conf. Embedded Netw. Sensor Syst.*, 2014, pp. 251–265.

[7] I. Jezdović, S. Popović, M. Radenković, A. Labus, and Z. Bogdanović, “A crowdsensing platform for real-time monitoring and analysis of noise pollution in smart cities,” *Sustain. Comput.: Inform. Syst.*, vol. 31, Sep. 2021, Art. no. 100588.

[8] Z. Liu, S. Jiang, P. Zhou, and M. Li, “A participatory urban traffic monitoring system: The power of bus riders,” *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 10, pp. 2851–2864, Oct. 2017.

[9] H. Li, X. Wu, L. Hou U, and K. P. Kou, “Near-optimal fixed-route scheduling for crowdsourced transit system,” in *Proc. IEEE 37th Int. Conf. Data Eng. (ICDE)*, 2021, pp. 2273–2278.

[10] P. Vitello et al., “Collaborative data delivery for smart city-oriented mobile crowdsensing systems,” in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, 2018, pp. 1–6.

[11] X. Wang et al., “Online spatial crowdsensing with expertise-aware truth inference and task allocation,” *IEEE J. Sel. Areas Commun.*, vol. 40, no. 1, pp. 412–427, Jan. 2022.

[12] X. Wei, Z. Li, C. Ren, T. Guo, and S. Gao, “HSM-SMCS: Task assignment based on hybrid sensing modes in sparse mobile crowdsensing,” *IEEE Internet Things J.*, vol. 10, no. 5, pp. 4034–4048, Mar. 2023.

[13] S. He and K. G. Shin, “Steering crowdsourced signal map construction via Bayesian compressive sensing,” in *Proc. IEEE INFOCOM 2018-IEEE Conf. Comput. Commun.*, 2018, pp. 1016–1024.

[14] T. Liu, Y. Zhu, Y. Yang, and F. Ye, “ALC^2: When active learning meets compressive crowdsensing for urban air pollution monitoring,” *IEEE Internet Things J.*, vol. 6, no. 6, pp. 9427–9438, Dec. 2019.

[15] W. Liu, L. Wang, E. Wang, Y. Yang, D. Zeghlache, and D. Zhang, “Reinforcement learning-based cell selection in sparse mobile crowdsensing,” *Comput. Netw.*, vol. 161, pp. 102–114, Oct. 2019.

Authorized licensed use limited to: Wuhan University. Downloaded on August 27,2024 at 01:31:37 UTC from IEEE Xplore. Restrictions apply.

[16] S. E. Awan, M. Bennamoun, F. Sohel, F. Sanfilippo, and G. Dwivedi, "Imputation of missing data with class imbalance using conditional generative adversarial networks," *Neurocomputing*, vol. 453, pp. 164–171, Sep. 2021.

[17] Y. Zhang, B. Zhou, X. Cai, W. Guo, X. Ding, and X. Yuan, "Missing value imputation in multivariate time series with end-to-end generative adversarial networks," *Inf. Sci.*, vol. 551, pp. 67–82, Apr. 2021.

[18] E. Wang, W. Liu, W. Liu, Y. Yang, B. Yang, and J. Wu, "Spatiotemporal urban inference and prediction in sparse mobile crowdsensing: A graph neural network approach," *IEEE Trans. Mobile Comput.*, early access, Aug. 5, 2022, doi: 10.1109/TMC.2022.3195706.

[19] E. Wang, M. Zhang, Y. Xu, H. Xiong, and Y. Yang, "Spatiotemporal fracture data inference in sparse urban crowdsensing," in *Proc. IEEE INFOCOM 2022-IEEE Conf. Comput. Commun.*, 2022, pp. 1499–1508.

[20] H. Yao et al., "Deep multi-view spatial-temporal network for taxi demand prediction," *Proc. AAAI Conf. Artif. Intell.*, vol. 32, no. 1, pp. 2588–2595, 2018.

[21] E. Wang et al., "Deep learning-enabled sparse industrial crowdsensing and prediction," *IEEE Trans. Ind. Informat.*, vol. 17, no. 9, pp. 6170–6181, Sep. 2021.

[22] W. Liu, Y. Yang, E. Wang, and J. Wu, "Fine-grained urban prediction via sparse mobile crowdsensing," in *Proc. IEEE 17th Int. Conf. Mobile Ad Hoc Sensor Syst. (MASS)*, 2020, pp. 265–273.

[23] Z. Zhu, B. Chen, W. Liu, Y. Zhao, Z. Liu, and Z. Zhao, "A cost-quality beneficial cell selection approach for sparse mobile crowdsensing with diverse sensing costs," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3831–3850, Mar. 2020.

[24] X. Li et al., "Multi-view matrix factorization for sparse mobile crowdsensing," *IEEE Internet Things J.*, vol. 9, no. 24, pp. 25767–25779, Dec. 2022.

[25] X. Wei, Z. Li, Y. Liu, S. Gao, and H. Yue, "SDLSC-TA: Subarea division learning based task allocation in sparse mobile crowdsensing," *IEEE Trans. Emerg. Topics Comput.*, vol. 9, no. 3, pp. 1344–1358, Jul./Sep. 2021.

