Prediction of COVID-19 in the population of Wisconsin and Michigan.

Ayush Paudel
Department of Electrical and Computer Engineering
Michigan State University
East Lansing, Michigan 48823, USA
paudela1@msu.edu

Abstract— **In this project, we use Convolutional-LSTM series Neural Network (NN) to predict Covid-19 cases in the combined population of Wisconsin and Michigan. The model structure and the performance are both investigated, and the results are presented.**

I. Introduction

The COVID-19 public health crisis has exposed our inherent lack of understanding or management of the spread of a global pandemic. With the advent of powerful machine learning tools, it would be beneficial if we can predict the cases in a population which will allow us to have some understanding and will to make better decision to protect vulnerable members of society.

COVID-19 took the world down to its knees. One astonishing thing about it is how differently it affected people. If we compare the trends of transmission of COVID-19 even between territories close to each other, we can clearly observe vastly different outcomes. Figure 1 shows how rampant the COVID-19 infection was in Michigan and Wisconsin compared to Ontario with both territories having equivalent populations. [1]

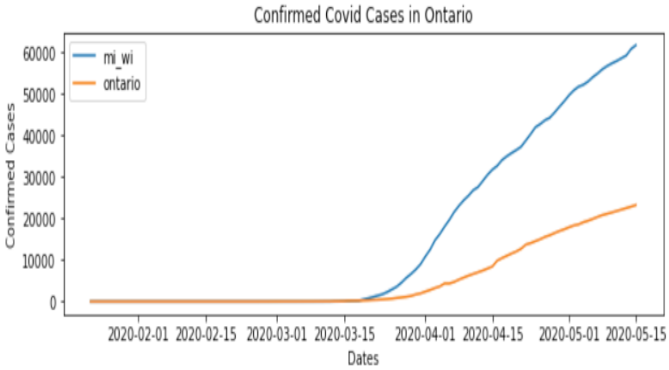


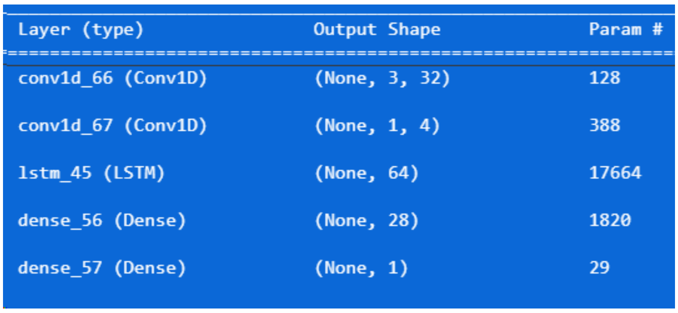
Fig 1: Comparison of covid cases in the US states Michigan +Wisconsin and the Canadian state of Ontario.

Our simple goal is to obtain a model that can accurately and efficiently predict the number of covid cases in a region. A neural network that is able to quantify this complex, intricate and astonishingly efficient transmission of COVID-19 might be very useful in our understanding and containment of it.

II. Model and Dataset

The dataset on which we'll be working is captured from Kaggle[2]. It compiles the COVID-19 confirmed cases and deaths from 184 countries worldwide including regions. The dataset starts from the advent of COVID-19 panic in North America, containing data from January 22, 2020 up to May 15, 2020. We are specially focused on Michigan and Wisconsin as these were the battle ground of bipartisan agendas during the lockdown.

Our model uses convolutional layer connected to the LSTM in series. The model characteristics are listed below:



```
Layer (type)                 Output Shape              Param #
================================================================
conv1d_66 (Conv1D)           (None, 3, 32)             128

conv1d_67 (Conv1D)           (None, 1, 4)              388

lstm_45 (LSTM)               (None, 64)                17664

dense_56 (Dense)             (None, 28)                1820

dense_57 (Dense)             (None, 1)                 29
```

Fig 2: Description of the NN model

The first layer is a one dimensional convolutional layer that is connected to a series of another 1-dimensional convolution layer, a LSTM and two fully connected layers with each layer having the 'RELU' activation function. A dropout layer wasn't included as the performance of the model was satisfactory.

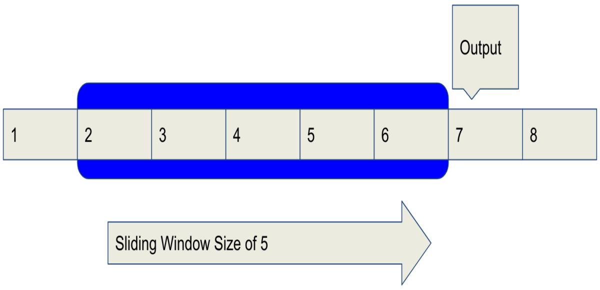


Fig 3: The Sliding Window algorithm for our model

The model uses a sliding window of size 5 to reduce the complexity and cost. The train and test data was split 90-10. Ninety percent of the train dataset was used to train the model and the 10 percent of the test dataset was used to test the prediction. The model is compiled using 'Mean-Squared Error' and 'NADAM' optimizer.

III. RESULTS

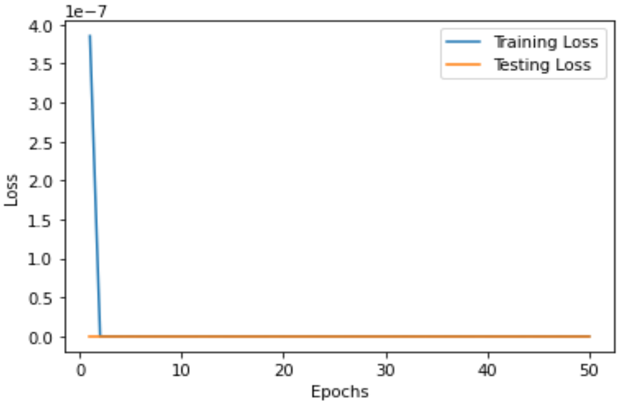


Fig 3: Training and Testing loss vs Epochs

Figure 3 shows the progression of training and testing losses for the NN. It is observed that the losses are very small. We suspect it is because of the data being linear and using RELU activation for all layers.

Then we move on to check the model's performance and accuracy.

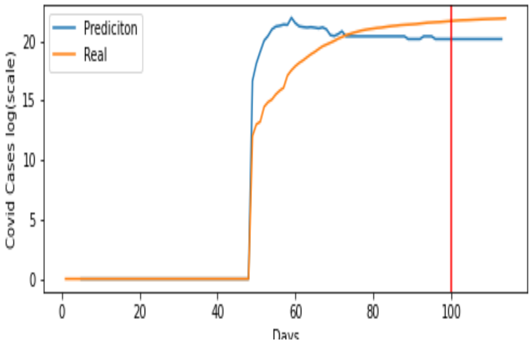


Fig 4: Predicted vs Real data for covid cases

As seen in figure 4, the prediction has some error at the start. As the RNN needs to be initialized with a series of data before it can finally begin predicting, the NN takes some time before it matches the trend. The red line is the 100-day mark. The data before the red line is training data while after the red line is the test data that we're trying to predict. Overall, the model shows satisfactory results.

IV. Conclusion

The trained model worked well to predict the future cases i.e. testing data. There is some fault at the beginning as the RNN needs some time to settle. Since, in real world, the data is already accessible, the model functioned as per our standard.

V. REFERENCES

[1] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9663291/

[2] https://www.kaggle.com/competitions/covid19-global-forecasting-week-4