HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY

ANALYTICAL REPORT

Subject: Discrete Structure

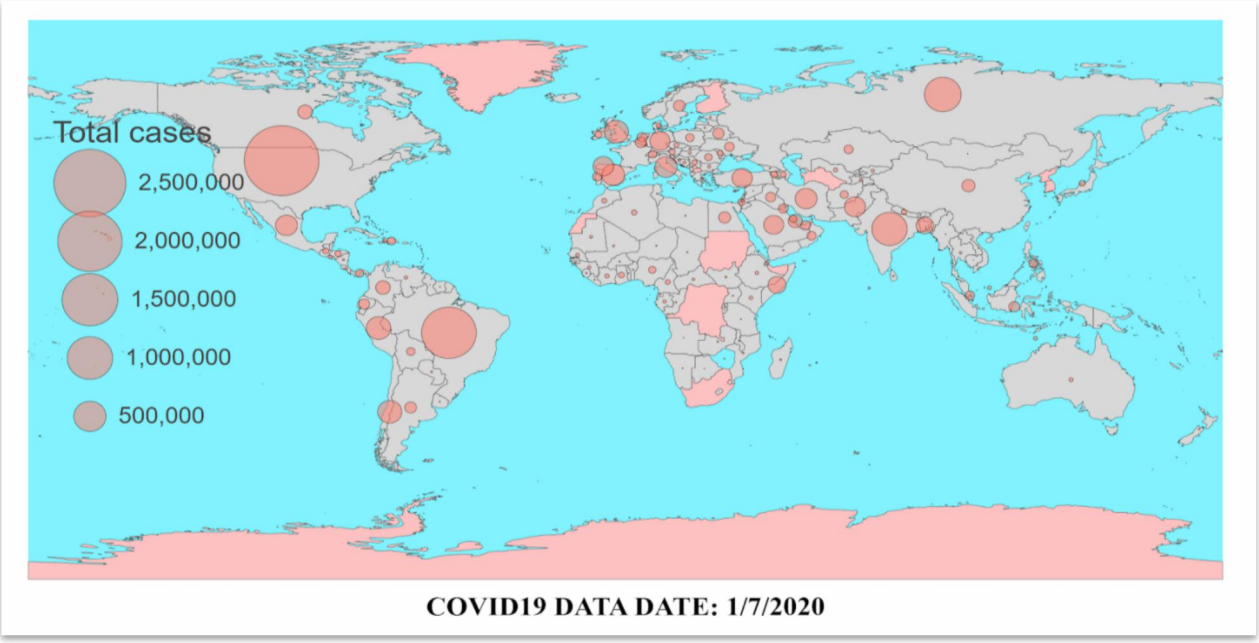
Class: CC03

Lecturer: Mr. Tran Tuan Anh

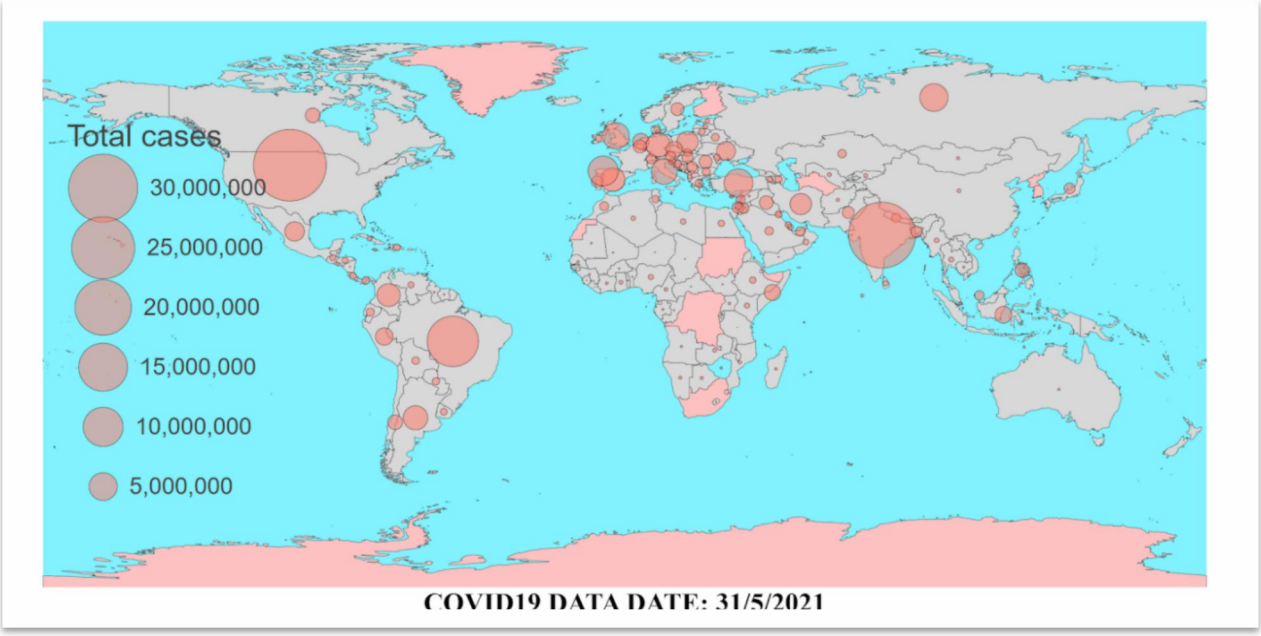
TEAM MEMBERS

Members	ID	Tasks	Effort Percentage
Nguyễn Thanh Duy Ân	2052858	Data analysis	25%
Nguyễn Đình Gia Lập	2052813	Data collection - Data analysis	25%
Võ Duy Hùng	2053625	Model prediction	25%
Lê Đức Toàn	2014771	Model analysis	25%

General Global Situation



COVID19 DATA DATE: 1/7/2020



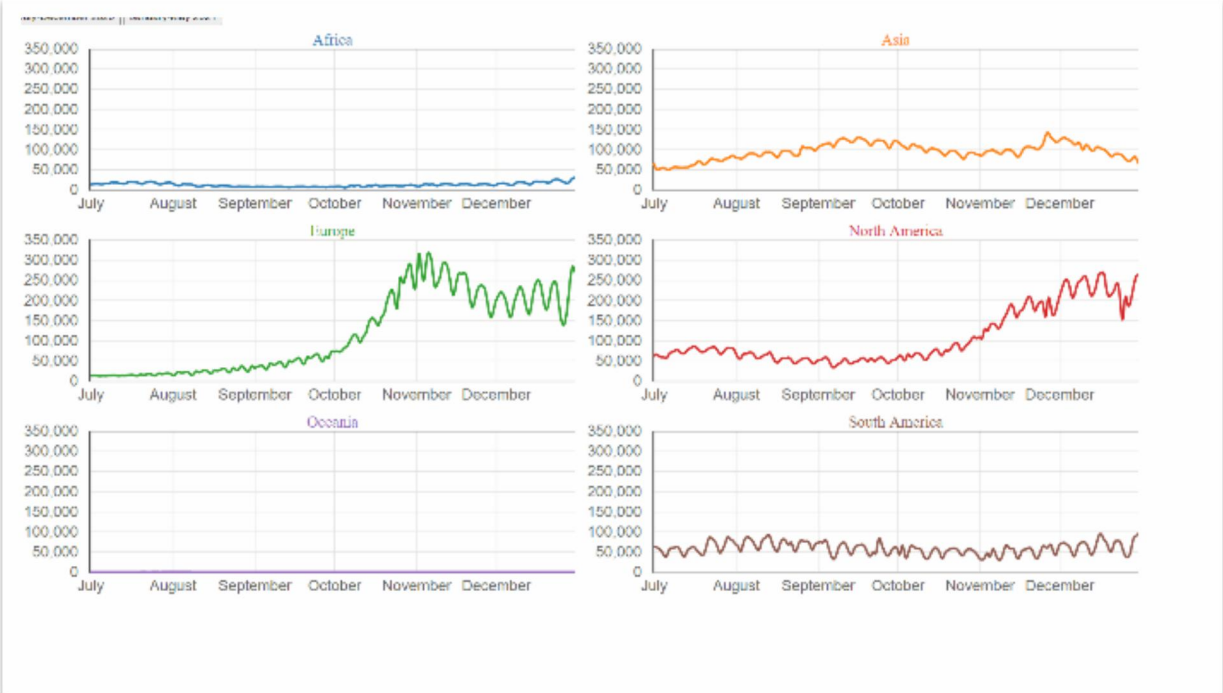
COVID19 DATA DATE: 31/5/2021

Over the period depicted, the number of COVID-19 cases worldwide increased dramatically. The center of the epidemic continued to be the United States of America, Brazil, and India. The COVID-19 infections ratio was more concentrated in European countries than before, while there seemed to be just minor changes on other continents.

Please **zoom in** and **hover over the regions** to have a better view of the data: https://munpro2002.github.io/COVID19-Data-Visualization/

I. Data Analysis

1. New COVID-19 Cases And Deaths In Six Regions From July To December 2020



The charts give information about new COVID-19 cases and deaths respectively in six regions between July and December 2020.

According to the chart table, there are similarities in the trends of victims and demises which are newly investigated. On the whole, some regions match others' tendency: Europe and North America appear to grow day by day; Asia shares the same propensity with South America, while Africa and Oceania have the earmarks of constant.

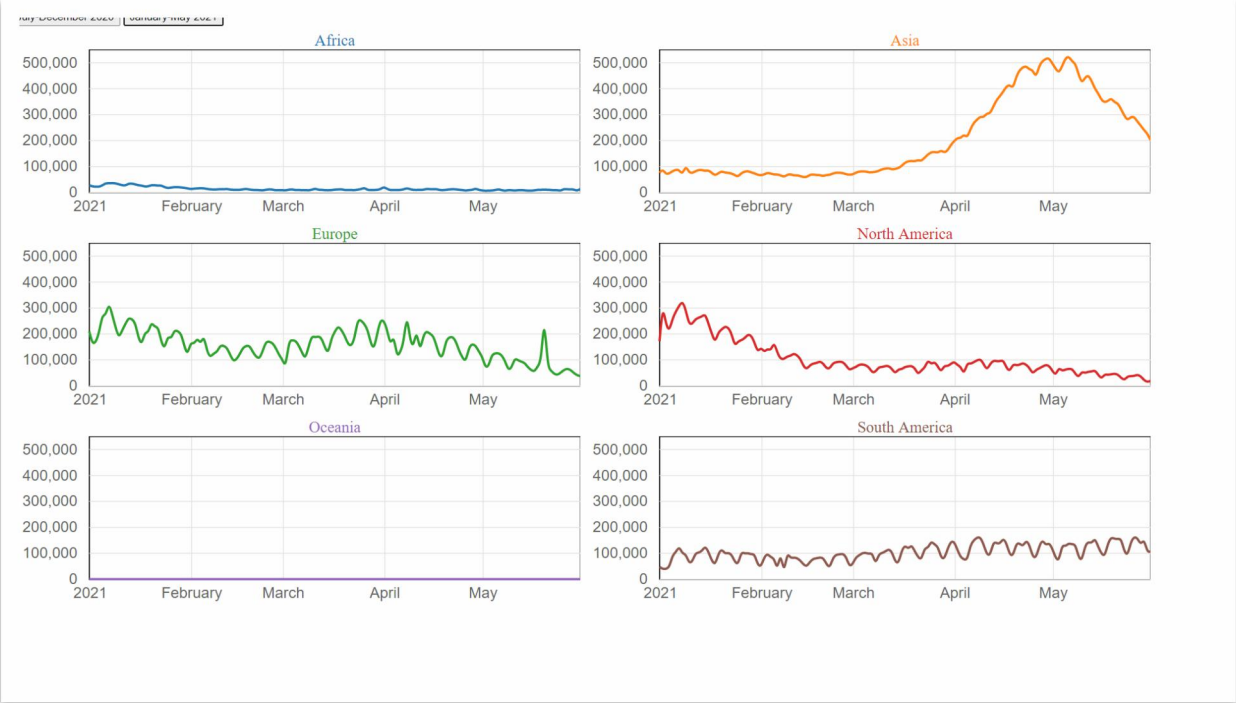
In July, North America had confirmed around 60000 new cases, a huge number when compared to Europe's which amounted to almost 20000 new cases. In the next three months, there were slight changes in the number of infected people found. Up to the middle of October, drastically the new-victim rate of Europe rose, peaking at above 300000 new patients in November and fluctuating till the end of 2020. Likewise, North America tended to change as the same as Europe, but less steep with a peak at about 250000 in December. When comparing the new death rate in Europe and North America, the graphs demonstrated an analogous trend with two peaks at 6000 and 5000 respectively.

