Hybrid VAR-LSTM Networks Modeling and Forecasting COVID-19 Data in Canada

Elham Afzali, Adeola Adegoke, Zhiyong Jin, Woming Qiu, Liqun Wang*
Department of Statistics, University of Manitoba

January 15, 2021

Abstract

The world is witnessing an unprecedented pandemic of COVID-19 which affects all branches of our society specifically the health care system. There is an urgent need for utilizing mathematical and statistical modeling methods to study the patterns of the transmission of COVID-19 and to forecast its future development. In this study, we use the time series modeling method combined with a neural network technique. Specifically, we use a hybrid VAR-LSTM model to model the COVID-19 data in Canada. The data are first trained by using the Vector Auto-regressive (VAR) technique, then the outputs are used as the inputs for the Long Short-Term Memory (LSTM) networks by using a deep learning (DL) approach. Furthermore, a stochastic simulation model is used to produce a dataset of the confirmed cases that can be used to check the accuracy of the LSTM forecasts. The simulated dataset covers the entire process of the pandemic. The proposed hybrid network model is found to be flexible enough to capture the main patterns in the COVID-19 data in Canada.

Keywords: ARIMA, VAR, COVID-19, Deep Learning, time series, infectious disease, Long short-term memory (LSTM) networks, stochastic simulation model.

1 Introduction

Since the COVID-19 was declared a pandemic by the World Health Organization (WHO), many stakeholders have continued to work to analyze the data, study the spread mechanism of the epidemic, and generally end it. Also, many inter-disciplinary studies based on the classical compartmental (SIR) models have been used to investigate the efficacy of some public intervention measures and controls.

In the past, many statistical techniques such as regression analysis have been utilized to model the transmission of infectious diseases. However, the major drawback of using these methods for modeling infectious disease is that epidemiological data are oftentimes

*Corresponding author: Department of Statistics, University of Manitoba, Winnipeg, Manitoba, Canada R3T 2N2. Email: liqun.wang@umanitoba.ca

nonlinear, non-temporal, and as such would require the use of too many assumptions to achieve a simplistic model (Chimmula and Zhang 2020).

Meanwhile, machine learning (ML) methodologies have the power to improve on the shortcomings of parametric statistical methods to predict the real-time transmission dynamics of infectious diseases. Some of the methods that have been explored in the modeling of COVID-19 include k-nearest neighborhood (KNN), random forests (RF), neural networks (NN), support vector machines (SVM), and gradient boosted trees (Schwab et al. 2020). Huang et al. (2020) proposed a Convolutional Neural Network which has the highest accuracy compared to other DL methods in predicting the number of the confirmed cases. Also, Jiang et al. (2020) explored different ML methods to predict the clinical severity of COVID-19 and reported that the SVM classifier has the best accuracy (80%) among different classifiers. Further, de Moraes Batista et al. (2020) used different ML classifiers to forecast the diagnosis of COVID-19 and reported SVM and RF classifiers (0.847) as the best AUC scores classifiers.

Deep learning methods are another powerful technique that can be used to predict COVID-19 cases with high accuracy. Bandyopadhyay and Dutta (2020) used a combination of the Long short-term memory (LSTM) and Gated Recurrent Unit for the dataset training; Tomar and Gupta (2020) applied the LSTM and curve-fitting estimation methods; both achieving predictions similar to the results reported by clinical doctors. Also, Pal et al. (2020) predicted the risk class of a country via a shallow LSTM using weather data and trend data for predictions, while Chimmula and Zhang (2020) used LSTM for predicting the trend and possible stopping point for the COVID-19 pandemic.

Time series analysis (TSA) is another powerful tool that can be used to extract the information of the agents in the past time slices and use the information for predicting present and future time slices information. The auto-regressive integrated moving average (ARIMA) is a TSA model that is employed in stationary data for linearity identifications. ARIMA has a variety of advantages such as flexibility, quality, and a wide range of applications. In this study, since we have three outcome variables of interest, namely the numbers of confirmed, death, and recovered cases, we implement the multivariate generalization of the ARIMA - Vector Auto-Regressive (VAR) method. To tackle the disadvantages of the VAR model, the VAR-LSTM hybrid model can be adopted in the analysis to address the accuracy of the model in the presence of both linearity and nonlinearity of the data.