[26] D. Wu et al., "When sharing economy meets IoT: Towards fine-grained urban air quality monitoring through mobile crowdsensing on bike-share system," *Proc. ACM Interactive, Mobile, Wearable Ubiquitous Technol.*, vol. 4, no. 2, pp. 1–26, 2020.

[27] L. Pan and J. Li, "K-nearest neighbor based missing data estimation algorithm in wireless sensor networks," *Wireless Sensor Netw.*, vol. 2, no. 02, p. 115, 2010.

[28] N. Marchang and R. Tripathi, "KNN-ST: Exploiting spatio-temporal correlation for missing data inference in environmental crowd sensing," *IEEE Sensors J.*, vol. 21, no. 3, pp. 3429–3436, Feb. 2021.

[29] K. Xie, X. Li, X. Wang, G. Xie, J. Wen, and D. Zhang, "Active sparse mobile crowd sensing based on matrix completion," in *Proc. 2019 Int. Conf. Manage. Data*, 2019, pp. 195–210.

[30] W.-C. Lin, C.-F. Tsai, and J. R. Zhong, "Deep learning for missing value imputation of continuous data and the effect of data discretization," *Knowl.-Based Syst.*, vol. 239, Mar. 2022, Art. no. 108079.

[31] T. Zhi-Xuan, H. Soh, and D. Ong, "Factorized inference in deep Markov models for incomplete multimodal time series," *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 6, 2020, pp. 10334–10341.

[32] N. Marchang, G. M. Meitei, and T. Thakur, "Task reduction using regression-based missing data imputation in sparse mobile crowdsensing," *J. Supercomput.*, vol. 78, no. 14, pp. 15995–16028, 2022.

[33] X. Chen et al., "PAS: Prediction-based actuation system for city-scale ridesharing vehicular mobile crowdsensing," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 3719–3734, May 2020.

[34] R. K. C. Chan, J. M.-Y. Lim, and R. Parthiban, "A neural network approach for traffic prediction and routing with missing data imputation for intelligent transportation system," *Expert Syst. Appl.*, vol. 171, Jun. 2021, Art. no. 114573.

[35] M. Gupta and R. Beheshti, "Time-series imputation and prediction with bi-directional generative adversarial networks," 2020, *arXiv:2009.08900*.

[36] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[37] M. Guo, A. Haque, D.-A. Huang, S. Yeung, and L. Fei-Fei, "Dynamic task prioritization for multitask learning," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 270–287.

[38] J. Shang, Y. Zheng, W. Tong, E. Chang, and Y. Yu, "Inferring gas consumption and pollution emission of vehicles throughout a city," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. discovery data mining*, 2014, pp. 1027–1036.

[39] Y. Zheng, F. Liu, and H.-P. Hsieh, "U-air: When urban air quality inference meets big data," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2013, pp. 1436–1444.

[40] X. Yan, W. Xiong, L. Hu, F. Wang, and K. Zhao, "Missing value imputation based on Gaussian mixture model for the Internet of Things," *Math. Problems Eng.*, vol. 2015, 2015, Art. no. 548605.

[41] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2017, *arXiv:1609.02907*.

[42] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," 2017, *arXiv:1710.10903*.

Jie Huo received the B.S. degree from Beijing Jiaotong University, Beijing, China, in 2019. She is currently pursuing the Ph.D. degree in information and communication engineering from Beijing University of Posts and Telecommunications, Beijing.

Her research interests mainly focus on the mobile crowdsensing, edge computing, and connected vehicles.

Luhan Wang received the Ph.D. degree from Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2017.

In 2016, he visited Eurecom, Biot, France, as a visiting Ph.D. student. In 2017, he joined the School of Information and Communication Engineering, BUPT as an Assistant Professor. He also works as a Researcher and the Project Manager with Joint BUPT-Eurecom Open5G Laboratory, Beijing. He currently leads a team in BUPT which is getting involved in OpenAirInterface 5G Core Network Development. His research interests include network architecture, network function virtualization, and software-defined networks.

Zhaoming Lu received the Ph.D. degree from Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2012.

He is currently an Associate Professor and a Faculty Member of Beijing Laboratory of Advanced Information Networks, BUPT. His research interests are in the areas of opensource 5G, network aided autonomous driving, WiFi sensing, and imaging.

Dr. Lu has got the Best Paper Award of IEEE ICEI 2018 and the Best Demo of IEEE ISWCS 2016. He has served as a general chair, a TPC chair, a TPC member for several international workshops and conferences. He is on the editorial board of two international journals.

Xiangming Wen (Senior Member, IEEE) received the M.S. and Ph.D. degrees in information and communication engineering from Beijing University of Posts and Telecommunications, Beijing, China, in 1992 and 2002, respectively.

He is currently the Director of Beijing Key Laboratory of Network System Architecture and Convergence, Beijing University of Posts and Telecommunications, where he has managed several projects related to open wireless networking. His current research interests focus on radio resource and mobility management, software-defined wireless networks, and broadband multimedia transmission technology.

Authorized licensed use limited to: Wuhan University. Downloaded on August 27,2024 at 01:31:37 UTC from IEEE Xplore. Restrictions apply.