Spatio-Temporal Machine Learning Algorithms for Influenza Prediction

Amy Paul
Department of Computer Science
University of Arizona
Tucson, AZ 85721
amypaul@email.arizona.edu

Connie Sun
Department of Computer Science
University of Arizona
Tucson, AZ 85721
conniesun@email.arizona.edu

Abstract

The recent rise in infectious diseases has motivated the need for accurate long-term prediction of the duration and intensity of pandemics. Although many time-series models have been developed for disease prediction, models that also consider spatial relationships of data are relatively new. We explored the utility of graph neural networks (GNNs) as a spatio-temporal machine learning model for long-term United States influenza prediction. We compared the results of the GNN to two baseline temporal models, Autoregressive Integrated Moving Average (ARIMA) and Random Forest (RF). RF is one of the most commonly used ML models for time-series prediction, and ARIMA was specifically developed as a statistical model for time-series forecasting. The task for all models was to predict the total number of US influenza cases for a full flu season. After training on historical data, we tested these models on the 2019 flu season; the results of our experiment indicate that the GNN outperforms the RF for this learning task, but ARIMA has the best prediction. Finally, we present a forecast of flu data for the 2021-2022 flu season using our trained models.

1 Introduction

The continual global impact of infectious diseases such as influenza, ebola, and Covid-19 drives the need for accurate long-term forecasting models. Predicting epidemic duration, peak intensity, and time of peak can inform public health decisions to prepare health services and minimize future spread. In particular, influenza (flu) is a seasonal disease endemic in the United States. Flu season occurs in the fall and winter, typically peaking in February [1]. Between 2010 and 2020, the CDC estimates that the flu spreads to about 9-41 million people annually, with 12,000-52,000 of these cases resulting in death [2].

Many machine learning (ML) approaches have been used to model the temporal relationships of infectious disease data [3; 4; 5]. Each of the machine learning models described and implemented in this paper were chosen for their perceived ability to usefully model and predict influenza data. RF and SARIMA models are commonly used to model data with temporal trends, as is the case with flu data [5; 6]. Since influenza is particularly not just time-series, but seasonal, using a SARIMA model was preferred over a regular ARIMA model since it would use that seasonality in its modeling. After running into time constraints and hyperparameter inefficacy with SARIMA, we elected to attempt to see just how useful a standard ARIMA model could be without the seasonal component.

Although many temporal ML models have been developed for time-series data, as is the case with RF, ARIMA, and SARIMA, spatio-temporal models are relatively new. Because infectious diseases spread through contact, we hypothesized that a model incorporating spatial relationships would be a more accurate predictor than a purely temporal model due to increased information about underlying

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

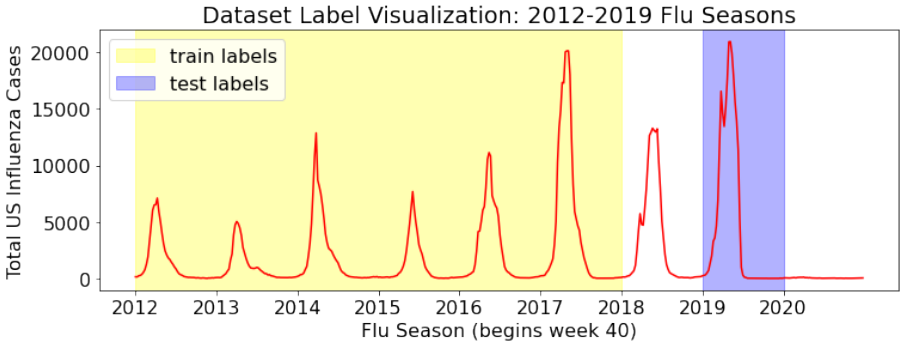

Figure 1: Visualization of the time-series labels used for training and testing.

features of the data. We explored the graph neural network (GNN) algorithm, a type of neural network that operates on graph-structured data, to leverage both spatial and temporal dependencies in United States influenza data. We separated the United States into regions (states) based on CDC data and created a graph representing the entire US. We compared this spatio-temporal model against two common baseline time-series machine learning models: random forest (RF) and autoregressive integrated moving average (ARIMA) (as well as seasonal autoregressive integrated moving average (SARIMA), an offshoot of ARIMA).

