Neural Network Comparison for Time Series Forecasting

Andy Chen, Boo Fullwood, Yunzhou Liu

May 05, 2020

Introduction

The COVID-19 (or coronavirus) pandemic has forced the widespread deployment of lockdowns, stay-at-home orders, and other social distancing measures to try and contain its spread, while policymakers and health researchers have been interested in forecasting the spread of the disease in order to help prepare and combat against it. This final project attempts to use machine learning techniques, specifically standard feed-forward and LSTM neural networks, to forecast future coronavirus cases in the United States for certain counties and across the state level in aggregate, based on data for past cases and deaths.

Additionally, we use the Google Community Mobility dataset to generate additional features for our neural networks in an attempt to improve forecast accuracy.

Related Work

A LSTM neural network addresses a particular shortcoming of traditional neural networks: an inability to learn long-term dependencies in the network. As a result, they are well-suited for modeling time series data, such as the number of cases per day in an epidemic. The coronavirus pandemic arrived only recently in the United States, but past work on applying feed-forward and LSTM neural network models to epidemic forecasting do exist in the literature. Wang et al. (2019) used multiple models, including LSTM neural networks, to model an HIV epidemic in Guangxi, China, finding evidence for its effectiveness over other time-series models. Mussumeci and Coelho (2020) used LSTM in comparison to LASSO and Random Forest regression to forecast weekly incidence of Dengue fever in Brazil, again finding evidence for LSTM's effectiveness. With regards to forecasting the coronavirus pandemic itself, Tomar and Gupta (2020) used an LSTM model to forecast coronavirus cases in India with somewhat decent results. Yang et al. (2020) found that an LSTM model trained on 2003 SARS epidemic data produced an incidence curve that fit surprisingly well to the real one. In contrast, Punn et al. (2020) predicted global cases over 10 days using only cases, deaths, and recovered data with both a standard deep neural network and an LSTM model, but found in comparison that polynomial regression was superior in the end. Nevertheless, these studies indicate potential for using an LSTM neural network for coronavirus cases forecasting.

Even non-LSTM models, however, can still demonstrate good prediction accuracy. A multilayer convolutional neural network using multiple inputs for cases and deaths forecasted the total number of confirmed cases in various Chinese cities with decent accuracy (Huang et al., 2020). As such, CNNs would also be good to try for predicting total cases.

Methodology

Data was drawn from U.S. COVID-19 case and death statistics from the Center for Systems Science and Engineering at Johns Hopkins University, already organized by state and county upon download. After exploring several models across different team members, using cumulative U.S. cases and deaths data for one set of models and county-level U.S. cases data for the other, we decided that the following data transformations should be applied for COVID-19 confirmed cases and deaths when exploring neural network structures:

1. Since there are too many counties in the dataset, we cannot create a one-hot encoding for each of the counties, and thus we tried to encode different counties by numbers, and view them as discrete quantitative variables. For the U.S. country-level aggregate cases and deaths data, this was obviously not needed.

2. Normalization of the records. For each county, we calculate their normalized cumulative confirmed cases by $\frac{x-\mu_x}{\sigma x}$. Although the data being a time series that continually increases (except for the odd instance of reported cases going

down in a day) means that this may not be the best way to apply normalization to our dataset, this still helped our models to converge (when compared to not normalizing).

We incorporated Google Community Mobility data, obtained using Google's BigQuery API, as additional inputs into our neural networks. Since social distancing, if executed properly, can greatly affect the rate of spread of the coronavirus in an area, mobility data showing how much people tend to visit certain locations would in theory serve as a proxy for the level of social distancing in an area, and help predict the number of future cases.

Model Structure

Overall, four models were created to evaluate both standard feed-forward and recurrent networks. The first standard model consists of two fully connected hidden layers with ReLU as the activation function, with each layer containing 128 units. The second is a convolutional neural net (CNN) of a 1 dimensional convolution aggregated by max pooling. The two recurrent networks depend on the Long Short Term Memory (LSTM) model and are built on identical layers of 128 LSTM units. Both a simple single layer and a stacked double layer were tested.