In this study, A hybrid DL model has been developed to predict the confirmed case of COVID-19 by implementing the VAR-LSTM method. The confirmed cases have been considered in the model as a sequence of data that are recorded sequentially in time. The number of recovered and death cases have also been considered in the model to feed up the training process to increase the accuracy of the model. The remaining parts of our paper are organized as follows. In the next section, we discuss our model in detail. Afterward, our method is applied to a case study, and the collected data are analyzed. Finally, the implications of the study and some general recommendations are discussed in the conclusive section.

2 Methods

2.1 Data

The dataset used in this research is collected from Johns Hopkins University and the Canadian Health authority, provided with the numbers of confirmed, death, and recovered cases from January 22 until August 29, 2020. Using the Augmented Dickey-Fuller (ADF; Cheung and Lai 1995) test, we checked for stationarity of the COVID-19 dataset. To remove non-stationarity, and also to mitigate the random noise of variables, we implemented the differencing technique. Then, the current dataset for VAR and LSTM parts is divided into an 80% training set and a 20% testing set that can be used to evaluate the performance of the model.

For ARIMA, exponential smoothing, and FNN models, the data from January 22 to August 16 of 2020 was used to forecast the next 15 days. The best forecasting model for ARIMA would be ARIMA(1,2,2) which is selected using the Akaike information criterion (AIC; Akaike 1974). Further, the Multilayer Perceptron (MLP) is used to determine the number of inputs, neurons, and outputs of the FNN model. Since there is no specific error distribution assumed, the plot of the FNN model does not include a confidence interval.

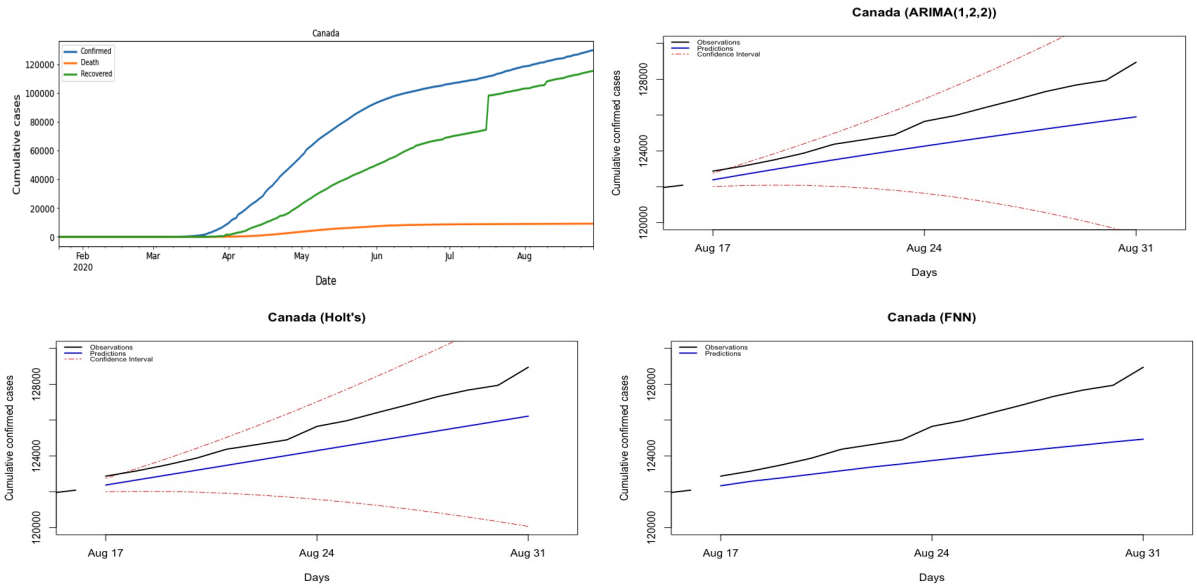

Figure 1: Top left: Cumulative confirmed, death and recovered cases in Canada from January 22 to August 29, 2020. Top right: 15-day forecasts using ARIMA model. Bottom left: 15-day forecasts using exponential smoothing (Holt's) method. Bottom right: 15-day forecasts using FNN model, all from August 17 to 29, 2020.

