Graph Message Passing with Cross-location Attentions for Long-term ILI Prediction

Songgaojun Deng,[1] Shusen Wang,[1] Huzefa Rangwala,[2] Lijing Wang,[3] Yue Ning[1]
[1]Stevens Institute of Technology
[2]George Mason University
[3]University of Virginia
sdeng4, shusen.wang, yue.ning@stevens.edu, rangwala@cs.gmu.edu, lw8bn@virginia.edu

January 1, 2020

ABSTRACT

Forecasting influenza-like illness (ILI) is of prime importance to epidemiologists and health-care providers. Early prediction of epidemic outbreaks plays a pivotal role in disease intervention and control. Most existing work has either limited long-term prediction performance or lacks a comprehensive ability to capture spatio-temporal dependencies in data. Accurate and early disease forecasting models would markedly improve both epidemic prevention and managing the onset of an epidemic. In this paper, we design a **cro**ss-**l**ocation **a**ttention based **g**raph **n**eural **n**etwork (Cola-GNN) for learning time series embeddings and location aware attentions. We propose a graph message passing framework to combine learned feature embeddings and an attention matrix to model disease propagation over time. We compare the proposed method with state-of-the-art statistical approaches and deep learning models on real-world epidemic-related datasets from United States and Japan. The proposed method shows strong predictive performance and leads to interpretable results for long-term epidemic predictions.

1 Introduction

Epidemic disease propagation that involves large populations and wide areas can have a significant impact on society. The Center for Disease Control and Prevention (CDC) estimates 79,400 deaths from influenza occurred during the 2017-2018 season in the United States [1]. Early forecasting of infectious diseases such as influenza-like illness (ILI) provides optimal opportunities for timely intervention and resource allocation. It helps with the timely preparation of corresponding vaccines in health care departments which leads to reduced financial burden. For instance, the World Health Organization (WHO) reports that Australia spent over 352 million dollars on routine immunization in the 2017 fiscal year [2]. We focus on the problem of long term ILI forecasting with lead time from 1 to 20 weeks based on the influenza surveillance data collected for multiple locations (states and regions). Given the process of data collection and surveillance lag, accurate statistics for influenza warning systems are often delayed by a few weeks, making early prediction imperative. However, there are a few challenges in long-term epidemic forecasting. First, the temporal dependency is hard to capture with short-term input data. Without manually added seasonal trends, most statistical models fail to provide high accuracy. Second, the influence from other locations has not been exhaustively explored with limited data input. Spatio-temporal effects have been studied but they usually require adequate data sources to achieve good performance [22].

Existing work on epidemic prediction has been focused on various aspects: 1) Traditional causal models [15, 8, 3], including compartmental models and agent-based models, employ disease progression mechanisms such as Susceptible-Infectious-Recovered (SIR) to capture the dynamics of ILI diseases. Compartmental models focus on mathematical modeling of population-level dynamics. Agent-based models simulate the propagation process at the individual level with contact networks. Calibrating these models is challenging due to the high dimensionality of the parameter space.

---

[1]https://tinyurl.com/y3tf8ebl
[2]https://tinyurl.com/y2duz5p8

2) Time series prediction with statistical models such as Autoregressive (AR) and its variants (e.g., VAR) are not suitable for long term ILI trend forecasting given that the disease activities and human environments evolve over time. 3) Machine learning and deep learning methods [21, 26, 31, 29] such as recurrent neural networks have been explored in recent years but they barely consider cross-spatial effects in long term disease propagation.

In this paper, we focus on long term (10-20 weeks) prediction of the count of ILI patients using data from a limited time range (20 weeks). To tackle this problem, we explore a graph propagation model with deep spatial representations to compensate the loss of temporal information. Assuming each location is a node, we design a graph neural network framework to model epidemic propagation at the population level. Meanwhile, we investigate recurrent neural networks for capturing sequential dependencies in local time series data and temporal convolutions for identifying short-window patterns. Our key contributions are summarized as follows:

- We propose a novel graph-based deep learning framework for long-term epidemic prediction from a time-series forecasting perspective. This is one of the first works of graph neural networks adapted to epidemic forecasting.

- We investigate a location-aware attention mechanism to capture location correlations. The influence of locations can be directed and automatically optimized in the model learning process. The attention matrix is further evaluated as an adjacency matrix in the graph neural network for modeling disease propagation.

- We design a temporal convolution module to automatically extract temporal dependencies and hidden features for time series data of multiple locations. The learned temporal features for each location are utilized as node attributes for the graph neural network.

- The proposed method, Cola-GNN, outperforms a broad range of state-of-the-art models on three real-word datasets with different long-term prediction settings. We also demonstrate the effectiveness of its learned attention matrix compared to a geographical adjacency matrix in an ablation study.

2 Related Work

2.1 Influenza Prediction

In many studies, forecasting influenza or influenza-like illnesses (ILI) case counts is formulated as time series regression problems, where autoregressive models are widely used [27, 1, 11, 30]. Instead of focusing on seasonal effects, Wang et al. [30] propose a dynamic poisson autoregressive model to improve short-term prediction accuracy (e.g. 1-4 weeks). Furthermore, variations of particle filters and ensemble filters have been used to predict influenza activities. Yang et al. [32] evaluate the performance of six state-of-the-art filters to forecast influenza activity and concluded that the models have comparable performance. Ensemble methods such as matrix factorization based regression and nearest neighbor based regression have been studied [5]. While autoregressive, filter-based, and ensemble models are simple and straightforward, they often neglect the geographical dependence in disease propagation.

Attempts to study spatio-temporal effects in influenza disease modeling are not rare. Waller et al. propose a hierarchical Bayesian parametric model for the spatio-temporal interaction of generic disease mapping [28]. A non-parametric Bayesian method [22] is proposed for predicting spatial and temporal variation of influenza cases. Venna et al. develop data-driven approaches involving climatic and geographical factors for real-time influenza forecasting [26]. Wu et al. use deep learning for modeling spatio-temporal patterns in epidemiological prediction problems [31]. Despite their impressive performance, these methods have limitations such as the requirement of additional data which are not readily available and long-term prediction performance is not satisfactory. Improving the long-term epidemiological prediction with restricted training data is an open research problem.

2.2 Long-term Epidemic Prediction

Long-term prediction (aka multi-step prediction), that is, predicting several steps ahead, is a challenge in time series prediction. Long-term prediction has to face growing uncertainties arising from various problems such as accumulation of errors and lack of information. Long-term prediction methods can be categorized into two types: (i) direct methods and (ii) iterative methods. Direct methods predict a future value using the past values in one shot. Iterative methods recursively invoke short-term predictors to make long-term predictions. Specifically, they use the observed data $x_1, \ldots, x_t$ to predict the next step x_{t+10}, then use $x_2, \cdots, x_{t+1}$ to predict x_{t+11}, and so on.

For long-term predictions using time series data, Sorjamaa et al. combine a direct prediction strategy and sophisticated input selection criteria [23]; Qian-Li et al. and Du et al. develop neural network based methods to improve the
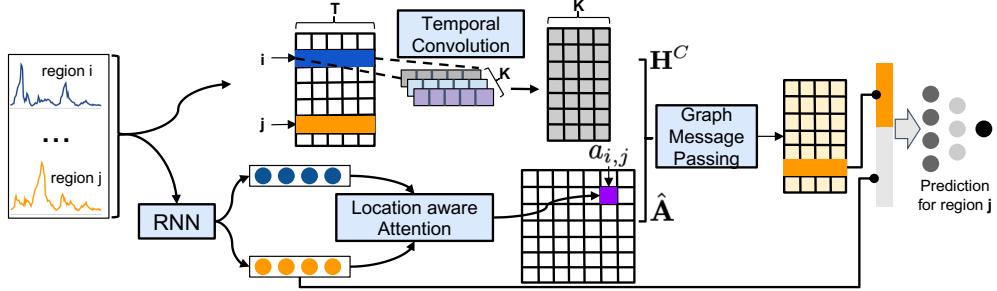
Figure 1: The overview of the proposed framework. Eq. 4-6 are skipped for brevity.

performance of long-term prediction [20, 10]. Recent works [26, 31] explore deep learning models for direct long-term epidemiological predictions. DEFSI [29] combines deep neural network methods with causal models to address high-resolution ILI incidence forecasting. Yet most of these models rely heavily on extrinsic data to improve accuracy.

3 The Proposed Method

3.1 Problem Formulation

We formulate the epidemic prediction problem as a regression task with multiple time series as input. Throughout the paper, we denote the number of locations by N and the time span for one input example as T. We use the terms region and location interchangeably.

At each time step t, the multi-location epidemiology profile is denoted by $\mathbf{x}_t \in \mathbb{R}^N$ whose elements are the observations from N sources/locations, e.g. the influenza patient counts per week (t) in N locations. We further denote the training data in a time-span of size T as $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_T] \in \mathbb{R}^{N \times T}$. The objective is to predict an epidemiology profile at a future time point $T + h$ where h refers to the horizon/lead time of the prediction.

