Deep Learning for Epidemiological Predictions

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

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

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

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

ACM Reference Format:
Yuexin Wu, Yiming Yang, Hiroshi Nishiura, and Masaya Saitoh. 2018. Deep Learning for Epidemiological Predictions. In *SIGIR '18: The 41st International ACM SIGIR Conference on Research and Development in Information Retrieval, July 8–12, 2018, Ann Arbor, MI, USA.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3209978.3210077

SIGIR '18, July 8–12, 2018, Ann Arbor, MI, USA

ACM ISBN 978-1-4503-5657-2/18/07…$15.00
https://doi.org/10.1145/3209978.3210077

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 approach[1] 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

[1] Code available at https://github.com/CrickWu/DL4Epi.

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 $\boldsymbol{x}_t \in \mathbb{R}^m$ 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 $X = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_T]$ 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 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:

$$\tilde{x}^{(i)}_{t+h} = \sum_{p=0}^{w-1} \alpha^{(i)}_p x^{(i)}_{t-p} + \varepsilon_{t+h} + c^{(i)} \tag{1}$$

where the prediction for the i-th signal of epidemiology profile $\boldsymbol{x}$ is the weighted sum of the data points in past *window* of size w, and ε_{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 $\{\alpha_p\}$ 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 profile as:

$$\tilde{\boldsymbol{x}}_{t+h} = \sum_{p=0}^{w-1} A_p \boldsymbol{x}_{t-p} + \boldsymbol{\varepsilon}_{t+h} + \boldsymbol{c} \tag{2}$$

where the signal-wise α_p in AR is replaced with matrix A_p to capture the correlation information. Notably, the number of parameters for $\{A_p\}$ is $O(m^2 w)$ 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.

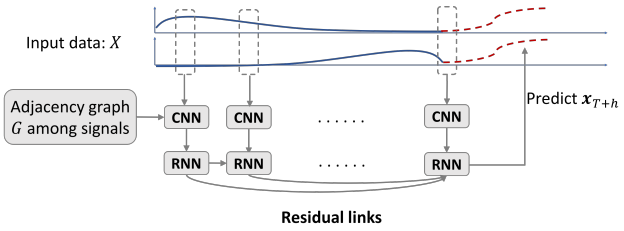

Figure 1: The proposed deep learning framework where the top portion is the temporal sequence of epidemiology profiles (input vectors), the middle portion consists of the CNN modules, and the bottom portion consists of the RNN modules with residual links in-between.

Suppose the future profiles are $\{y_1, \ldots, y_n\}$ and their past histories are $\{z_1, \ldots, z_n\}$ correspondingly where $y_i \in \mathbb{R}$ and $z_i \in \mathbb{R}^T$. Then,

$$\begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{pmatrix} K(z_1, z_1) & \ldots & K(z_1, z_n) \\ & \ddots & \\ K(z_n, z_1) & \ldots & K(z_n, z_n) \end{pmatrix}\right) \tag{3}$$

where K is a kernel function (e.g. radial basis function) computing the covariance of two past histories. Non-linearity, thus, appears along with this function as long as the kernel is beyond dot product. Such non-linear design would yield more accurate predictions than linear models when the dependency patterns are complex.

3 PROPOSED METHOD

Our model framework is shown in Figure 1. The overall structure is composed of 3 parts: a CNN for capturing correlation between signals, a RNN for linking up the dependencies in the temporal dimension and the residual links for fast training and overfitting prevention. We carefully restrain the parameter space, making the total model have a similar size as AR.

3.1 CNN Module

We use a convolutional Neural Network (CNN) module to fuse the information across different sources. In the deep learning literature, CNN modules are known to be small in the number of parameters and effective in capturing local dependency patterns. However, direct application of CNNs in our framework would not work well as conventional CNNs are designed for a grid structure of neighborhood in data (such as in images), but in our data the grid-structure assumption does not hold. In order to preserve the ability to model local feature, we propose a new structure. Precisely, we utilize a given adjacent nearest neighbor matrix G to regularize the number of parameters while mimicking the convolution behavior. Let

$$\boldsymbol{h}_t = \sigma(\Phi_G \boldsymbol{x}_t) \tag{4}$$

where $\boldsymbol{h}_t$ is the transformed feature map and Φ_G is the parameter matrix. Φ_G's entries can only be non-zero if and only if the corresponding entry in G is non-zero. σ 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

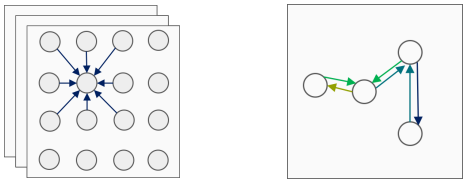

Figure 2: Left: Image CNN filter, a uniform grid filter is applied on each node; the filter is computed over each node one-by-one. Right: Adjacency CNN filter, a one-time node-specific filter defined on the whole irregular graph is applied; the filter is computed over all nodes at once.

kept in G. This number is only of comparable size of AR which is well within the acceptable range, and we could gain more flexible non-linear representing power by stacking multiple CNNs hierarchically. Besides, as adjacent convolution gets more node-specific parameters than grid convolution, it is possible to use just one filter (i.e. one Φ_G) to represent complex patterns which can only be captured by multiple filters in the grid form.