We tested our models on the 2019 and 2021 flu seasons and present a comparison of their performance and discussion of the limitations/potential pitfalls of our methods. All of our code for this project, including models, results, and data, can be found at: `https://github.com/alpshad/cs580_project`.

2 Data and Methods

In this section, we describe data processing and the machine learning architectures we used.

2.1 Data Preparation

Influenza data was collected for the United States from the Center for Disease Control FluView site for the 2010-2020 flu seasons [7]. The data contains reported case counts for each week of the year. Flu season begins on week 40 of the year and ends on week 39 of the next year (for example, the 2019 flu season begins on week 39 of 2019 and ends on week 40 of 2020). In the FluView data set, the case counts are divided into 54 U.S. regions: the 50 states, the District of Columbia, the Virgin Islands, Puerto Rico, and New York City.

The original CDC data set separated the total number of influenza cases by type and subtype (e.g., A and B), and data prior to 2015 had a different format from the later data. To preprocess the CDC data set for our purposes, we summed the total number of reported flu cases for each region for each week, ignoring subtype. For our temporal features, we used a window size of two full flu seasons (52 weeks each) for all 54 locations, for a total of 5616 features (52 * 2 * 54). The label was the number of total U.S. cases (summed across all regions) for each week of the next flu season, a vector of length 52. Both features and labels were standardized using the mean and standard deviation of the train set.

Our training set contained each window size of two years from the 2010 to 2016 flu seasons (totaling six training examples) with labels of total US cases from 2012 to 2017, respectively. We tested using 2017-2018 data to predict the 2019 flu season. A visualization of the time-series labels that we used for this task is shown in Figure 1. For the purely temporal models, feature data was sorted first by week and then by location as a one-dimensional vector. For the spatio-temporal GNN, feature data was sorted by week for each location (graph node). The label data was sorted by week so that the final prediction of each machine learning model was a time series forecast of the next year's flu season. Additionally, after training the RF and GNN, we predicted the 2021 US flu season using the labels for 2019-2020.

2.2 Baseline Models

2.2.1 Random Forest

Random Forest is a machine learning model that involves the construction of many decision trees to find the best representation of the data. RF is an ensemble ML algorithm, meaning it combines the predictions of several smaller models to create a final prediction. In a regression task like the one we performed, the average of each tree is used to predict the total number of cases.

We implemented scikit-learn's RandomForestRegressor as a baseline machine learning model trained on temporal data. We conducted a randomized hyperparameter search on the training data to tune the RF. The parameters that we searched were the number of estimators (decision trees) in the ensemble (n_estimators), minimum number of samples needed to split an inner node (min_samples_split), minimum number of samples required at a leaf node (min_samples_leaf), the number of features to consider when looking for the best split (max_features), the maximum depth of the tree (max_depth), and whether to use bootstrap samples or the whole data set when building the tree (bootstrap). We then used the hyperparameter setting that performed the best on the training data as our chosen model for the 2019 and 2021 tests.

2.2.2 Seasonal AutoRegressive Integrated Moving Average

Seasonal AutoRegressive Integrated Moving Average (SARIMA) models use time-series data trends to make predictions. They do this by taking previous data points, or lagged data, and differencing them, while taking into account how data changes in a seasonal fashion.

Initially, we wanted to use a SARIMA model as opposed to a regular ARIMA model. The initial assumption we made was that since influenza is seasonal, using a SARIMA model would result in better predictions since it takes seasonality into account. To make ours, we implemented statsmodels' SARIMA version, and conducted a grid hyperparameter search on the training data to optimize the model. The parameters that we searched were the trend autoregressive order, the trend difference order, and the trend moving average order, along with four seasonal hyperparameters: seasonal autoregressive order, seasonal difference order, seasonal moving average order, and the number of time steps for a single seasonal period.