The proposed framework as shown in Figure 1 consists of three modules: 1) location-aware attention to capture location wise interactions, 2) temporal convolutional layer to capture local temporal features, 3) global graph message passing to combine the temporal features and the location-aware attentions to generate further hidden features and make predictions. The pseudocode is described in Algorithm 1 and each module is described as below.

Algorithm 1: Cola-GNN

Input: Time series data $\{\mathbf{X}, \mathbf{y}\}$ from multiple locations, geographical adjacency matrix $\mathbf{A}^g$
Output: Model parameters Θ

1 **for** *each epoch* **do**
2   Randomly sample a mini batch
3   **for** *each region i* **do**
4           $\mathbf{h}_{i,T} \leftarrow$ RNN module$(\mathbf{x}_{i:})$
5           $\mathbf{h}_i^C \leftarrow$ Temporal Conv$(\mathbf{x}_{i:})$
6   **for** *each region pair (i, j)* **do**
7           $\hat{a}_{i,j} \leftarrow$ Loc-Aware Attn$(\mathbf{h}_{i,T}, \mathbf{h}_{j,T}, \mathbf{A}^g)$
                                          ▷ Simultaneous calculations for all regions
8   **for** *each region i* **do**
9           $\mathbf{h}_i^l \leftarrow$ Graph Message Passing$(\mathbf{h}_i^C, \hat{\mathbf{A}})$
10          $\hat{y}_i \leftarrow$ Output$\big([\mathbf{h}_{i,T}; \mathbf{h}_i^{(l)}]\big)$
11      $\Delta\mathcal{L}(\Theta) \leftarrow$ BackProp$\big(\mathcal{L}(\Theta), \mathbf{y}, \hat{\mathbf{y}}, \Theta\big)$
12      $\Theta \leftarrow \Theta - \eta\Delta\mathcal{L}(\Theta)$ ▷ SGD step

---

3.2 Location-aware Attention

In this study, without precise population movement data, we dynamically model the impact of one area on other areas during the epidemics of infectious disease. We first learn hidden states for each location given a time period using a Recurrent Neural Network (RNN) given its great success in sequential (temporal) data prediction. Specifically, we use a simple and classic vanilla RNN in this module. The RNN module can be replaced by Gated Recurrent Unit (GRU) [7] or Long short-term memory (LSTM) [14]; however, in this application, RNN achieves the best performance compared to GRU and LSTM.

Given the multi-location time series data $\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_T] \in \mathbb{R}^{N \times T}$, we employ a global RNN model to capture the temporal dependencies of all locations. For location i, an instance of a time series is represented by $\mathbf{x}_{i:} = [x_{i,1}, ..., x_{i,T}] \in \mathbb{R}^{1 \times T}$. Let D be the dimension of the hidden state. For each element $x_{i,t}$ in the input sequence, the RNN updates its hidden state according to

$$\mathbf{h}_{i,t} = \tanh\left(\mathbf{w}x_{i,t} + \mathbf{U}\mathbf{h}_{i,t-1} + \mathbf{b}\right) \in \mathbb{R}^D, \tag{1}$$

where $\mathbf{h}_{i,t}$ is the hidden state vector at time t and $\mathbf{h}_{i,t-1}$ is the hidden state vector at time $t-1$; $\tanh$ is the non-linear activation function; $\mathbf{w} \in \mathbb{R}^D$, $\mathbf{U} \in \mathbb{R}^{D \times D}$, and $\mathbf{b} \in \mathbb{R}^D$ determine the adaptive weight and bias vectors of the RNN. Let $\mathbf{h}_i = \mathbf{h}_{i,T}$ be the last hidden state and we will use it to represent location i.

Next, we define an attention coefficient $a_{i,j}$ for measuring the impact of location j on location i.

Additive attention (or multi-layer perceptron attention) [2] and multiplicative attention (or dot-product attention) [25, 24] are the two most commonly used attention mechanisms. They share the same idea of computing the alignment score between elements from two sources, but with different compatibility functions. We utilize the compatibility function of additive attention due to its better predictive quality, which is defined as:

$$a_{i,j} = \mathbf{v}^T g(\mathbf{W}^s \mathbf{h}_i + \mathbf{W}^t \mathbf{h}_j + \mathbf{b}^s) + b^v, \tag{2}$$

where g is an activation function that is applied element-wise; $\mathbf{W}^s, \mathbf{W}^t \in \mathbb{R}^{d_a \times D}$, $\mathbf{v} \in \mathbb{R}^{d_a}$, $\mathbf{b}^s \in \mathbb{R}^{d_a}$, and $b^v \in \mathbb{R}$ are trainable parameters. d_a is a hyperparameter that controls the dimensions of the parameters in Eq. 2. Assuming that the impact of location i on location j is different than vice versa, we obtain an asymmetric attention coefficient matrix $\mathbf{A}$ where each row indicates the degree of influence by other locations on the current location. Usually, a softmax function is used to transform the attention scores to a probability distribution. In our problem, the overall impact of other locations vary for different places. For instance, compared to New York, Hawaii may be less affected overall by other states. Instead, we perform normalization over the rows of $\mathbf{A}$ to normalize the impact of other locations on one location:

$$\mathbf{a}_{i:} \longleftarrow \frac{\mathbf{a}_{i:}}{\max(\|\mathbf{a}_{i:}\|_p, \epsilon)}, \tag{3}$$

where ϵ is a small value to avoid division by zero, and $\|\cdot\|_p$ denotes the ℓ_p-norm.

Given the geographic nature of this task, we also consider the spatial distance between two locations. We use $\mathbf{A}^g$ to indicate the connectivity of locations: $a_{i,j}^g = 1$ means locations i and j are neighbors [3]. The correlation of the two locations may be affected by their geographic distance, i.e. nearby areas may have similar topographic or climatic characteristics that make them have similar flu outbreaks. Non-adjacent areas may also have potential dependencies due to population movements and similar geographical features. Simulating all the factors related to a flu outbreak is difficult. Therefore, we consider both the attention derived from historical data and the geographical distances of the locations. The final location-aware attention matrix is obtained by combining the geographical adjacency matrix $\tilde{\mathbf{A}}^g$ and the attention matrix $\mathbf{A}$. The combination is accomplished by an element-wise gate $\mathbf{M}$, learned from the attention matrix which evolves over time. We consider the attention matrix to be a feature matrix with gate $\mathbf{M}$ being adapted from the feature fusion gate [13]:

$$\tilde{\mathbf{A}}^g = \mathbf{D}^{-\frac{1}{2}} \mathbf{A}^g \mathbf{D}^{-\frac{1}{2}}, \tag{4}$$

$$\mathbf{M} = \sigma(\mathbf{W}^m \mathbf{A} + b^m \mathbf{1}_N \mathbf{1}_N^T), \tag{5}$$

$$\hat{\mathbf{A}} = \mathbf{M} \odot \tilde{\mathbf{A}}^g + (\mathbf{1}_N \mathbf{1}_N^T - \mathbf{M}) \odot \mathbf{A}, \tag{6}$$

where Eq. 4 is for normalization, $\mathbf{D}$ is the degree matrix defined as $d_{ii} = \sum_{j=1}^N a_{ij}^g$. $\mathbf{W}^m \in \mathbb{R}^{N \times N}$ and $b^m \in \mathbb{R}$ are trainable parameters.

---

[3]By default, each location is adjacent to itself.

3.3 Temporal Convolution Layer

Besides the spatial dependencies, the outbreak of influenza also has its unique characteristics over time. For instance, the United States experiences annual epidemics of seasonal flu. Most of the time flu activity peaks between December and February, and it can last as late as May [4]. Convolutional Neural Networks (CNN) have shown successful results in capturing various important local patterns from grid data and sequence data. We apply 1D CNN filters to every row of $\mathbf{X}$ to capture the temporal dependency; note that the row $\mathbf{x}_{s:}$ is the observed sequential data at location s. Specifically, we define K filters where each filter $\mathbf{c}_k \in \mathbb{R}^{1 \times Q}$ and Q is chosen to be the maximum window length T in our experiments. Convolutional operations yield $\mathbf{H}^C \in \mathbb{R}^{N \times K}$, where $h_{i,k}^C$ represents the convolutional value of the i-th row vector and the k-th filter. Formally, this convolution operation is given by

$$h_{i,k}^C = \text{ReLU}\left(\text{MaxPool}\left(\sum_{\tau=1}^{Q} x_{i,\tau} \times c_{k,\tau} \right) \right). \tag{7}$$

Max pooling is needed when $Q < T$ as in Kim [16]. To constrain the data, we also apply a nonlinearity to the convolution results. Then the new detected temporal feature of each row/location is $\mathbf{h}_i^C = [h_{i,1}^C, ..., h_{i,K}^C] \in \mathbb{R}^{1 \times K}$.