2.2 Auto Regressive Integrated Moving Average (ARIMA) Method

Time series (TS) data are kind of data which are collected over temporal order. A basic feature of time series data is that they contain temporal correlations. Therefore the TS

models are sensitive to temporal correlation patterns, which include trends, seasonal, and/or other cyclical variations. The commonly used TS model is the class of ARMA models. An auto-regressive integrated moving average model (ARIMA) is used for non-stationary time series. The standard notation for the ARIMA model is ARIMA(p,d,q), where the parameters are integer values which are defined as follows:

- p- the order of the auto-regressive part.
- d- the number of times that observations are differenced.
- q- the order of the moving average part.

Specifically, let Y_t be the stationary process after necessary differencing from the original time series. Then Y_t can be modeled using an ARMA model as

$$Y_t = \alpha + \beta_1 Y_{t-1} + \cdots + \beta_p Y_{t-p} + \epsilon_t + \phi_1 \epsilon_{t-1} + \cdots + \phi_q \epsilon_{t-q}.$$

Sometimes, the AR models and MA models are used in practice, as

$$Y_t = \alpha + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \cdots + \beta_p Y_{t-p} + \epsilon_t$$

and

$$Y_t = \alpha + \epsilon_t + \phi_1 \epsilon_{t-1} + \phi_2 \epsilon_{t-2} + \cdots + \phi_q \epsilon_{t-q}$$

respectively.

2.3 Long Short-Term Memory (LSTM)

Recurrent Neural Network (RNN) is similar to a traditional neural network; however, it has a memory-state feature that is added to the neurons. This feature contributes to its internal memory to process sequences of input. RRN contains loops in its structure which allow for the memory of previous calculations. In a traditional neural network, the model generates the output by multiplying the input with the weight and implementing the activation function. However, RNN employs the result obtained via the hidden layers to process future input.

LSTM networks are specific types of Recurrent Neural Networks (RNNs) architecture that can capture long-term dependencies to predict future events. This approach extends the memory of recurrent neural networks and introduces long-term memory into recurrent neural networks. LSTM networks can solve the limitations of traditional time series forecasting methods by adapting nonlinearities of the dataset and can provide precise results on temporal data.

The fundamental component of LSTM networks is memory blocks, which were implemented to tackle vanishing gradients. Vanishing gradient happens when small weights through several steps are multiplied over and over and it makes the gradients approach zero, which makes the neural network stops learning. LSTM contributes to solving this problem by memorizing network parameters in a series of gates for a long duration. It controls the information that has to be kept over time by utilizing forgetting and saving mechanism (Bandara et al. 2017).

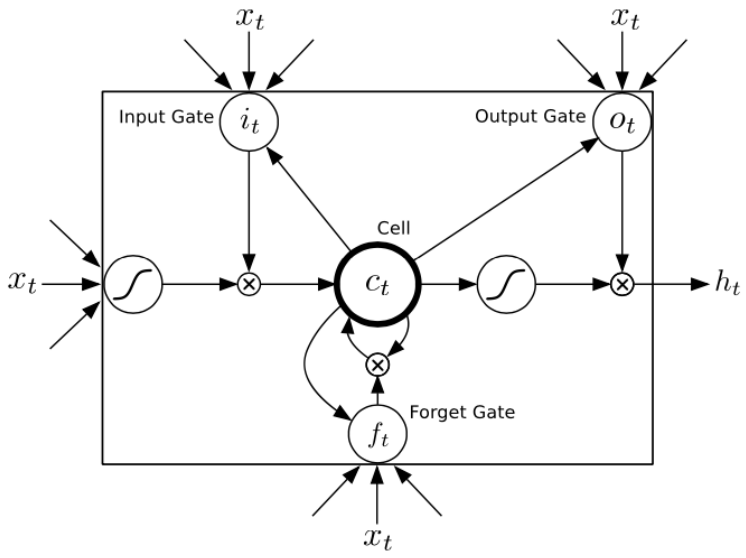

Figure 2: LSTM unit structure from Graves et al. 2013

LSTM operates at different time steps within each block and passes its output to the next block until the final LSTM block generates the sequential output. Memory blocks in LSTM architecture are similar to the differential storage systems of digital systems. Gates in LSTM helps in processing the information with the help of (sigmoid) activation function and output is between 0 and 1. The rationale for using a sigmoid activation function is that we need to pass only positive values to the next gates for getting a clear output.

