Cola-GNN: Cross-location Attention based Graph Neural Networks for Long-term ILI Prediction

Songgaojun Deng
Stevens Institute of Technology
Hoboken, New Jersey
sdeng4@stevens.edu

Shusen Wang
Stevens Institute of Technology
Hoboken, New Jersey
shusen.wang@stevens.edu

Huzefa Rangwala
George Mason University
Fairfax, Virginia
rangwala@cs.gmu.edu

Lijing Wang
University of Virginia
Charlottesville, Virginia
lw8bn@virginia.edu

Yue Ning
Stevens Institute of Technology
Hoboken, New Jersey
yue.ning@stevens.edu

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 fails to capture spatio-temporal dependencies in data. In this paper, we design a **c**r**o**ss-**l**ocation **a**ttention based **g**raph **n**eural **n**etwork (Cola-GNN) for learning time series embeddings in long-term ILI predictions. We propose a graph message passing framework to combine graph structures (e.g., geolocations) and time-series features (e.g., temporal sequences) in a dynamic propagation process. We compare the proposed method with state-of-the-art statistical approaches and deep learning models. We conducted a set of extensive experiments on real-world epidemic-related datasets from the United States and Japan. The proposed method demonstrated strong predictive performance and leads to interpretable results for long-term epidemic predictions.

CCS CONCEPTS

• **Information systems** → **Spatial-temporal systems**; *Data mining*; • **Computing methodologies** → *Neural networks.*

KEYWORDS

ILI prediction; dynamic graph neural network; spatial attention

ACM Reference Format:
Songgaojun Deng, Shusen Wang, Huzefa Rangwala, Lijing Wang, and Yue Ning. 2020. Cola-GNN: Cross-location Attention based Graph Neural Networks for Long-term ILI Prediction. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM '20), October 19–23, 2020, Virtual Event, Ireland*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3340531.3411975

CIKM '20, October 19–23, 2020, Virtual Event, Ireland

ACM ISBN 978-1-4503-6859-9/20/10…$15.00
https://doi.org/10.1145/3340531.3411975

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 35.5 million people getting sick with influenza and 34,200 deaths from influenza occurred during the 2018-2019 season in the United States [5]. 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 burdens. For instance, the World Health Organization (WHO) reports that Australia spent over 352 million dollars on routine immunization in the 2017 fiscal year [34]. In this work, we focus on the problem of long term ILI forecasting with lead time from 2 to 15 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 long-term forecasting imperative.

Existing work on epidemic prediction has been focused on various aspects: 1) Traditional causal models [3, 9, 16], 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. 2) 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) Deep learning methods [23, 28, 31, 35] such as recurrent neural networks have been explored in recent years yet they barely consider cross-spatial effects in long term disease propagation.

There are several challenges in long-term epidemic forecasting. First, the temporal dependency is hard to capture with short-term input data. Without manual integration of seasonal trends, most statistical models fail to achieve high accuracy. Second, the influence of other locations often changes over time. Dynamic spatial effects have not been exhaustively explored with limited data input. Spatio-temporal effects have been studied while they usually

require adequate data sources to achieve decent performance in epidemic forecasting [24].

In this paper, we focus on long term (2-15 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 for 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 dilated temporal convolutions for identifying short and long-term patterns. Our key contributions can be 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 dynamic location-aware attention mechanism to capture spatial 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 dilated convolution module to automatically extract both short and long temporal dependencies from time-series data of multiple locations. The learned features for each location are utilized as node attributes for the graph neural network.
- We evaluate the proposed method on a broad range of state-of-the-art models on three real-world 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.

The rest of this paper is organized as follows: Section 2 summarizes the related works on influenze prediction, long-term epidemic prediction, and spatio-temporal prediction. Section 3 introduces the formulation of this research problem and the details of our proposed model. Then, the experimental settings, datasets, and evaluation metrics are shown in Section 4 followed by analytical results in Section 5. Finally, we summarize our work and discuss the potential future directions in Section 6.

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 [1, 11, 29, 32]. Instead of focusing on seasonal effects, Wang et al. [32] proposed 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. [38] evaluated the performance of six state-of-the-art filters to forecast influenza activity and concluded that the models have comparable performance. Matrix factorization and nearest-neighbor based regression have been studied in ensemble methods [6]. There are attempts to study the spatio-temporal effects in influenza disease modeling. Waller et al. developed a hierarchical Bayesian parametric model for the spatio-temporal interaction of generic disease mapping [30]. A non-parametric model based on Gaussian Process regression is introduced to capture the complex spatial and temporal dependencies present in ILI data [24]. Venna et al. developed data-driven approaches involving climatic and geographical factors for real-time influenza forecasting [28]. Wu et al. used deep learning for modeling spatio-temporal patterns in epidemiological prediction problems [35]. Yang et al. presented ARGO (AutoRegression with GOogle search data) for the estimation of influenza epidemics. Despite their impressive performance, these methods have limitations such as the requirement of additional data that are not readily available, and long-term prediction is not satisfactory. For example, Google Correlate data used in ARGO [37] has been shut down in Dec. 2019. Improving the long-term epidemiological prediction with limited training data is an open research problem.