3.4 Graph Message Passing (the Propagation Model)

After learning the cross-location attentions (Section 3.2) and the local hidden features (Section 3.3), we design a flu propagation model using graph neural networks. Graph neural networks iteratively update the node features from their neighbors. When generalized to irregular domains, this operation is often referred to as message passing or neighbor aggregation. Epidemic disease propagation at the population level is usually affected by human connectivity and transmission. Considering each location as a node in a graph, we take advantage of graph neural networks to model the epidemic disease propagation among different locations. We model the adjacency matrix using the cross-location attention matrix and the nodes' initial features using the the temporal convolutional features. With $\mathbf{h}_i^{(l-1)} \in \mathbb{R}^{F^{(l-1)}}$ denoting node features of node i in layer $(l-1)$ and $\hat{a}_{i,j}$ denoting the location-aware attention coefficient from node j to node i, the message passing graph neural network can be described as

$$\mathbf{h}_i^{(l)} = g\left(\sum_{j \in \mathcal{N}} \hat{a}_{i,j} \mathbf{W}^{(l-1)} \mathbf{h}_j^{(l-1)} + \mathbf{b}^{(l-1)} \right), \tag{8}$$

where g denotes a nonlinear activation function, $\mathbf{W}^{(l-1)} \in \mathbb{R}^{F^{(l)} \times F^{(l-1)}}$ is the weight matrix for hidden layer l with with $F^{(l)}$ feature maps, and $\mathbf{b}^{(l-1)} \in \mathbb{R}^{F^{(l)}}$ is a bias. $\mathcal{N}$ is the set of locations. $\mathbf{h}_i^{(0)}$ is initialized with $\mathbf{h}_i^C$ at the first layer.

3.5 Output Layer (Prediction)

For each location, we learn the RNN hidden states ($\mathbf{h}_{i,T} \in \mathbb{R}^D$) from its own historical sequence data, as well as the graph features ($\mathbf{h}_i^{(l)} \in \mathbb{R}^{F^{(l)}}$) learned from other locations' data in our propagation model. We combine these two features and feed them to the output layer for prediction, which is defined as:

$$\hat{y}_i = \phi\left(\boldsymbol{\theta}^\top [\mathbf{h}_{i,T}; \mathbf{h}_i^{(l)}] + b^\theta \right), \tag{9}$$

where ϕ is the activation function (identity or nonlinear) and $\boldsymbol{\theta} \in \mathbb{R}^{D+F^{(l)}}, b^\theta \in \mathbb{R}$ are model parameters.

3.6 Optimization

We compare the prediction value of each location with the corresponding ground truth and then optimize a regularized ℓ_1-norm loss:

$$\mathcal{L}(\Theta) = \sum_{i=1}^{N} \sum_{m=1}^{n_i} |y_{i,m} - \hat{y}_{i,m}| + \lambda \mathcal{R}(\Theta), \tag{10}$$

where n_i is the number of samples in location i obtained by a moving window, shared by all locations, $y_{i,m}$ is the true value of location i in sample m, and $\hat{y}_{i,m}$ is the model prediction. Θ stands for all training parameters and $\mathcal{R}(\Theta)$ is the regularization term (e.g. ℓ_2-norm). All model parameters can be trained via back-propagation and optimized by the Adam algorithm [18] given its efficiency and ability to avoid overfitting.

---

[4] https://tinyurl.com/yxevpqs9

Table 1: Dataset statistics: min, max, mean, and standard deviation (SD) of patient counts; dataset size means number of locations multiplied by # of weeks.

Data set	Size	Min	Max	Mean	SD
Japan-Prefectures	47×348	0	26635	655	1711
US-Regions	10×785	0	16526	1009	1351
US-States	49×360	0	9716	223	428

4 Experiment Setup

4.1 Datasets

We prepare three real-world datasets for experiments: Japan-Prefectures, US-States and US-Regions and their data statistics are shown in Table 1.

- **Japan-Prefectures** We collect this data from the Infectious Diseases Weekly Report (IDWR) [5] in Japan. This dataset contains weekly influenza-like-illness statistics (patient counts) from 47 prefectures in Japan, ranging from August 2012 to March 2019.
- **US-States** We collect the influenza disease data from the Center for Disease Control (CDC) [6]. It contains the count of patient visits for ILI for each week and each state in United States from 2010 to 2017. After removing a state with missing data we kept 49 states remaining in this dataset.
- **US-Regions** This dataset is the ILINet portion of the US-HHS (Department of Health and Human Services) dataset [6], consisting of weekly influenza activity levels for 10 HHS regions of U.S. mainland for the period of 2002 to 2017. Each HHS region represents some collection of associated states. We use flu patient counts for each region, which is calculated by combining state-specific data.

Data is normalized to 0-1 range for each region. The maximum value of the region is set to 1, and the minimum value of the region is set to 0. After ordering the data by time, the first 50% is used for training, next 20% for validation, and the last 30% for testing. Validation data is used to determine the number of epochs that should be run to avoid overfitting. We fixed the validation and test sets by dates for different lead time values. In this case, the test data covers 2.1, 4.5, and 2.1 flu seasons in Japan-Prefectures, US-States and US-Regions respectively. Accordingly, there are at least 3, 7.2 and 3 flu seasons in the three training sets. All data is normalized based on the maximum and minimum values of the training data.

4.2 Evaluation Metrics

In the experiments, we adopt the following metrics for evaluation. Denote the prediction and true values to be $\{\hat{y}_1, ..., \hat{y}_n\}$ and $\{y_1, ..., y_n\}$, respectively. We do not distinguish regions in evaluation.

The **Root Mean Squared Error (RMSE)** measures the difference between predicted and true values after projecting the normalized values into the real range:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}.$$

The **Mean Absolute Error (MAE)** is a measure of difference between two continuous variables:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i|.$$

The **Pearsons Correlation (PCC)** is a measure of the linear dependence between two variables:

$$\text{PCC} = \frac{\sum_{i=1}^{n}(\hat{y}_i - \bar{\hat{y}})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(\hat{y}_i - \bar{\hat{y}})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}.$$

---

[5] `https://tinyurl.com/y5dt7stm`
[6] `https://tinyurl.com/y39tog3h`

Leadtime is the number of weeks that the model predicts in advance. For instance, if we use $X_{N,T}$ as input and predict the infected patients of the fifth week (leadtime = 5) after current week T, the ground truth (expected output) is $X_{N,T+5}$.

4.3 Comparison Methods

We compare our model with several state-of-the-art methods and their variants listed as below.

- **Autoregressive (AR)** Autoregressive models have been widely applied for time series forecasting [4, 30]. Basically, the future state is modeled as a linear combination of past data points. We train an autoregressive model for each location. No data and parameters are shared among locations.

- **Global Autoregression (GAR)** This model is mainly used when training data is limited. We train one global model using the data available from each location.

- **Vector Autoregression (VAR)** The VAR models cross-signal dependence to address the potential drawback of the AR model, i.e. the signal sources are processed independently of each other. Therefore, it introduces more parameters and is more expensive in training.

- **Autoregressive Moving Average (ARMA)** ARMA contains the autoregressive terms and moving-average terms together. A considerable amount of preprocessing has to be performed before such model fitting. The order of the moving average is set to 2 in implementation.

- **Recurrent Neural Network (RNN)** RNNs have demonstrated powerful abilities to predict temporal dependencies. We employ a global RNN for our problem, that is, parameters are shared across different regions. RNN can be be replaced by GRU or LSTM. Experimentally, fancy RNN models did not achieve better results, so we only consider simple RNN for comparison.

- **RNN+Attn [6]** This model considers the self-attention mechanism in a global RNN. In the calculation of rnn units, the hidden state is replaced by a summary vector, which uses the attention mechanism to aggregate all the information of the previous hidden state.

- **CNNRNN-Res [31]** A deep learning framework that combines CNN, RNN and residual links to solve epidemiological prediction problems.

- **GCNRNN-Res** A variation of CNNRNN-Res. We change the CNN module to a GCN [19] module with two hidden layers, the feature dimensions of which remain unchanged. We utilize the given geographical adjacent matrix.

