MepoGNN: Metapopulation Epidemic Forecasting with Graph Neural Networks

Qi Cao[1], Renhe Jiang[1(✉)], Chuang Yang[1], Zipei Fan[1], Xuan Song[1,2], and Ryosuke Shibasaki[1]

[1] The University of Tokyo, Tokyo, Japan
{caoqi,jiangrh,chuang.yang,songxuan,shiba}@csis.u-tokyo.ac.jp, fanzipei@iis.u-tokyo.ac.jp
[2] Southern University of Science and Technology, Shenzhen, China

Abstract. Epidemic prediction is a fundamental task for epidemic control and prevention. Many mechanistic models and deep learning models are built for this task. However, most mechanistic models have difficulty estimating the time/region-varying epidemiological parameters, while most deep learning models lack the guidance of epidemiological domain knowledge and interpretability of prediction results. In this study, we propose a novel hybrid model called MepoGNN for multi-step multi-region epidemic forecasting by incorporating Graph Neural Networks (GNNs) and graph learning mechanisms into Metapopulation SIR model. Our model can not only predict the number of confirmed cases but also explicitly learn the epidemiological parameters and the underlying epidemic propagation graph from heterogeneous data in an end-to-end manner. Experiment results demonstrate our model outperforms the existing mechanistic models and deep learning models by a large margin. Furthermore, the analysis on the learned parameters demonstrates the high reliability and interpretability of our model and helps better understanding of epidemic spread. Our model and data have already been public on GitHub https://github.com/deepkashiwa20/MepoGNN.git.

Keywords: Epidemic forecasting · Hybrid model · Metapopulation epidemic model · Graph Neural Networks · Deep learning · COVID-19

1 Introduction

The coronavirus disease 2019 (COVID-19) pandemic has caused around 500 million confirmed cases and more than 6 million deaths in the global, and it is still ongoing. Due to this circumstance, epidemic forecasting has been a key research topic again as it can guide the policymakers to develop effective interventions and allocate the limited medical resources. Many mechanistic models and deep learning models have been built for the epidemic prediction task. In particular, human mobility is seen as one of the most important factors to understand and forecast the epidemic propagation among different regions. In this study, we

M.-R. Amini et al. (Eds.): ECML PKDD 2022, LNAI 13718, pp. 453–468, 2023.
https://doi.org/10.1007/978-3-031-26422-1_28

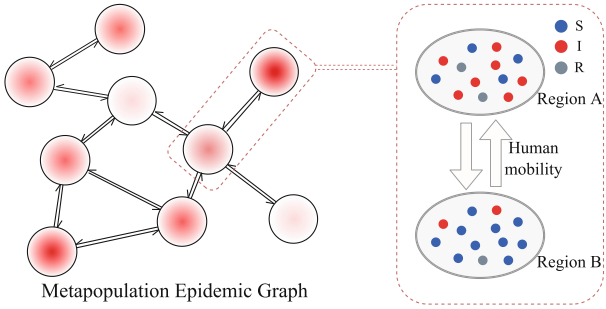

Fig. 1. Illustration of metapopulation epidemic propagation among regions [1].

employ metapopulation SIR model [1,2] as the base model for our task, which extends the most fundamental compartmental model (i.e., SIR [11]) in epidemiology with metapopulation epidemic propagation. As illustrated in Fig. 1, it divides the total population under the epidemic into several sub-populations (e.g., by regions). Each sub-population consists of three compartments, S (susceptible individuals), I (infectious individuals), R (removed individuals, including deaths and recovery cases), and the human mobility between sub-populations is modeled as a directed graph. Thus, it can well model the epidemic propagation in a large-scale area. The metapopulation epidemic models have achieved great success in modeling and analyzing the propagation of epidemic diseases, such as SARS, H1N1, and Malaria [3–5].

However, it is always a non-trivial task to build a metapopulation epidemic model, especially for new emerging epidemics such as the COVID-19 due to the following reasons. First, the epidemiological parameters in metapopulation model keep varying from region to region and time to time. As we all know, the Coronavirus keeps evolving, and the transmissibility and mortality of the variants (e.g., Alpha, Delta, and Omicron) are significantly different. Besides, the intervention policies and the human movements also vary over different periods and regions. Second, due to the mixed factors mentioned above, the epidemic propagation effects via human mobility in metapopulation model are also difficult to be obtained or estimated. In the case of prefecture-level prediction in Japan, we need to collect the large-scale human mobility data of the entire Japan and obtain the amount of human movements between each pair of prefectures. Then how to accurately infer the underlying disease propagation network becomes another intractable task. Third, besides the daily infection data, external features such as date information (e.g., $dayofweek$) and daily movement change patterns should also be involved.

To tackle these challenges, we incorporate deep learning modules into metapopulation SIR model to form a novel hybrid epidemic model. Specifically, we first learn the time/region-varying epidemiological parameters from multiple data features through a spatio-temporal module, which consists of Temporal Convolutional Networks (TCN) and Graph Convolutional Networks (GCN).

Next, we design two types of graph learning module to automatically approximate the underlying epidemic propagation graph based on the countrywide human mobility data. Furthermore, we let the learned latent graph be shared by the spatio-temporal module and the metapopulation SIR module, which further enhances the model interpretability and reliability. Previous deep learning methods [6–10] simply treat the epidemic forecasting as time-series prediction task or spatio-temporal prediction task, which can only output the predicted number of infections in a pure black-box manner. Recent study [29] involves the classical epidemic modeling into deep neural networks, however, it does not explicitly consider the epidemic propagation among regions via metapopulation modeling like ours, which largely limits the model interpretability for multi-region epidemic forecasting. *To the best of our knowledge, our work is the first hybrid model that couples metapopulation epidemic model with spatio-temporal graph neural networks*. In summary, our work has the following contributions:

- We propose a novel hybrid model along with two types of graph learning module for multi-step multi-region epidemic prediction by mixing metapopulation epidemic model and spatio-temporal graph convolution networks.
- Our model can explicitly learn the time/region-varying epidemiological parameters as well as the latent epidemic propagation among regions from the heterogeneous inputs like infection related data, human mobility data, and meta information in a completely end-to-end manner.
- We collect and process the big human GPS trajectory data and other COVID-19 related data that covers the 47 prefectures of Japan from 2020/04/01 to 2021/09/21 for countrywide epidemic forecasting.
- We conduct comprehensive experiments to validate not only the superior forecasting performance but also the high interpretability of our model. Our model and data have already been public on GitHub https://github.com/deepkashiwa20/MepoGNN.git.

2 Related Work

The models for epidemic simulation and forecasting can be divided into two types: *mechanistic approaches* and *deep learning approaches*.