2.2 Long-term Epidemic Prediction

Long-term prediction (aka multi-step prediction) is challenging because of the growing uncertainties arising from the accumulation of errors and lack of complete 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+1}, then use $x_2, \cdots, x_{t+1}$ to predict x_{t+2}, and so on.

Sorjamaa et al. proposed a sophisticated strategy for selecting input variables by optimizing different criteria and using Least Squares Support Vector Machine (LS-SVM) for direct multi-step forecasting [25]. Different kernels were employed to address seasonality, nonstationarity, short and long-term variations in a non-parametric Bayesian method [24]. Recent works [28, 35] explored deep learning models for direct long-term epidemiological predictions and achieved good performance. DEFSI [31] combined deep neural network methods with causal models to address high-resolution ILI incidence forecasting. Yet the majority of these models rely heavily on extrinsic data to improve accuracies, such as longitude and latitude [24], and climate information [28].

2.3 Spatio-temporal Prediction

With the increasing growth of spatio-temporal data nowadays, machine learning models for predicting spatio-temporal events are developed and evaluated on many fields such as societal event forecasting [22, 41], air quality prediction [12, 19], and traffic forecasting [18, 20, 20, 36, 39]. In the prediction of social events, text data such as news articles and tweets are often used as features, which is usually a weak auxiliary feature for ILI prediction as influenza often occurred periodically at the population level. Collecting and processing relevant external data such as news or tweets is also expensive. Latitude, longitude, and climate information usually refer to a small area, and the granularity of statistics is lower than that of influenza. Meanwhile, climate information would be inaccurate due to human error or mechanical failure. In recent studies of air quality and traffic prediction, researchers model spatio-temporal

Table 1: Important notations and descriptions

Notation	Description
T	window size of training data
N	number of locations
h	horizon/lead time of a prediction
$D, F^{(l-1)}, F^{(l)}$	feature dimensions
$\mathbf{X} \in \mathbb{R}^{N \times T}$	training data for N locations
$\mathbf{x}_{i:} \in \mathbb{R}^{1 \times T}$	training data for location i
$\mathbf{A}^g \in \mathbb{R}^{N \times N}$	geographical adjacency matrix
$\mathbf{A} \in \mathbb{R}^{N \times N}$	general attention matrix
$\hat{\mathbf{A}} \in \mathbb{R}^{N \times N}$	location-aware attention matrix
$\mathbf{h}_{i,t}$	RNN hidden states at time t of location i
$\mathbf{h}_i^C$	dilated convolution features of location i
$\mathbf{h}_i^{(l)}$	graph features of location i in l-th layer

dependence between different sensors by integrating graph convolutional networks into recurrent neural networks or convolutional neural networks. However, data sampling for ILI data is different than air or traffic data. For instance, traffic sensors transmit data at 5-minute intervals. ILI data collection usually shows a larger granularity (e.g., weeks) with a delay. Influenza outbreaks also exhibit long seasonality (e.g., about 13 weeks in the United States). It is of great significance to introduce an effective influenza prediction model for long-term ILI prediction given limited data.

3 THE PROPOSED METHOD

3.1 Problem Formulation

We formulate the epidemic prediction problem as a graph-based propagation model. We have N locations in total. Each location (e.g., a city or a state) is a node, and it is associated with a time series input for a window T, e.g., the ILI patient counts for T weeks. We 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 (i.e., the ILI patient counts) at a future time point $T + h$ where h refers to the horizon/lead time of the prediction. The necessary mathematical notations are in Table 1. The proposed framework as shown in Figure 1 consists of three modules: 1) location-aware attention to capture location-wise interactions (edge weights), 2) dilated convolution layer to capture short-term and long-term local temporal dependencies (node attributes), 3) global graph message passing to combine the temporal features and the location-aware attentions, to learn hidden location embeddings and make predictions.

3.2 Directed Spatial Influence Learning

In this study, we dynamically model the impact of one location on other locations during the epidemics of infectious disease. The correlation of two locations can be affected by their geographic distance, i.e. nearby areas may have similar topographic or climatic characteristics that make them have similar flu outbreaks. However, 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 propose a location-aware attention mechanism, which takes into account the temporal dependencies of locations from historical data, as well as the geographical information.

Initially, we learn hidden states for each location given a time period using a Recurrent Neural Network (RNN) [33]. The RNN module can be replaced by Gated Recurrent Unit (GRU) [8] or Long short-term memory (LSTM) [15]. 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 to capture the temporal dependencies cross all locations. An instance for location i 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, 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 at time t and $\mathbf{h}_{i,t-1}$ is from 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. The last hidden state $\mathbf{h}_{i,T}$ is used as the representation ($\mathbf{h}_i$) of location i later.

Next, we define a general attention coefficient $a_{i,j}$ to measure the impact of location j on location i from the hidden states learned from RNN. Additive attention [2] and multiplicative attention [26, 27] are the two most commonly used attention mechanisms. We utilize 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. In our problem, the overall impact of other locations varies for different places. For instance, compared to New York, Hawaii may be less affected overall by other states. Instead of using softmax, 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.