Hyper-parameter Setting & Implementation Details In our model, we adopt exponential linear unit (ELU) [9] as nonlinearity for function g in Eq. 2, and idendity for function ϕ in Eq. 9. In the experiment, the input window size is 20 weeks, which spans roughly five months. The hyperparameter d_a in the location-aware attention is set to $\frac{D}{2}$ to reduce the number of parameters compared to standard additive attention. The order of the norm p in Eq. 3 is set to 2, and ϵ is 1e-12. The number of filters K is 10 in Eq. 7. For all methods using the RNN module, we tune the hidden dimensions of the RNN module from $\{10, 20, 30\}$, and 20 yields the best performance in most cases. The number of RNN hidden layers and graph layers is optimized to 1 and 2 respectively. In the training process, the best models are selected by early stopping when the validation accuracy does not increase for 200 consecutive epochs, and the maximum epoch is 1500. All the parameters are initialized with Glorot initialization [12] and trained using the Adam [17] optimizer with weight decay 5e-4, and dropout rate 0.2. The initial learning rate of all methods is searched from the set $\{0.001, 0.005, 0.01\}$. The batch size is set to 32 across all datasets. All experimental results are the average of 10 randomized trials.

Suppose the dimension of weight matrices in graph message passing is set to $D \times D$, the number of parameters of the proposed model is $O(D^2 + N^2)$. In our epidemiological prediction problems, D and N are limited by relatively small numbers.

5 Results

5.1 Prediction Performance

We evaluate our approach in short-term (leadtime = 2, 3, 4) and long-term (leadtime = 5, 10, 15) lead time settings. We ignore the case of leadtime = 1, because symptom monitoring data is usually delayed by at least one week. Table 2 summarizes the results of all the methods in terms of RMSE and PCC in short-term settings. We can observe that when the lead time is relatively small, our method achieves the most stable and optimal performance on all datasets. In

Table 2: RMSE and PCC performance of different methods on the three datasets with leadtime = 2, 3, 4. Bold face indicates the best result of each column and underlined the second-best. **(Short-term)**

RMSE(↓)	**Japan-Prefectures**			**US-Regions**			**US-States**		
	2	3	4	2	3	4	2	3	4
GAR	1232	1628	1865	536	715	859	150	187	213
AR	1377	1705	1901	570	757	888	161	204	231
VAR	1361	1711	1910	741	870	967	290	276	283
ARMA	1371	1703	1902	560	742	874	161	200	228
RNN	1001	1259	1366	<u>513</u>	<u>689</u>	805	<u>149</u>	<u>181</u>	<u>204</u>
RNN+Attn	1166	1572	1706	613	753	962	152	186	210
CNNRNN-Res	1133	1550	1795	571	738	<u>802</u>	205	239	253
GCNRNN-Res	<u>1031</u>	<u>1129</u>	<u>1133</u>	736	847	935	194	210	236
Cola-GNN	**919**	**1060**	**1072**	**483**	**633**	**765**	**136**	**167**	**191**

PCC(↑)	2	3	4	2	3	4	2	3	4
GAR	0.804	0.626	0.461	0.932	0.881	0.835	0.945	0.914	0.893
AR	0.752	0.579	0.428	0.927	0.878	0.834	0.94	0.909	0.885
VAR	0.754	0.585	0.419	0.859	0.797	0.741	0.765	0.79	0.78
ARMA	0.754	0.579	0.428	0.927	0.876	0.833	0.939	0.909	0.886
RNN	0.892	0.833	0.813	<u>0.94</u>	<u>0.895</u>	<u>0.855</u>	<u>0.948</u>	<u>0.922</u>	0.9
RNN+Attn	0.85	0.668	0.604	0.887	0.859	0.774	0.947	0.922	<u>0.903</u>
CNNRNN-Res	0.852	0.673	0.513	0.92	0.862	0.829	0.904	0.86	0.842
GCNRNN-Res	<u>0.893</u>	<u>0.889</u>	<u>0.886</u>	0.871	0.831	0.796	0.903	0.884	0.854
Cola-GNN	**0.911**	**0.893**	**0.894**	**0.944**	**0.905**	**0.863**	**0.955**	**0.933**	**0.907**

Table 3: RMSE and PCC performance of different methods on the three datasets with leadtime = 5, 10, 15. Bold face indicates the best result of each column and underlined the second-best. **(Long-term)**

RMSE(↓)	**Japan-Prefectures**			**US-Regions**			**US-States**		
	5	10	15	5	10	15	5	10	15
GAR	1988	2065	2016	991	1377	1465	236	314	340
AR	2013	2107	2042	997	1330	1404	251	306	327
VAR	2025	1942	1899	1059	1270	1299	295	324	352
ARMA	2013	2105	2041	989	1322	1400	250	306	326
RNN	1376	1696	1629	<u>896</u>	1328	1434	<u>217</u>	274	315
RNN+Attn	1746	1612	1823	1065	1367	1368	234	315	334
CNNRNN-Res	1942	1865	1862	936	<u>1233</u>	<u>1285</u>	267	<u>260</u>	<u>250</u>
GCNRNN-Res	<u>1178</u>	**1384**	**1457**	1051	1298	1402	248	275	288
Cola-GNN	**1156**	<u>1403</u>	<u>1500</u>	**871**	**1126**	**1218**	**202**	**241**	**232**

PCC(↑)	5	10	15	5	10	15	5	10	15
GAR	0.339	0.288	0.47	0.79	0.581	0.485	0.875	0.777	0.742
AR	0.310	0.238	0.483	0.792	0.612	<u>0.527</u>	0.863	0.773	0.723
VAR	0.3	0.426	0.474	0.685	0.508	0.467	0.758	0.709	0.6529
ARMA	0.31	0.253	0.486	0.792	<u>0.614</u>	0.52	0.862	0.773	0.725
RNN	0.821	0.616	0.709	<u>0.821</u>	0.587	0.499	**0.886**	<u>0.821</u>	0.758
RNN+Attn	0.59	0.741	0.522	0.752	0.554	0.552	<u>0.884</u>	0.78	0.739
CNNRNN-Res	0.38	0.438	0.467	0.782	0.552	0.4851	0.822	0.82	<u>0.847</u>
GCNRNN-Res	<u>0.875</u>	**0.823**	**0.774**	0.739	0.554	0.4471	0.844	0.814	0.814
Cola-GNN	**0.883**	<u>0.818</u>	<u>0.754</u>	**0.832**	**0.719**	**0.639**	0.897	**0.822**	**0.859**

this case, most of the methods can capture relatively good performance in the three datasets, which is due to the small information gap between the history window and the predicted time, thus the models can fit the temporal pattern more easily. The one exception is that in the Japan-Prefectures dataset, the results of most baseline methods deteriorate with a slight increase in lead time. A possible reason for this phenomenon in the Japan-Prefectures dataset is that the seasonal influenza curve in the dataset is less predictive, even for short-term forecasts. The dataset statistic also shows that Japan-Prefectures dataset has the largest standard deviation.

Table 3 reports the RMSE and PCC results in long-term settings. Overall, the proposed method achieves best performance for most datasets with long lead time windows (leadtime = 5, 10 or 15 weeks). Autoregression models have poor performance, especially VAR which has the largest number of model parameters. This suggests the importance of controlling the model complexity for data insufficiency problems. Recurrent neural network models only achieve
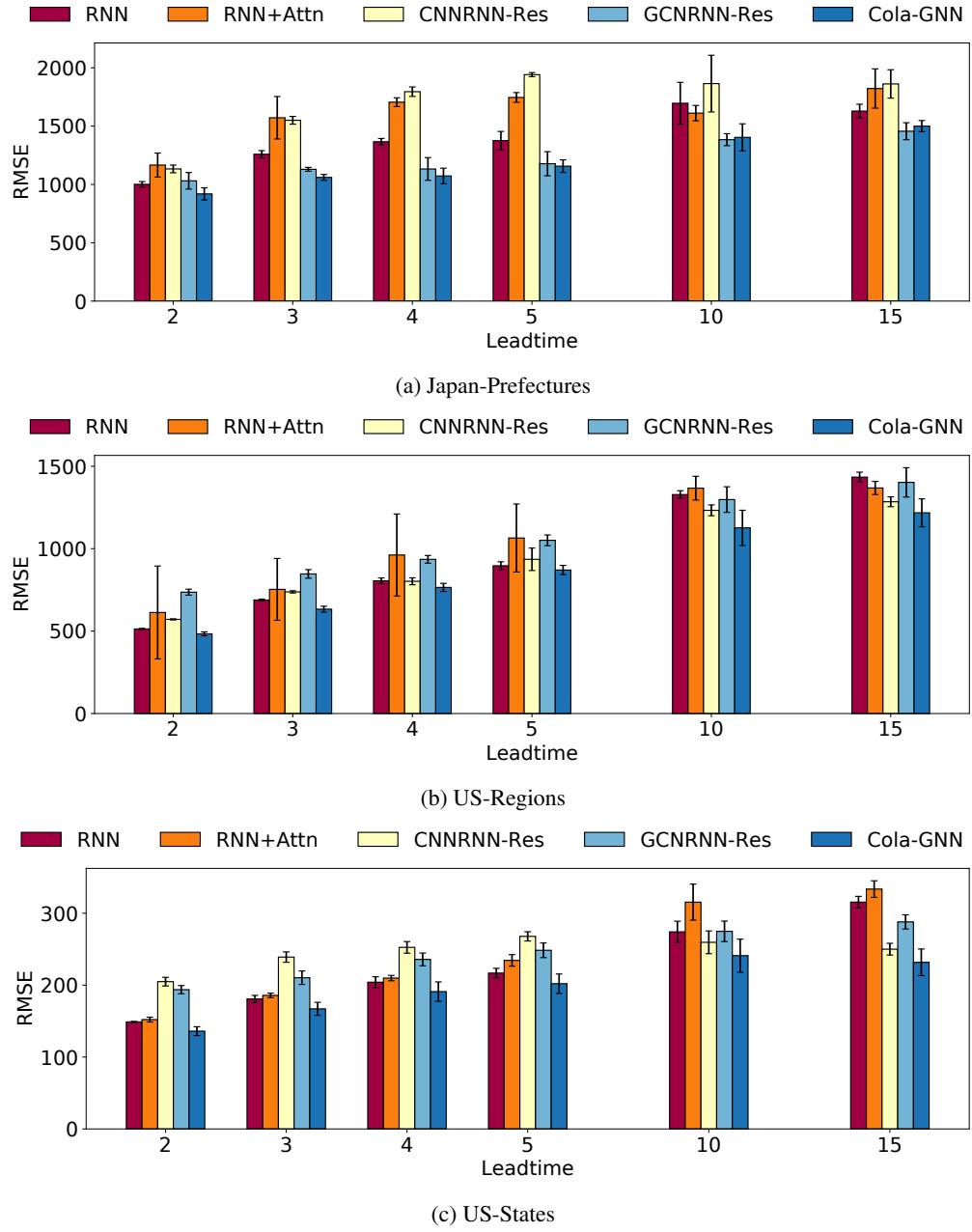
Figure 2: RMSE of the flu prediction models with different leadtimes on three datasets.