LSTM network typically consists of: a cell that works as a memory part, and three gates (input, output, forget) which funnels the information through the LSTM network(layers) as shown in figure 2. According to figure 2, $\mathbf{C_t}$ is the cell state that contains the information. $\mathbf{h_t}$ is hidden state which is regulated by gates, through *sigmoid* and *tanh* activation functions. In general, the gates take in, as input, the hidden states from the previous time step $\mathbf{h_{t-1}}$ and the current input $\mathbf{X_t}$ and multiply them pointwise by weight matrices, $\mathbf{W}$, and a bias $\mathbf{b}$ is added to the product.

We assume that there are $\mathbf{h}$ hidden units, the mini-batch is of size $\mathbf{n}$, and number of inputs is $\mathbf{d}$. Thus, the input is $\mathbf{X}_t \in \mathbb{R}^{n\times d}$ and the hidden state of the last step is $\mathbf{H}_{t-1} \in \mathbb{R}^{n\times h}$. Correspondingly, the gates are defined as follows: the input gate is $\mathbf{I}_t \in \mathbb{R}^{n\times h}$, the forget gate is $\mathbf{F}_t \in \mathbb{R}^{n\times h}$, and the output gate is $\mathbf{O}_t \in \mathbb{R}^{n\times h}$. They are calculated as follows:

$$\mathbf{I}_t = \sigma\left(\mathbf{X}_t\mathbf{W}_{xi} + \mathbf{H}_{t-1}\mathbf{W}_{hi} + \mathbf{b}_i\right)$$
$$\mathbf{F}_t = \sigma\left(\mathbf{X}_t\mathbf{W}_{xf} + \mathbf{H}_{t-1}\mathbf{W}_{hf} + \mathbf{b}_f\right)$$
$$\mathbf{O}_t = \sigma\left(\mathbf{X}_t\mathbf{W}_{xo} + \mathbf{H}_{t-1}\mathbf{W}_{ho} + \mathbf{b}_o\right)$$

where $\mathbf{W}_{xi}, \mathbf{W}_{xf}, \mathbf{W}_{xo} \in \mathbb{R}^{d\times h}$ and $\mathbf{W}_{hi}, \mathbf{W}_{hf}, \mathbf{W}_{ho} \in \mathbb{R}^{h\times h}$ are weight parameters and $\mathbf{b}_i, \mathbf{b}_f, \mathbf{b}_o \in \mathbb{R}^{\mathrm{i}\times h}$ are bias parameters.

The forget gate determines what information will be deleted from the cell state. The result in the output is a number between zero and 1. Zero means deleting all while 1 implies

remember all. The input gate utilizes *tahn* activation layer to create a vector of potential candidate as follows:

$$\tilde{\mathbf{C}}_t = \tanh\left(\mathbf{X}_t\mathbf{W}_{xc} + \mathbf{H}_{t-1}\mathbf{W}_{hc} + \mathbf{b}_c\right)$$