In the final step, we involve the spatial distance between two locations. Let $\mathbf{A}^g$ indicate the connectivity of locations: $a^g_{i,j} = 1$ means locations i and j are neighbors.[1] The location-aware attention matrix is obtained by combining the geographical adjacency matrix $\tilde{\mathbf{A}}^g$, and the attention coefficient matrix $\mathbf{A}$. The combination is accomplished by an element-wise gate $\mathbf{M}$, learned from the general attention matrix that evolves over time. We consider the general attention matrix to be a feature matrix with gate $\mathbf{M}$ being adapted from the feature fusion gate [14]:

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

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

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

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

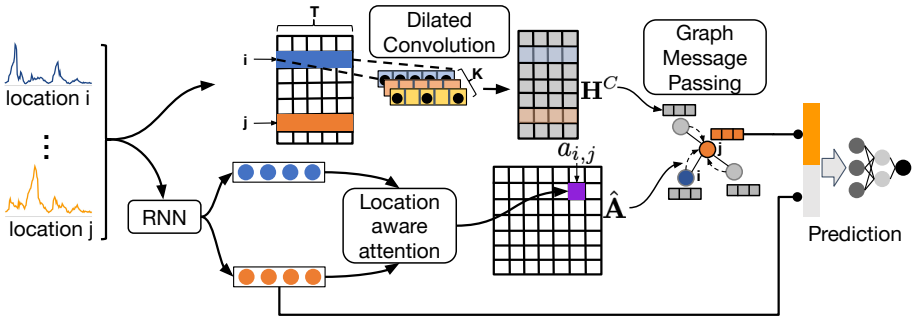


Figure 1: The overview of the proposed framework. The original time series for each location are copied to two components: (1) an RNN model (bottom) for learning directed spatial influence; and (2) a dilated convolution model (top) for learning multi-level temporal features.

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. The feature fusion gate is dynamically learned, and it weighs the contribution of geographic and historical information when modeling the influence of two locations.

3.3 Multi-Scale Dilated Convolution

Besides the spatio-temporal 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.[2] Convolutional Neural Networks (CNN) have shown successful results in capturing various important local patterns from grid data and sequence data. RNN in the location-aware attention module aggregates and learns all the features of historical time steps equally, while the convolutional layer captures important local feature patterns from the original time series. We aim to use the latter to extract important features for graph message passing. Yu and Koltun demonstrated the effectiveness of Dilated Convolution to extract local patterns on images [40]. Thus, we adapt a multi-scale dilated convolutional module [21] which consists of multiple parallel convolutional layers with the same filter and stride size but different dilation rates. We apply 1D CNN filters with different dilation rates to every row of $\mathbf{X}$ to capture temporal dependencies at different levels of granularity; note that the row $\mathbf{x}_{s:}$ is the observed sequential data at location s. Formally, the dilated convolution is a convolution applied to input with defined gaps. Dilated convolution on 1D data is defined as:

$$\mathbf{d}_s[i] = \sum_{l=1}^{L} \mathbf{x}_s[i + k \times l] \times \mathbf{c}[l], \tag{7}$$

where $\mathbf{d}_s$ is the output feature vector, $\mathbf{c}$ represents the convolutional filter of length L, and k is the dilation rate. We use multiple filters to generate different filter vectors. Specifically, for short-term and long-term patterns, we define K filters with dilation rates k_s and k_l ($k_l > k_s$), respectively. Each filter size L is chosen to be the maximum window length T in our experiments. We concatenate the multiple filter vectors to obtain the final convolution output, denoted as $\mathbf{h}_{s:}^{C}$ for location s. The output encodes local patterns with short-term and long-term trends. To constrain the data, we also apply a nonlinear layer to the convolution results.

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

3.4 Graph Message Passing – Propagation

After learning the cross-location attentions (Section 3.2) and the local temporal 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, which is often referred to as message passing. 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 dilated 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 from node j to node i, the message passing can be described as:

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

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 $F^{(l)}$ 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. We use the dilated convolved features instead of the original time series because they capture hidden temporal features with multiple levels of granularity.

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:

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

Algorithm 1: Cola-GNN training

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

for *each epoch* **do**
  Randomly sample a mini batch
  for *each region i* **do**
    $\mathbf{h}_{i,T} \leftarrow$ RNN module$(\mathbf{x}_{i:})$
    $\mathbf{h}_i^C \leftarrow$ Dilated Conv$(\mathbf{x}_{i:})$
  for *each region pair* (i, j) **do**
    $\hat{a}_{i,j} \leftarrow$ Loc-Aware Attn$(\mathbf{h}_{i,T}, \mathbf{h}_{j,T}, \mathbf{A}^{\mathcal{G}})$
    ▷ `Simultaneous calculations for all locations`
  for *each region i* **do**
    $\mathbf{h}_i^l \leftarrow$ Graph Message Passing$(\mathbf{h}_i^C, \hat{\mathbf{A}})$
    $\hat{y}_i \leftarrow$ Output$([\mathbf{h}_{i,T}; \mathbf{h}_i^{(l)}])$
  $\Theta \leftarrow$ BackProp$(\mathcal{L}(\Theta), \mathbf{y}, \hat{\mathbf{y}}, \Theta)$ ▷ `SGD step`

where ϕ is the activation function 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 via gradient descent:

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

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). The pseudocode of the algorithm is described in Algorithm 1.

4 EXPERIMENT SETUP

4.1 Datasets

We prepare three real-world influenza datasets for experiments: Japan-Prefectures, US-States, and US-Regions, and their data statistics are shown in Table 2.