good predictive performance when lead time is small, which demonstrates that long-term predictions require a better design to capture spatial and temporal dependencies. CNNRNN-Res uses geographic location information and it only performs well in the US-States dataset. In the Japan-Prefectures and US-Regions datasets, the model performs poorly when having long lead time windows. Its variant GCNRNN-Res contains a graph convolutional module that learns the features from adjacent regions. GCNRNN-Res has achieved good results in Japan-Prefectures and US-States datasets. It proves that the graph convolution module can help capture long-term dependencies. The performances of CNNRNN-Res and GCNRNN-Res are unstable on three datasets and often show large variance in multiple rounds of training. To better visualize the results, we show the mean value and standard deviation of the 10 different runs of some models in Figure 2 and Figure 3.

If we look at the big picture of the prediction performance, the performance difference of all methods is relatively small when the lead time is 2, but as the lead time increases, the predictive power of simple methods (such as autoregressive)

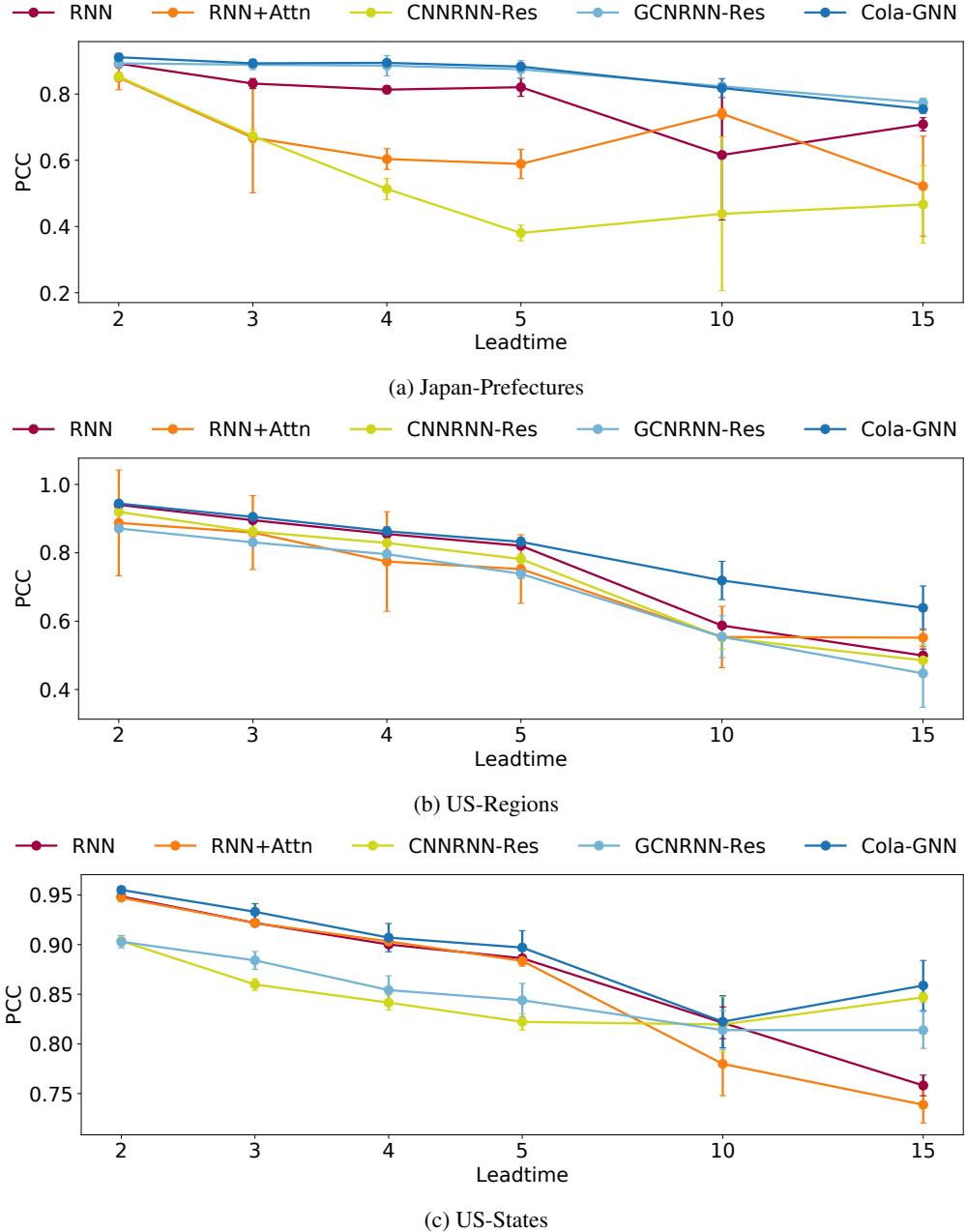(a) Japan-Prefectures

(b) US-Regions

(c) US-States

Figure 3: PCC of the flu prediction models with different leadtimes on three datasets.

decreases significantly. This suggests that modeling temporal dependence is challenging when a relatively large gap exists between the historical window and the expected prediction time.