Mechanistic approaches are built based on the domain knowledge of epidemiology which employ pre-defined physical rules to model infectious diseases' transmission dynamics, mainly *classical compartmental models* [11,12], *metapopulation models* [2,13–15] and *agent-based models* [16–18]. The classical compartmental models simulate the spread of infectious diseases in a homogeneous population which are unable to model epidemic spread between regions. The metapopulation models assume the heterogeneity of sub-populations and use the human mobility pattern between regions to model the spread of the epidemic [1,2]. The agent-based models directly use the individual-level movement pattern [16,17] or trajectories [18] to emulate the contagion process. Our work is related to the metapopulation model which is most suitable for multi-region epidemic forecasting task. To implement epidemic modeling, it needs to be calibrated first using

historical observations and use the optimized or manually modified parameters to make prediction. These efforts are hardly applicable for multi-step forecasting tasks. The parameters calibration process needs high computational complexity, especially when facing huge parameter state space [13,16]. Moreover, in most mechanistic models, epidemiological parameters keep fixed during forecasting. The variation of parameters through time is not considered which leads to the problem of cumulative error on multi-step prediction.

Deep learning approaches have shown excellent performance in the modeling and forecasting on time series prediction tasks. As a typical time series, several research efforts utilizing deep learning techniques, such as LSTM [6,8], have been conducted for epidemic forecasting over a single region [6,8,19,20]. Nevertheless, the epidemic propagation is often spatially dependent, i.e., co-evolving over regions. Thus, treating epidemic forecasting as a multivariate time-series prediction task, performing collaborative forecasting over multiple geographical units should be a more reasonable choice. For such tasks, a key challenge is to model the complex and implicit spatio-temporal dependencies among the observations, on which much evidence shows that GNN can perform very well for modeling the inter-series relationships. A series of state-of-the-art solutions based on GNN have been proposed for multivariate time-series prediction tasks, such as STGCN [21], DCRNN [22], GraphWaveNet [23], ColaGNN [9], and CovidGNN [10]. In particular, ColaGNN [9] and CovidGNN [10] were explicitly designed for the epidemic prediction. However, these works ignore the domain knowledge of epidemiology and are hard to interpret from the epidemiological perspective. STAN [19] incorporates epidemiological constraints into deep learning models, but it can only predict infections of a single region. CausalGNN [29] embeds single-patched SIRD model into GNN for multi-region epidemic forecasting.

Overall, we distinguish our work from existing ones in the following ways: Compared with the mechanistic models, MepoGNN adopts an end-to-end framework that can predict the dynamic change of epidemiological parameters and use predicted parameters to produce multi-region and multi-step prediction; Compared with the deep learning models for the multi-region prediction task, MepoGNN incorporates the domain knowledge of epidemiology and enhances the interpretability by combining spatio-temporal deep learning model with the metapopulation model; Furthermore, MepoGNN can output the prediction of infections through the metapopulation epidemic model and learn the interpretable epidemiological parameters and the latent graph of epidemic propagation simultaneously.

3 Problem

In this study, we focus on forecasting the number of daily confirmed cases for multi-region and multi-step simultaneously. For a single region, the historical daily confirmed cases from timestep $t - T_{in} + 1$ to t can be represented as $\mathbf{x}^{t-(T_{in}-1):t} \in \mathbb{R}^{T_{in}}$. Then, the historical daily confirmed cases

of N regions can be denoted as $\mathbf{X}^{t-(T_{in}-1):t} = \{\mathbf{x}_1^{t-(T_{in}-1):t}, \mathbf{x}_2^{t-(T_{in}-1):t}, ..., \mathbf{x}_N^{t-(T_{in}-1):t}\} \in \mathbb{R}^{N \times T_{in}}$. Besides the historical observations, we also incorporate the external factors to form a multi-channel input as $\mathcal{X}^{t-(T_{in}-1):t} = \{\mathbf{X}_1^{t-(T_{in}-1):t}, \mathbf{X}_2^{t-(T_{in}-1):t}, ..., \mathbf{X}_C^{t-(T_{in}-1):t}\} \in \mathbb{R}^{N \times T_{in} \times C}$. Details of the input features will be introduced in Sect. 5.1. Additionally, human mobility between regions (static flow data $\mathbf{U} \in \mathbb{R}^{N \times N}$ or dynamic flow data $\mathcal{O}^{t-(T_{in}-1):t} \in \mathbb{R}^{N \times N \times T_{in}}$) is used as another type of input. The prediction target is the daily confirmed cases of N regions in next T_{out} timesteps $\mathbf{Y}^{t+1:t+T_{out}} \in \mathbb{R}^{N \times T_{out}}$. The problem can be formulated as follows:

$$\{\mathcal{X}^{t-(T_{in}-1):t}, \mathbf{U}\} \text{ or } \{\mathcal{X}^{t-(T_{in}-1):t}, \mathcal{O}^{t-(T_{in}-1):t}\} \xrightarrow{f(\cdot)} \mathbf{Y}^{t+1:t+T_{out}} \quad (1)$$

4 Methodology

We present Metapopulation Epidemic Graph Neural Networks (MepoGNN), demonstrated in Fig. 2, for spatio-temporal epidemic prediction. MepoGNN consists of three major components: metapopulation SIR module, spatio-temporal module and graph learning module. These three components tightly cooperate with each other. Graph learning module learns the mobility intensity between

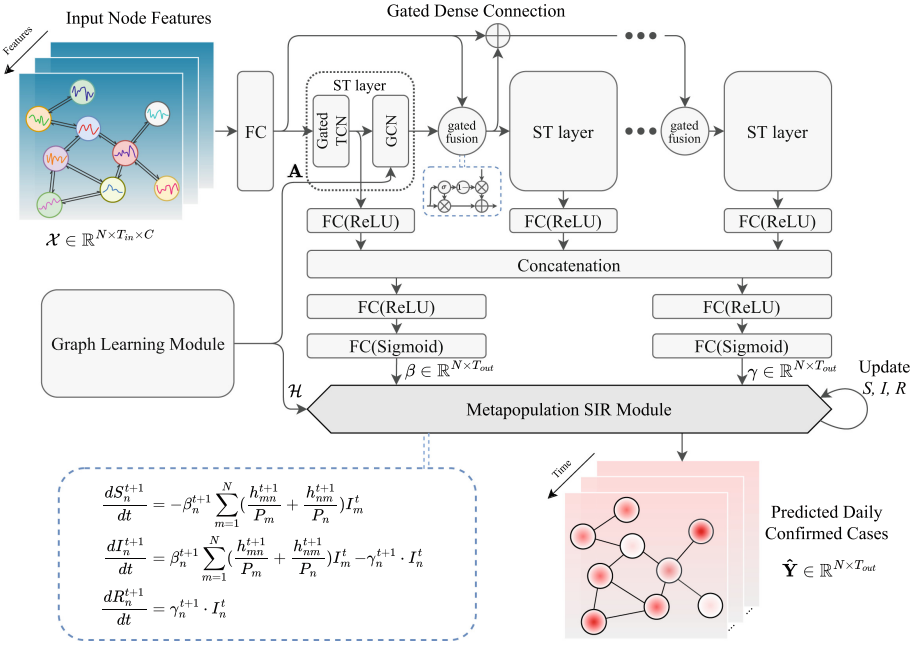

Fig. 2. Proposed metapopulation epidemic graph neural networks (MepoGNN) for spatio-temporal epidemic prediction.

