CPSC-672: FUNDAMENTALS OF SOCIAL NETWORK ANALYSIS AND DATA MINING

FORECASTING AND ANALYSIS OF COVID19 TOTAL CASES USING RNN – LSTM.

TEAM MEMBER: **VIVEK SURESH RAJ (UCID: 30109204)**

DATE : 17-DEC-2020

Keywords

Recurrent Neural Network.

Long Short Term Memory.

Mean squared Error.

Abstract

A Recurrent Neural Network based Long Short-Term Memory is being applied on the dataset to predict the future number of total cases. The dataset is made use from an authentic and reliable source. Since, the study is focussed on Canada, the data-preprocessing involves in creating a Data Frame with the necessary attributes. Upon then, LSTM model is built and the best performed model is selected. The selected model with minimum errors is chosen to predict the number of total cases. It is observed that the final mean squared error is observed to be less than 3%. This study also suggests the possible ways to prevent the spread of COVID-19. So, this model could also be applied on predicting the covid19 cases with other countries.

1. INTRODUCTION:

Covid19 is a pandemic disease caused by newly discovered coronavirus which was originated from Wuhan, China. The deadly virus was responsible for the death of approximately 10M+ people globally. Most of the people, infected with this disease might experience mild illness with short respiratory problem and would recover without any special treatment. However, older people with prolonged health issues like Cardio-vascular illness and other health issues might experience severe problems which might lead to death. (WHO, n.d.)

Currently, there are vaccines that are at their final stages. There are even a couple of vaccines like *Pfizer* and *Moderna* that are stated to be 90% and above effective (Moderna covid19 vaccine highly effective, n.d.). However, it is necessary to remain safe and healthy until the vaccines are distributed to all of the citizens of the country. To achieve this, it is required to practice a very good social distancing and to practise wearing masks (Centers for Disease Control and Prevention, n.d.). But for a country to sustain this pandemic it necessary that they act clearly in knowing the future cases and to implement counter-measures so that it might benefit it's citizens as well as the country's economy.

The main objective of the study is to analyse the existing cases on covid19 with Canada and to develop a model that could predict the optimized value of the total cases with minimum Mean Squared Error (mse).

2. METHOD:

2.1 RNN – Recurrent Neural Network:

The RNN model differs from the conventional Neural Network with the 'hidden state' – h(s). This is called as the memory or the information in real-time examples. It carries the information that is

helpful in predicting the output, y(s). Since the hidden layers are connected to the next input token, x(s) is could be used for the hierarchical or sequential data predictions. The RNN's contains three weight matrices namely U, V and Ws. The hidden state's weight matrix W, plays an important role in passing the information from previous hidden state, h(s-1). In the study experiment the inputs were taken as the 'Date' of observed cases and 'total cases' that was recorded sequentially. However, the hidden is generated by the function of

$$H(s) = W*X + B$$

Where H(s) is the actual hidden state with respect to the given input token, W is the Weight matrix and X(i) is the input token and B is the Bias to be added.

2.2. LONG SHORT TERM MEMORY:

For prediction events, the LSTM's are considered to be the most feasible models as they predict the future information with the various input features of the dataset that are highlighted. The fundamental principle of the LSTM model is to ***'forget'*** the information that is no longer needed for the output to be predicted and to ***'memorize'*** the information for a better prediction. This design is useful than the RNN's as the LSTM allow for the information for a long-term storage than the RNN's. Also, an advantage of LSTM is that, it overcomes vanishing gradient and exploding gradient which are considered to be the major trouble with RNN. (Long Short Term Memory - Concept, n.d.)

In the design described below, it could be seen that an LSTM cell contains 'gate' functions to 'add' and to 'remove' information. It also uses 'tanh' and 'sigmoidal' functions as activation function to normalize the value between '-1' to '+1' with respect to the tanh and '0' to'+1' with respect to the sigmoidal function.