Then, the old cell state C_{t-1} is updated as follows:

$$C_t = f_t * C_{t-1} + I_t * \tilde{C}_t$$

Finally, the scaled cell state is multiplied by the filtered output to obtain the hidden state h_t to be passed on to the next cell:

$$h_t = O_t * \tanh\left(C_t\right)$$

2.4 Hybrid VAR-LSTM-model

The main idea of the hybrid VAR-LSTM model is explained in the work titled "Stock Price Prediction Based on ARIMA-RNN Combined Model" by Shui-Ling and Li (2017). This model contains a moving average filter which is used to separate the low fluctuation times series from the high fluctuation time series. To forecast COVID-19 confirmed cases, we implement the VAR method to the training dataset to improve the training of our neural network. At this point, the VAR is learning the internal behavior of our multivariate dataset, adjusting the outlier data, properly handling the NaNs, and correcting the anomalous trends. All this information is stored in the fitted values, such that they are a modified copy of the real data which have been implemented by the model, during the training procedure. This study aims to predict the COVID-19 confirmed cases by implementing the hybrid VAR-LSTM method. We consider the confirmed cases as a sequence of data that are recorded sequentially in time with the help of death and recovered cases. (This study aims to predict the COVID-19 confirmed cases by implementing the hybrid VAR-LSTM method, where we consider the confirmed, death, and recovered cases as a sequence of data that are recorded sequentially with time.)

3 Results and discussion

In this section we implement the model on two different datasets; the real dataset described in section 2.1 and a simulated dataset in section 3.2 so that we may investigate the performance of our method.

3.1 Real Data

We split data into training and testing sets. Then we trained our network with 80% of data which is until 15^{th} July 2020. We examined our model predictions using Mean Square Error (MSE) and Mean Absolute Percentage Error (MAPE). In figure 1 we plotted the total number of confirmed, death, and recovered cases in Canada during the January 22 until August 29, 2020. According to figure 1, we can observe that, from the beginning of June, there is a change in the slope of Canada's confirmed cases, and the confirmed cases are going

to increase more sharply(exponentially) due to changing the policy about opening public places.

Our model in comparison with other forecasting models, such as LSTM, performs much better. In our model, we perform a two-step training process. First, we implement the VAR method on the trained dataset, then evaluated the VAR method using the test dataset. The RMSE is 254.81 while the MAPE is 10.36%. Then we extract what VAR has learned, and use our fitted VAR to feed our LSTM model to improve the training of an LSTM network.

Before fitting the VAR model, we differenced the data two times to achieve a stationary time-series, then check the stationarity by ADF test. With LSTM we also combine time attributes like lagged data(2 steps) for all three variables (Confirmed, Deaths, Recovered).

While we are performing multiple steps training we may be exposed to the Catastrophic Forgetting problem- a problem faced by many models and algorithms. When the model is trained on one task, and then trained on a second task, many machine learning models forget how to perform the first task which would be a serious problem for neural networks. We use dropout to tackle this problem. Moreover, to have a properly tuned network, we preserve a final part of our previous training as validation. Finally, to minimize the bias on our training algorithm we performed the regularization part in the LSTM network.

Meanwhile, based on our testing/validation dataset, the RMSE error is about 985.66 with an accuracy of 93.07% for long-term predictions. The predictions of the LSTM model and VAR-LSTM model are shown in figure 3 with solid orange and blue lines respectively. It shows that our model was able to predict the confirmed cases better than the LSTM model with minimum fluctuation. From figure 1 we can say that confirmed cases in Canada were growing linearly until March 16, 2020. While the number of infections in Canada started to follow exponential distribution after March 17, 2020.

The benefit of this hybrid model is that this approach doesn't need any assumption for estimating optimal parameters, which are necessary for statistical methods. VAR-LSTM model is different from statistical methods since it was able to overcome the parameter assumptions using cross-validation and achieved better performance by reducing the uncertainty.

The actual number of cases might be higher than the reported cases since some people may be recovered before even testing. So, this fact may lead to a different result of our model estimations.