regions as a graph and output it to spatio-temporal module and metapopulation SIR module. Spatio-temporal module captures the spatio-temporal dependency to predict the sequences of parameters for metapopulation SIR module. Then, metapopulation SIR module takes the learned graph and the predicted parameters to produce the multi-step prediction of daily confirmed cases.

4.1 Metapopulation SIR Module

SIR model is one of the most fundamental compartmental models in epidemiology, used for modeling the epidemic spread [11]. However, it can only model the epidemic spread for a homogeneous population, which ignores the epidemic propagation between sub-populations. Metapopulation SIR model [2] fills this gap by assuming the heterogeneity of sub-populations and using human mobility to model the propagation between sub-populations. Metapopulation SIR model, consists of three compartments for each sub-population: S_n^t for number of susceptible individuals, I_n^t for number of infectious individuals, R_n^t for the number of recovered or deceased individuals of sub-population n at time t. P_n represents the size of sub-population n which is assumed to be a constant number, where $P_n = S_n^t + I_n^t + R_n^t$. β is the rate of infection, and γ is the rate of recovery and mortality. Furthermore, it uses h_{nm} to represent the epidemic propagation from sub-population (also called patch) n to m. The original metapopulation SIR model [2] is shown as follows:

$$\begin{aligned}
\frac{dS_n^{t+1}}{dt} &= -\beta \cdot S_n^t \sum_{m=1}^{N} \left(\frac{h_{mn}}{P_m} + \frac{h_{nm}}{P_n}\right) I_m^t \\
\frac{dI_n^{t+1}}{dt} &= \beta \cdot S_n^t \sum_{m=1}^{N} \left(\frac{h_{mn}}{P_m} + \frac{h_{nm}}{P_n}\right) I_m^t - \gamma \cdot I_n^t \\
\frac{dR_n^{t+1}}{dt} &= \gamma \cdot I_n^t
\end{aligned} \tag{2}$$

In this study, we model population of each region as sub-population in metapopulation SIR model. So, the h_{nm} can be represented by human mobility between regions. Because of different characteristics of regions, policy changes with time and so on, there is spatio-temporal heterogeneity of epidemic spread. In our model, β, γ and h_{nm} are assumed to vary over time and regions. In addition, to prevent β to be extremely small and make it be in a relatively stable magnitude, S_n^t is omitted from the equations. Thus, we extend the original metapopulation SIR in Eq. 2 as follows:

$$\begin{aligned}
\frac{dS_n^{t+1}}{dt} &= -\beta_n^{t+1} \sum_{m=1}^{N} \left(\frac{h_{mn}^{t+1}}{P_m} + \frac{h_{nm}^{t+1}}{P_n}\right) I_m^t \\
\frac{dI_n^{t+1}}{dt} &= \beta_n^{t+1} \sum_{m=1}^{N} \left(\frac{h_{mn}^{t+1}}{P_m} + \frac{h_{nm}^{t+1}}{P_n}\right) I_m^t - \gamma_n^{t+1} \cdot I_n^t \\
\frac{dR_n^{t+1}}{dt} &= \gamma_n^{t+1} \cdot I_n^t
\end{aligned} \tag{3}$$

With predicted β_n^{t+1}, γ_n^{t+1} and $\mathcal{H}^{t+1}$ (the epidemic propagation matrix formed by $\{h_{nm}^{t+1}|n,m \in \{1,2,...,N\}\}$), S, I, R can be updated iteratively:

$$[S_n^t, I_n^t, R_n^t] \xrightarrow[\beta_n^{t+1},\gamma_n^{t+1},\mathcal{H}^{t+1}]{Eq.(3)} [S_n^{t+1}, I_n^{t+1}, R_n^{t+1}] \tag{4}$$

The final prediction output of daily confirmed cases can be formed as:

$$\begin{aligned} \hat{y}_n^{t+1} &= \beta_n^{t+1} \sum_{m=1}^{N} \left(\frac{h_{mn}^{t+1}}{P_m} + \frac{h_{nm}^{t+1}}{P_n}\right) I_m^t \\ \hat{\mathbf{Y}} &= \begin{bmatrix} \hat{y}_1^{t+1} & \dots & \hat{y}_1^{t+T_{out}} \\ \vdots & \ddots & \vdots \\ \hat{y}_n^{t+1} & \dots & \hat{y}_n^{t+T_{out}} \end{bmatrix}_{N \times T_{out}} \end{aligned} \tag{5}$$

4.2 Spatio-Temporal Module for Epidemiological Parameters

Spatio-temporal module takes the node input features $\mathcal{X} \in \mathbb{R}^{N \times T_{in} \times C}$ and the weighted adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ as input and output the predicted parameters $\beta \in \mathbb{R}^{N \times T_{out}}$ and $\gamma \in \mathbb{R}^{N \times T_{out}}$. We use the spatio-temporal layer (ST layer) combining Gated TCN and GCN (same as in GraphWaveNet [23]) to capture the spatio-temporal dependency. Gated TCN [24] is used to capture temporal dependency:

$$\mathcal{Q}_l = g(\Theta_{l1} \star \mathcal{Z}_l + \mathbf{b}_{l1}) \odot \sigma(\Theta_{l2} \star \mathcal{Z}_l + \mathbf{b}_{l2}) \tag{6}$$

where $\mathcal{Z}_l$ is input of l-th layer, Θ_1 and Θ_2 are temporal convolution kernels, $\mathbf{b}_1$ and $\mathbf{b}_2$ are biases, $g(\cdot)$ is tanh activation function for output, $\sigma(\cdot)$ is sigmoid function to form the gate, $\star$ is convolution, $\odot$ is element-wise product. Next, we model the regions and the interactions between regions as a graph and use diffusion graph convolution [22,23] to capture the spatial dependency:

$$\mathbf{P}_f = \mathbf{A}/rowsum(\mathbf{A}), \quad \mathbf{P}_b = \mathbf{A}^{\mathbf{T}}/rowsum(\mathbf{A}^{\mathbf{T}}) \tag{7}$$

$$\tilde{\mathcal{Z}}_l = \sum_{k=0}^{K} \mathbf{P}_f^k \mathcal{Q}_l \mathbf{W}_{lk1} + \mathbf{P}_b^k \mathcal{Q}_l \mathbf{W}_{lk2} \tag{8}$$

where $\mathbf{A} \in \mathbb{R}^{N \times N}$ is weighted adjacency matrix, $\mathbf{P}_f$ is forward transition matrix, $\mathbf{P}_b$ is backward transition matrix, $\tilde{\mathcal{Z}}_l$ is output of l-th layer.