2.2.1. SUMMARY OF THE LSTM MODEL:

The first step is the 'forget layer' that makes the network to forget the information that are no longer needed to predict the information. The previous hidden state h(t-1) is subjected to the multiplication with the input token x(t) with a weight matrix to 'MULTIPLICATION GATE' which erases the information that are not needed further. (Understanding LSTM networks, n.d.)

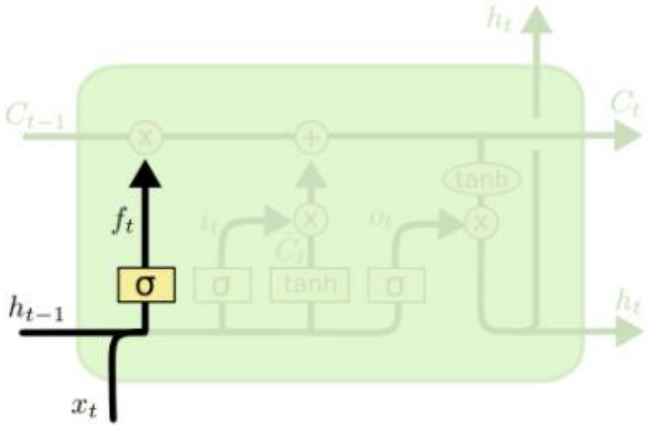

Fig.1 Forgot layer in LSTM cell

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

The second step is to Add/Store the information that is required to predict the output. The Ć(t) contains the 'actual' data to be added to C(t) – Long-Term memory. But the action is done through an input gate called i(t).

$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

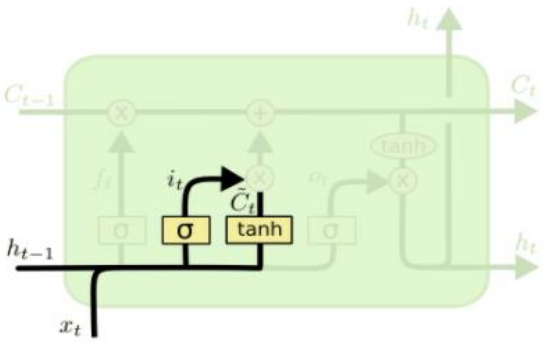

Fig.2 ADD layer in LSTM cell

Now that, the new information is updated at the C(t) while the forget layer has removed the un-related data from the LSTM cell. This step combines the previous steps 1 and 2 together through the 'MULTIPLICATION GATE' and 'ADDITION GATE' as represented in the below equation.

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

The C(t) – Long term memory is updated by the 2nd step.

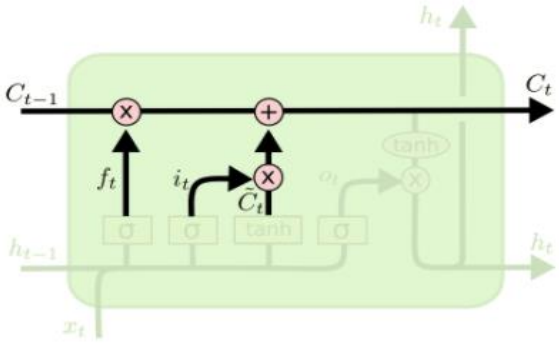

Fig.3 ADD/STORE layer in LSTM cell

The final layer in the LSTM cell operations is called the Output layer. O(t) is determined through the transpose operation with previous hidden state h(s-1) and present input x(t) which is subjected to sigmoidal function to stretch the variety of values obtained between 0 to 1. This is employed to get the short-term memory h(t) along with the previous step's value which is subjected to tanh operation. This outputs the short-term memory which is a part of the long term memory.