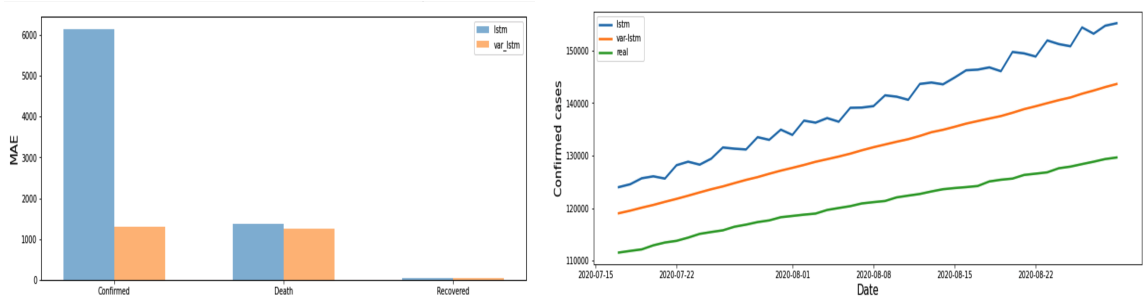

Figure 3: Left: The comparison between the MAE of the LSTM and the VAR-LSTM models. Right: The comparison between the MAE of the LSTM and the VAR-LSTM models.

3.2 Stochastic Simulated Data

In this section, we mainly introduce the method of how we generate the simulated data. Then we carry out our method on this dataset the same as the real dataset.

Similar to compartmental models, it is a direct idea to divide the population into different groups according to the status. This simulation assumes that susceptible individuals (S) turn to exposed individuals (E) that have been infected but can not infect susceptible individuals. After a latent period, the exposed individuals turn to infectious (I). The infectious individuals end up with 3 results: hospitalized and recovered, hospitalized but unfortunately deceased, never hospitalized and recovered; we call this group removed individuals (R). As a result of the rapid spreading, it is assumed that the population is constant. For immunity, this simulation assumes the population is susceptible in general. For adaptive immunity, studies show that people who have recovered from infection have antibodies to the virus, but some studies also show that the recovered individuals may only get 6-12 months of immunity. We simulated both permanent immunity and 6-12 months of immunity scenarios.

For the random period of latency and actual infectious period without quarantine, we use the following settings. Studies show that the period of latency is a random variable that follows a log-normal distribution. We have explained that the removed group includes three possible outcomes, depending on whether it is hospitalized and survival status. We assume an individual that is hospitalized can not infect others, although we know the respectful medical workers are taking risks. Therefore, when it is without public health measures, for example, isolation, we assume the actual infectious period follows a mixed distribution, which in our simulation is mixed up with three distributions: two gamma distributions with different parameters and a log-normal distribution. For the random period of latency and social

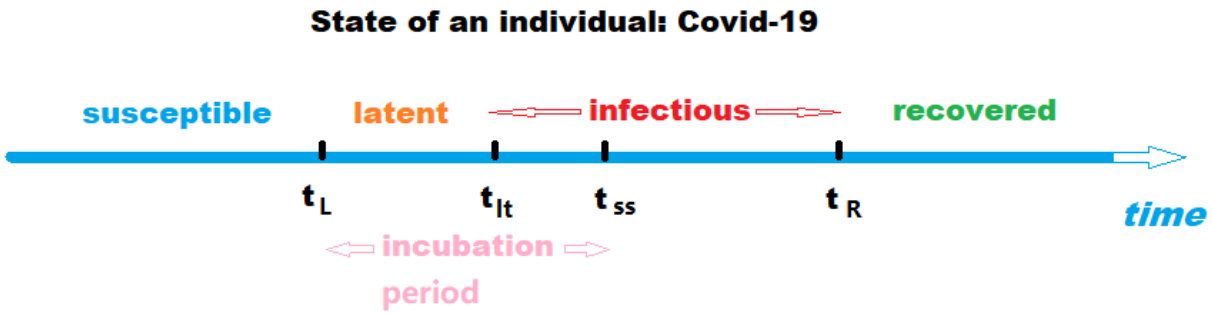

Figure 4: The progress of COVID-19 in an individual

infectious period with quarantine, we consider public health measures. Quarantine is used to keep someone who might have been exposed to COVID-19 away from others. Isolation is used to separate people infected with the virus (those who are sick with COVID-19 and those with no symptoms) from people who are not infected" (cited from CDC). Considering the public health measures, the exposed individuals (E) and infectious individuals (I) may get quarantined or isolated (Q) so that they do not have contact with susceptible individuals. We include the virus carriers with no symptoms in consideration: an infected individual may never suffer any symptoms or on the other hand it may infect others before symptoms show. It is reasonable to assume that when symptoms show the infected individual will get isolated

thus removed. Please note that the time of quarantine t_Q may be earlier than the time t_{lt} when an individual gets infectious, so an infected individual may never infect others because of the quarantine. We call $\max\{t_Q - t_{lt}, 0\}$ as the public health infectious period, which also follows a mixed distribution. Here we simplify as a log-normal distribution in the following simulations.