Multiple ST layers can be stacked to capture the spatio-temporal dependency in different scales. We use a gated dense connection to bridge different ST layers. It can extract important information from previous ST layers and pass it to next layer:

$$\mathcal{D}_l = \begin{cases} \mathcal{X}, & \text{if } l = 1, \\ \mathcal{D}_{l-1} + \mathcal{Z}_l, & \text{otherwise.} \end{cases} \tag{9}$$

$$\mathcal{Z}_{l+1} = \begin{cases} \mathcal{X}, & \text{if } l = 0 \ , \\ \tilde{\mathcal{Z}}_l \odot \sigma(\tilde{\mathcal{Z}}_l) + \mathcal{D}_l \odot (1 - \sigma(\tilde{\mathcal{Z}}_l)), & \text{otherwise.} \end{cases} \tag{10}$$

where $\mathcal{D}_l$ stores the information from previous layers. Then, we concatenate the output from different layers through skip connections to fuse the information of different scales. Finally, the parameters $\beta \in \mathbb{R}^{N \times T_{out}}$ and $\gamma \in \mathbb{R}^{N \times T_{out}}$ are produced through two fully connected layers, respectively.

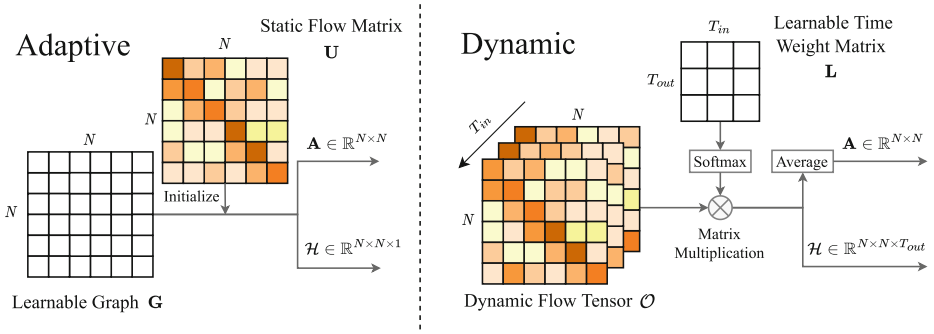

Fig. 3. Two types of graph learning: adaptive and dynamic.

4.3 Graph Learning Module for Epidemic Propagation

There are two different graphs used in metapopulation SIR module and spatio-temporal module, respectively. Unlike the trivial method which input two fixed graphs to each module separately, we make two modules share a single learnable graph. With the shared learnable graph, the spatial dependency used in spatio-temporal module would be consistent with epidemic propagation in metapopulation SIR module which can improve the interpretability of our model. Furthermore, the parameters of graph learning module can be updated by gradients from both spatio-temporal module and metapopulation SIR module which make learned graph more realistic.

As shown in Fig. 3, there are two types of graph learning module to deal with different input data. The first type is adaptive graph learning module which takes the static flow data (e.g., commuter survey data) as input. Intuitively, we initialize an adaptive graph $\mathbf{G}$ with static flow matrix $\mathbf{U}$ and make it learnable through training. Then, the adaptive graph can be output to spatio-temporal module (Eq. 7) as $\mathbf{A} \in \mathbb{R}^{N \times N}$ and to metapopulation SIR module (Eq. 3) as $\mathcal{H} \in \mathbb{R}^{N \times N \times 1}$ (which means we use same h_{nm} for all timesteps). The second type is dynamic graph learning module which takes the dynamic OD flow tensor as input. Although the OD flow and epidemic spread status are both dynamic,

but they are not necessarily one-to-one temporally corresponding. Considering the delayed effect, influence of mobility on epidemic spread can be seen as a weighted average of the given past values (T_{in} days). So, we initialize a learnable time weight matrix $\mathbf{L} \in \mathbb{R}^{T_{out} \times T_{in}}$ and normalize it as $\tilde{\mathbf{L}}$ through a softmax function. The normalized time weight matrix can map the historical dynamic flow $\mathcal{O}^{t-(T_{in}-1):t} \in \mathbb{R}^{N \times N \times T_{in}}$ to its influence on future epidemic spread. The output of $\mathcal{H}^{t+1:t+T_{out}} \in \mathbb{R}^{N \times N \times T_{out}}$ and $\mathbf{A} \in \mathbb{R}^{N \times N}$ can be calculated as follows:

$$\tilde{\mathbf{L}} = Softmax_{:,j}(\mathbf{L}) \tag{11}$$

$$\mathcal{H}^{t+1:t+T_{out}} = \tilde{\mathbf{L}}\mathcal{O}^{t-(T_{in}-1):t}, \quad \mathbf{A} = \frac{\sum_{i=1}^{T_{out}} \mathcal{H}^{t+i}}{T_{out}} \tag{12}$$

Why Propose Two Types of Graph Learning? Dynamic graph learning module can illustrate the dynamic change of epidemic propagation. But it requires dynamic flow data which is not available in most cases. To improve the applicability of our model, we propose adaptive graph learning module to address this problem. With two types of graph learning module, our model can handle different situations of data availability in the best way possible.

5 Experiment

5.1 Data

We set 47 prefectures of Japan and 2020/04/01 $\sim$ 2021/09/21 (539 d) as our study area and time period, respectively. The number of daily confirmed cases and cumulative cases and deaths are collected from the NHK COVID-19 database[1]. The number of recovered cases is collected from Japan LIVE Dashboard[2] [25] (original data source is from Ministry of Health, Labour and Welfare, Japan). The population of each prefecture is collected from 2020 census data. With above-mentioned data, daily S, I, R of each prefecture can be calculated. Apart from the number of daily confirmed cases, the input node features also include daily movement change, the ratio of daily confirmed cases in active cases, and $dayofweek$. The movement change data is collected from Facebook Movement Range Maps[3]. It records the change of people movement range compared to a baseline period. Because it is not provided at prefecture level, we use population weighted average to get data at prefecture level. The input static flow data for adaptive graph learning module is the number of commuters between prefectures, which is collected from 2015 census data. The input dynamic flow data for dynamic graph learning module is the daily OD flow data among 47 prefectures, which is generated from human GPS trajectory data provided by

[1] https://www3.nhk.or.jp/news/special/coronavirus/data/.
[2] https://github.com/swsoyee/2019-ncov-japan.
[3] https://data.humdata.org/dataset/movement-range-maps.

