Deep Learning for Epidemiological Predictions

Yuexin Wu
Carnegie Mellon University
yuexinw@andrew.cmu.edu

Hiroshi Nishiura
Hokkaido University
nishuirah@med.hokudai.ac.jp

Yiming Yang
Carnegie Mellon University
yiming@cs.cmu.edu

Masaya Saitoh
The Institute of Statistical Mathematics
saitohm@ism.ac.jp

ABSTRACT
Predicting new and urgent trends in epidemiological data is an important problem for public health, and has attracted increasing attention in the data mining and machine learning communities. The temporal nature of epidemiology data and the need for real-time prediction by the system makes the problem residing in the category of time-series forecasting or prediction. While traditional autoregressive (AR) methods and Gaussian Process Regression (GPR) have been actively studied for solving this problem, deep learning techniques have not been explored in this domain. In this paper, we develop a deep learning framework, for the first time, to predict epidemiology profiles in the time-series perspective. We adopt Recurrent Neural Networks (RNNs) to capture the long-term correlation in the data and Convolutional Neural Networks (CNNs) to fuse information from data of different sources. A residual structure is also applied to prevent overfitting issues in the training process. We compared our model with the most widely used AR models on USA and Japan datasets. Our approach provides consistently better results than these baseline methods.

CCS CONCEPTS
• Mathematics of computing → Time series analysis; • Computing methodologies → Neural networks; • Applied computing → Bioinformatics;

KEYWORDS
deep learning, epidemiology prediction, time series

1 INTRODUCTION
Epidemic prediction over the world is an important problem for public health. Timely detection, tracking and forecasting of key information of epidemics such as peak intensity and outbreak time are crucial for effective health intervention. Classic work in computational epidemiology mainly focused on compartmental models where the whole population is divided into different groups (of susceptible, infective and recovered), and the transition among groups are modeled by differential equations[5]. While being intuitive and popular, such models have limited prediction power due to the rather narrow function space, lack of the ability to model individual-level information, and do not embrace new developments in recent machine learning and data mining technologies.

A recent interdisciplinary effort is to approach this problem from a time-series perspective, as the temporal nature of epidemic observations and the need for real-time alerts makes the problem residing in the scope of time-series prediction. Autoregressive (AR) models and their variants (e.g., VAR), as the representing approaches, have been widely used to capture spatio-temporal patterns [1, 6]. AR models use history data to make a (usually linear) prediction about the future, and adapt the model parameters using updated history over time. Gaussian Process Regression (GPR) [7] is another representing method, which extends the prediction power by utilizing a non-linear kernel (e.g. radial basis function) for modeling complex temporal patterns. The adaptive nature of time-series models is a major departure from classic compartment models, which fix the model parameters in the entire process. Both AR and GPR require a relative small number of parameters due to their simplicity (relying on linear combination or predefined kernels). This makes them popular in epidemiology prediction as weekly sampled epidemic statistics usually provide limited training instances. However, such simplicity also limits the expressiveness of those models. How to further enhance the prediction power for epidemic prediction with restricted training data is an open question for research.

In this paper we propose a deep learning approach1 for the epidemic prediction problem from a time-series forecasting perspective. Deep neural networks have not been studied for epidemic modeling so far, to our best knowledge. With a non-trivial adaptation of deep learning methods from other application domains to computational epidemiology, and more specifically by using adjacent graph convolution and a recurrent module, our proposed method shows significant and consistent performance improvement over other representative baseline methods on multiple real-world datasets in our evaluation. Furthermore, our ablation test shows

that we can effectively address the overfitting issue which is general in deep learning, by introducing densely-connected residual links in our networks.

2 BACKGROUND

2.1 Task Definition

Let us define the epidemic prediction problem precisely as a time series forecasting task. Denote by \( x_t \in \mathbb{R}^m \) the multi-variate epidemiology profile, whose elements are the observations from \( m \) different sources/signals at time stamp \( t \), e.g., the influenza patient counts per week \( t \) in \( m \) states of the U.S.. Further denote by \( X = [x_1, x_2, \ldots, x_T] \) the available training data in a time-span of size \( T \). The task then is to predict epidemiology profile at a future time point \( T + h \) where \( h \) is the refer to the horizon of the prediction.