Data Structure

Again, we note that different team members initially explored forecast models in different ways, with the idea being to do a model comparison at the end. The models were initially constructed so that, for a 6-day period, the first 5 days of country-level cases/deaths data were used as inputs, while the 6th day was used as the label, when constructing the CNN and LSTM models. For the basic 2-layer model, a 11-day period of data was used, with the first 10 days of county-level cases data being used as inputs to the neural net and the 11th day being used as the label. Essentially, each model predicted the number of cases/deaths on the next day from the number of cases/deaths from prior days. Since there are too many counties in total for the 2-layer NN, we only chose counties in ten states, including North Carolina, New York, California, and Washington, to use for training the NN. We used counties in Illinois, Texas and Nevada as testing sets. For all neural networks trained, the error function was the mean-squared error (MSE) of normalized predicted cases.
In the final step, to compare our models using the same data (as opposed to country-level for some and county-level for the others), we took the 2-layer NN, CNN, LSTM, and stacked LSTM and trained them on state-level COVID-19 cases data, normalized in the manner described earlier, and using the past 10 days of data to predict cases on the 11th day. In addition, we trained the models again using the data on percent changes in mobility from a certain baseline for certain categories (retail and recreation, grocery and pharmacy, parks, transit stations, workplaces, and residential areas), from the Google Community Mobility dataset. These were also normalized similarly to the cases data, as described above. For the LSTM model, the mobility data for a certain data was simply appended to the end of the cases data for that day, so that a time series of 10 days of inputs together with the number of cases on the 11th day was formed. For the 2-layer NN, each observation used to predict cases on the 11th day was already a row vector with each of the past 10 days of cases data as a column. As a result, we added each of the past 10 days of data for each mobility category as its own column, effectively adding 60 more features to each observation.

Results

2 Layer Neural Net

This model demonstrated remarkable predictive power despite lacking advanced methods for correlating adjacent time points. Overall accuracy was good, but the model lagged behind actual counts as the predicted date moved away from the training time frame. While less accurate, especially at those distant points, than the more advanced models, this is a good demonstration of the power of simple, dense neural networks.

Convolutional Neural Net

The CNN model performed significantly better than expected for a simple model. It achieved comparable test loss results as the more complex LSTM models below and produced reasonable predictions out nearly 20 days from the last trained data point. The model was mostly linear, which matched the nature of the training data well, though it did diverge slightly from the test data as the result of moderate non-linear fluctuations.

LSTM Neural Net

The single layer LSTM model achieved moderate predictive performance. With data scaled to the range $[0, 1]$, the models predictive power declined more rapidly from the last trained data point than the other models. This resulted in an overall lower performance on the test data. The stacked LSTM model gave markedly better performance, maintaining low error even at high distances from the training point. However, both LSTM models demonstrated an interesting pattern in that both produced a model that decreases in growth rate as the days progress. This may be the influence of the memory capability of the LSTM layers, as the CNN did not produce this effect. It is this curve that primarily contributes to the single layer LSTM model's difficulty maintaining accuracy.

Augmentation with Mobility Data

Across the board, the models demonstrated significant improvement when incorporating the mobility data. This is a good indication of a strong, predictive relationship between case and death statistics and the mobility data. This intuitively makes sense given the probable origin of these changes, but the resilience to time delays between influences and the ability of relatively simple models to capture relationships between multiple time series is promising.

Additional Comparisons

We did unfortunately run out of time to properly set up a comparison between our neural network models and time-series forecasting models like ARIMA or VAR, or a mathematical logistic growth model. With more time, we would have set up such comparisons for our report.

Extensions

Although predicting the number of cases on the next day can still be useful, a more relevant metric to train our model on would be the accuracy of the predictions for multiple days ahead.
Additionally, major limitations exist on the in-person testing that generated the coronavirus cases data used, as different states and counties may have had different levels of access to testing kits or laboratories to process test results, or may have had differences in how or when cases were reported. The Google Mobility dataset also was collected from users that opted into a certain program, restricting how representative it may be. The hope of this project was to nevertheless produce a neural network model that could predict future coronavirus cases with some degree of accuracy in spite of the noise in the data.

Impact

These models represent relatively accurate forecasting tools in the near term. However, accuracy degrades as the prediction date distances from the last trained date. Caution should be used when interpreting long term predictions as trends may not be immediately apparent to the observer or the model. This does offer strong support for the incorporation of tangentially related data to improve the short term forecasting ability of similar models, as well as a way to identify data with significant predictive power relative to the statistic of interest.

Model	MSE	MAE
2 Layer NN	0.00066	0.01556
2 Layer NN (No Mobility)	0.00234	0.03068
CNN	0.00072	0.01617
CNN (No Mobility)	0.00579	0.03479
LSTM (Single)	0.00021	0.00788
LSTM (Single, No Mobility)	0.00210	0.01892
LSTM (Stacked)	0.00022	0.00812
LSTM (Stacked, No Mobility)	0.00201	0.01837