- **Japan-Prefectures** We collect this data from the Infectious Diseases Weekly Report (IDWR) in Japan.[3] 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).[4] It contains the count of patient visits for ILI (positive cases) 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, [4]consisting of weekly influenza activity levels for 10 HHS regions of the 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.

[3]https://tinyurl.com/y5dt7stm
[4]https://tinyurl.com/y39tog3h

Table 2: Dataset statistics: min, max, mean, and standard deviation (SD) of patient counts; dataset size means the 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

We split the data into training, validation, and test set in chronological order at a ratio of 50%-20%-30%. All data are normalized to 0-1 range for each location based on the training data. 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. 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.

4.2 Evaluation Metrics

In the experiments, we denote the prediction and true values to be $\{\hat{y}_1, ..., \hat{y}_n\}$ and $\{y_1, ..., y_n\}$, respectively. We do not distinguish locations in evaluations. We adopt the following metrics for 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 **Pearson's Correlation (PCC)** is a measure of the linear dependence between two variables:

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

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 the 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 below.

- **Autoregressive (AR)** Autoregressive models have been widely applied for time series forecasting [4, 32]. Basically, the future state is modeled as a linear combination of past data points. We

Table 3: RMSE and PCC performance of different methods on three datasets with leadtime = 2, 3, 5, 10, 15. Bold face indicates the best result of each column and underlined the second-best. Relative gain is compared with the second best result.

	Japan-Prefectures					US-Regions					US-States				
RMSE(↓)	2	3	5	10	15	2	3	5	10	15	2	3	5	10	15
GAR	1232	1628	1988	2065	2016	536	715	991	1377	1465	150	187	236	314	340
AR	1377	1705	2013	2107	2042	570	757	997	1330	1404	161	204	251	306	327
VAR	1361	1711	2025	1942	1899	741	870	1059	1270	1299	290	276	295	324	352
ARMA	1371	1703	2013	2105	2041	560	742	989	1322	1400	161	200	250	306	326
RNN	1001	1259	1376	1696	1629	513	689	896	1328	1434	149	181	217	274	315
LSTM	1052	1246	1335	1622	1649	507	688	975	1351	1477	150	180	213	276	307
RNN+Attn	1166	1572	1746	1612	1823	613	753	1065	1367	1368	152	186	234	315	334
DCRNN	1502	1769	2024	2019	1992	711	874	1127	1411	1434	165	209	244	299	298
CNNRNN-Res	1133	1550	1942	1865	1862	571	738	936	1233	1285	205	239	267	260	250
LSTNet	1133	1459	1883	1811	1884	554	801	998	1157	1231	199	249	299	292	292
ST-GCN	996	1115	1129	1541	1527	697	807	1038	1290	1286	189	209	256	289	292
Cola-GNN	**929**	**1051**	**1117**	**1372**	**1475**	**480**	**636**	**855**	**1134**	**1203**	**136**	**167**	**202**	**241**	**237**
% relative gain	6.7%	5.7%	1.1%	11.0%	3.4%	5.3%	7.6%	4.6%	2.0%	2.3%	8.7%	7.2%	5.2%	7.3%	5.2%
PCC(↑)	2	3	5	10	15	2	3	5	10	15	2	3	5	10	15
GAR	0.804	0.626	0.339	0.288	0.470	0.932	0.881	0.790	0.581	0.485	0.945	0.914	0.875	0.777	0.742
AR	0.752	0.579	0.310	0.238	0.483	0.927	0.878	0.792	0.612	0.527	0.940	0.909	0.863	0.773	0.723
VAR	0.754	0.585	0.300	0.426	0.474	0.859	0.797	0.685	0.508	0.467	0.765	0.790	0.758	0.709	0.653
ARMA	0.754	0.579	0.310	0.253	0.486	0.927	0.876	0.792	0.614	0.520	0.939	0.909	0.862	0.773	0.725
RNN	0.892	0.833	0.821	0.616	0.709	0.940	0.895	0.821	0.587	0.499	0.948	0.922	0.886	0.821	0.758
LSTM	0.896	0.873	0.853	0.681	0.695	0.943	0.895	0.812	0.586	0.488	0.948	0.922	0.889	0.820	0.771
RNN+Attn	0.850	0.668	0.590	0.741	0.522	0.887	0.859	0.752	0.554	0.552	0.947	0.922	0.884	0.780	0.739
DCRNN	0.697	0.537	0.292	0.342	0.525	0.897	0.849	0.760	0.604	0.558	0.941	0.886	0.886	0.829	0.837
CNNRNN-Res	0.852	0.673	0.380	0.438	0.467	0.920	0.862	0.782	0.552	0.485	0.904	0.860	0.822	0.820	0.847
LSTNet	0.846	0.728	0.432	0.518	0.515	0.935	0.868	0.746	0.609	0.533	0.913	0.850	0.759	0.760	0.802
ST-GCN	0.902	0.880	0.872	0.735	**0.773**	0.879	0.840	0.741	0.644	0.619	0.907	0.778	0.823	0.769	0.774
Cola-GNN	**0.915**	**0.901**	**0.890**	**0.813**	0.753	**0.946**	**0.909**	**0.835**	**0.717**	**0.639**	**0.955**	**0.933**	**0.897**	**0.822**	**0.856**
% relative gain	1.4%	2.4%	2.1%	9.7%	-	0.6%	1.6%	1.7%	10.2%	3.2%	0.7%	1.2%	0.9%	0.1%	1.1%