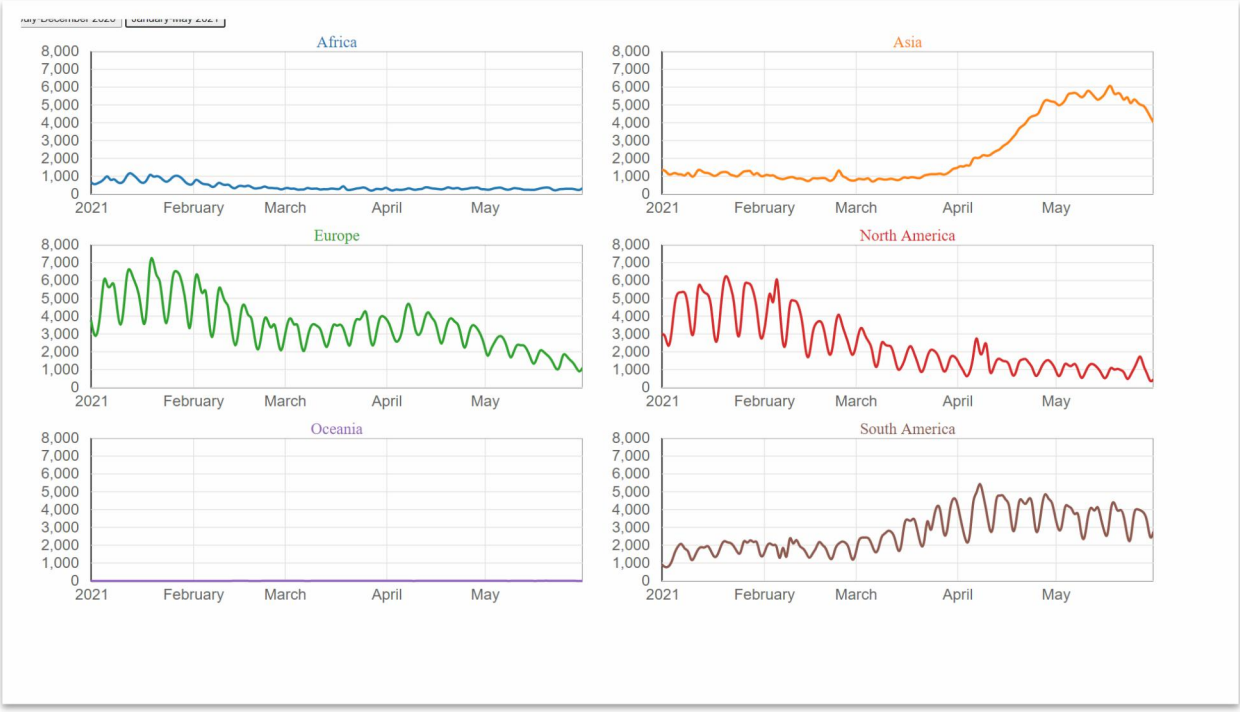
Both Asia and South America started with 50000 new infected cases in July, but Asia gradually increased in the following months while South America remained at the same pace. Asia reached the highest number of 150000 new patients and that number in South America was 100000 at nearly the same point of time, which is December. With the new-death rate, Asia varied from 1000 to 2000 over that period of time. By contrast, South America illustrated a stark trend with three remarkable peaks at about 4500 and one at 3500, then decreased in the rest of 2020.

Such an elated signal that both Africa and Oceania displayed when the rate, including new cases and new deaths, was very low, especially Oceania with a nearly zero number.

All in all, there are changes, big and small, among six regions about new-case and new-death rate. As the center of the world with many populous countries, Europe and America undoubtedly had difficulty handling the pandemic. Asia had shown an advance in pandemic prevention work when the rate decreased significantly after a blast in the early of 2020. Meanwhile, Africa and Oceania both showed a steady 'defense', confronting the virus.

2. New COVID-19 Cases And Deaths In Six Regions From January To May 2021





The line graphs illustrate changes in the Covid-19 new infections and new deaths from January to May 2021.

Over the period shown, there are trend parallels between new cases and new deaths per day in six regions. While the number of new cases and deaths grew in Asia and South America, the converse was true in Europe and North America, Oceania and Africa appear to have stabilized over the last five months.

In January, about 100000 new cases were reported in Asia, more than twice the amount reported in South America, which was around 50000. Over the next five months, the number of new cases in Asia gradually grew, peaking at over 500000 in May, but the number of new cases in South America fluctuated throughout the period, peaking at around 100000 per day in May. When comparing new deaths data to new cases data, there were minor differences in trending form. In Asia, 1000 people died each day at first, peaking at around 6000 people per day in May, whereas in South America, 1000 people died per day at first, peaking at over 4000, then somewhat decreasing to over 2000 people per day.

Europe and North America both reported nearly the same number of new cases of COVID-19 in January, over 200000 people per day then showed variability in number over the next five months, both peaking at 300000 cases and reporting around 50000 people per day in May. In comparison to new cases data, new deaths data from these two regions showed the same fluctuation trends. Europe reached a peak of 7000 cases in May and ended with 1000 cases, while North America reached a peak of 6000 cases in May and ended with 1000 cases.

In addition, while there were few infected cases in Oceania and Africa, there were few changes in COVID-19 data. In fact, there haven't been any infected cases in Oceania since 2020.

For further information, RNA vaccines were the first COVID-19 vaccines to be authorized in the United Kingdom, the United States, and the European Union in January 2021, but it only widely used worldwide since March 2021, though, Asia still gradually increased to its peak of new infection cases and deaths before starting to decrease in June. South America data was irregular despite the appearance of the COVID-19 vaccine. Europe and America, fortunately, witnessed a decrease in data.

II. Model Prediction

In machine learning, there's something called the "No Free Lunch" theorem. In a nutshell, it states that no one machine learning algorithm works best for every problem, and it's especially relevant for supervised learning (i.e. predictive modeling).

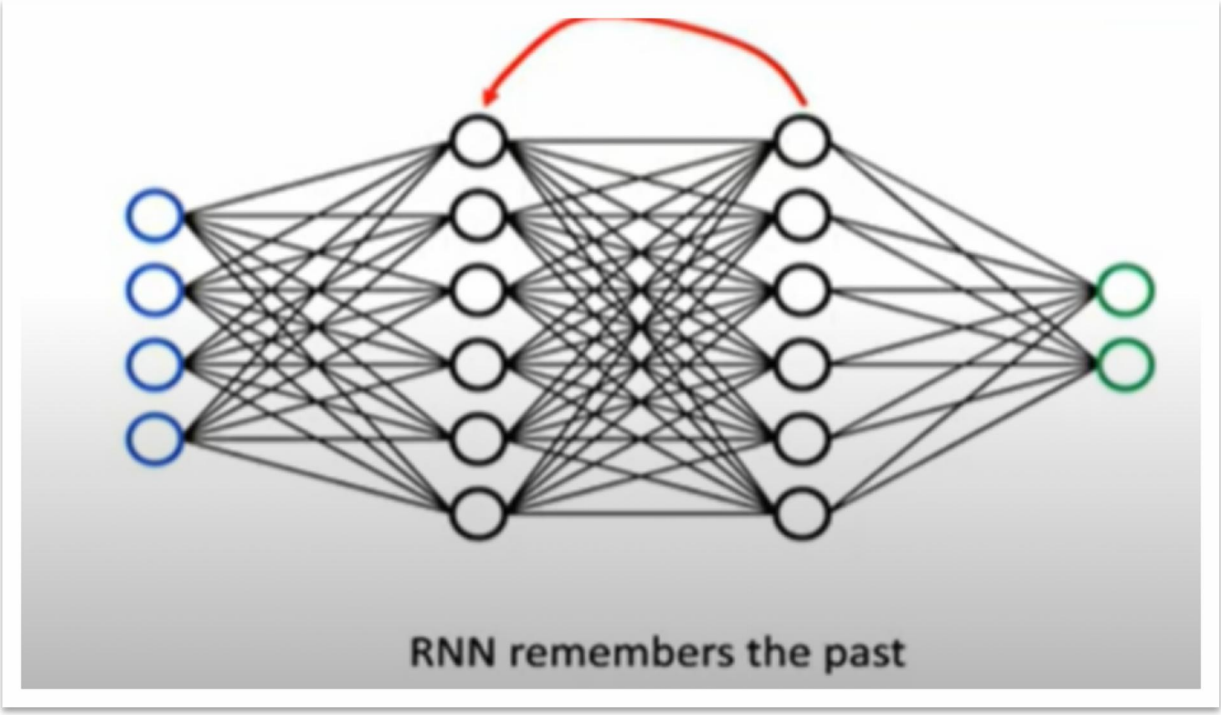
As a result, you should try many different algorithms for your problem, while using a hold-out "test set" of data to evaluate performance and select the winner. Of course, the algorithms you try must be appropriate for your problem, which is where picking the right machine learning task comes in.

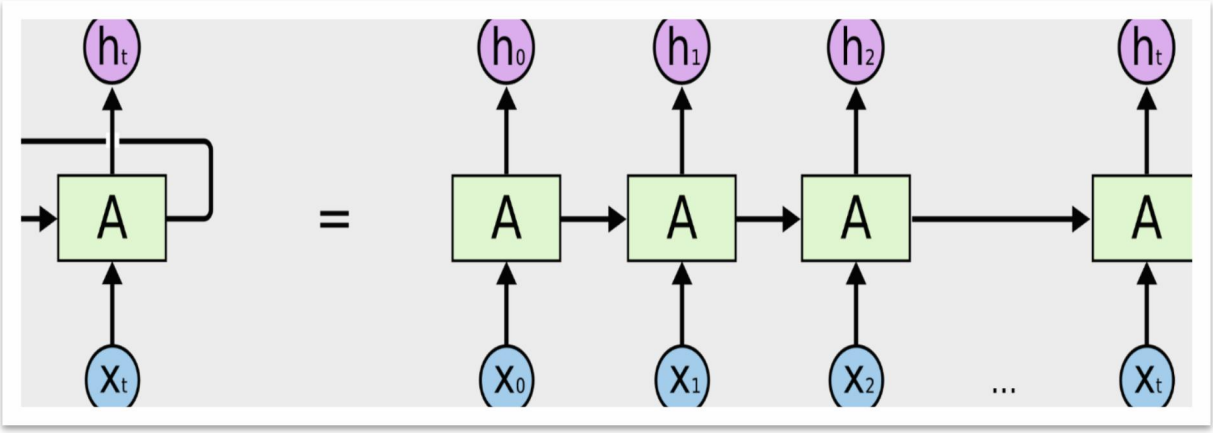
For the aforementioned requirement, we will try **RNN (Recurrent Neural Networks)-LSTM (Long-Short-Term Memory)** for **Covid-19 Prediction.**

1. Theory summary:

A. Introduction to RNN (Recurrent Neural Networks):

RNNs are designed to make use of sequential information.