In the following demonstration, we use balls to replace individuals in a population and assume all balls move within a certain area following a certain moving trace pattern. An event of contact is called hitting. At certain observation times, if the distance of two balls is less than a certain value, we call a hitting happening between these two balls. This model assumes that a hitting event will not change the movement of balls in contact. In this model, the distance we now are using is euclidean distance. Hittings may only happen at certain observation times. We are using this assumption for the discretization of continuous-time and movements.

While the period of latency and infectious period are biological features of the disease, the infectious rate is affected by the transmission risk factor (the probability p of disease transmission in a contact, which is a biological feature of the disease), as well as a social factor (the average number of contacts per person per time). The average number of contacts per person per time depends on the social patterns of contact in a particular society, which can be time-varying as a result of public health measures and knowledge of the new disease in the pandemic.

In our simulation we assume all balls move within a certain area following a moving trace pattern. The movement frequency in each pattern is an important parameter that can be used to measure the movement of society (e.g., changes due to lockdown). We mainly discuss the following three moving patterns:

- Uniformly distributed: at every moment, the location of each ball is i.i.d uniformly distributed within the area and are not dependent on their previous locations;
- Brownian motion trace patterns: every ball does 2-dimensional Brownian motion (with reflection) in the area independently;
- Levy fight trace pattern: every ball does 2-dimensional levy flight (with reflection) in the area independently.

To check the ability of the LSTM network, we implement the network on a simulated dataset; represented in the top left of figure 5. This figure represents the cumulative confirmed cases in a complete process. We considered the whole process as three subsets(first 65 data points(days), first 115 data points(days) and first 165 data points(days)) and in each subset, we used the first 80% of the data for training, and the remaining 20% for testing. According to the results in the top right, bottom left, and bottom right of figure 5, the tolerance for the prediction by the LSTM network is around 100. As shown in the bottom left of figure 5, the network could predict the end of the process in which the line would be flattened.

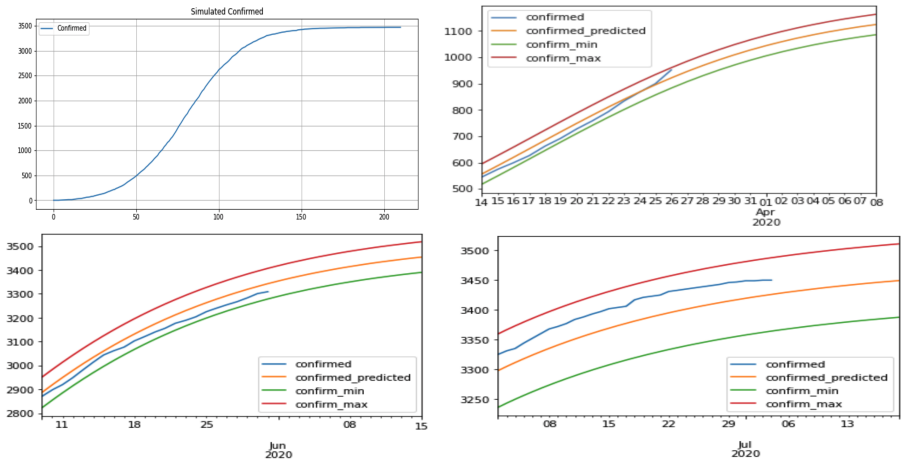

Figure 5: Top left: Simulation of whole process. Top right: prediction using the first subset data(65 days). Bottom left: prediction using the second subset data(115 days). Bottom right: prediction using the third subset data(165 days).