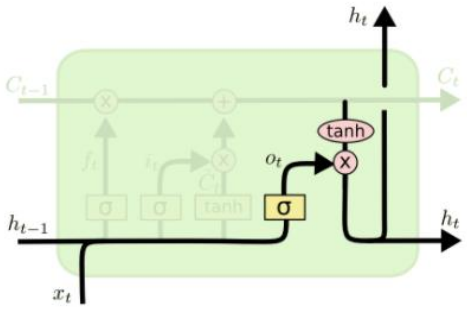

Fig.4 Output layer in LSTM

$$o_t = \sigma\left(W_o \; [h_{t-1}, x_t] \; + \; b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

2.2.2. VARIANTS ON LSTM:

Some variants of LSTM are that the C(t-1) has a peephole connected through which the gates are connected. It makes the C(t-1) pass through every gate in the LSTM cell. However, the LSTM is considered as much superior and powerful in comparison with the other network model like the

GRU - Gated Recurrent Unit though the GRU's are considered as the simplest ones (TowardsDataScience - LSTM vs GRU, n.d.).

3. EXPERIMENTAL DESIGN:

3.1. DATASET DESCRIPTION:

The dataset for the study is made use from *'ourworlddata'* which updates the data with the covid19 on a daily basis (ourworlddata, n.d.). The global dataset that is available is made use to study. The dataset is imported in the project for the analysis. It is a sequential data that has the number of total cases recorded from January, 2020 to Nov, 2020. The number of total cases that has been entered on a daily basis is visualized through 'matplotlib' for further analysis of the data.

```
In [19]: plot_col = ['total_cases', 'total_deaths']
         plot_features = df[plot_col]
         plot_features.index = date_time
         _ = plot_features.plot(subplots=True)

         plot_features = df[plot_col][:480]
         plot_features.index = date_time[:480]
         _ = plot_features.plot(subplots=True)
```

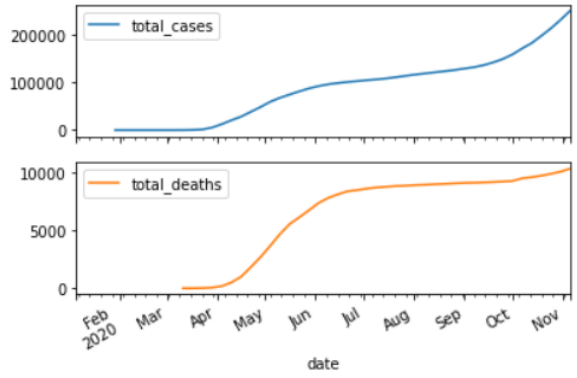

Fig. 5 Number of Total cases vs Date plot of Canada from Feb, 2020 to Nov, 2020.

3.2. LSTM MODEL DESIGN:

The study experiment is conducted on python which is a high-level general-purpose programming language with open-source libraries including NumPy, Pandas, Matplotlib, Seaborn, Keras and TensorFlow. The high-level API's which in the study are the deep learning packages are used to built LSTM model. The RNN's model is understood through which its variants including Bi-directional RNNs, Stacked RNN, Vanilla RNN, Convolutional RNNs are studied upon which LSTM is considered for the sequential data analysis and forecasting of covid19 data (RNN and LSTM, n.d.). The train data and test data are trained on the models for prediction. The parameters on the model like the Batch size, Epoch limit, Dropout rate, learning rate, Number of hidden layers are adjusted for better accuracy ***(refer to fig.6).*** The efficiency of the model is studied using the EPOCH vs LOSS curve.

In these models, the optimizers to be used is the 'adam' optimizer for optimizing the mean squared error (mse) ***(refer to fig.7)***. Thus finally, based on the accuracy the best model is chosen for prediction.