5.2 Case Studies

To evaluate the long-term predictive performance of the proposed model, we plot a sequence of predictions, where lead time is 15, in the test set. Four better baselines were chosen and the comparison on the three datasets is shown in Figure 4. We randomly select three locations from each dataset and observe that even though we are using a relatively small window (20) to predict long-term flu count (leadtime = 15), our model is able to better capture the trend and outbreak time of the epidemic outbreak.
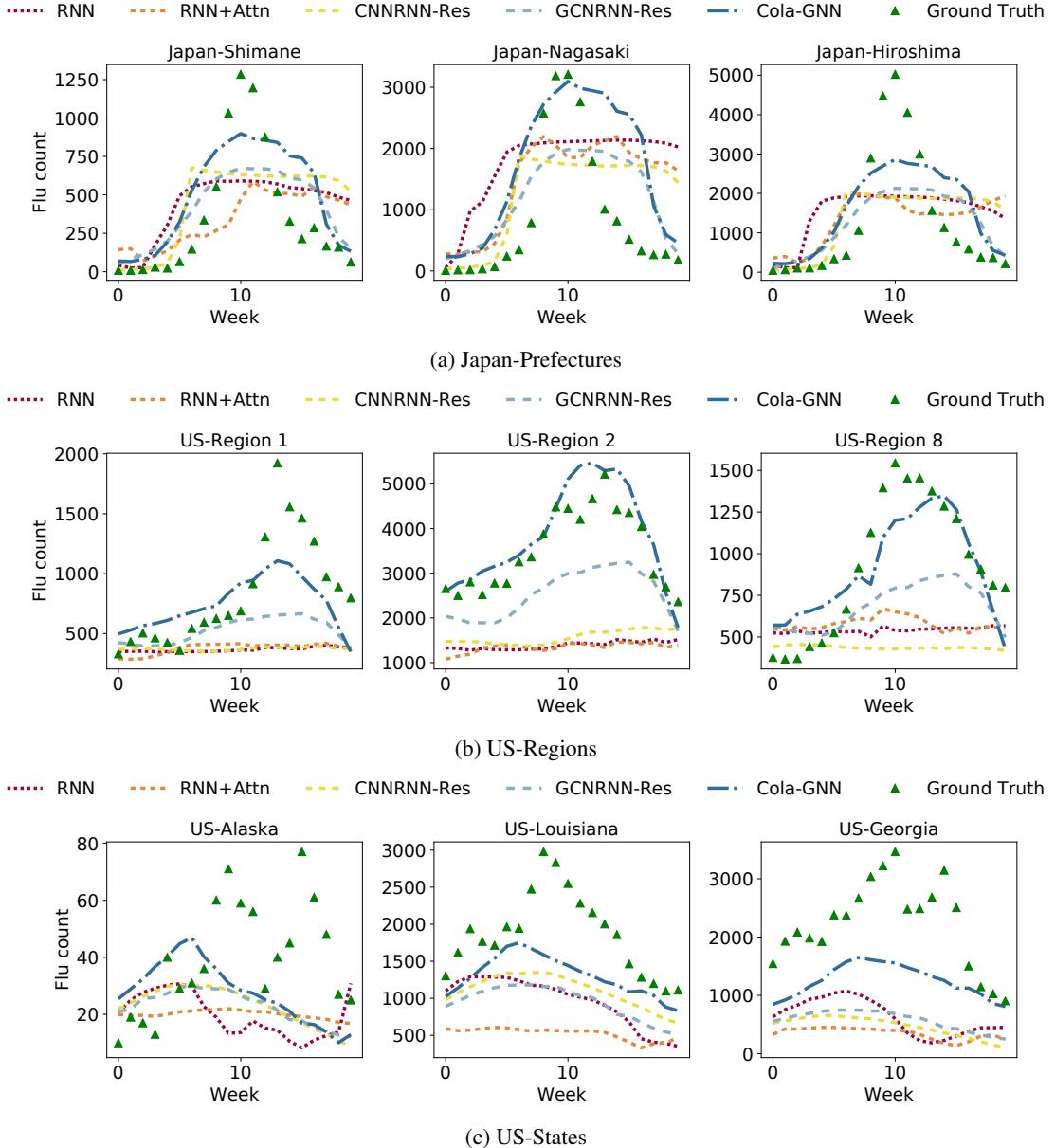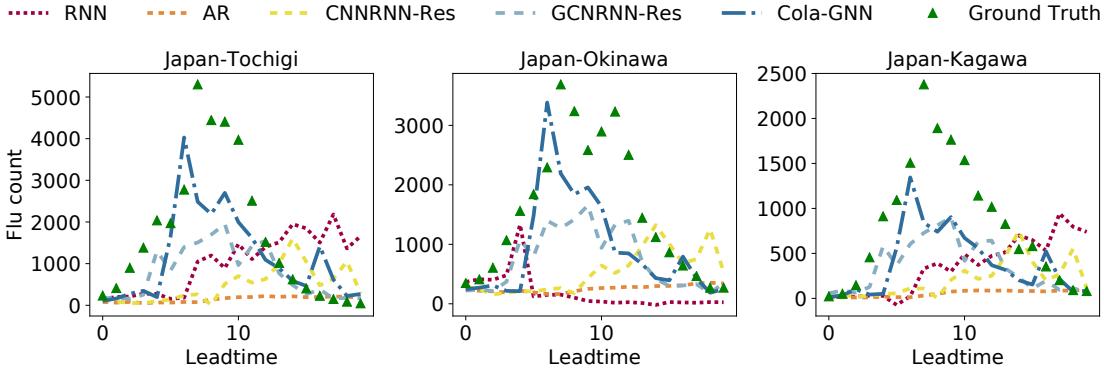
Figure 4: Iterative prediction results when leadtime = 15. We test the models trained in leadtime = 15 by moving the history window.
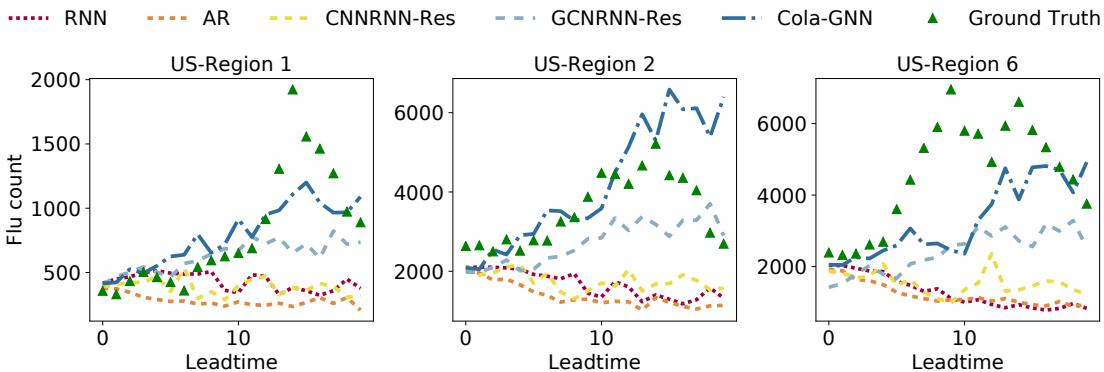
We fix the input window and plot the prediction curve of leadtime from 1 to 20. Likewise, we also randomly sample three locations from each dataset. From the observation in Figure 5, our model tends to capture the peaks and trends in future time based on given historical data.
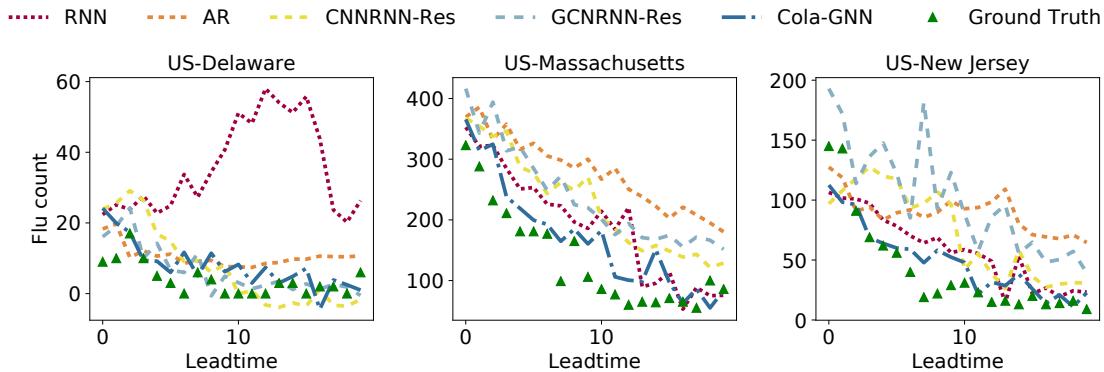
5.3 Attention Visualization

Figure 6 shows an example of the location-aware attention mechanism with a lead time of 15 in the US-Regions dataset. In this example, we focus on *region 5*. We visualize the input data of *region 5* and two regions {*region 3*, *region 4*} with highest attention values for *region 5*, as well as two regions that has lowest attention values {*region 1*, *region 8*}. We normalize the data by regions to better compare flu outbreaks across regions. The time period of the light yellow shade is the input sequence of window = 20. The vertical line indicates the predicted time. We are using only a small part of the sequence of all regions to predict the epidemic outbreak of *region 5* in 15 weeks. The regions

Figure 5: Direct prediction curves with fixed input windows. We fix the input window and test the models trained with a lead time from 1 to 20.

with higher attention share same early epidemic outbreak as *region 5* while regions with lower attention values have later outbreak times.