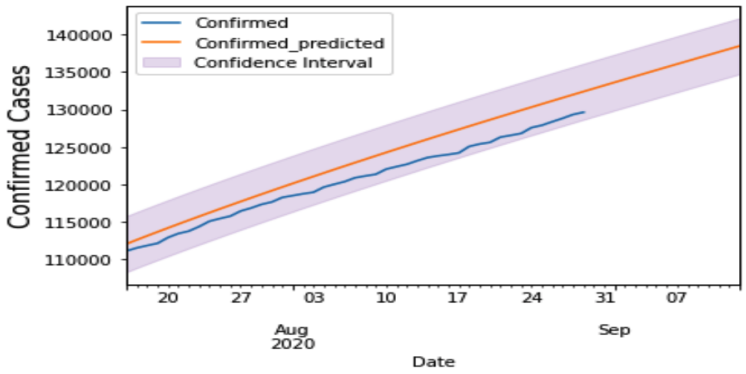

Figure 6: Confidence Interval for predicting confirmed cases until September 15^{th}, 2020.

4 Conclusion and future work

The trend of data shows that measures such as social distancing taken by Canadian public health authorities, which minimized human exposure showed a positive impact to control the transmission rate. While after opening the public places, the rate of transmission in Canada follows an exponential trend. After simulations and data fitting, according to figure 6 our model predicted Canada would witness a peak in the second week of September (September 15^{th}, 2020) which would be the start of the second wave and it would be different for each province. Our result could help the Canadian government to monitor the current situation and use our forecasts to prevent further transmissions. Although we couldn't predict patients who are in the incubation period, yet our proposed model can predict the size of an epidemic, and also predict the start of the second wave of the epidemic with high accuracy.

References

Akaike, H. (1974). A new look at the statistical model identification. *IEEE transactions on automatic control 19*(6), 716–723.

Bandara, K., C. Bergmeir, and S. Smyl (2017). Forecasting across time series databases using long short-term memory networks on groups of similar series. *arXiv preprint arXiv:1710.03222 8*, 805–815.

Bandyopadhyay, S. K. and S. Dutta (2020). Machine learning approach for confirmation of covid-19 cases: Positive, negative, death and release. *medRxiv*.

Cheung, Y.-W. and K. S. Lai (1995). Lag order and critical values of the augmented dickey–fuller test. *Journal of Business & Economic Statistics 13*(3), 277–280.

Chimmula, V. K. R. and L. Zhang (2020). Time series forecasting of covid-19 transmission in canada using lstm networks. *Chaos, Solitons & Fractals*, 109864.

de Moraes Batista, A. F., J. L. Miraglia, T. H. R. Donato, and A. D. P. Chiavegatto Filho (2020). Covid-19 diagnosis prediction in emergency care patients: a machine learning approach. *medRxiv*.

Graves, A., N. Jaitly, and A.-r. Mohamed (2013). Hybrid speech recognition with deep bidirectional lstm. In *2013 IEEE workshop on automatic speech recognition and understanding*, pp. 273–278. IEEE.

Huang, C.-J., Y.-H. Chen, Y. Ma, and P.-H. Kuo (2020). Multiple-input deep convolutional neural network model for covid-19 forecasting in china. *medRxiv*.

Jiang, X., M. Coffee, A. Bari, J. Wang, X. Jiang, J. Huang, J. Shi, J. Dai, J. Cai, T. Zhang, et al. (2020). Towards an artificial intelligence framework for data-driven prediction of coronavirus clinical severity. *CMC: Computers, Materials & Continua 63*, 537–51.

Pal, R., A. A. Sekh, S. Kar, and D. K. Prasad (2020). Neural network based country wise risk prediction of covid-19. *arXiv preprint arXiv:2004.00959*.

Schwab, P., A. D. Schütte, B. Dietz, and S. Bauer (2020). predcovid-19: A systematic study of clinical predictive models for coronavirus disease 2019. *arXiv preprint arXiv:2005.08302*.

Shui-Ling, Y. and Z. Li (2017). Stock price prediction based on arima-rnn combined model. *DEStech Transactions on Social Science, Education and Human Science* (icss).

Tomar, A. and N. Gupta (2020). Prediction for the spread of covid-19 in india and effectiveness of preventive measures. *Science of The Total Environment*, 138762.