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) [33].** 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.
- **Long short-term memory (LSTM) [15]** As a special kind of RNN, LSTM is capable of learning long-term dependencies. We train a vanilla LSTM in implementation.
- **RNN+Attn [7]** 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.
- **DCRNN [20]** A diffusion convolution recurrent neural network, which combines graph convolution networks with recurrent neural networks in an encoder-decoder manner.
- **CNNRNN-Res [35]** A deep learning framework that combines CNN, RNN, and residual links to solve epidemiological prediction problems. It employs CNN to fuse information from data of different locations.
- **LSTNet [18]** This model uses CNN and RNN to extract short-term local dependency patterns among variables and to discover long-term patterns for time series trends.
- **ST-GCN [39]** A deep learning framework for traffic prediction which integrates graph convolution and gated temporal convolution through spatio-temporal convolutional blocks.

Hyper-parameter Setting & Implementation Details In our model, we adopt exponential linear unit (ELU) [10] as nonlinearity for g in Eq. 2, and idendity for ϕ in Eq. 9. In the experiment, the input window size T is 20 weeks, which spans roughly five months. The hyperparameter d_a in the location-aware attention is set to $\frac{D}{2}$. The order of the norm p and ϵ in Eq. 3 are set to 2 and 1e-12. The number of filters K is 10 in multi-scale dilated convolution. The dilation rates of short-term k_s and long-term k_l are set to 1 and 2, respectively. For all methods using the RNN module, we

Table 4: Ablation test results on three datasets.

RMSE(↓)	2	3	5	10	15
	Japan-Prefectures				
Cola-GNN w/o *temp*	**912**	1115	1310	1388	1517
Cola-GNN w/o *loc*	942	1154	1199	1470	1576
Cola-GNN w/o *geo*	1075	1105	1219	1417	1502
Cola-GNN	929	**1050**	**1117**	**1372**	**1475**
	US-Regions				
Cola-GNN w/o *temp*	485	662	889	1144	1228
Cola-GNN w/o *loc*	499	666	891	1177	1292
Cola-GNN w/o *geo*	507	665	945	1264	1296
Cola-GNN	**480**	**636**	**855**	**1134**	**1203**
	US-States				
Cola-GNN w/o *temp*	138	169	194	250	251
Cola-GNN w/o *loc*	138	169	**202**	245	246
Cola-GNN w/o *geo*	145	188	211	262	249
Cola-GNN	**136**	**167**	**202**	**241**	**232**
PCC(↑)	2	3	5	10	15
	Japan-Prefectures				
Cola-GNN w/o *temp*	0.910	0.867	0.818	0.793	0.744
Cola-GNN w/o *loc*	0.914	0.881	0.880	0.781	0.727
Cola-GNN w/o *geo*	0.864	0.870	0.853	0.800	0.755
Cola-GNN	**0.915**	**0.901**	**0.890**	**0.813**	**0.753**
	US-Regions				
Cola-GNN w/o *temp*	0.944	0.902	0.824	0.712	0.588
Cola-GNN w/o *loc*	0.942	0.898	0.824	0.682	0.582
Cola-GNN w/o *geo*	0.943	0.901	0.806	0.606	0.574
Cola-GNN	**0.946**	**0.909**	**0.835**	**0.717**	**0.639**
	US-States				
Cola-GNN w/o *temp*	0.953	0.930	**0.908**	0.833	0.836
Cola-GNN w/o *loc*	**0.955**	0.931	0.904	**0.856**	0.855
Cola-GNN w/o *geo*	0.950	0.914	0.888	0.818	0.840
Cola-GNN	**0.955**	**0.933**	0.897	0.822	**0.859**

tune the hidden dimensions of the RNN module from {10, 20, 30}. The feature dimension of the last graph layer $F^{(l)}$ is equal to the lead time. The number of RNN and graph layers is optimized to 1 and 2, respectively. We performed early stopping according to the loss on the validation set. All the parameters are initialized with Glorot initialization [13] and trained using the Adam [17] optimizer with weight decay 5e-4, and dropout rate 0.2. The initial learning rate is searched from the set {0.001, 0.005, 0.01}, and the batch size is 32. 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) and long-term (leadtime = 5, 10, 15) settings. We ignore the case of leadtime equals to 1 because symptom monitoring data is usually delayed by at least one week. Table 3 summarizes the results of all the methods

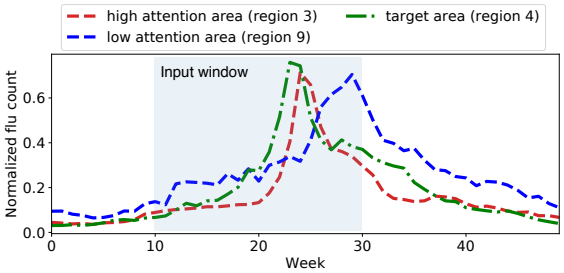
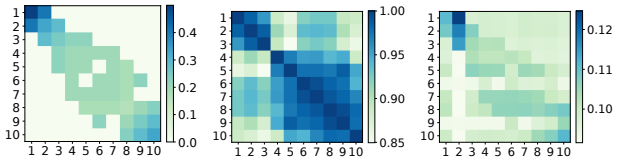