We show the normalized geolocation distance matrix in Figure 7a, which is calculated according to Eq. 4, and the Pearson correlation coefficient of input time series in Figure 7b. The learned attention matrix (Figure 7c) utilizes geolocation information as well as additive attentions among regions. From the learned attention matrix, we observe that adjacent regions sometimes get higher attention values. Meanwhile, non-adjacent regions can also receive high attention values given their similar long-term influenza trends. The learned attention reveals hidden dynamics (e.g., epidemic outbreaks and peak time) among regions.
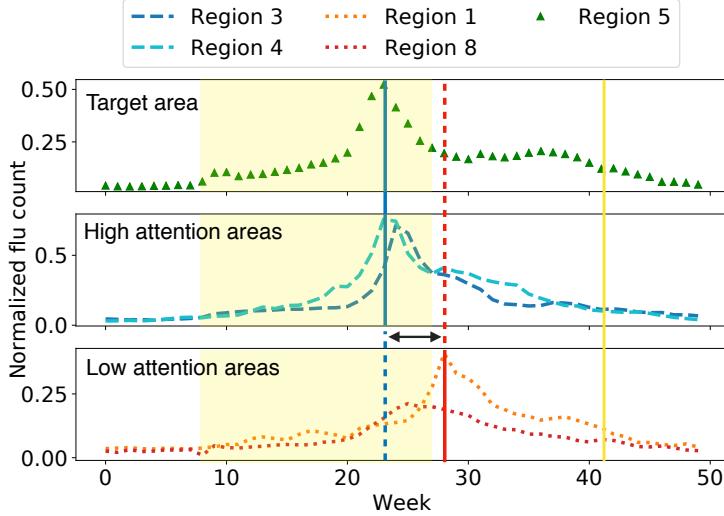
Figure 6: An example of the location-aware attention mechanism with a lead time of 15 in the US-Regions data set. Yellow line indicates prediction time. Shaded area is the input.
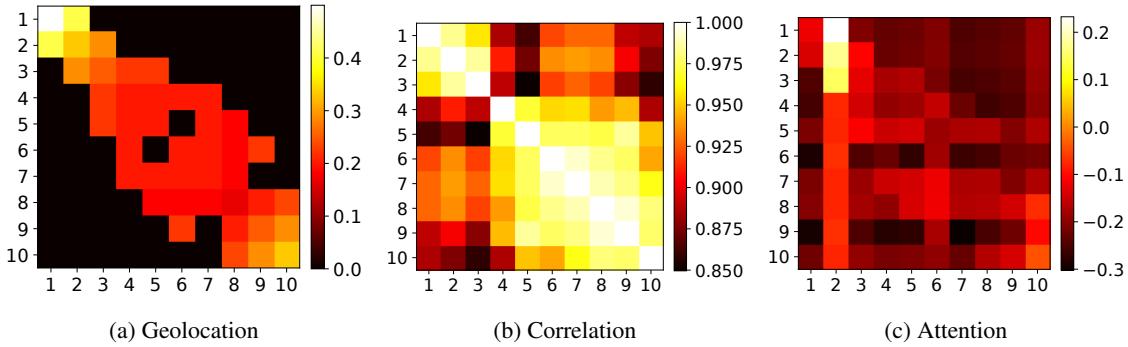

| (a) Geolocation | (b) Correlation | (c) Attention |

Figure 7: Comparison of original geolocation matrix (7a), input correlation matrix (7b), and learned attention matrix (US region).

5.4 Ablation Tests

To analyze the effect of each component in our framework, we perform the ablation tests on all the datasets with the follow settings:

- Cola-GNN w/o $temp$: Remove the temporal convolution module from the proposed model, and use the raw time series input as features in graph message passing.

- Cola-GNN w/o loc: Remove the location-aware attention module and directly use the geographical adjacent matrix which defines the spatial distance between pairs of locations.

The results of RMSE and PCC are shown in Table 4. We can observe that in most cases, variant versions of the proposed method can achieve very good performance. In the US-states dataset, models without temporal or location-aware attention modules are sometimes slightly better than the full model. The US-states dataset has the lowest number of reported influenza cases compare with two other datasets, and the standard deviation is small. Overall, the full model achieves optimal performance across all datasets. Note that all datasets are relatively small in size, which means that adding more parameters may affect the performance due to overfitting. However, adding temporal and spatial modules does not change the short-term (leadtime = 2,3,4) prediction very much. Instead, for long-term predictions (leadtime = 15), involving these two modules produces better results.

Table 4: Ablation test results in RMSE(top) and PCC(bottom) when leadtime=2,3,4,5,10,15 for three datasets.

RMSE($\downarrow$)	2	3	4	5	10	15
	Japan-Prefectures					
Cola-GNN w/o $temp$	911	1115	1204	1310	1388	1517
Cola-GNN w/o loc	942	1154	1164	1195	1473	1576
Cola-GNN	919	1060	1072	1156	1403	1500
	US-Regions					
Cola-GNN w/o $temp$	485	662	772	888	1144	1228
Cola-GNN w/o loc	499	666	782	890	1179	1292
Cola-GNN	483	633	765	871	1126	1218
	US-States					
Cola-GNN w/o $temp$	138	169	188	194	251	251
Cola-GNN w/o loc	138	169	193	202	246	246
Cola-GNN	136	167	191	202	241	232
PCC($\uparrow$)	2	3	4	5	10	15
	Japan-Prefectures					
Cola-GNN w/o $temp$	0.91	0.867	0.846	0.818	0.793	0.744
Cola-GNN w/o loc	0.914	0.881	0.89	0.88	0.781	0.727
Cola-GNN	0.911	0.893	0.894	0.883	0.818	0.754
	US-Regions					
Cola-GNN w/o $temp$	0.944	0.902	0.861	0.824	0.712	0.588
Cola-GNN w/o loc	0.942	0.898	0.858	0.824	0.682	0.582
Cola-GNN	0.944	0.905	0.863	0.832	0.719	0.639
	US-States					
Cola-GNN w/o $temp$	0.953	0.93	0.908	0.908	0.833	0.836
Cola-GNN w/o loc	0.955	0.931	0.913	0.904	0.856	0.855
Cola-GNN	0.955	0.933	0.907	0.897	0.822	0.859

5.5 Sensitivity Analysis

In this section, we investigate how the prediction performance varies with some hyperparameters.

Size of History Windows To test if our model is sensitive to the length of historical data, we evaluate different window sizes from 10 to 50 with step 5. The experiment was conducted on US-Regions and US-States datasets as shown in Figure 8. The predictive performance in RMSE and MAE with different window sizes are fairly stable. We can avoid training with very long sequences and achieve relatively comparable results.

Size of Graph Features We learn the RNN hidden states from the historical sequence data $h_{i,T}$ and the graph features $h_i^{(l)}$ which involves features of other regions by message passing over location-aware attentions. We vary the dimension of the graph feature from 1 to 15 and evaluate the predictive performance in US-States dataset when leadtime is 15. Figure 9 reports RMSE and MAE results. Features of smaller dimensions result in poor predictive performance due to limited encoding power. The model produces better predictive power when the feature dimension is larger.

RNN Modules The RNN module is used to output a hidden state vector for each location based on given historical data. The hidden state vector is then provided to the location-aware attention module. We replaced the RNN modules with GRU and LSTM to assess their impact on model performance. Figure 10 shows RMSE results for leadtime = 2,5,10,15 in US-Regions and US-States datasets. We found that the performance of GRU and LSTM is not better than a simple RNN. The likely reason is that they involve more model parameters and tend to overfit in the epidemiological datasets.
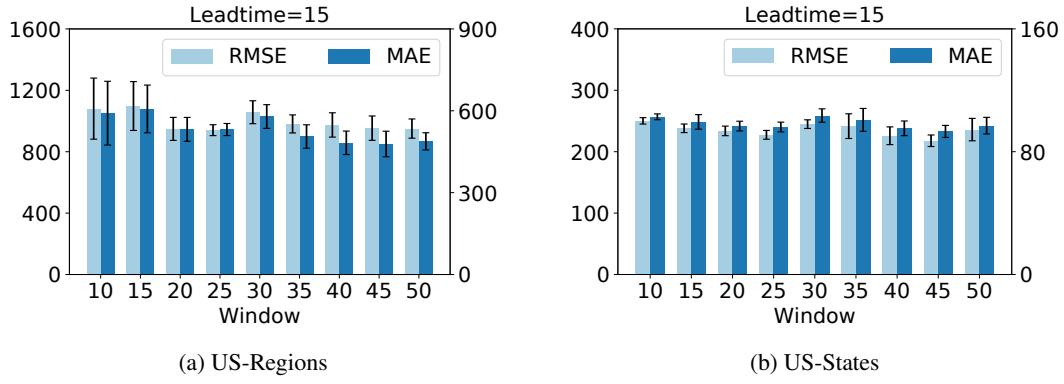
(a) US-Regions   (b) US-States

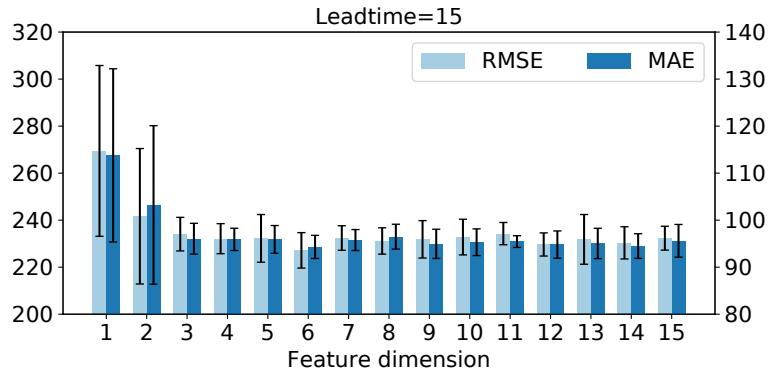Figure 8: Sensitivity analysis on window size.