2.3 AutoRegressive Integrated Moving Average

AutoRegressive Integrated Moving Average (ARIMA) models are the same as SARIMA but without the added seasonal component. They still use time-series data trends to make predictions, use lagging data points, and difference the data, but seasonal changes are not taken into account.

Issues with the SARIMA model results led us to implement and train a regular ARIMA model as well. We implemented statsmodels' ARIMA model, and just as with SARIMA grid searched the hyperparameters on the training data to optimize the model. The parameters that we searched were the number of lag observations, the number of times that the raw observations are differenced (degree of differencing) and the size of the moving average window (order of moving average).

2.4 Graph Neural Network

Graph neural networks (GNNs) are a type of neural network that operate on graph-structured data [8]. Unlike a typical neural network, where nodes are organized into layers that pass information forward and backward, a GNN has nodes organized into a graph structure from computer science. A graph consists of a set of vertices (nodes) connected to each other by edges; information is passed to neighboring nodes through these edges. This ML model captures dependencies between nodes and as a result allows for spatial relationships to be considered when making predictions.

The GNN we used was a modification of STAN, a spatio-temporal attention network [3]. The original STAN model is a hybrid machine learning and epidemiological (mathematical) model. The mathematical model used in STAN is susceptible-infectious-recovered (SIR), a basic compartmental model that predicts the long-term progression of an infectious disease in a human population. We removed the epidemiological aspect of STAN to utilize it as a pure ML algorithm. Note that this modification should be considered when evaluating the predictive performance of STAN, as the

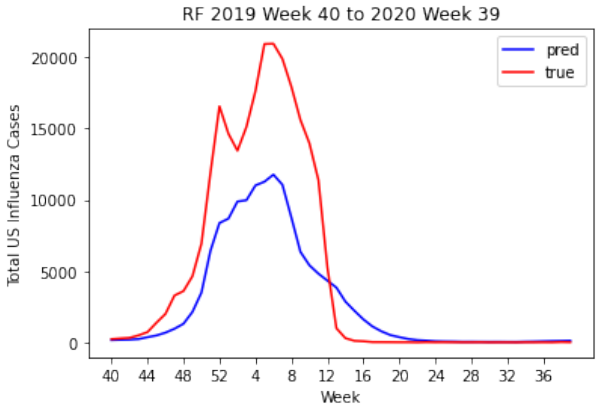

Figure 2: Random Forest test results for the 2019 flu season.

authors originally intended long-term prediction of the model to be informed by mathematical modeling.

STAN uses a graph attention network (GAT), which is a form of GNN that learns attention scores for each edge (edge weights). In the original STAN architecture, input data is fed through two GAT layers and then a gated recurrent unit (GRU) to create an embedding for the entire graph. (GRUs are commonly used for sequence learning tasks, such as speech recognition and time-series forecasting.) The output of the GRU is fed through a simple linear neural network to make the final prediction of total US cases.

Our graph neural network had 54 nodes, one for each location in the CDC data set. Each node received temporal features for the window length of two full flu seasons (104 features per node). Additionally, we included edges between every node to model full connectivity within the United States, for a total of 2916 edges (54 * 54).

3 Results

3.1 Random Forest

The random forest was able to correctly predict the time of the 2019 flu season peak, as well as the general pattern of decline (Figure 2). However, it under-predicted the severity of the peak and performed the worst out of the three models that we tested. The random forest was very efficient to train, taking approximately 3.5 seconds on a standard laptop.