Figure 2: An example of location-aware attentions. Shaded area is the input. Attention scores are learned from the model.

in terms of RMSE and PCC. We also provide the relative performance gain of our method to the best baseline model. The large difference in RMSE values across different datasets is due to the variance of the data, i.e., the variance of the Japan-Prefectures is greater than the US-Regions and US-States datasets. When the lead time is relatively small (leadtime = 2, 3), our method achieves the most stable and optimal performance on all datasets. Most of the methods show relatively good performance, 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. Given long lead time windows (leadtime = 5, 10, 15), the proposed method achieves the best performance for most datasets. Statistical 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. RNN models only achieve good predictive performance when lead time is small, which demonstrates that long-term ILI predictions require a better design to capture spatial and temporal dependencies. CNNRNN-Res uses geographic location information and they only perform well on the US-States and Japan-Prefectures datasets, respectively. In our experiments, simple models (e.g., ARMA, RNN) achieves better performance than deep learning models in some cases. A possible reason is that deep learning-based models with high model complexity tend to overfit due to the small size of training data in the epidemic domain. The DCRNN model performs very unstably on these three data sets, especially in long-term settings. For the model LSTNet designed to capture long-term and short-term local dependency patterns, and the spatio-temporal model ST-GCN, which uses CNN and GCN to extract temporal and spatial dependencies, they do not perform well on all influenza datasets. The possible reason is that the complexity of these models is very high, leading to overfitting on the flu prediction task.

Overall, 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 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.



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

Figure 3: Comparison of geolocation matrix (3a), input correlation matrix (3b), and learned attention matrix (3c) for the US-Regions dataset.

5.2 Ablation Tests

To analyze the effect of each component in our framework, we perform the ablation tests on Japan-Prefectures and US-Regions datasets with the following settings:

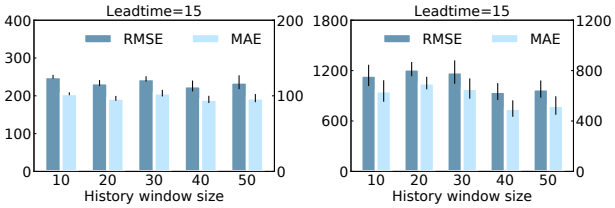
- Cola-GNN w/o *temp*: Remove the dilated temporal convolutional layer from the proposed model, and use the raw time-series input as features in graph message passing.
- Cola-GNN w/o *loc*: Remove the location-aware attention module and directly use the geographical adjacent matrix that saves distance between pairs of locations.
- Cola-GNN w/o *geo*: Remove the geographical adjacent matrix in the location-aware attention module (Eq. 4,5, and 6), and only use the general attention matrix **A** in the propagation.

The results of RMSE and PCC are shown in Table 4. We observe that in most cases, variant versions of the proposed method can achieve good performance. However, adding temporal and spatial modules does not change the short-term prediction very much. Instead, for long-term predictions, involving these two modules produces better results. The variant model that removing geographical information produces considerable performance when lead time is large. This shows that, in the absence of geolocation information, the dynamic correlation between the locations learned from our model is helpful for long term influenza prediction.

5.3 Interpretability

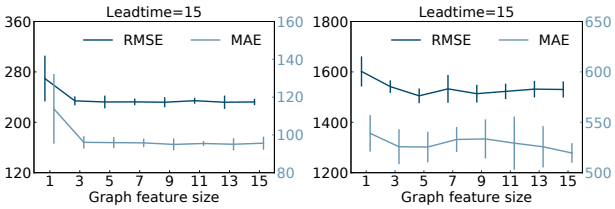
Figure 2 shows an example of the location-aware attention of *region 4* on the US-Regions dataset. When predicting the patient count for *region 4*, we visualize the normalized patient count of the highest attention area, *region 3*, as well as the region that has the lowest attention score, *region 9*. The region with a higher attention score shares more similar trends with *region 4*, while the low attention region has a visibly different pattern (the peak is 8 weeks later). The input time-series, which is the shaded light blue section in the figure, is 20 weeks. We use only a small part of the sequence of each region to predict the ILI patient count of *region 4* in a future week (e.g., 15 weeks later). In our model, spatial attentions provide indicators for future event predictions.

We also compare the geographical adjacency matrix (Figure 3a), which is calculated according to Eq. 4, the Pearson's correlation of input data (Figure 3b), and the location-aware attention matrix (Figure 3c). The attention matrix utilizes geolocation information as well as additive attentions among regions. From the attention matrix, we observe that some non-adjacent regions also receive high attention values given their similar long-term influenza trends.



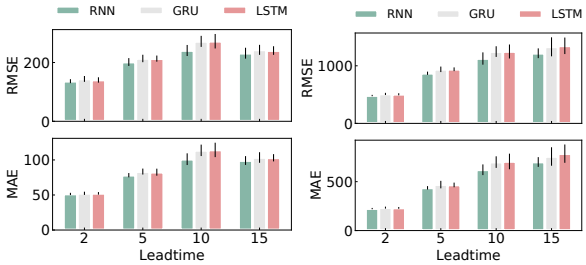
(a) US-States (b) US-Regions

Figure 4: Sensitivity analysis on window size T.