Blogwatcher Inc. To mitigate the spatio-temporal imbalance in our data, we use stay put ratio (ratio of people staying in a single location all day) in Facebook Movement Range Maps to get the ratio of active users and use it to normalize the OD flow. Finally, the input features of 47 prefectures are generated as a (539, 47, 4) tensor, the static flow is a (47, 47) matrix, and the dynamic flow is a (539, 47, 47) tensor.

5.2 Setting

The input time length T_{in} and output time length T_{out} are both set to 14 d which means we use two-week historical observations to do the two-week prediction of daily confirmed cases. Then, we split the data with ratio 6:1:1 to get training/validation/test datasets, respectively. The fifth wave of infection in Japan is included in test dataset to test the model performance on a real outbreak situation. During training, we use the curriculum learning strategy [26] which increases one prediction horizon every two epochs starting from one day ahead prediction. The batch size is set to 32. The loss function is set as *MAE* (Mean Absolute Error). Adam is set as the optimizer, where the learning rate is 1e-3 and weight decay is 1e–8. The training algorithm would either be early-stopped if the validation error did not decrease within 20 epochs or be stopped after 300 epochs. PyTorch is used to implement our model. Then experiments are performed on a server with four 2080Ti GPUs. Finally, we evaluate the performance of model on 3 d, 7 d, 14 d ahead prediction and overall 14 steps prediction. The four metrics are used to qualify the performance: *RMSE* (Root Mean Square Error), *MAE* (Mean Absolute Error), *MAPE* (Mean Absolute Percentage Error) and *RAE* (Relative Absolute Error). To mitigate the influence of randomness, we perform 5 trials for each model and calculate the mean and 95% confidence interval of results. The used random seeds are 0, 1, 2, 3, 4.

5.3 Evaluation

We implement three classes of baselines to compare and evaluate our model on epidemic prediction task:

Mechanistic Models: (1) SIR [11]. SIR model is one of most basic compartmental models in epidemiology. We use optimized β and γ of each regions to produce the prediction. **(2) SIR(Copy)**. Because of weekly periodicity, we copy the β and γ of last week to produce the prediction. **(3) MetaSIR** [2]. Metapopulation SIR model considers the heterogeneity of sub-populations and models the interaction between sub-populations. We use the commuter survey data as $\mathcal{H}$ and optimize β and γ for each region to produce the prediction. **(4) MetaSIR(Copy)**. We copy the β and γ of last week to produce the prediction.

Spatio-Temporal Deep Learning Models: (5) STGCN [21]. STGCN is one of the earliest models which applies GCN and TCN to do spatio-temporal prediction. **(6) DCRNN** [22]. DCRNN proposes a variant of GCN, called diffusion

Table 1. Performance comparison with baselines

Model	3 d Ahead				7 d Ahead			
	RMSE	MAE	MAPE	RAE	RMSE	MAE	MAPE	RAE
SIR	429.4 ± 23.2	153.9 ± 5.2	83.8 ± 0.7	0.47 ± 0.02	507.5 ± 29.6	191.4 ± 7.7	111.4 ± 3.8	0.57 ± 0.02
SIR(Copy)	248.1	97.4	57.4	0.29	318.5	127.1	67.2	0.38
MetaSIR	336.0 ± 21.6	126.8 ± 3.5	72.2 ± 0.9	0.38 ± 0.01	429.8 ± 25.5	166.9 ± 3.7	92.9 ± 0.8	0.50 ± 0.01
MetaSIR(Copy)	236.5	92.2	54.1	0.28	307.6	120.0	62.7	0.36
STGCN	375.6 ± 18.8	118.6 ± 10.8	45.3 ± 2.8	0.36 ± 0.03	381.1 ± 17.7	128.0 ± 6.6	52.5 ± 3.0	0.38 ± 0.02
DCRNN	305.0 ± 9.8	89.3 ± 4.4	37.3 ± 0.7	0.27 ± 0.01	323.8 ± 15.9	107.6 ± 5.3	47.3 ± 1.4	0.32 ± 0.02
AGCRN	223.5 ± 28.5	80.0 ± 7.8	56.6 ± 13.2	0.24 ± 0.02	253.1 ± 37.7	97.9 ± 7.6	60.8 ± 10.1	0.29 ± 0.02
GraphWaveNet	223.8 ± 46.6	70.6 ± 11.7	35.4 ± 1.2	0.21 ± 0.04	259.9 ± 52.2	89.2 ± 15.2	42.3 ± 1.5	0.27 ± 0.05
MTGNN	297.6 ± 19.2	102.4 ± 6.7	40.6 ± 0.8	0.31 ± 0.02	363.5 ± 37.9	130.9 ± 13.1	49.1 ± 1.7	0.39 ± 0.04
CovidGNN	261.9 ± 55.5	88.4 ± 16.7	43.3 ± 3.8	0.27 ± 0.05	305.4 ± 70.6	116.5 ± 23.8	60.9 ± 5.3	0.35 ± 0.07
ColaGNN	221.7 ± 40.7	72.7 ± 7.2	38.9 ± 1.5	0.22 ± 0.02	300.6 ± 61.2	109.4 ± 16.4	49.3 ± 1.5	0.33 ± 0.05
MepoGNN(Adp)	141.0 ± 7.2	54.3 ± 2.3	34.9 ± 0.8	0.16 ± 0.01	174.6 ± 10.1	69.7 ± 4.2	**41.4 ± 1.6**	0.21 ± 0.01
MepoGNN(Dyn)	**135.9 ± 17.8**	**52.7 ± 4.6**	**34.2 ± 0.7**	**0.16 ± 0.01**	**160.6 ± 4.5**	**67.6 ± 1.2**	41.7 ± 0.9	**0.20 ± 0.00**
Model	14 d Ahead				Overall			
	RMSE	MAE	MAPE	RAE	RMSE	MAE	MAPE	RAE
SIR	890.2 ± 83.8	314.5 ± 16.9	228.3 ± 11.8	0.94 ± 0.05	595.0 ± 43.5	210.0 ± 9.2	128.2 ± 4.7	0.63 ± 0.03
SIR(Copy)	835.5	332.6	183.2	1.00	539.1	190.2	102.7	0.57
MetaSIR	766.1 ± 58.5	279.1 ± 8.2	177.4 ± 4.5	0.84 ± 0.02	500.4 ± 33.9	182.1 ± 4.4	104.9 ± 1.3	0.55 ± 0.01
MetaSIR(Copy)	786.4	302.7	161.9	0.91	503.7	175.6	92.7	0.53
STGCN	430.2 ± 15.8	159.4 ± 6.0	74.7 ± 3.7	0.48 ± 0.02	389.5 ± 7.9	132.0 ± 2.9	55.6 ± 2.4	0.40 ± 0.01
DCRNN	377.9 ± 11.1	146.0 ± 5.0	69.5 ± 4.0	0.44 ± 0.01	335.0 ± 11.8	112.5 ± 4.5	49.5 ± 1.3	0.34 ± 0.01
AGCRN	390.4 ± 105.8	149.0 ± 11.4	88.0 ± 12.8	0.45 ± 0.03	322.7 ± 136.7	108.0 ± 9.9	67.9 ± 15.6	0.32 ± 0.03
GraphWaveNet	389.8 ± 20.8	144.4 ± 7.3	60.2 ± 4.2	0.43 ± 0.02	294.7 ± 40.9	100.1 ± 11.1	44.7 ± 1.4	0.30 ± 0.03
MTGNN	443.5 ± 15.4	168.3 ± 8.1	68.0 ± 2.9	0.50 ± 0.02	363.2 ± 20.5	130.0 ± 8.3	50.7 ± 1.6	0.39 ± 0.03
CovidGNN	414.7 ± 59.8	177.4 ± 15.9	111.2 ± 6.6	0.53 ± 0.05	329.6 ± 59.8	124.2 ± 19.2	66.9 ± 4.2	0.37 ± 0.06
ColaGNN	388.3 ± 23.2	153.4 ± 10.2	75.5 ± 10.8	0.46 ± 0.03	310.7 ± 31.4	110.2 ± 7.2	51.9 ± 3.7	0.33 ± 0.02
MepoGNN(Adp)	261.1 ± 16.0	**105.1 ± 7.3**	**60.1 ± 3.2**	**0.32 ± 0.02**	196.2 ± 11.3	75.4 ± 4.7	**44.0 ± 1.6**	0.23 ± 0.01
MepoGNN(Dyn)	**253.2 ± 7.5**	107.0 ± 3.0	62.0 ± 2.0	0.32 ± 0.01	**186.1 ± 5.0**	**74.3 ± 2.0**	44.4 ± 0.8	**0.22 ± 0.01**