The RF model was difficult to tune due to a very small number of training samples, so the randomized search of hyperparameter space was not very informative. However, we still used the best architecture from the search, with 300 individual estimators and no bootstrapping. Hyperparameter tuning for the random forest model is computationally intensive; we did not conduct any hyperparameter tuning for the GNN, and it still outperformed the tuned RF. Further hyperparameter tuning beyond what we were able to accomplish would likely improve the prediction results of this ML algorithm. However, we did not have the time to conduct a more exhaustive search.

3.2 SARIMA

The SARIMA model was extraordinarily inefficient and quite inaccurate, requiring several hours to run and not giving good results. Likely, the issues with it stemmed from incorrect hyperparameters or insufficient data. Due to how long it took to run, we ran out of time to continue adjusting it, and had technical issues graphing the data. Figure 3 shows the attempted result of graphing SARIMA models with three different sets of hyperparameters.

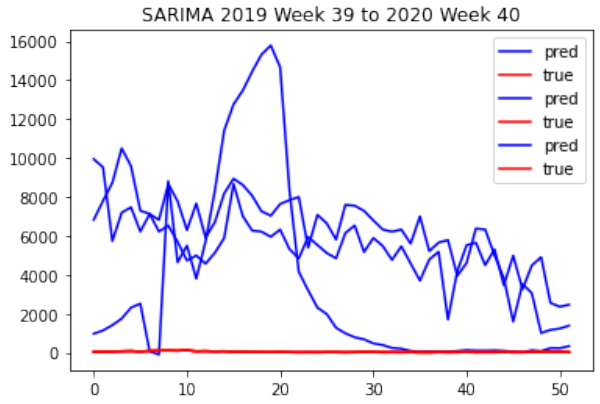

Figure 3: SARIMA test results for the 2019 flu season.

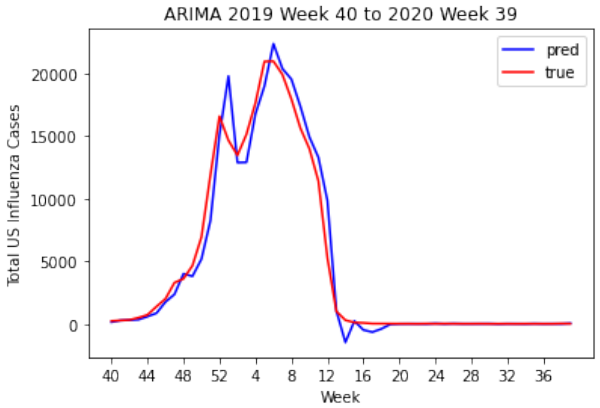

Figure 4: ARIMA test results for the 2019 flu season.

3.3 ARIMA

The ARIMA model performed the best out of all the models. It was computationally efficient, taking approximately 10 seconds to train on a standard laptop. The model did an excellent job of predicting the peak and decline of the 2019 flu season overall and was the most accurate of all the models, though it did overpredict slightly on the peaks and underpredict on the bottom of the decline (Figure 4).

Tuning the model's hyperparameters was computationally inefficient due to grid search; however, we were fortunately able to find optimal ones quickly with an order of (5, 1, 0).

3.4 Graph Neural Network

The graph neural network outperformed the tuned random forest. In terms of performance efficiency, the GNN took approximately 20.5 seconds to train on a standard laptop GPU. The GNN correctly predicted the time of peak and decline in the 2019 flu season, though it somewhat underpredicted the intensity of the peak. Additionally, the GNN prediction appears less "smooth" than the other models, especially when there are few cases (i.e., weeks 16-39 of 2020). This might be because the model learns noise in the training data.