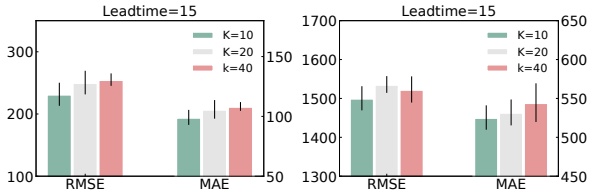
(a) US-States (b) Japan-Prefectures

Figure 5: Sensitivity analysis on graph feature size $F^{(l)}$.



(a) US-States (b) US-Regions

Figure 6: Sensitivity analysis on RNN modules.



(a) US-States (b) Japan-Prefectures

Figure 7: Sensitivity analysis on filter number K.

For instance, region 1 and 8 are not adjacent (Figure 3a), but they have a relative high correlation (Figure 3b). Note that the attention score of region 8 to region 1 is also relatively high (Figure 3c). The learned attention reveals hidden dynamics (e.g., epidemic outbreaks and peak time) among regions.

5.4 Sensitivity Analysis

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

Size of History Windows T. To test if our model is sensitive to the length of historical data, we evaluate different window sizes from 10 to 50 with step 10. The results of RMSE and MAE are shown in Figure 4. On the US-States dataset, the predictive performance in RMSE and MAE with different window sizes are quite stable. We can avoid training with very long sequences and achieve relatively comparable results. On the US-Regions dataset, our model reaches the best performance when the window size is 40.

Size of Graph Features $F^{(l)}$. The proposed model learns 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 when leadtime is 15. Figure 5 reports RMSE and MAE results on the US-States and Japan-Prefectures datasets. 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 6 shows RMSE results for leadtime = 2,5,10,15 on the 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.

Number of Filters K. In the convolutional layer, we apply 1D CNN filters to each row of $\mathbf{X}$ (i.e., the multi-location time series data) to capture the temporal dependency. We perform sensitivity analysis on different numbers of filters and the results are shown in Figure 7. In both the US-States and the Japan-Prefectures datasets, the proposed model achieves the best performance when $K = 10$. More filters tend to reduce the predictive power of the model, indicating that for limited influenza training data, the size of the model should not be too large.

5.5 Model Complexity

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 baseline methods, our model has no obvious adverse effect on training efficiency. It controls the size of model parameters well to prevent overfitting. All programs are implemented using Python 3.7.4 and PyTorch 1.0.1 with CUDA 9.2 in an Ubuntu server with an Nvidia 1080Ti GPU.

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

Methods	Parameters	Runtime(s)
GAR	21	0.01
AR	1K	0.02
VAR	48K	0.02
ARMA	1K	0.03
RNN	481	0.04
LSTM	1K	0.05
RNN+Attn	1K	0.58
DCRNN	5K	2.70
CNNRNN-Res	7K	0.04
LSTNet	12K	0.05
ST-GCN	14K	0.24
Cola-GNN	3K	0.21

5.6 Summary Statistics of Models

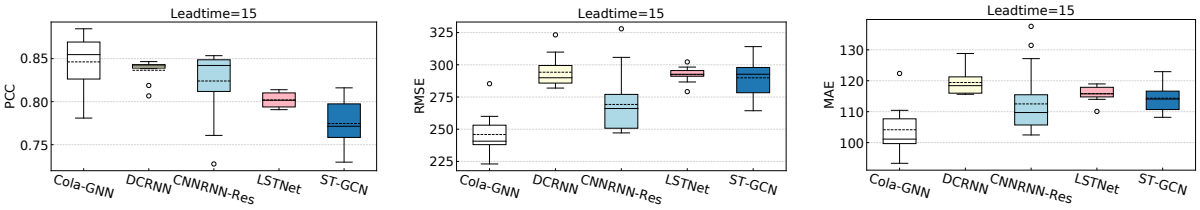
Figure 8 shows the statistical analysis of PCC, RMSE, and MAE for different deep learning models on the US-States dataset when the lead time is 15. The scores are evaluated in the test set of 10 randomized trials. The proposed model achieves stable predictive performance in terms of these three metrics.

6 CONCLUSION

In this work, we propose a graph-based deep learning framework with time series attributes for each node to study the spatio-temporal influence of long-term ILI predictions. We design a new dynamic adjacency matrix using cross-location attention scores to identify directed spatial effects. We also adopt a multi-scale dilated convolution layer on time series to capture both short and long-term patterns. We demonstrate the effectiveness of the proposed model on real-world epidemiological datasets. We also evaluated the interpretability of the dynamic graph learning method with case studies. One shortcoming of the proposed method is training flexibility. Separate models are trained for different lead time settings. In the future, we will consider iterative predictions to increase model flexibility. Another research direction is to involve more complex dependencies such as social factors, climate changes, 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 location by learning multiple locations simultaneously.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers and NVIDIA Corporation with the donation of the Titan V GPU. This work is supported in part by the US National Science Foundation under grant IIS-1948432. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.



(a) PCC (US-States) **(b) RMSE (US-States)** **(c) MAE (US-States)**

Figure 8: Statistics analysis of the prediction results from deep models.

REFERENCES

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

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. 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. 2009. EpiFast: a fast algorithm for large scale realistic epidemic simulations on distributed memory systems. In *ICS*. ACM.