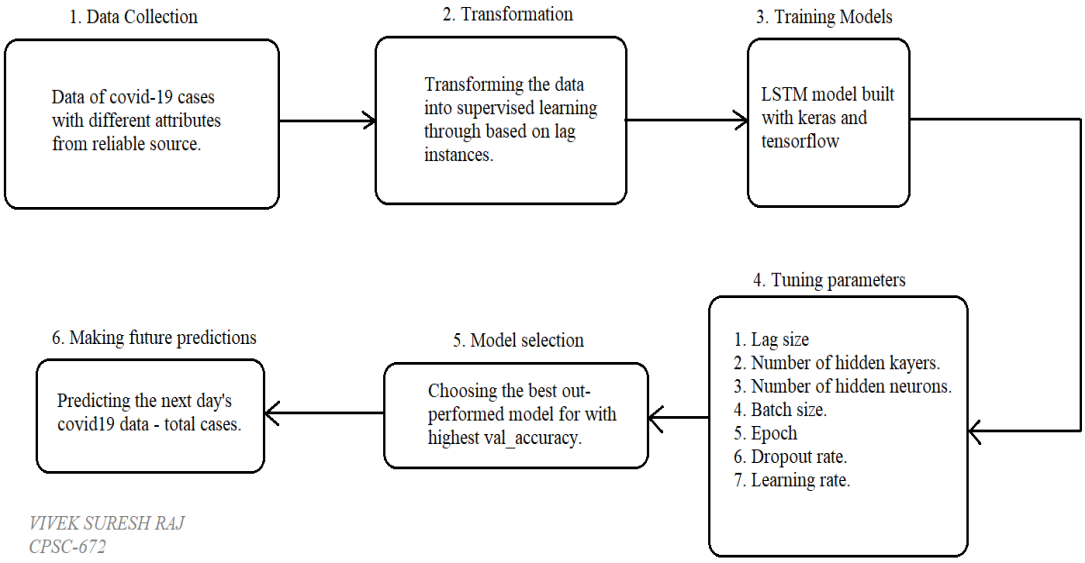

Fig. 6 Proposed layout for the study experiment.

3.3.ERROR – MSE:

The mean squared error is the parameter to test the performance of the model. The mean squared error is the SUM OF ALL THE DATA POINTS of the SQUARED DIFFERENCE BETWEEN THE PREDICTED VALUE AND THE ACTUAL VALUE divided by the amount of data's. (Loss functions, n.d.)

$$\sum_{i=1}^{n} \frac{\left(w^T x(i) - y(i)\right)^2}{n}$$

```
In [8]: lstm_model.fit_generator(generator, epochs=28)

C:\Users\Vivek\anaconda3\envs\tf-gpu\lib\site-packages\tensorflow\python\keras\engine\tr
_generator` is deprecated and will be removed in a future version. Please use `Model.fit
  warnings.warn('`Model.fit_generator` is deprecated and '

Epoch 1/28
314/314 [==============================] - 5s 4ms/step - loss: 0.0592
Epoch 2/28
314/314 [==============================] - 1s 4ms/step - loss: 1.4422e-05
Epoch 3/28
314/314 [==============================] - 1s 4ms/step - loss: 1.0418e-05
Epoch 4/28
314/314 [==============================] - 1s 4ms/step - loss: 8.3592e-06
Epoch 5/28
314/314 [==============================] - 1s 4ms/step - loss: 8.2867e-06
Epoch 6/28
314/314 [==============================] - 1s 4ms/step - loss: 5.7819e-06
Epoch 7/28
314/314 [==============================] - 1s 4ms/step - loss: 5.5388e-06
Epoch 8/28
314/314 [==============================] - 1s 4ms/step - loss: 8.2712e-06
Epoch 9/28
314/314 [==============================] - 1s 4ms/step - loss: 1.0907e-05
Epoch 10/28
314/314 [==============================] - 1s 4ms/step - loss: 3.3479e-05
Epoch 11/28
314/314 [==============================] - 1s 4ms/step - loss: 8.6027e-06
Epoch 12/28
314/314 [==============================] - 1s 4ms/step - loss: 2.3694e-05
Epoch 13/28
```

Fig. 7 Generating Models of LSTM

4. RESULTS AND DISCUSSIONS:

4.1. PREDICTION:

Upon observing the number of total cases predicted by the model resembles closely with the actual numerical value. It is observed upon the visualization that there are chances of cases to rise by an approximate average rate of 25%. Upon observing the graph ***(refer to fig.8)***, it shows a peak raise in the number of cases for the next day. This show the learning rate of the model through which the optimized value for rise in the confirmed cases could be considered. If the model is good with learning and understanding the dataset then the generated prediction would be optimum.

```
test_data.plot()
```

```
<AxesSubplot:xlabel='date'>
```