3.5 2019 Comparison

Table 1 summarizes the mean squared error results of each model on the test set (de-normalized). The GNN showed a 53% reduction in MSE compared to the RF, while ARIMA had an 89% reduction in comparison to the RF and a 77% reduction compared to the GNN. Figure 6 compares the predictions

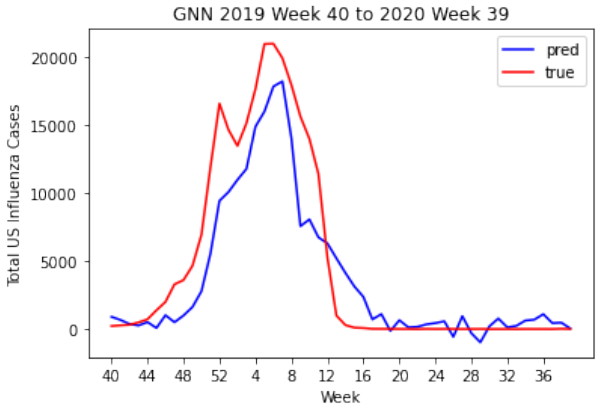

Figure 5: GNN test results for the 2019 flu season.

Table 1: Mean Squared Error Results

Model	MSE
RF	16859243
ARIMA	1814207
GNN	7918473

of the three viable models (GNN, RF, and ARIMA) in one graph. ARIMA closely follows the true number of total cases for the entire flu season. The GNN and RF are similar until the beginning of 2020, when the GNN more accurately predicts the severity of the season's peak, whereas the RF suffers from underprediction. However, all models predict the peak to occur at similar times.

3.6 2021 Forecasting

After training our models, we input the 2019 and 2020 flu seasons as labels to predict the 2021-2022 flu season, which is currently ongoing and thus does not have associated data. The RF and GNN models did a good job with this, and predict a spike in influenza cases at the end of December / beginning of January before tapering down until the end of the season 7. These trends are similar to the ground truth labels of previous years (Figure 1).

We ran into problems with using the ARIMA model to predict the 2021-2022 flu season due to its lack of seasonality. The 2019-2020 flu season data the model was trained on showed a downward trend before flattening out, as is the case with most flu seasons. Because ARIMA does not care about

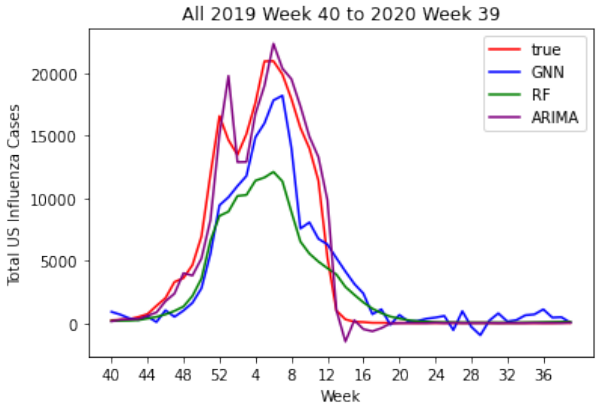

Figure 6: All ML model predictions for the 2019 flu season.

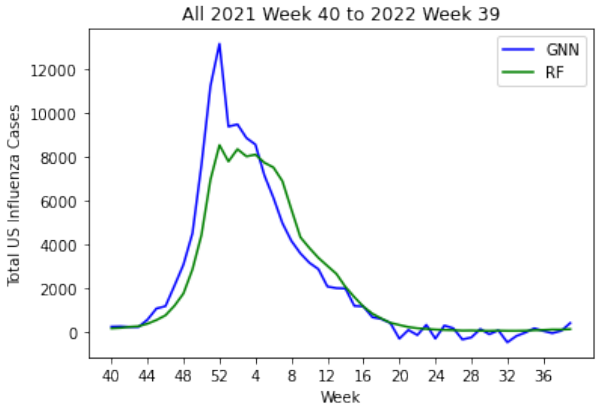

Figure 7: RF and GNN model predictions for the 2021 flu season.