[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. 2017. Combining participatory influenza surveillance with modeling and forecasting: Three alternative approaches. *JMIR public health and surveillance* 3, 4 (2017).

[5] CDC. 2020. Estimated Influenza Illnesses, Medical visits, Hospitalizations, and Deaths in the United States – 2018-2019 influenza season. https://www.cdc.gov/flu/about/burden/2018-2019.html.

[6] 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. 2014. Forecasting a moving target: Ensemble models for ILI case count predictions. In *SDM*.

[7] Jianpeng Cheng, Li Dong, and Mirella Lapata. 2016. Long Short-Term Memory-Networks for Machine Reading. In *EMNLP*. ACL, Austin, Texas.

[8] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In *EMNLP*. ACL, Doha, Qatar. https://doi.org/10.3115/v1/D14-1179

[9] GMAM Chowell, MA Miller, and C Viboud. 2008. Seasonal influenza in the United States, France, and Australia: transmission and prospects for control. *Epidemiology & Infection* 136, 6 (2008).

[10] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. 2015. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). In *ICLR*, Vol. abs/1511.07289.

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

[12] Yuyang Gao, Liang Zhao, Lingfei Wu, Yanfang Ye, Hui Xiong, and Chaowei Yang. 2019. Incomplete Label Multi-Task Deep Learning for Spatio-Temporal Event Subtype Forecasting. In *AAAI*.

[13] Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS*.

[14] Yichen Gong and Samuel Bowman. 2018. Ruminating Reader: Reasoning with Gated Multi-hop Attention. In *Proceedings of the Workshop on Machine Reading for Question Answering*. ACL, Melbourne, Australia. https://doi.org/10.18653/v1/W18-2601

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

[16] William Ogilvy Kermack and Anderson G McKendrick. 1927. A contribution to the mathematical theory of epidemics. *Proceedings of the royal society of london* 115, 772 (1927).

[17] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization, In ICLR. *ICLR* abs/1412.6980.

[18] Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. 2018. Modeling long-and short-term temporal patterns with deep neural networks. In *SIGIR*. 95–104.

[19] Xiang Li, Ling Peng, Yuan Hu, Jing Shao, and Tianhe Chi. 2016. Deep learning architecture for air quality predictions. *Environmental Science and Pollution Research* 23, 22 (2016).

[20] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. 2017. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926* (2017).

[21] Liantao Ma, Junyi Gao, Yasha Wang, Chaohe Zhang, Jiangtao Wang, Wenjie Ruan, Wen Tang, Xin Gao, and Xinyu Ma. 2020. AdaCare: Explainable Clinical Health Status Representation Learning via Scale-Adaptive Feature Extraction and Recalibration. In *AAAI*.

[22] Yue Ning, Rongrong Tao, Chandan K Reddy, Huzefa Rangwala, James C Starz, and Naren Ramakrishnan. 2018. STAPLE: Spatio-Temporal Precursor Learning for Event Forecasting. In *SDM*. SIAM, 99–107.

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

[24] Ransalu Senanayake, Simon O'Callaghan, and Fabio Ramos. 2016. Predicting spatio-temporal propagation of seasonal influenza using variational Gaussian process regression. In *AAAI*.

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

[26] Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-to-end memory networks. In *NeurIPS*.

[27] Ashish Vaswani, Noam Shazeer, et al. 2017. Attention is all you need. In *NeurIPS*. 5998–6008.

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

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

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

[31] Lijing Wang, Jiangzhuo Chen, and Madhav Marathe. 2019. DEFSI: Deep Learning Based Epidemic Forecasting with Synthetic Information. In *AAAI*, Vol. 33. https://doi.org/10.1609/aaai.v33i01.33019607

[32] Zheng Wang, Prithwish Chakraborty, et al. 2015. Dynamic poisson autoregression for influenza-like-illness case count prediction. In *KDD*. ACM.

[33] Paul J Werbos. 1990. Backpropagation through time: what it does and how to do it. *Proc. IEEE* 78, 10 (1990), 1550–1560.

[34] WHO. 2019. Immunization Financing Indicators. https://www.who.int/immunization/programmes_systems/financing/data_indicators/en/.

[35] Yuexin Wu, Yiming Yang, Hiroshi Nishiura, and Masaya Saitoh. 2018. Deep Learning for Epidemiological Predictions. In *SIGIR*. ACM, 1085–1088.

[36] Zonghan Wu, Shirui Pan, Guodong Long, Jing Jiang, and Chengqi Zhang. 2019. Graph wavenet for deep spatial-temporal graph modeling. *arXiv preprint arXiv:1906.00121* (2019).

[37] Shihao Yang, Mauricio Santillana, and Samuel C Kou. 2015. Accurate estimation of influenza epidemics using Google search data via ARGO. *Proceedings of the National Academy of Sciences* 112, 47 (2015).

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

[39] Bing Yu, Haoteng Yin, and Zhanxing Zhu. 2017. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875* (2017).

[40] Fisher Yu and Vladlen Koltun. 2015. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122* (2015).

[41] Liang Zhao, Qian Sun, Jieping Ye, Feng Chen, Chang-Tien Lu, and Naren Ramakrishnan. 2015. Multi-task learning for spatio-temporal event forecasting. In *KDD*. 1503–1512.