Table 1: Model Loss and Error Values

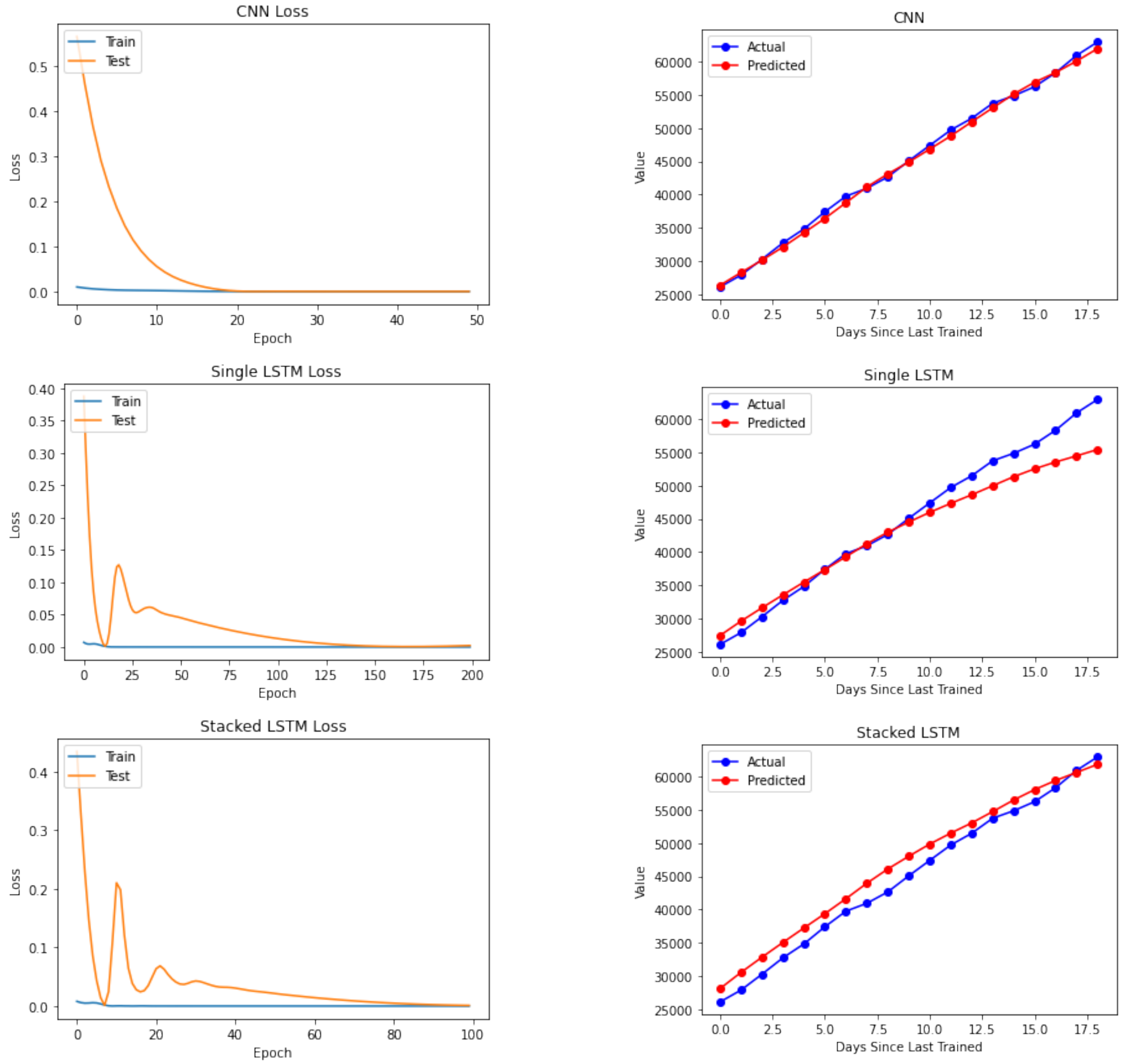

Figure 1: Model Training and Predictive Performance

References

[1] Ayyoubzadeh, S. M., Ayyoubzadeh, S. M., Zahedi, H., Ahmadi, M., & Kalhori, S. R. N. (2020). Predicting COVID-19 Incidence Through Analysis of Google Trends Data in Iran: Data Mining and Deep Learning Pilot Study. JMIR Public Health and Surveillance, 6(2), e18828.

[2] Huang, C. J., Chen, Y. H., Ma, Y., & Kuo, P. H. (2020). Multiple-Input Deep Convolutional Neural Network Model for COVID-19 Forecasting in China. medRxiv.

[3] Mussumeci, E., & Coelho, F. C. (2020). Machine-learning forecasting for Dengue epidemics-Comparing LSTM, Random Forest and Lasso regression. medRxiv.

[4] Punn, N. S., Sonbhadra, S. K., & Agarwal, S. (2020). COVID-19 Epidemic Analysis using Machine Learning and Deep Learning Algorithms. medRxiv.

[5] Tomar, A., & Gupta, N. (2020). Prediction for the spread of COVID-19 in India and effectiveness of preventive measures. Science of The Total Environment, 138762.

[6] Wang, G., Wei, W., Jiang, J., Ning, C., Chen, H., Huang, J., ... & Lai, J. (2019). Application of a long short-term memory neural network: a burgeoning method of deep learning in forecasting HIV incidence in Guangxi, China. Epidemiology & Infection, 147.

[7] Yang, Z., Zeng, Z., Wang, K., Wong, S. S., Liang, W., Zanin, M., ... & Liang, J. (2020). Modified SEIR and AI prediction of the epidemics trend of COVID-19 in China under public health interventions. Journal of Thoracic Disease, 12(3), 165.