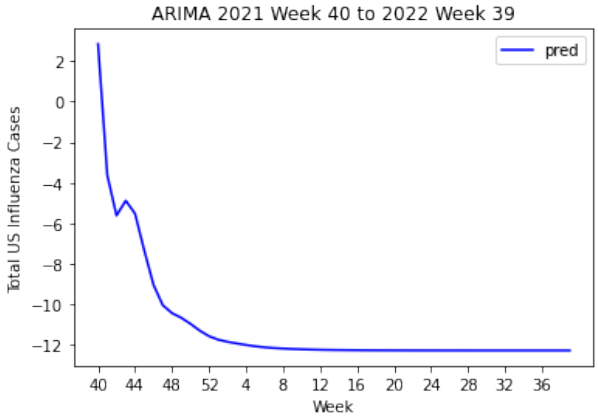

Figure 8: ARIMA model predictions for the 2021 flu season.

seasonality, only the most recent trends in data, the resulting predictions were unusable, as shown in Figure 8. This was the problem we had intended to solve by using a SARIMA model instead, but technical difficulties prevented us from doing so.

4 Discussion and Conclusion

The results of our application study reinforce the different capabilities, advantages, and disadvantages of major machine learning methods. The random forest algorithm requires significant hyperparameter tuning for this learning task, and even after a randomized grid search, the performance was significantly worse than the other models. The ARIMA model was developed specifically for time series data and was able to accurately predict the number of cases for the entire 2019 flu season. However, this model is limited because it takes as input an entire series of time instead of separate training samples like the other models. In contrast, the RF and GNN models would be able to predict historical trends in data if given previous years' data as labels. The graph neural network performed well in this influenza learning task; however, it is unclear whether the results of our experiment support our hypothesis that spatially-informed machine learning models will perform better for spatially-dependent data. Further tests would need to be conducted to determine if separating influenza data by region improves the predictive power of machine learning models.

Our experiment was most severely limited by lack of training samples. We could not find consistent flu data for years prior to 2010, and the data for the 2020 flu season was impacted by the Covid-19 pandemic, so we were limited to only ten full typical flu seasons of data. Because we wanted to predict long-term patterns in the data, our training set contained only six examples. As more flu data

becomes available in the following years, these models will have access to more training data, so they will likely generalize better to the pattern of flu seasons in the United States. We will also be able to assess the performance of our trained models on the 2021 forecasting task.

In our literature review, we did not find previous research that trained ML models to predict future total US influenza cases for a full season. Our work supports the use of ML models for studying infectious disease; in particular, our experiments suggest that GNN models have strong predictive power for this type of data and confer benefits that other forecasting models do not. Future work experimenting with hyperparameter settings, graph construction, and hybrid models may increase predictive power. We hope that our application study will inform public health decisions to prepare for flu season in the US. Additionally, this work supports the use of GNNs to predict the progression of other infectious diseases that have real-world spatial and temporal relationships.

References

[1] CDC, "Flu season," 2021.

[2] CDC, "Disease burden of flu," 2021.

[3] J. Gao, R. Sharma, C. Qian, L. M. Glass, J. Spaeder, J. Romberg, J. Sun, and C. Xiao, "STAN: spatio-temporal attention network for pandemic prediction using real-world evidence," *Journal of the American Medical Informatics Association*, vol. 28, pp. 733–743, 01 2021.

[4] D. Benvenuto, M. Giovanetti, L. Vassallo, S. Angeletti, and M. Ciccozzi, "Application of the arima model on the covid-2019 epidemic dataset," *Data in Brief*, vol. 29, p. 105340, 2020.

[5] S. Uddin, A. Khan, M. E. Hossain, and M. A. Moni, "Comparing different supervised machine learning algorithms for disease prediction," *BMC Medical Informatics and Decision Making*, vol. 19, p. 281, Dec 2019.

[6] Z. He and H. Tao, "Epidemiology and ARIMA model of positive-rate of influenza viruses among children in wuhan, china: A nine-year retrospective study," *International Journal of Infectious Diseases*, vol. 74, pp. 61–70, Sept. 2018.

[7] CDC, "National, regional, and state level outpatient illness and viral surveillance," 2021.

[8] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, pp. 61–80, Jan. 2009.