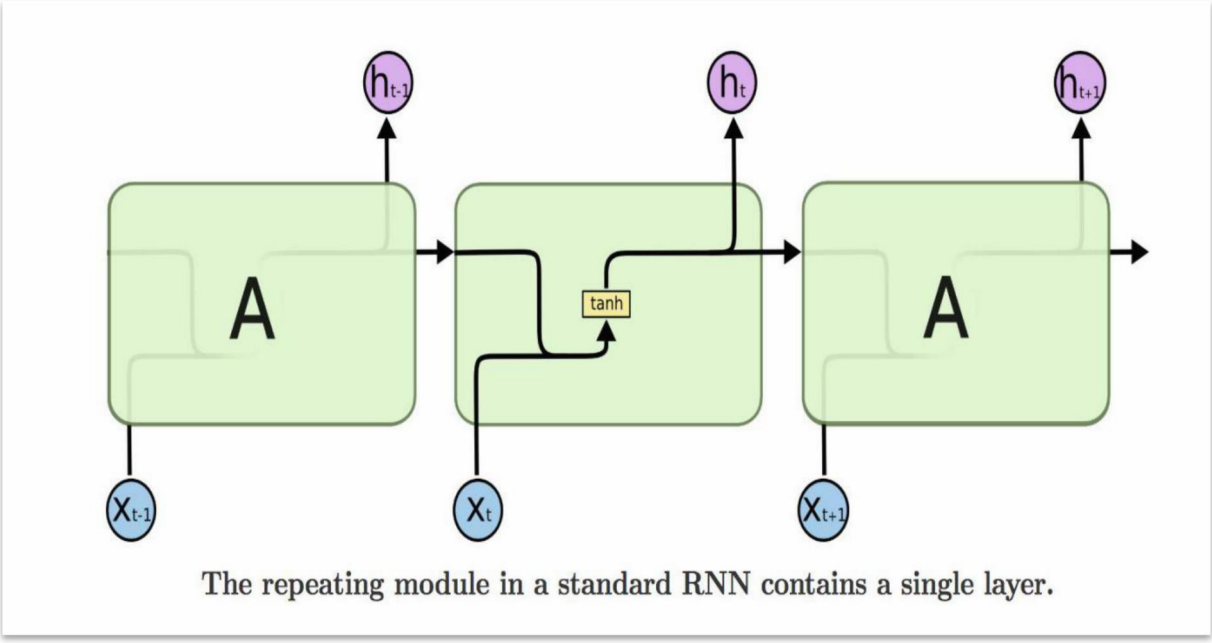




RNNs are “unrolled” programmatically during training and prediction.

RNN is like multiple copies of the same network where each copy passes to the other. The list-like structure makes them appropriate for sequential data.

There are many types of RNN: one to one, one to many, many to one, many to many.



The repeating module in a standard RNN contains a single layer.

The problem with RNN:

RNNs are good with short sequences. However, for long sequences, RNNs seem to fail. Therefore, we use LSTM, a special kind of RNN.

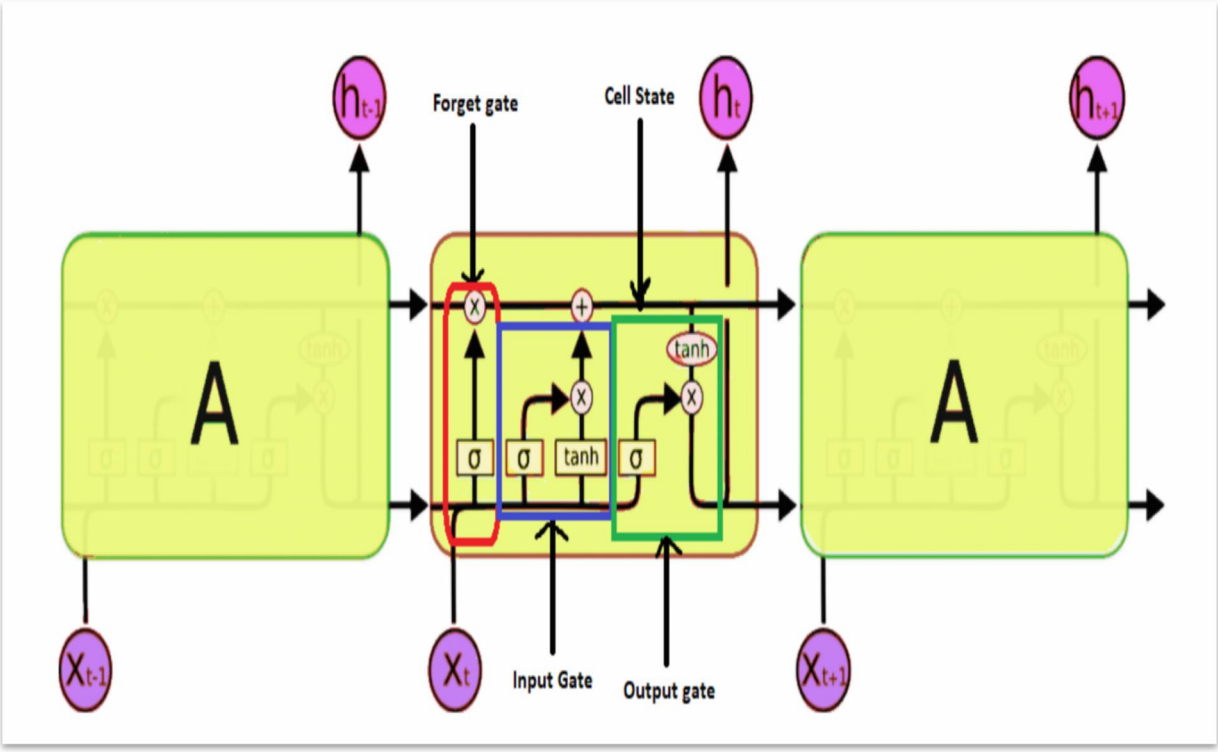
B. Introduction to LSTM (Long Short-Term Memory):

LSTM is a **special kind** of RNN.

Designed to overcome limitations of RNNs such as:

- Gradient vanishing and exploding.
- Complex training.
- Difficulty to process very long sequences.

Remembering information for long periods of time is intrinsic to LSTM.



Cell State:

Gates let information through the cell state.

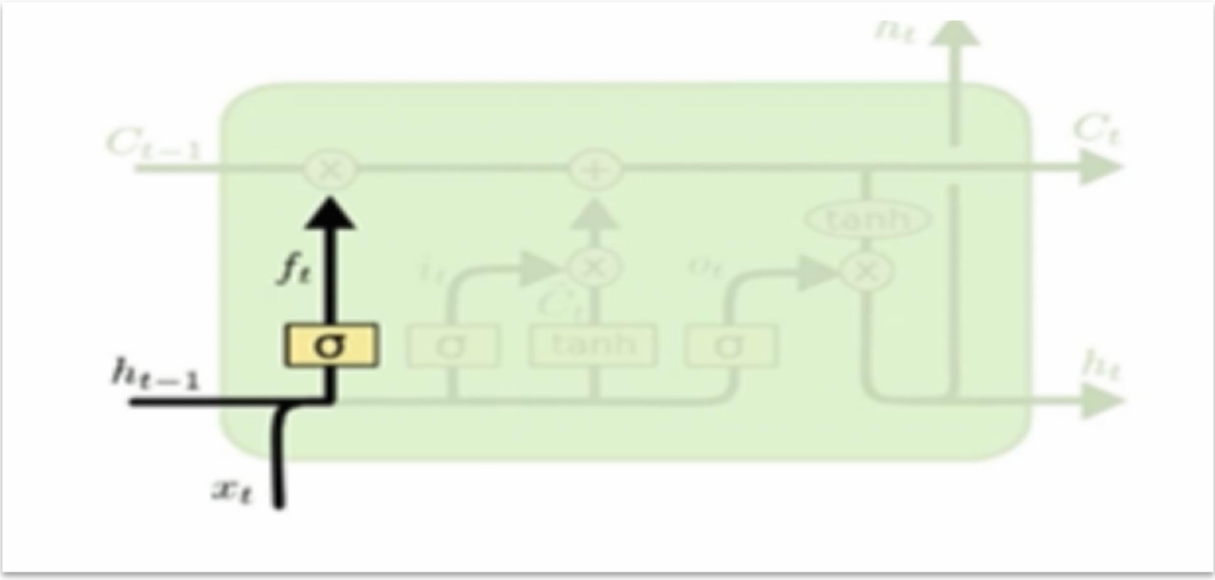
Information flows through the line path.

Sigmoid: can output 0 to 1, it can be used to forget or remember the information.

Tanh: to overcome the vanishing gradient problem. Tanh's second derivative can sustain for a long range before going to zero.

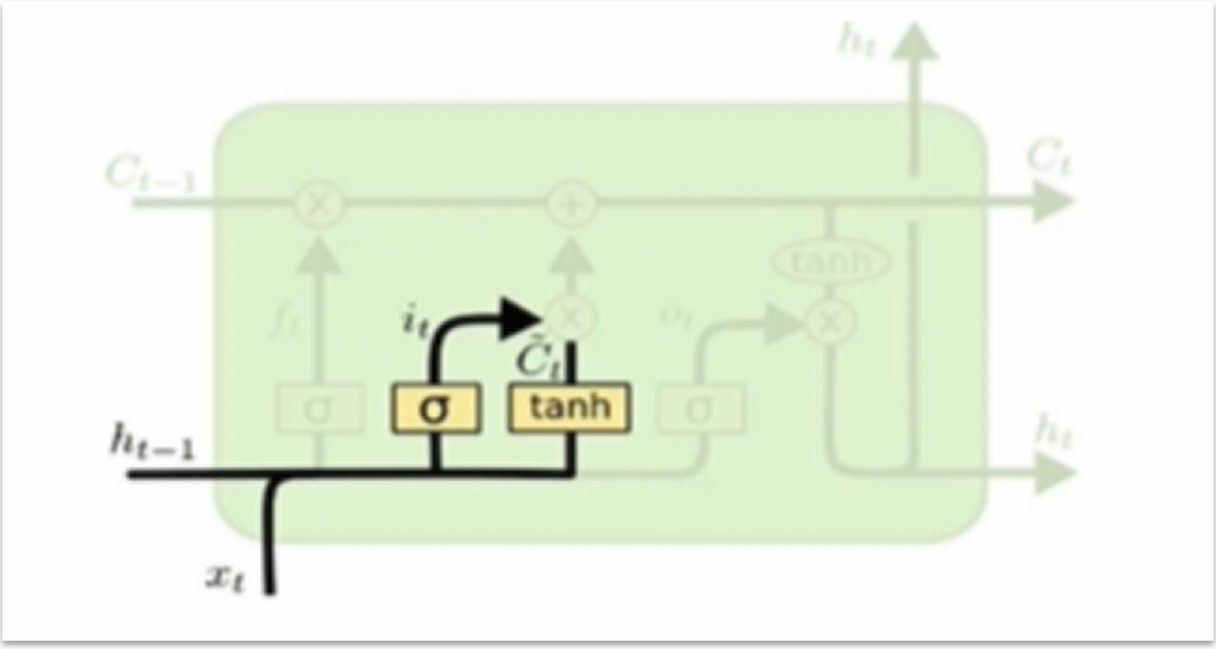
Forget gate:

How much of the past to forget:



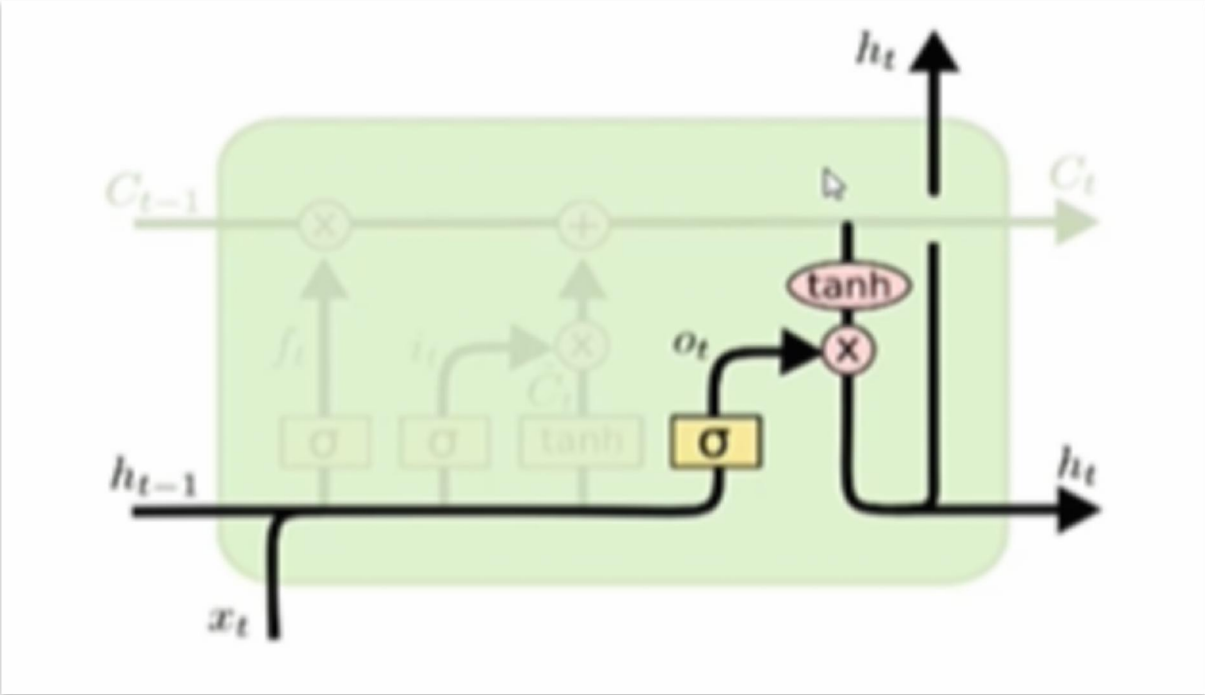
Outputs a number between 0 and 1 for each number in the cell state. 0 to completely forget and 1 to keep all information.

Input gate:



What new information will be stored in the cell state.

Output gate:
Decide what part of the current cell makes the output.



2. Model Forecasting:

Because the source code and source data are too long that I will send through our zip files.

My source code for predicting **has been published at**:

https://github.com/HungVoCs47/COVID-19-DATA-PREDICTION/blob/main/Model%20Prediction/prediction.py

My data for model prediction **has been published at:**

https://github.com/HungVoCs47/COVID-19-DATA-PREDICTION/blob/main/Data/Covid-19-data.csv

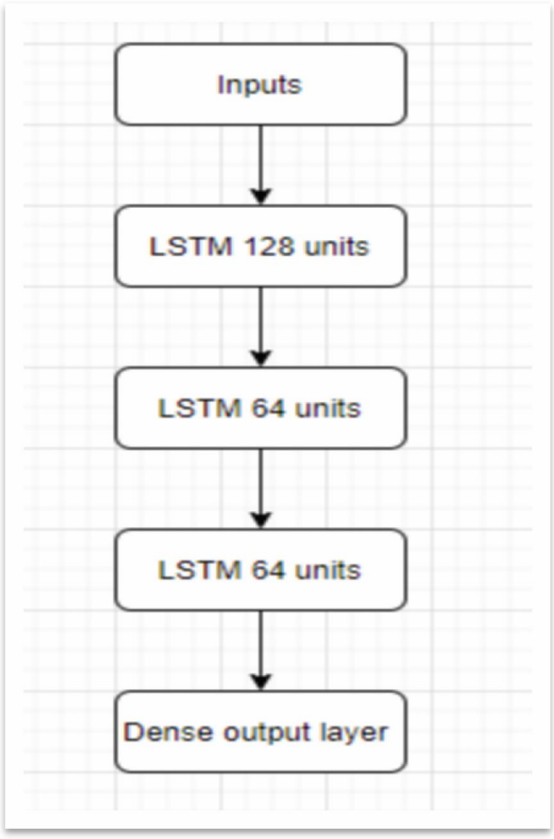
Or if you want to see **the raw data:**

https://raw.githubusercontent.com/HungVoCs47/COVID-19-DATA-PREDICTION/main/Data/Covid-19-data.csv

This model will predict the confirmed cases in the next one week, starting from **26/06/2021** to **03/07/2021**.

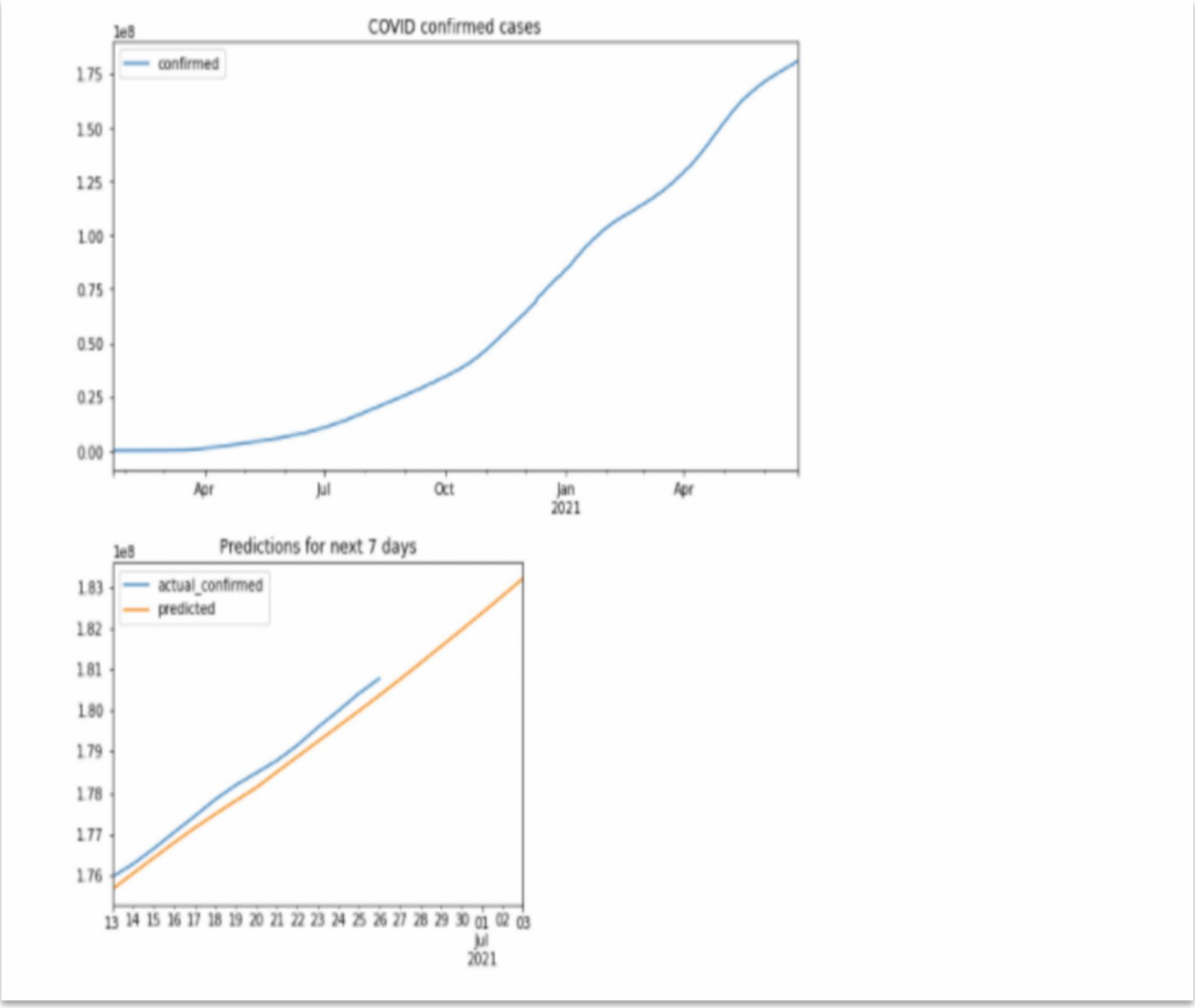
A. Model Overview:

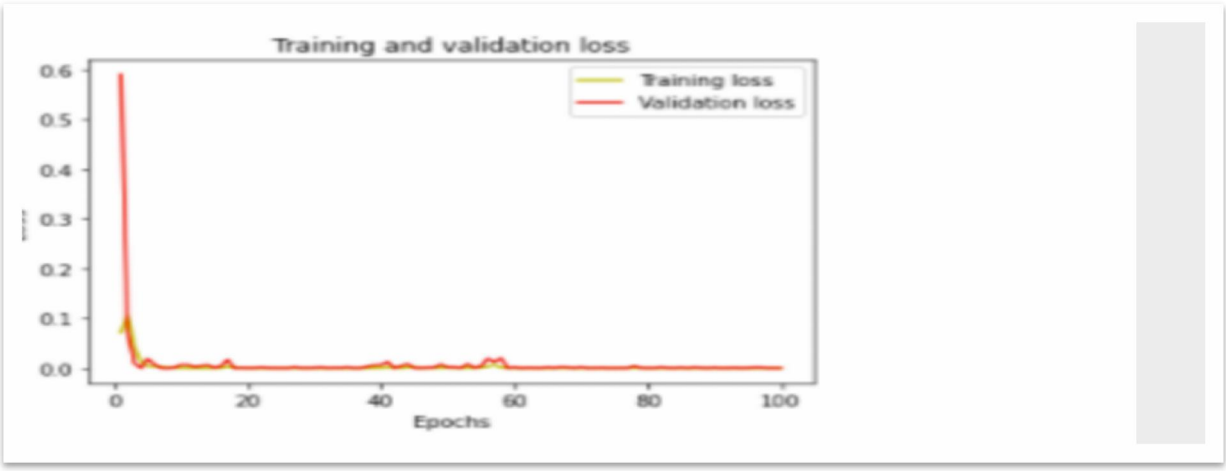
```
Total days in the dataset 522
Total number of samples in the original training data =  508
Total number of samples in the generated data =  501
Total number of samples in the original training data =  14
Total number of samples in the generated data =  7
Model: "sequential_6"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm_12 (LSTM)               (None, 7, 128)            66560
_________________________________________________________________
lstm_13 (LSTM)               (None, 64)                49408
_________________________________________________________________
dense_12 (Dense)             (None, 64)                4160
_________________________________________________________________
dense_13 (Dense)             (None, 1)                 65
=================================================================
```