convolution and combines it with gated recurrent unit (GRU) to build a spatio-temporal prediction model. **(7) GraphWaveNet** [23]. GraphWaveNet proposes an adaptive learnable graph and uses GCN and TCN to capture spatio-temporal dependency. **(8) MTGNN** [26]. MTGNN uses a graph learning module to learn spatial correlation and fuse different spatial hops and different TCN kernels to enhance the model capacity. **(9) AGCRN** [27]. AGCRN uses GCN and GRU along with a graph learning module and a node adaptive parameter learning module to capture spatio-temporal dependency.

GNN-Based Epidemic Models: (10) CovidGNN [10]. CovidGNN is one of the earliest GNN-based epidemic models. It embeds temporal features on node and uses GCN with skip connections to capture spatial dependency. **(11) ColaGNN** [9]. ColaGNN uses the location-aware attention to extract spatial dependency and uses GCN to integrate the spatio-temporal information.

Performance Evaluation: In Table 1, we compare the performance on three different horizons and overall performance for multi-step prediction among the above-mentioned three classes of baseline models and proposed MepoGNN with two types of graph learning module. Generally, the spatio-temporal deep learning models and GNN-based epidemic models outperform the mechanistic models,

Table 2. Ablation study

Graph	Model	Mean RMSE	Mean MAE	Mean MAPE	Mean RAE
Adaptive	w/o glm	209.51 ± 22.70	81.85 ± 6.69	47.51 ± 2.62	0.25 ± 0.02
	w/o propagation	203.23 ± 24.70	82.05 ± 8.05	45.84 ± 1.68	0.25 ± 0.02
	w/o SIR	318.05 ± 16.30	108.53 ± 5.26	46.07 ± 0.53	0.33 ± 0.02
	MepoGNN	$\mathbf{196.16 \pm 11.33}$	$\mathbf{75.45 \pm 4.65}$	$\mathbf{44.02 \pm 1.55}$	$\mathbf{0.23 \pm 0.01}$
Dynamic	w/o glm	194.50 ± 17.65	76.84 ± 6.04	$\mathbf{43.63 \pm 1.59}$	0.23 ± 0.02
	w/o propagation	200.55 ± 17.00	80.73 ± 5.54	45.16 ± 1.24	0.24 ± 0.01
	w/o SIR	290.78 ± 33.92	102.00 ± 9.93	45.79 ± 1.61	0.31 ± 0.03
	MepoGNN	$\mathbf{186.07 \pm 4.99}$	$\mathbf{74.30 \pm 1.99}$	44.43 ± 0.77	$\mathbf{0.22 \pm 0.01}$

especially for long horizons. Among all baseline models, GraphWaveNet gets the best performance. However, our proposed two MepoGNN models get the very significant improvement over all baseline models. For two types of graph learning module, dynamic one gets slightly better performance than adaptive one. Figure 4 compares the 7 d ahead prediction results of Tokyo and Hyogo of the top two baseline models and MepoGNN model with dynamic graph learning module. From the prediction results, GraphWaveNet and ColaGNN can not produce accurate predictions for high daily confirmed cases during the outbreak. This phenomenon could be explained by different data distributions of daily confirmed cases in training dataset and test dataset. The test dataset covers the period of fifth epidemic wave in Japan which is much more severe than previous ones. Deep learning models have difficulty to predict these high daily confirmed cases that never happened before the fifth wave. However, with the help of metapopulation SIR module, our proposed MepoGNN model can handle this problem and make significantly better prediction for unprecedented surge of cases. This capability is very crucial for a trustworthy epidemic forecasting model.

Ablation Study: To demonstrate the effect of different components of our model, we conduct an ablation study for MepoGNN models with two different graph learning modules, respectively. The variants are as follows: **(1) w/o glm**: Remove the graph learning module of MepoGNN model; **(2) w/o propagation**: Remove the metapopulation propagation from metapopulaiton SIR module (which means metapopulation SIR model is reduced to SIR model); **(3) w/o SIR**: Remove the metapopulation SIR module completely. Table 2 demonstrates that all three components can bring significant boost of performance for our model. Particularly, it is easy to find that the biggest performance drop happens when removing the metapopulation SIR module. Because the metapopulation SIR module enables the capability of MepoGNN model to handle the unprecedented surge of cases.