2.2 Autoregressive Methods

Autoregressive (AR) models have been most popular for time series forecasting [6, 8]. The basic idea is to model the future state as a linear combination of past data points. For example, the basic order-\( p \) autoregressive model can be formalized as:

\[
\hat{x}_{t+h}^{(i)} = \sum_{p=0}^{w-1} a_p^{(i)} x_{t-p} + \varepsilon_{t+h} + c^{(i)}
\]

where the prediction for the \( i \)-th signal of epidemiology profile \( x \) is the weighted sum of the data points in past window of size \( w \), and \( \varepsilon_{t+h} \) is a small random noise which is used to explain the deviation between the linear sum and the true value; \( c \) is the intercept term. When training data are limited and the signals from different sources exhibit similar patterns, we may train the system with only one set of \( \{a_p\} \) and \( c \) for all the sources; such a model is called Global Autoregression (GAR) in the literature.

A potential shortcoming of AR models is that the signal sources are treated independently from each other during the training process, which would be too simplistic. A direct extension of AR is to model cross-signal dependencies via Vector Autoregression (VAR).

It predicts the future as:

\[
\hat{x}_{t+h} = \sum_{p=0}^{w-1} A_p x_{t-p} + \varepsilon_{t+h} + c
\]

where the signal-wise \( a_p \) in AR is replaced with matrix \( A_p \) to capture the correlation information. Notably, the number of parameters for \( \{A_p\} \) is \( O(m^2w) \) which is far larger than that of AR (\( O(mw) \)). Thus VAR models are more expressive than AR models in general, with a higher chance of overfitting as potential trade-off.

2.3 Gaussian Process Regression

Both AR and VAR methods rely on the linear combination of past signals in making predictions, which may not be sufficiently expressive for some complicated real-world scenarios. A common approach to go beyond is to apply kernel tricks in a Gaussian Process Regression (GPR) [7]. Specifically, GPR assumes that the future predicting profiles altogether are sampled from a Gaussian distribution, where the variance is specified by its past history. For the clarity of explanation, consider a dataset with one-dimension signals.

\[ y_i = \sigma(G(x_i)) \]

where \( h_t \) is the transformed feature map and \( \Phi_G \) is the parameter matrix. \( \Phi_G \)'s entries can only be non-zero if and only if the corresponding entry in \( G \) is non-zero. \( \sigma \) is an activation function (e.g. sigmoid) making the transformation nonlinear.

Comparing to the image CNN, the grid filters are replaced by the adjacent parameter graph, enlarging the parameter number from \( O(1) \) to \( O(km) \) where \( k \) is the number of nearest neighbors.
We prepared three real-world datasets for experiments.

3.2 RNN Module
We employ an recurrent neural network (RNN) module to capture the temporal dependencies in the data. Specifically, we utilize an Gated Recurrent Unit (GRU) [2] in our framework. The input data are passed through a gate, which is then used to compute the new state in the memory cell of GRU given the old value. This process is repeated carried out along with new inputs. Compared to the traditional Long-Short Time Machine (LSTM) where there are 3 gates, an GRU has fewer parameters (2 gates) to be trained and thus is more suitable in the data-deficient case. Moreover, as each gate links to the hidden memory, by effectively constraining its size to a small number q, the parameter number can be limited to $O(qm)$ which is still of a similar size as AR.

3.3 Residual Module
For deep neural networks, it is well-known that overfitting issues arise when the amount of data does not scale accordingly with the number of parameters. Therefore, we utilize the residual links to let the training process bypass some of the intermediate layers, which can effectively mitigate the overfitting issue. Instead of using the standard residual links that each layer may only connect to its neighbors within 2-4 layers [3], we use a similar structure where the final layer “densely” links to nearly all previous layers [4]. The benefits of such design are two-fold: such design alleviates the gradient vanishing phenomenon during training which stabilizes the process; also the links may possibly introduce highly relevant long-jump data information to the final output (e.g. the annual epidemiology patterns), thus giving out a more accurate predictor. Similarly, to regularize the parameter number, we only introduce one scaling factor for each residual link, contributing to at most $O(w)$ parameters, which is smaller than AR.