B. Model results:

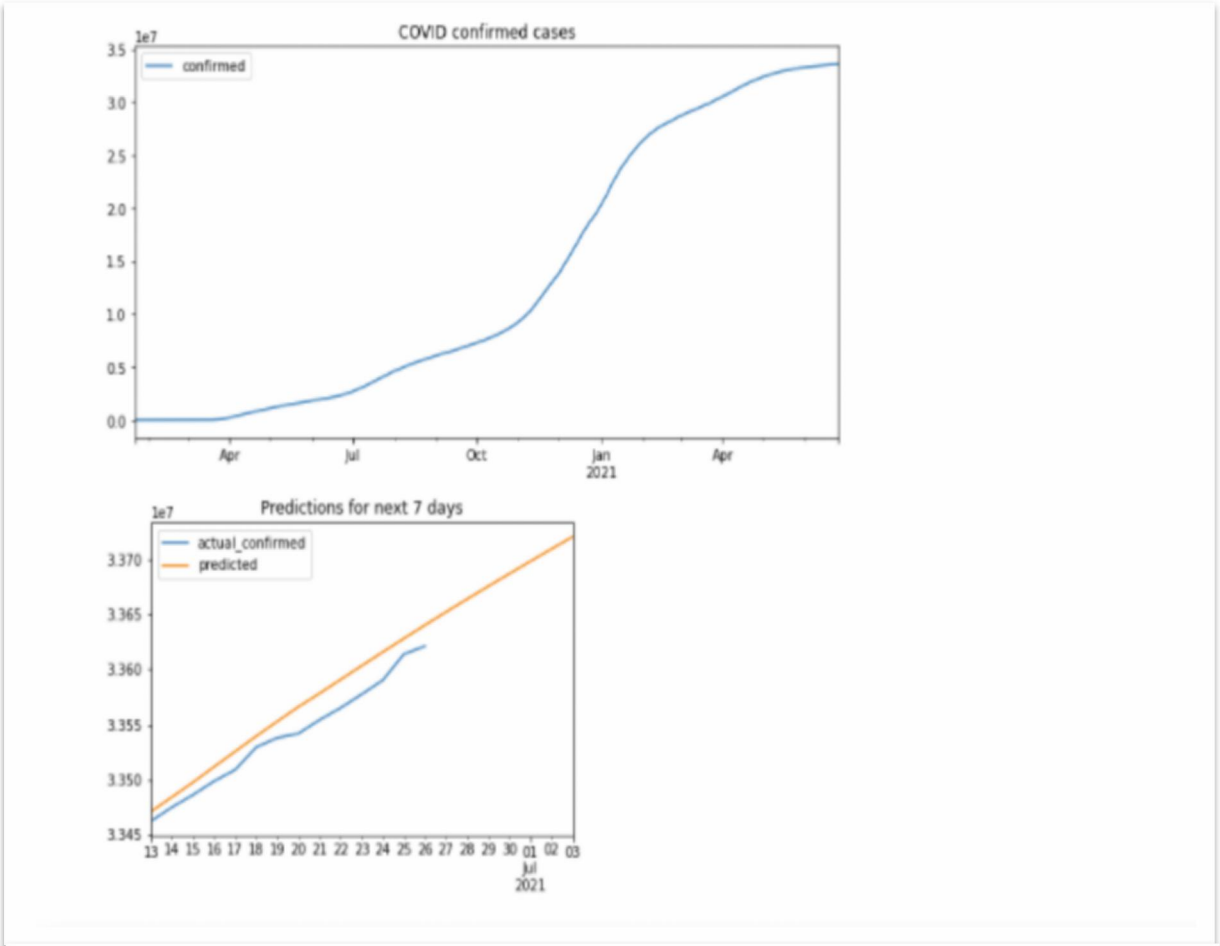
- **Global:**

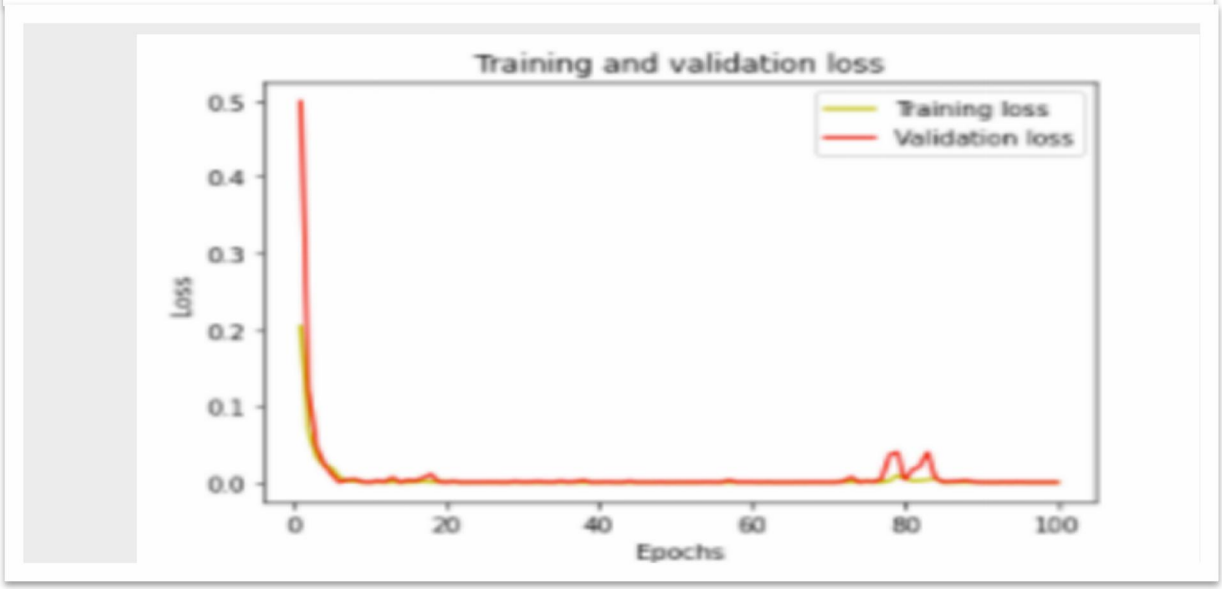




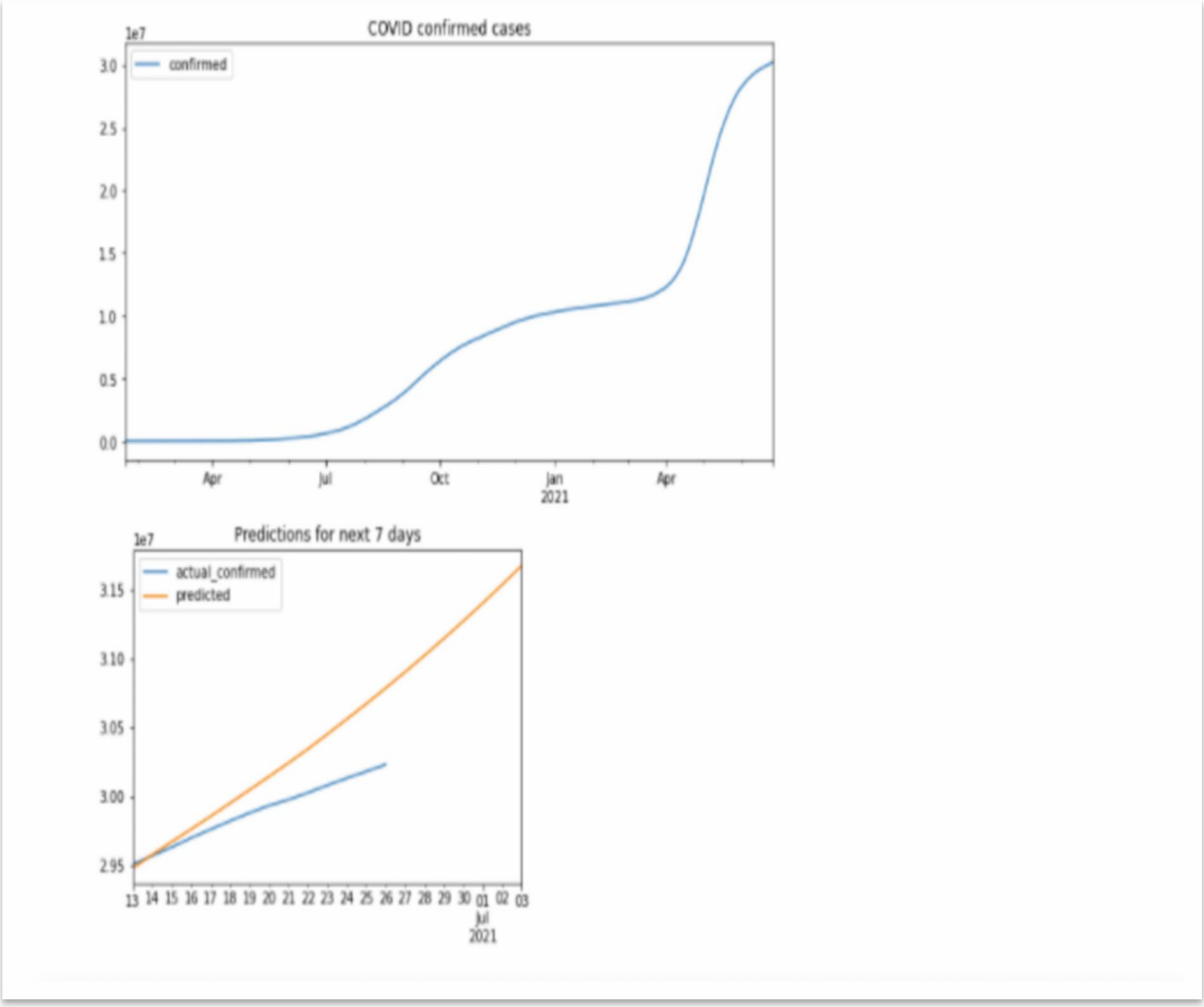
❖ 3 countries with the highest number of infections: USA, India, Brazil:

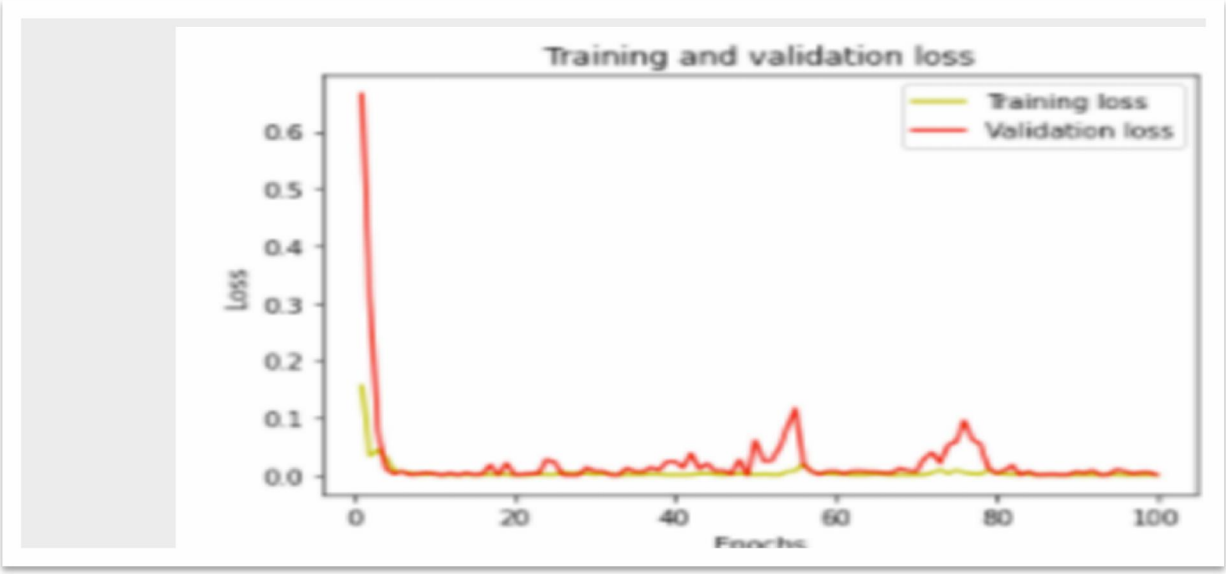
- **USA:**



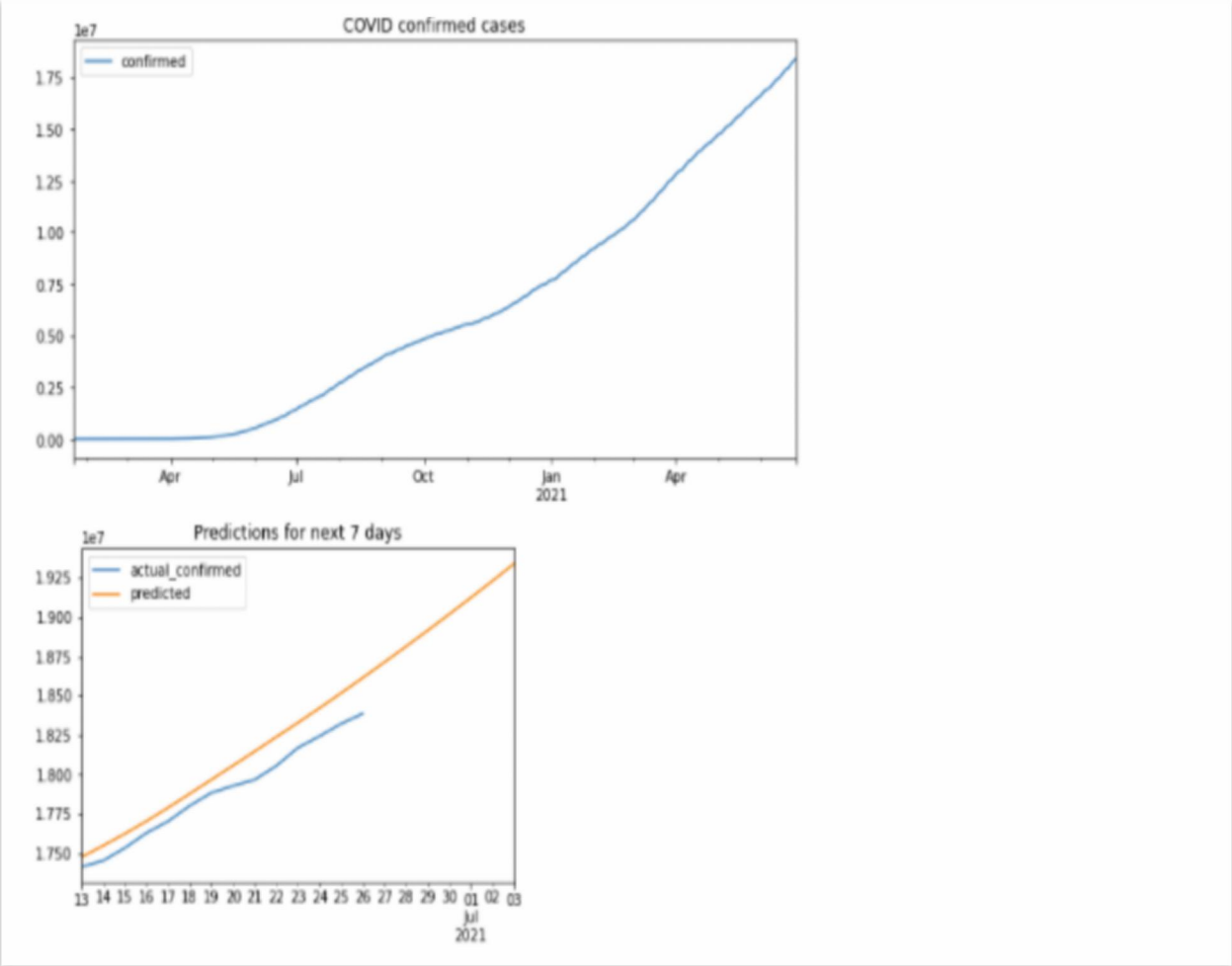


- **India:**


COVID confirmed cases
1e7
confirmed
3.0
2.5
2.0
1.5
1.0
0.5
0.0
Apr
Jul
Oct
Jan
2021
Apr
Predictions for next 7 days
1e7
actual_confirmed
predicted
3.15
3.10
3.05
3.00
2.95
13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 01 02 03
Jul
2021


Training and validation loss
Training loss
Validation loss
Loss
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0
20
40
60
80
100
Epochs

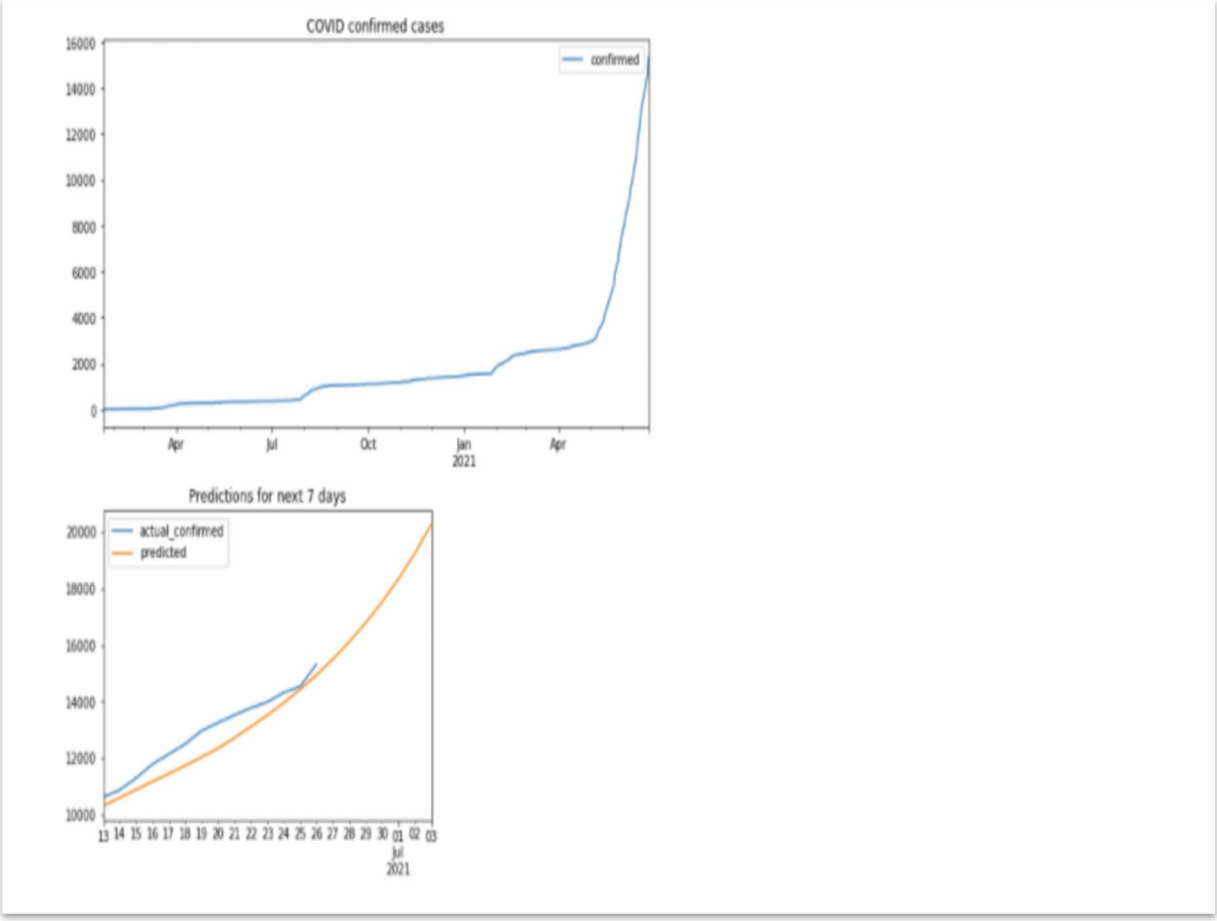
- **Brazil:**

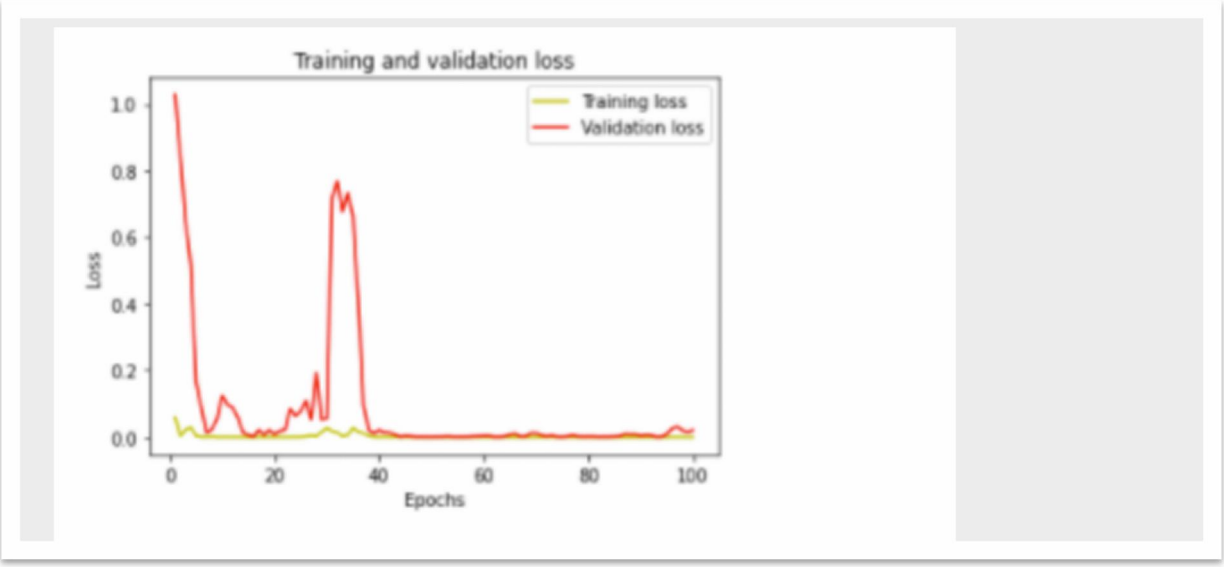

COVID confirmed cases
1e7
confirmed
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
Apr
Jul
Oct
Jan
2021
Apr
Predictions for next 7 days
1e7
actual_confirmed
predicted
1.925
1.900
1.875
1.850
1.825
1.800
1.775
1.750
13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 01 02 03
Jul
2021


Training and validation loss
Training loss
Validation loss
Loss
0.5
0.4
0.3
0.2
0.1
0.0
0
20
40
60
80
100