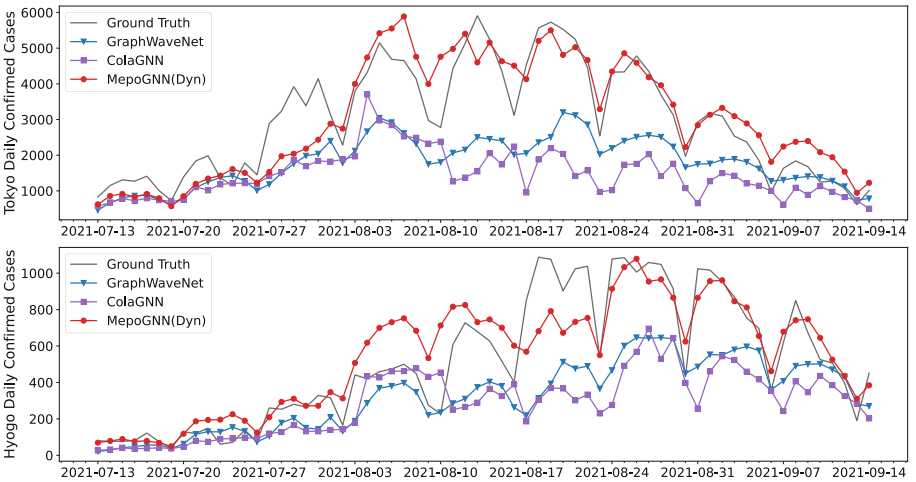

Fig. 4. Predicted daily confirmed cases of Tokyo and Hyogo with horizon = 7.

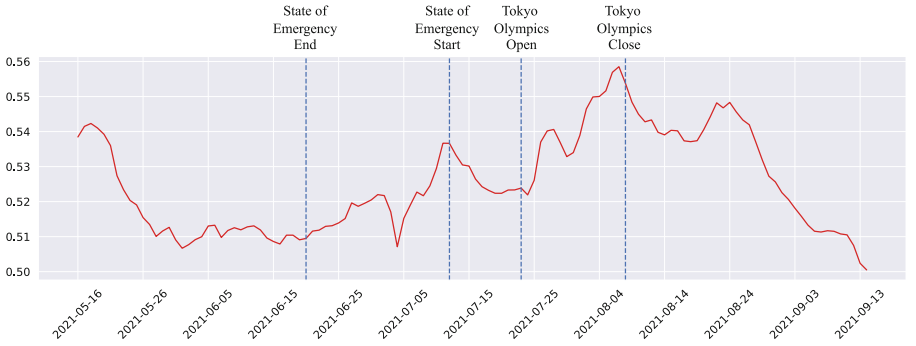

Fig. 5. 7-day moving average of predicted β of Tokyo with horizon = 7.

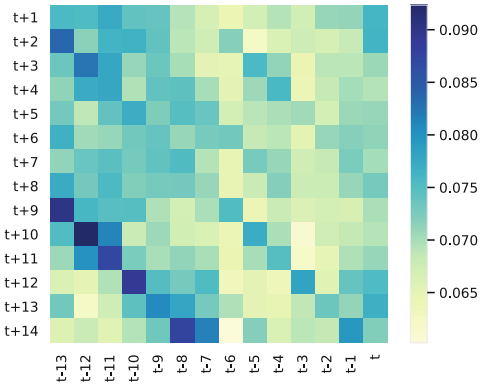

Fig. 6. Learned time weight matrix in dynamic graph learning module.

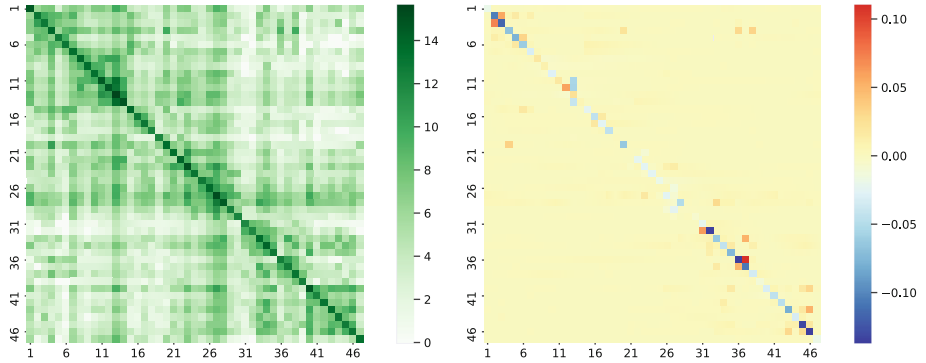

Fig. 7. Learned adaptive mobility graph of the 47 prefectures of Japan with log transformation (left) and its difference with static commuter graph (right).

5.4 Case Study

The final output of MepoGNN model is fully produced by metapopulation SIR module. It brings significant interpretability for our model. We conduct an analysis for the predicted parameters of metapopulation SIR module to demonstrate the interpretability. As shown in Fig. 5, we plot weekly average of predicted β of Tokyo at 7 d ahead horizon in validation and test dataset and label major events and policy changes on timeline. β starts to increase when state of emergency ends and starts to decrease when state of emergency starts. β rapidly increases during Tokyo Olympics, and decreases after it. It demonstrates the predicted β is consistent with reality. Figure 6 shows the learned time weight matrix of dynamic graph learning module. The most significant time lag of mobility effect on epidemic spread is 22 d. This result is consistent with a public health research [28] which states that the effective reproduction number significantly increased 3 weeks after the nightlife places mobility increased in Tokyo. Although the used indicator is different from our research, the mechanisms behind time lag could be similar. Figure 7 shows the learned graph of adaptive graph learning module and the difference between it and commuter graph. The learned adaptive mobility graph keeps the major structure of commuter graph. And the minor change from initialization can reflect the difference between commuter graph and spatial epidemic propagation.

6 Conclusion

Since the outbreak of COVID-19, epidemic forecasting has become a key research topic again. In this study, we propose a novel hybrid model for epidemic forecasting that incorporates spatio-temporal graph neural networks and graph learning mechanisms into metapopulation SIR model. Our model can not only predict the number of confirmed cases but also explicitly learn the time/region-varying epidemiological parameters and the underlying epidemic propagation graph from heterogeneous data in an end-to-end manner. Then, we evaluate our model by

using real COVID-19 infection data and big human mobility data of 47 prefectures in Japan. The evaluation results demonstrate the superior performance as well as the high reliability and interpretability of our model.

Acknowledgment. This work was partially supported by JST SICORP Grant Number JPMJSC2104.