Figure 9: Sensitivity analysis of graph feature size.

5.6 Model Complexity

Table 5: Runtime comparison of models on the US-States dataset. Runtime is the time spent on a single GPU per epoch.

Architecture	Parameters	Runtime(s)
GAR	21	0.01
AR	1,029	0.02
VAR	48,069	0.02
ARMA	1,960	0.03
RNN	481	0.04
RNN+Attn	1,321	0.58
CNNRNN-Res	7,695	0.04
GCNRNN-Res	6,214	0.04
Cola-GNN	3,778	0.21

Table 5 shows the comparison of runtimes and numbers of parameters for each model on the US-States dataset, which has the largest number of *regions* among the three datasets. In this task, all methods can be effectively trained due to the nature of the datasets. Meanwhile, we only utilize flu disease data and geographic location data, while ignoring other external features. Compared with other methods, the proposed method has no significant effect on training efficiency. It can also control the size of the model parameters to prevent overfitting.

6 Conclusion

In this work, we propose a graph-based deep learning framework with cross-location attentions to study the spatio-temporal influence of long-term epidemiological predictions. We demonstrate the effectiveness of the proposed model
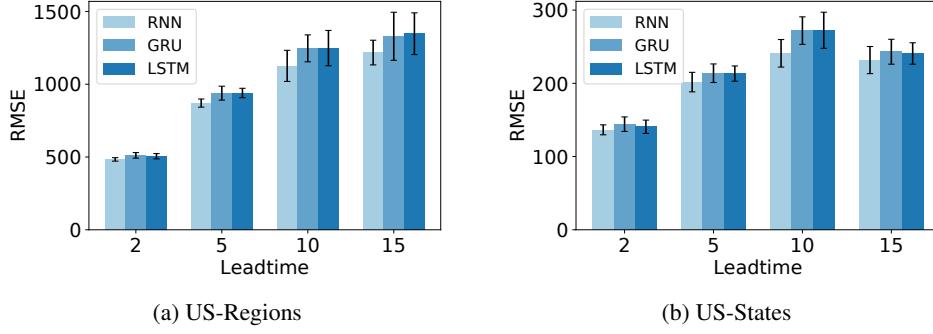
|(a) US-Regions | (b) US-States|

Figure 10: Sensitivity analysis on RNN modules.

on real-world epidemiological datasets. The proposed method is not flexible enough in the case that different models are trained for different lead time settings. Future work will consider iterative predictions to increase the flexibility of the model. Another research direction is to involve more complex dependencies such as weather, social factors, and population migration. We intend to determine if the prediction accuracy is improved when using external indicators. Furthermore, it is also essential to identify the main factors affecting the epidemic outbreak of one area by learning multiple areas simultaneously.

References

[1] Harshavardhan Achrekar, Avinash Gandhe, Ross Lazarus, Ssu-Hsin Yu, and Benyuan Liu. Predicting flu trends using twitter data. In *2011 IEEE conference on computer communications workshops (INFOCOM WKSHPS)*, pages 702–707. IEEE, 2011.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[3] Keith R Bisset, Jiangzhuo Chen, Xizhou Feng, VS Kumar, and Madhav V Marathe. Epifast: a fast algorithm for large scale realistic epidemic simulations on distributed memory systems. In *Proceedings of the 23rd international conference on Supercomputing*, pages 430–439. ACM, 2009.

[4] John S Brownstein, Shuyu Chu, Achla Marathe, Madhav V Marathe, Andre T Nguyen, Daniela Paolotti, Nicola Perra, Daniela Perrotta, Mauricio Santillana, Samarth Swarup, et al. Combining participatory influenza surveillance with modeling and forecasting: Three alternative approaches. *JMIR public health and surveillance*, 3(4):e83, 2017.

[5] Prithwish Chakraborty, Pejman Khadivi, Bryan Lewis, Aravindan Mahendiran, Jiangzhuo Chen, Patrick Butler, Elaine O Nsoesie, Sumiko R Mekaru, John S Brownstein, Madhav V Marathe, et al. Forecasting a moving target: Ensemble models for ili case count predictions. In *Proceedings of the 2014 SIAM international conference on data mining*, pages 262–270. SIAM, 2014.

[6] Jianpeng Cheng, Li Dong, and Mirella Lapata. Long short-term memory-networks for machine reading. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 551–561, Austin, Texas, November 2016. ACL. doi: 10.18653/v1/D16-1053.

[7] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar, October 2014. ACL. doi: 10.3115/v1/D14-1179.

[8] GMAM Chowell, MA Miller, and C Viboud. Seasonal influenza in the united states, france, and australia: transmission and prospects for control. *Epidemiology & Infection*, 136(6):852–864, 2008.

[9] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). In *Proceedings of the 2015 International Conference on Learning Representations*, volume abs/1511.07289, 2015.

[10] Baoxiang Du, Wei Xu, Bingbing Song, Qun Ding, and Shu-Chuan Chu. Prediction of chaotic time series of rbf neural network based on particle swarm optimization. In *Intelligent Data analysis and its Applications, Volume II*, pages 489–497. Springer, 2014.

[11] Andrea Freyer Dugas, Mehdi Jalalpour, Yulia Gel, Scott Levin, Fred Torcaso, Takeru Igusa, and Richard E Rothman. Influenza forecasting with google flu trends. *PloS one*, 8(2):e56176, 2013.

[12] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256, 2010.

[13] Yichen Gong and Samuel Bowman. Ruminating reader: Reasoning with gated multi-hop attention. In *Proceedings of the Workshop on Machine Reading for Question Answering*, pages 1–11, Melbourne, Australia, July 2018. ACL. doi: 10.18653/v1/W18-2601.

[14] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[15] William Ogilvy Kermack and Anderson G McKendrick. A contribution to the mathematical theory of epidemics. *Proceedings of the royal society of london. Series A, Containing papers of a mathematical and physical character*, 115(772):700–721, 1927.

[16] Yoon Kim. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, October 2014. ACL. doi: 10.3115/v1/D14-1181.

[17] D Kinga and J Ba Adam. A method for stochastic optimization. In *International Conference on Learning Representations*, volume 5, 2015.

[18] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the 2015 International Conference on Learning Representations*, volume abs/1412.6980, 2015.

[19] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

[20] Ma Qian-Li, Zheng Qi-Lun, Peng Hong, Zhong Tan-Wei, and Qin Jiang-Wei. Multi-step-prediction of chaotic time series based on co-evolutionary recurrent neural network. *Chinese Physics B*, 17(2):536, 2008.

[21] José Carlos Santos and Sérgio Matos. Analysing twitter and web queries for flu trend prediction. *Theoretical Biology and Medical Modelling*, 11(1):S6, 2014.

[22] Ransalu Senanayake, Simon O'Callaghan, and Fabio Ramos. Predicting spatio-temporal propagation of seasonal influenza using variational gaussian process regression. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[23] Antti Sorjamaa, Jin Hao, Nima Reyhani, Yongnan Ji, and Amaury Lendasse. Methodology for long-term prediction of time series. *Neurocomputing*, 70(16-18):2861–2869, 2007.

[24] Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448, 2015.

[25] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[26] Siva R Venna, Amirhossein Tavanaei, Raju N Gottumukkala, Vijay V Raghavan, Anthony S Maida, and Stephen Nichols. A novel data-driven model for real-time influenza forecasting. *IEEE Access*, 7:7691–7701, 2018.

[27] Cécile Viboud, Pierre-Yves Boëlle, Fabrice Carrat, Alain-Jacques Valleron, and Antoine Flahault. Prediction of the spread of influenza epidemics by the method of analogues. *American Journal of Epidemiology*, 158(10):996–1006, 2003.

[28] Lance A Waller, Bradley P Carlin, Hong Xia, and Alan E Gelfand. Hierarchical spatio-temporal mapping of disease rates. *Journal of the American Statistical association*, 92(438):607–617, 1997.

[29] Lijing Wang, Jiangzhuo Chen, and Madhav Marathe. Defsi: Deep learning based epidemic forecasting with synthetic information. volume 33, pages 9607–9612, Jul. 2019. doi: 10.1609/aaai.v33i01.33019607.

[30] Zheng Wang, Prithwish Chakraborty, Sumiko R Mekaru, John S Brownstein, Jieping Ye, and Naren Ramakrishnan. Dynamic poisson autoregression for influenza-like-illness case count prediction. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1285–1294. ACM, 2015.

[31] Yuexin Wu, Yiming Yang, Hiroshi Nishiura, and Masaya Saitoh. Deep learning for epidemiological predictions. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 1085–1088. ACM, 2018.

[32] Wan Yang, Alicia Karspeck, and Jeffrey Shaman. Comparison of filtering methods for the modeling and retrospective forecasting of influenza epidemics. *PLoS computational biology*, 10(4):e1003583, 2014.