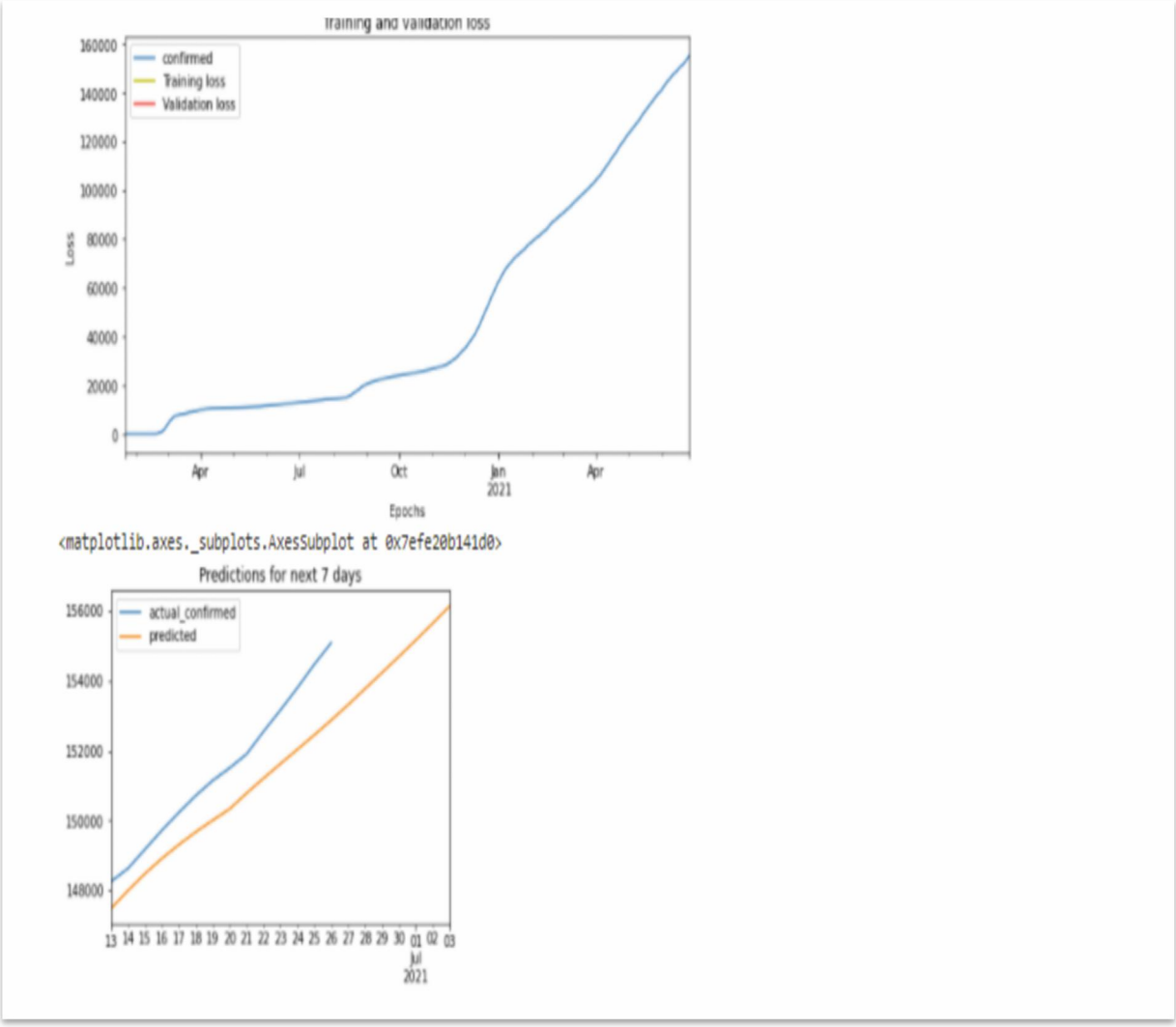
❖ Asian Countries:

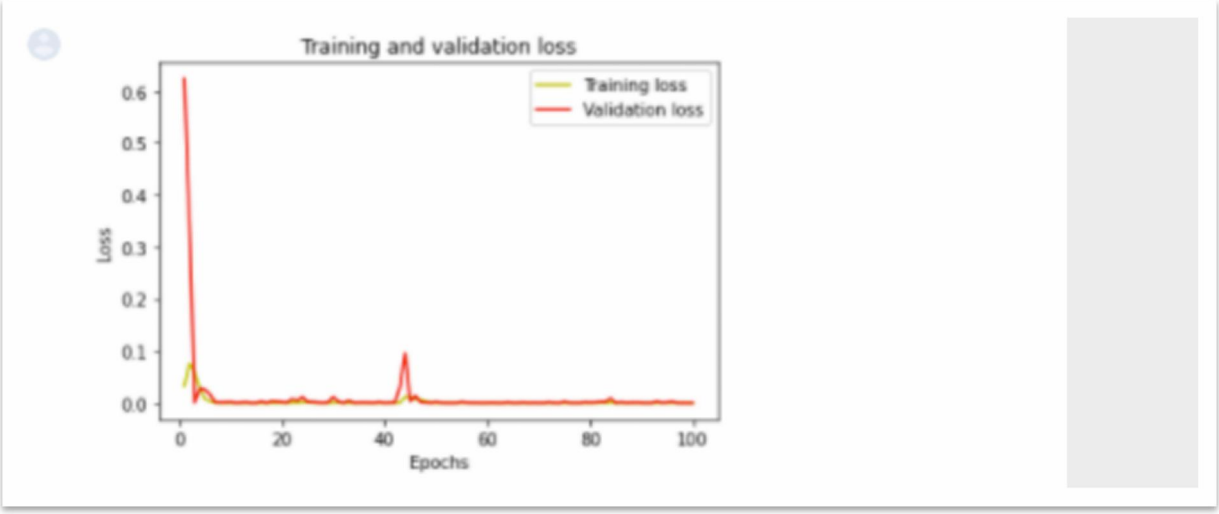
- **Viet Nam:**



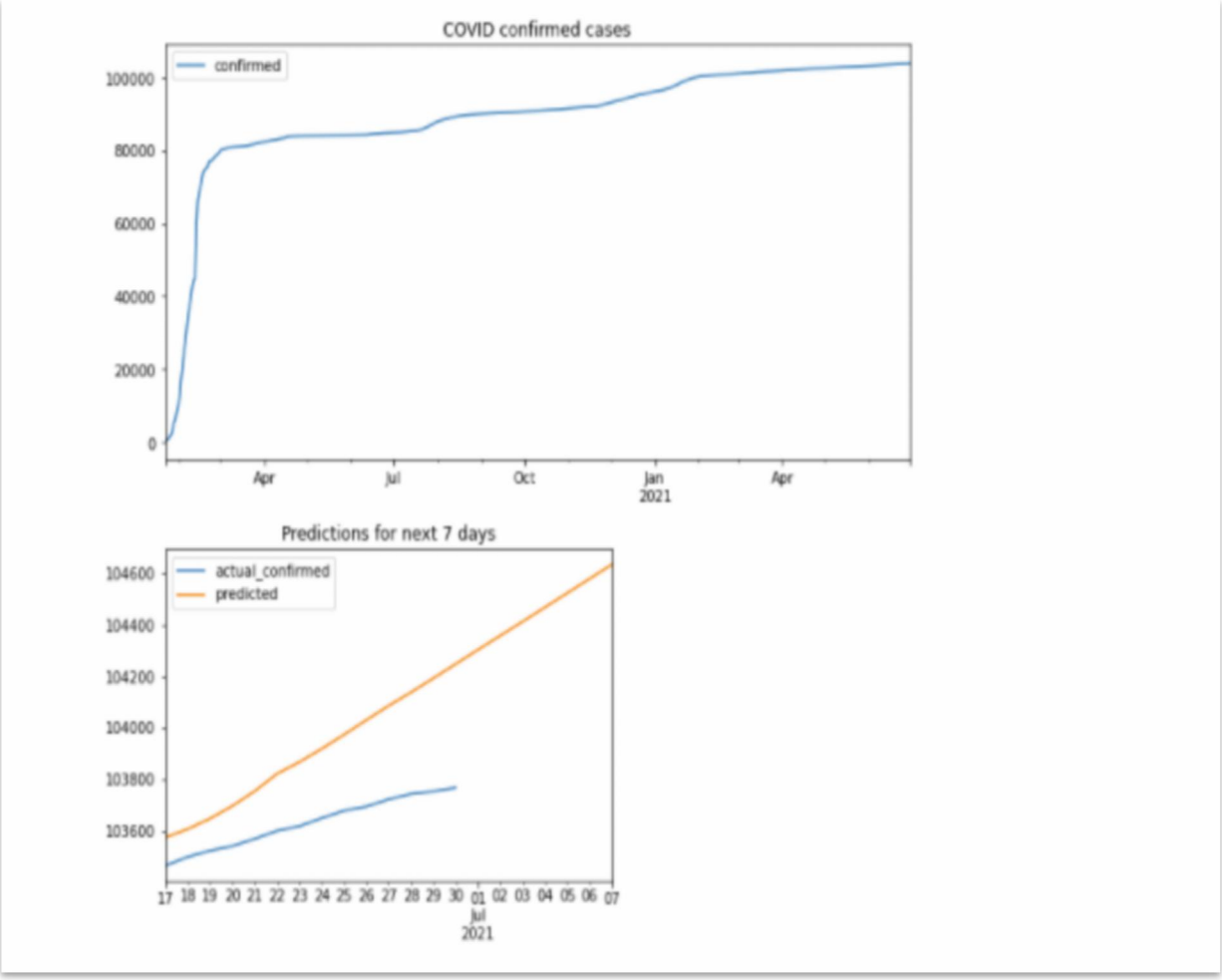


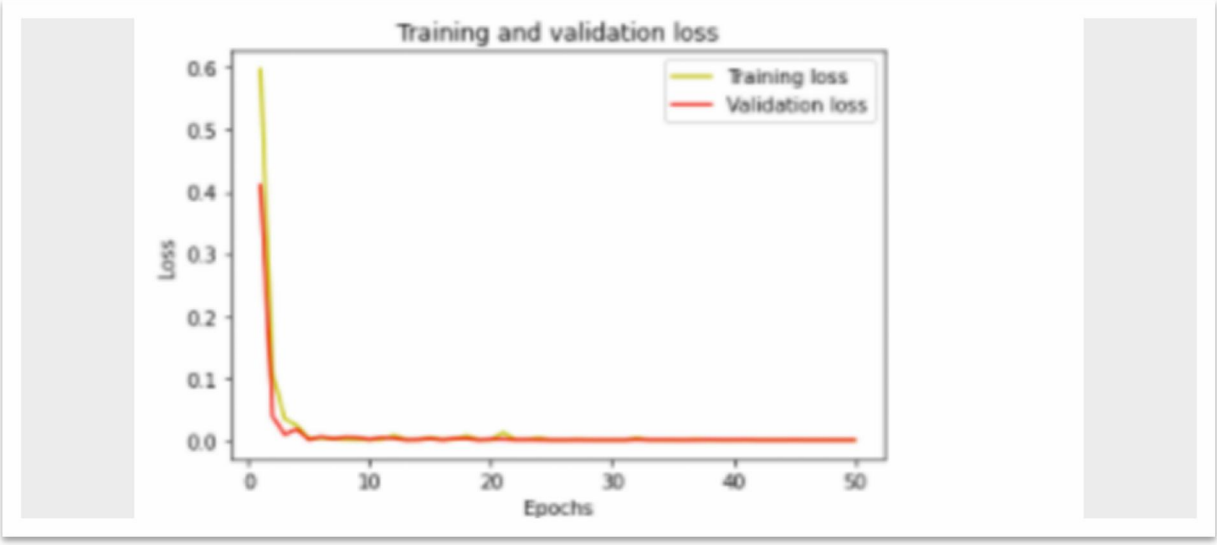
- **South Korea:**


Training and validation loss
confirmed
Training loss
Validation loss
Loss
Epochs
Apr
Jul
Oct
Jan
2021
Apr
<matplotlib.axes._subplots.AxesSubplot at 0x7efe20b141d0>
Predictions for next 7 days
actual_confirmed
predicted
13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 01 02 03
Jul
2021


Training and validation loss
Training loss
Validation loss
Loss
Epochs

- **China:**


COVID confirmed cases
confirmed
100000
80000
60000
40000
20000
0
Apr
Jul
Oct
Jan
2021
Apr
Predictions for next 7 days
actual_confirmed
predicted
104600
104400
104200
104000
103800
103600
17 18 19 20 21 22 23 24 25 26 27 28 29 30 01 02 03 04 05 06 07
Jul
2021


Training and validation loss
Training loss
Validation loss
Loss
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0
10
20
30
40
50
Epochs

III. Model Analysis

1. Recurrent Neural Network – Long-Short-Term Memory Network:

A. Recurrent Neural Network

Advantages

- RNN can process inputs of any length.
- An RNN model is modeled to remember each information throughout the time which is very helpful in any time series predictor.
- Even if the input size is larger, the model size does not increase.
- The weights can be shared across the time steps.
- RNN can use their internal memory for processing the arbitrary series of inputs which is not the case with feedforward neural networks.