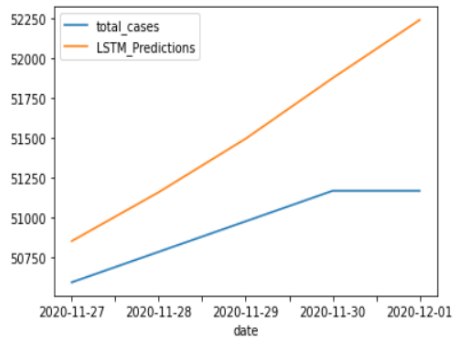

Fig. 8 Graphical representation of Prediction vs Actual value

```
test_data['LSTM_Predictions'] = lstm_predictions
test_data
```

```
C:\Users\Vivek\anaconda3\envs\tf-gpu\lib\site-pack
A value is trying to be set on a copy of a slice f
Try using .loc[row_indexer,col_indexer] = value ir

See the caveats in the documentation: https://panc
rsus-a-copy
  """Entry point for launching an IPython kernel.
```

date	total_cases	LSTM_Predictions
2020-11-27	50595	50853.114678
2020-11-28	50786	51159.031804
2020-11-29	50977	51494.277029
2020-11-30	51168	51873.607536
2020-12-01	51168	52237.441799

Fig. 9 LSTM predicted value vs total cases.

4.2. DISCUSSIONS:

In this section, we will discuss on the developed model and results. The model is developed with three hidden dense layers. It has an EPOCH size of 28. Now, that the activation function for the model is chosen. The model is given with a 'ReLU' layer at the output. Rectified Linear Unit (ReLU) is responsible for bring the output as a real numerical value. This make it clear that a linear function at the output layer is not necessary. Thus, it could be observed that there is a rise of an average of ~2% numerical values rise in the LSTM predictions per day. With a constant rise in the daily numerical value under 'LSTM prediction' the model has predicted the output with a steady rise in the number of cases with the next day. The output of the developed model and its predictions will help the higher authorities in implementing some serious measures that would not put the lives of public at stake. It might also be useful if a provincial data is made available so that the density of number of confirmed cases could be found exactly. This would guide the authorities in

considering the respective province or its city to be given some special lockdown to make sure that the situation is under control.

5. CONCLUSION:

This study has addressed and analysed the impacts of covid19 virus. A complete analysis of Canada's covid19 data was performed. Recurrent Neural Network based Long Short Term Memory (LSTM) cells were used to predict the results. The best model with less mean squared error is used for predicting the numerical values. The results were observed and it could used for implementing counter-measures in controlling the spread of covid-19.

6. REFERENCES:

1. Centers for Disease Control and Prevention. (n.d.). Retrieved from https://www.cdc.gov/coronavirus/2019-ncov/prevent-getting-sick/index.html
2. Long Short Term Memory - Concept. (n.d.). Retrieved from https://medium.com/@kangeugine/long-short-term-memory-lstm-concept-cb3283934359#:~:text=LSTM%20is%20well%2Dsuited%20to,and%20other%20sequence%20learning%20methods.&text=The%20structure%20of%20RNN%20is,hidden%20Markov%20model.
3. Loss functions. (n.d.). Retrieved from https://heartbeat.fritz.ai/5-regression-loss-functions-all-machine-learners-should-know-4fb140e9d4b0
4. Moderna covid19 vaccine highly effective. (n.d.). Retrieved from https://www.wsj.com/articles/modernas-covid-19-vaccine-is-next-in-line-for-authorization-11608028201
5. ourworlddata. (n.d.). Retrieved from https://ourworldindata.org/coronavirus-data
6. RNN and LSTM. (n.d.). Retrieved from https://medium.com/@purnasaigudikandula/recurrent-neural-networks-and-lstm-explained-7f51c7f6bbb9
7. TowardsDataScience - LSTM vs GRU. (n.d.). Retrieved from https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21
8. Understanding LSTM networks. (n.d.). Retrieved from http://colah.github.io/posts/2015-08-Understanding-LSTMs/
9. WHO. (n.d.). Retrieved from overview of coronavirus: https://www.who.int/countries/can/