4 EMPIRICAL EVALUATION

4.1 Datasets
We prepared three real-world datasets for experiments.

- **Japan-Prefectures** This dataset contains the weekly influenza-like-illness statistics (patient counts) from 47 prefectures in Japan, ranging from 2009 to 2015.
- **US-Regions** This dataset, collected from the CDC FluView website, contains the weekly influenza activity levels (from 1 to 10) for all the states in U.S. from 2009 to 2016. After removing the states with missing data we kept 29 states remaining in this dataset.
- **US-HHS** This dataset is the ILINet portion of the US-HHS dataset, consisting of the weekly influenza activity levels for the 10 districts of the mainland U.S. for the period of 2009 to 2016 measured using the weighted ILI metric.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Size</th>
<th>Min</th>
<th>Max</th>
<th>Mean</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Japan-Prefectures</td>
<td>47×312</td>
<td>0</td>
<td>18939</td>
<td>503.54</td>
<td>1368.31</td>
</tr>
<tr>
<td>US-Regions</td>
<td>29×451</td>
<td>1</td>
<td>10</td>
<td>2.17</td>
<td>2.39</td>
</tr>
<tr>
<td>US-HHS</td>
<td>10×364</td>
<td>0.05</td>
<td>10.62</td>
<td>1.52</td>
<td>1.17</td>
</tr>
</tbody>
</table>

Table 1: Dataset statistics include min, max, mean and standard deviation (SD) of patient counts or activity levels; dataset size means # of regions multiplied by # of weeks.

4.2 Experiment Setup
For comparative evaluation we include GAR, AR and VAR as representative baselines of the autoregressive family, GPR as a representative of non-linear models. For all datasets, we split them into three sets: training (60%), validation (20%) and test (20%) in chronological order. We tune the window size for all methods from the set {2, 8, 32, 64, 128}. To make GAR, AR and VAR more robust, we adopt a L2-regularization term during training, where its coefficient is searched from the set {0.01, 0.1, 1}. For GPR, we use the radial basis function (RBF) as its kernel function. The kernel bandwidth hyper-parameter for RBF is chosen from {2^{-5}, 2^{-4}, \ldots, 2^{2}}. For our method (CNNRNN-Res), we tune the hidden dimension for GRU from {5, 10, 20, 40}. The number of residual links are searched from set {4, 8, 16}. We construct the adjacency matrix G based on the real-world location of signals.

We adopt two evaluation metrics for comparison: Root Mean Squared Error (RMSE) and Pearson’s Correlation Coefficient (CORR). Denote the prediction and true values to be $\{\hat{y}_i, \ldots, \hat{y}_n\}$ and $\{y_1, \ldots, y_n\}$ respectively. The calculation for these metrics are defined as:

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

\[
\text{CORR} = \frac{\sum_{i=1}^{n} (\hat{y}_i - \text{mean}(\hat{y}))(y_i - \text{mean}(y))}{\sqrt{\sum_{i=1}^{n} (\hat{y}_i - \text{mean}(\hat{y}))^2 \sum_{i=1}^{n} (y_i - \text{mean}(y))^2}}
\]

4.3 Results
Table 2 summarizes the results of all the methods, where the proposed CNNRNN-Res has the dominating performance. Notice that CNNRNN-Res has similar number of model parameters as AR does, but a much better performance; VAR has the largest number of model parameters but the worst results on two out of the three
datasets. This suggests the importance of controlling the model complexity (the effective number of model parameters) for data insufficient problems. Also notice that on the Japan-Prefectures dataset, which has the largest standard deviation and hence a hard dataset, our method has the strongest results in terms of relative improvements over other methods on average. This suggest that our method can successfully capture nonlinear features with deep learning, outperforming non-linear GP.

### 4.4 Ablation Tests

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

- **RNN (GRU):** Only keeping the RNN layer but removing the CNN layer and the residual links among the RNN modules;
- **CNNRNN:** Keeping both the CNN and RNN layers but removing the residual links among the RNN modules;
- **CNNRNN-Res:** this is the full model.

The results are measured in RMSE in Fig. 3. It is interesting to see that CNNRNN does not consistently improve the performance of using RNN only. Besides, adding both the CNN layer and the residual modules improve the robustness. Notice that all the datasets are of a relative small size (hundreds of training samples) which are of a relative small size (hundreds of training samples) which. Adding the residual links among the RNN modules would hurt the performance due to overfitting. The CNNRNN-Res (the full model) offers a remedy for this issue.

5 CONCLUSION

In this paper, we presented the first study on deep learning to the epidemic prediction problem from a time-series prediction perspective. Our method combines the strengths of CNN, RNN and residual links for enhanced model expressiveness and robust prediction. Our experimental results showed the consistent performance improvements by the proposed approach over representative linear and non-linear methods on multiple real-world datasets.

ACKNOWLEDGMENTS

We thank the reviewers for their helpful comments. This work is supported in part by the National Science Foundation (NSF) under grant IIS-1546329 and Japan Science and Technology Agency CREST program.

REFERENCES