References

1. Wang, L., Li, X.: Spatial epidemiology of networked metapopulation: an overview. Chin. Sci. Bull. **59**(28), 3511–3522 (2014)
2. Wang, J., Wang, X., Wu, J.: Inferring metapopulation propagation network for intra-city epidemic control and prevention. In: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 830–838 (2018)
3. Brockmann, D., Helbing, D.: The hidden geometry of complex, network-driven contagion phenomena. Science **342**(6164), 1337–1342 (2013)
4. Hufnagel, L., Brockmann, D., Geisel, T.: Forecast and control of epidemics in a globalized world. Proc. Natl. Acad. Sci. **101**(42), 15124–15129 (2004)
5. Wesolowski, A., Eagle, N., Tatem, A.J., Smith, D.L., Noor, A.M., Snow, R.W., et al.: Quantifying the impact of human mobility on malaria. Science **338**(6104), 267–270 (2012)
6. Venna, S.R., Tavanaei, A., Gottumukkala, R.N., Raghavan, V.V., Maida, A.S., et al.: A novel data-driven model for real-time influenza forecasting. IEEE Access **7**, 7691–7701 (2018)
7. Wu, Y., Yang, Y., Nishiura, H., Saitoh, M.: Deep learning for epidemiological predictions. In: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, pp. 1085–1088 (2018)
8. Arora, P., Kumar, H., Panigrahi, B.K.: Prediction and analysis of covid-19 positive cases using deep learning models: a descriptive case study of India. Chaos, Solitons Fractals **139**, 110017 (2020)
9. Deng, S., Wang, S., Rangwala, H., Wang, L., Ning, Y.: Cola-gnn: cross-location attention based graph neural networks for long-term ili prediction. In: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, pp. 245–254 (2020)
10. Kapoor, A., Ben, X., Liu, L., Perozzi, B., Barnes, M., Blais, M., et al.: Examining covid-19 forecasting using spatio-temporal graph neural networks. arXiv preprint arXiv:2007.03113 (2020)
11. Kermack, W.O., McKendrick, A.G.: A contribution to the mathematical theory of epidemics. In: Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character, vol. 115, no. 772, pp. 700–721 (1927)
12. Dehning, J., Zierenberg, J., Spitzner, F.P., Wibral, M., Neto, J.P., Wilczek, M., et al.: Inferring change points in the spread of covid-19 reveals the effectiveness of interventions. Science **369**(6500), eabb9789 (2020)
13. Chang, S., Pierson, E., Koh, P.W., Gerardin, J., Redbird, B., Grusky, D., et al.: Mobility network models of covid-19 explain inequities and inform reopening. Nature **589**(7840), 82–87 (2021)
14. Chinazzi, M., Davis, J.T., Ajelli, M., Gioannini, C., Litvinova, M., Merler, S., et al.: The effect of travel restrictions on the spread of the 2019 novel coronavirus (covid-19) outbreak. Science **368**(6489), 395–400 (2020)

15. Jiang, R., et al.: Countrywide origin-destination matrix prediction and its application for covid-19. In: Dong, Y., Kourtellis, N., Hammer, B., Lozano, J.A. (eds.) ECML PKDD 2021. LNCS (LNAI), vol. 12978, pp. 319–334. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-86514-6_20
16. Aleta, A., Martin-Corral, D., Pastore y Piontti, A., Ajelli, M., Litvinova, M., Chinazzi, M., et al.: Modelling the impact of testing, contact tracing and household quarantine on second waves of covid-19. Nat. Hum. Behav. **4**(9), 964–971 (2020)
17. Chang, S.L., Harding, N., Zachreson, C., Cliff, O.M., Prokopenko, M.: Modelling transmission and control of the covid-19 pandemic in Australia. Nat. Commun. **11**(1), 1–13 (2020)
18. Yang, C., Zhang, Z., Fan, Z., Jiang, R., Chen, Q., Song, X., et al.: Epimob: interactive visual analytics of citywide human mobility restrictions for epidemic control. IEEE Trans. Vis. Comput. Graph. 1 (2022)
19. Gao, J., Sharma, R., Qian, C., Glass, L.M., Spaeder, J., Romberg, J., et al.: Stan: spatio-temporal attention network for pandemic prediction using real-world evidence. J. Am. Med. Inf. Assoc. **28**(4), 733–743 (2021)
20. Cui, Y., Zhu, C., Ye, G., Wang, Z., Zheng, K.: Into the unobservables: a multi-range encoder-decoder framework for covid-19 prediction. In: Proceedings of the 30th ACM International Conference on Information & Knowledge Management, pp. 292–301 (2021)
21. Yu, B., Yin, H., Zhu, Z.: Spatio-temporal graph convolutional networks: a deep learning framework for traffic forecasting. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence, pp. 3634–3640 (2018)
22. Li, Y., Yu, R., Shahabi, C., Liu, Y.: Diffusion convolutional recurrent neural network: data-driven traffic forecasting. In: International Conference on Learning Representations (2018)
23. Wu, Z., Pan, S., Long, G., Jiang, J., Zhang, C.: Graph wavenet for deep spatial-temporal graph modeling. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence, pp. 1907–1913 (2019)
24. Dauphin, Y.N., Fan, A., Auli, M., Grangier, D.: Language modeling with gated convolutional networks. In: International conference on machine learning, pp. 933–941. PMLR (2017)
25. Su, W., Fu, W., Kato, K., Wong, Z.S.Y.: "Japan live dashboard" for covid-19: a scalable solution to monitor real-time and regional-level epidemic case data. In: Context Sensitive Health Informatics: The Role of Informatics in Global Pandemics, pp. 21–25. IOS Press (2021)
26. Wu, Z., Pan, S., Long, G., Jiang, J., Chang, X., Zhang, C.: Connecting the dots: multivariate time series forecasting with graph neural networks. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 753–763 (2020)
27. Bai, L., Yao, L., Li, C., Wang, X., Wang, C.: Adaptive graph convolutional recurrent network for traffic forecasting. Adv. Neural Inf. Process. Syst. **33**, 17804–17815 (2020)
28. Nakanishi, M., Shibasaki, R., Yamasaki, S., Miyazawa, S., Usami, S., Nishiura, H., et al.: On-site dining in Tokyo during the covid-19 pandemic: time series analysis using mobile phone location data. JMIR mHealth and uHealth **9**(5), e27342 (2021)
29. Wang, L., Adiga, A., Chen, J., Sadilek, A., Venkatramanan, S., Marathe, M.: Causal-gnn: causal-based graph neural networks for spatio-temporal epidemic forecasting. In: Proceedings of the AAAI Conference on Artificial Intelligence (2022)