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 repeatedly 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[2], contains the weekly influenza activity levels (from 1 to 10) for all the states in U.S. from 2009 to 2016After 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 [3], 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[4].

Dataset	Size	Min	Max	Mean	SD
Japan-Prefectures	47×312	0	18939	503.54	1368.31
US-Regions	29×451	1	10	2.17	2.39
US-HHS	10×364	0.05	10.62	1.52	1.17

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 $\{\hat{y}_1, \ldots, \hat{y}_n\}$ and $\{y_1, \ldots, y_n\}$ respectively. The calculation for these metrics are defined as:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_i (\hat{y}_i - y_i)^2} \tag{5}$$

$$\text{CORR} = \frac{\sum_i (\hat{y}_i - \text{mean}(\hat{y}))(y_i - \text{mean}(y))}{\sqrt{\sum_i (\hat{y}_i - \text{mean}(\hat{y}))^2}\sqrt{\sum_i (y_i - \text{mean}(y))^2}} \tag{6}$$

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

[2] https://gis.cdc.gov/grasp/fluview/main.html
[3] https://gis.cdc.gov/grasp/fluview/fluportaldashboard.html
[4] https://www.cdc.gov/flu/pdf/weekly/overview.pdf

		Japan-Prefectures				US-Regions				US-HHS			
		Horizon				Horizon				Horizon			
Methods	Metrics	1	2	4	8	1	2	4	8	1	2	4	8
GAR (3)	RMSE	584	786	932	949	**1.2883**	1.7513	2.1967	**2.2538**	0.2596	0.3798	0.5217	0.603
	CORR	0.9127	0.8393	0.7655	0.7582	0.7917	0.6542	0.4692	**0.5**	0.9422	0.8813	0.7722	0.729
AR (2)	RMSE	652	839	1061	1061	1.3533	1.7685	2.3414	2.4983	0.2597	**0.3667**	0.472	0.5816
	CORR	0.8725	0.7426	0.5779	0.5861	0.7655	0.6157	0.3311	0.3539	0.9438	**0.892**	0.8226	0.7277
VAR (0)	RMSE	627	754	1014	1007	1.6158	1.9144	2.3455	2.4417	0.3	0.4134	0.5039	0.5712
	CORR	0.9212	0.8715	0.6538	0.6721	0.7461	0.6433	0.4528	0.3136	0.9318	0.868	0.8072	0.7441
GP (1)	RMSE	573	676	857	1022	1.3599	1.7279	2.2834	2.4084	0.2648	0.3736	0.4659	0.5719
	CORR	0.9423	0.9043	0.7714	0.6237	0.7614	0.6312	0.3516	0.3465	0.9396	0.8921	**0.8536**	0.8096
CNNRNN-Res (18)	RMSE	**500**	**561**	**691**	**644**	1.3147	**1.6783**	**2.1613**	2.3465	**0.259**	0.3717	**0.4451**	**0.4638**
	CORR	**0.9461**	**0.9254**	**0.9095**	**0.9236**	**0.8033**	**0.6942**	**0.5564**	0.4298	**0.9466**	0.8919	0.8509	**0.8538**

Table 2: Results summary. Bold face indicates the best result of each column in a particular metric and the total number of bold-faced results of each method is listed after the method name within parentheses.

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 follow 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 means that adding more parameters (the consequence of adding the CNN modules) would hurt the performance due to overfitting. The CNNRNN-Res (the full model) offers a remedy for this issue.

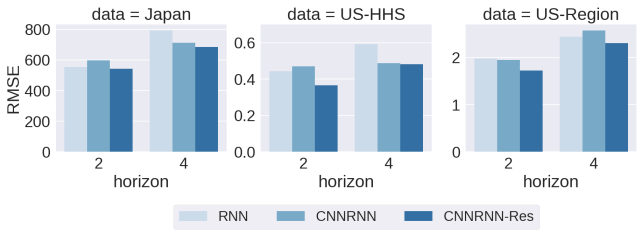

Figure 3: Ablation test results in RMSE – lower scores mean better performance.

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

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

[2] 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. *arXiv preprint arXiv:1406.1078* (2014).

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[4] Gao Huang, Zhuang Liu, Kilian Q Weinberger, and Laurens van der Maaten. 2017. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, Vol. 1. 3.

[5] William O Kermack and Anderson G McKendrick. 1927. A contribution to the mathematical theory of epidemics. In *Proceedings of the Royal Society of London A: mathematical, physical and engineering sciences*, Vol. 115. The Royal Society, 700–721.

[6] Daniela Perrotta, Michele Tizzoni, and Daniela Paolotti. 2017. Using Participatory Web-based Surveillance Data to Improve Seasonal Influenza Forecasting in Italy. In *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 303–310.

[7] Ransalu Senanayake, O'Callaghan Simon Timothy, and Fabio Ramos. 2016. Predicting Spatio-Temporal Propagation of Seasonal Influenza Using Variational Gaussian Process Regression.. In *AAAI*. 3901–3907.

[8] Zheng Wang, Prithwish Chakraborty, Sumiko R Mekaru, John S Brownstein, Jieping Ye, and Naren Ramakrishnan. 2015. 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*. ACM, 1285–1294.