Disadvantages

- Due to its recurrent nature, the computation is slow.
- Training of RNN models can be difficult.
- If we are using relu or tanh as activation functions, it becomes very difficult to process sequences that are very long.
- Prone to problems such as exploding and gradient vanishing.

B. Long-Short Term Memory

LSTM networks are an extension of recurrent neural networks (RNNs) mainly introduced to handle situations where RNNs fail. It has been so designed that the vanishing gradient problem is almost completely removed, while the training model is left unaltered. Long time lags in certain problems are bridged using LSTMs where they also handle noise, distributed representations, and continuous values. With LSTMs, there is no need to keep a finite number of states from beforehand. LSTMs provide us with a large range of parameters such as learning rates, and input and output biases. Hence, no need for fine adjustments.

2. Training loss and validation loss:

As you can see in our model prediction section, we have graphs called Training and Validation loss. Here is some further information

Cross-validation: is a technique for evaluating ML models by training several ML models on subsets of the available input data and evaluating them on the complementary subset of the data. The data is split to train / validation / test sets.

Training Dataset: The sample of data used to fit the model. Training a model simply means learning (determining) good values for all the weights and the bias from labeled examples.

Validation Dataset: The sample of data used to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyperparameters. The evaluation becomes more biased as skill on the validation dataset is incorporated into the model configuration.

Loss: is the result of a bad prediction. A loss is a number indicating how bad the model's prediction was on a single example.

If the model's prediction is perfect, the loss is zero; otherwise, the loss is greater. The goal of training a model is to find a set of weights and biases that have low loss, on average, across all examples. Higher loss is the worse (bad prediction) for any model.

The loss is calculated on training and validation and its interpretation is how well the model is doing for these two sets. Unlike accuracy, a loss is not a percentage. It is a sum of the errors made for each example in training or validation sets.

In machine learning and deep learning there are basically three cases

- ***Overfitting***: training loss << validation loss

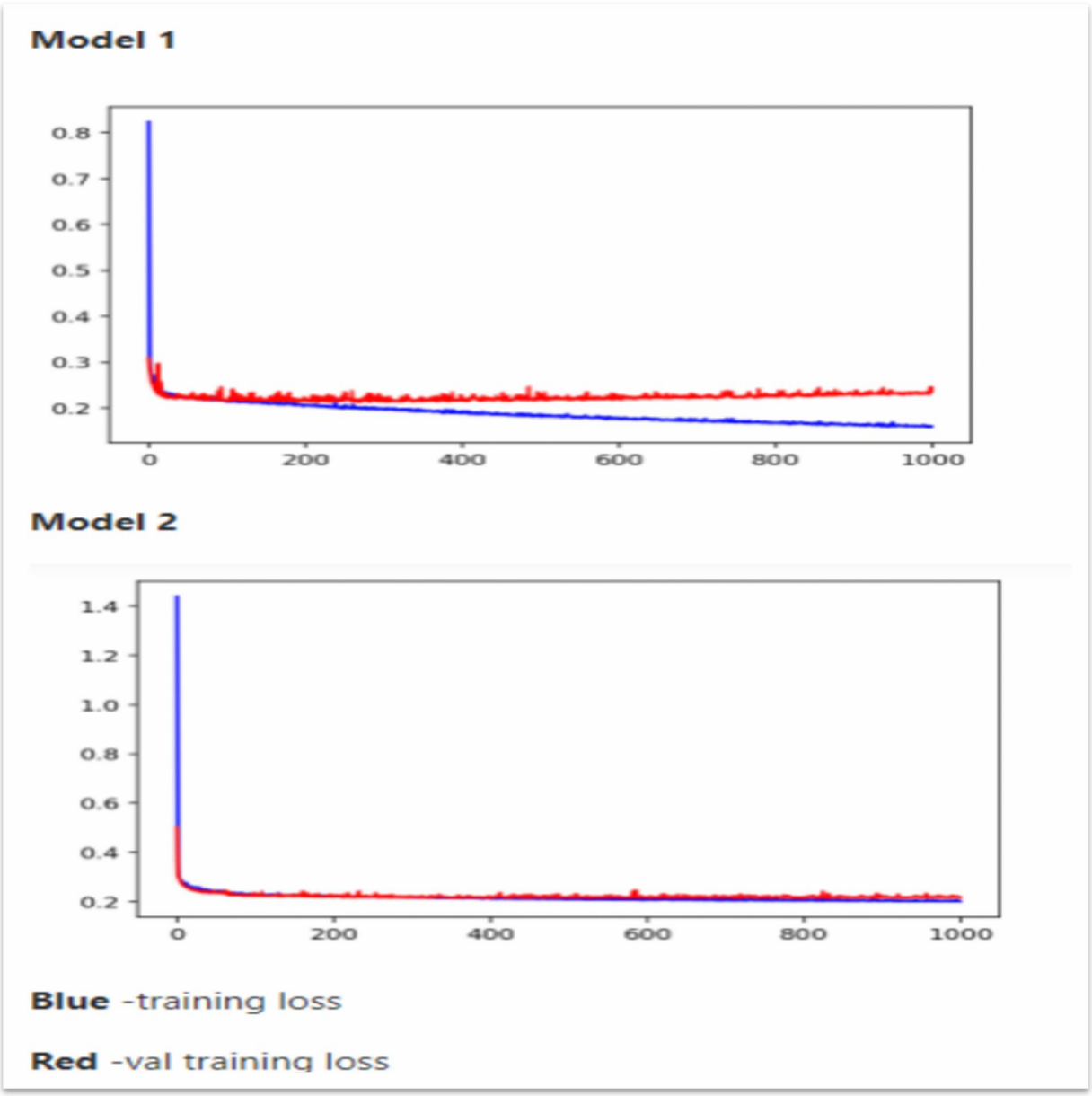
This means that your model is fitting very nicely the training data but not at all the validation data, in other words it's not generalizing correctly to unseen data

- ***Underfitting***: training loss >> validation loss

Refers to a model that can neither model the training data nor generalize to new data. An underfit machine learning model is not a suitable model and will be obvious as it will have poor performance on the training data

- ***Just right***: training loss ~ validation loss

If both values end up to be roughly the same and also if the values are converging (plot the loss over time) then chances are very high that you are doing it right.



Graph for model 1

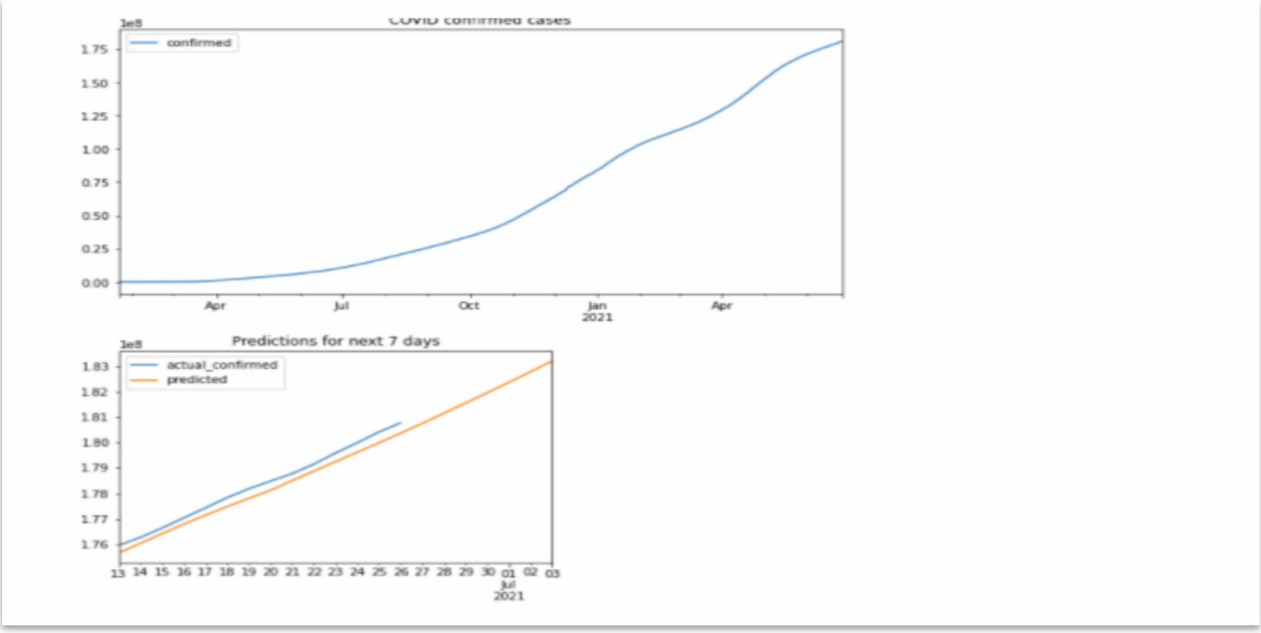
We notice that the training loss and validation loss aren't correlated. This means the as the training loss is decreasing, the validation loss remains the same of increases over the iterations.

Graph for model 2

In this case, there is clearly a health correlation between training loss and the validation loss. They both seem to reduce and stay at a constant value

3. Example in our Model Prediction:

Global:



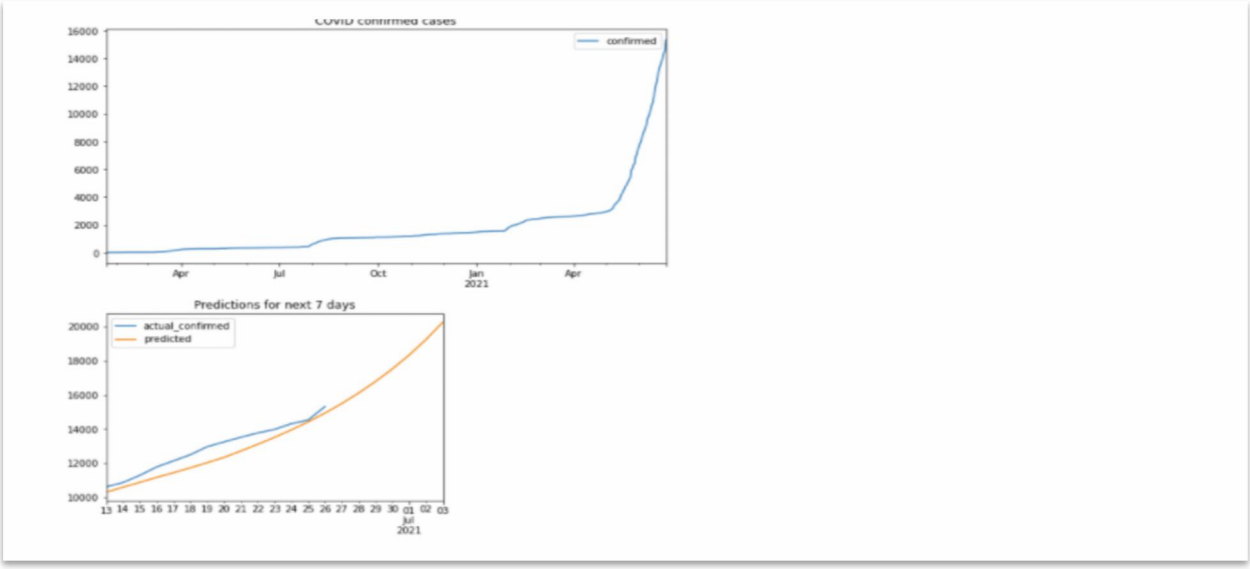
As you can see, on July 3, the global cases reach 184M as our model predicted it to be, not exactly the same but still pretty accurate.

Even though the validation loss sets tend to be higher at first but they still meet the same level at the end and the value approaches zero, which is said the model is doing its best to classify the input data and the output targets. This produces the almost accurate results as mentioned above.

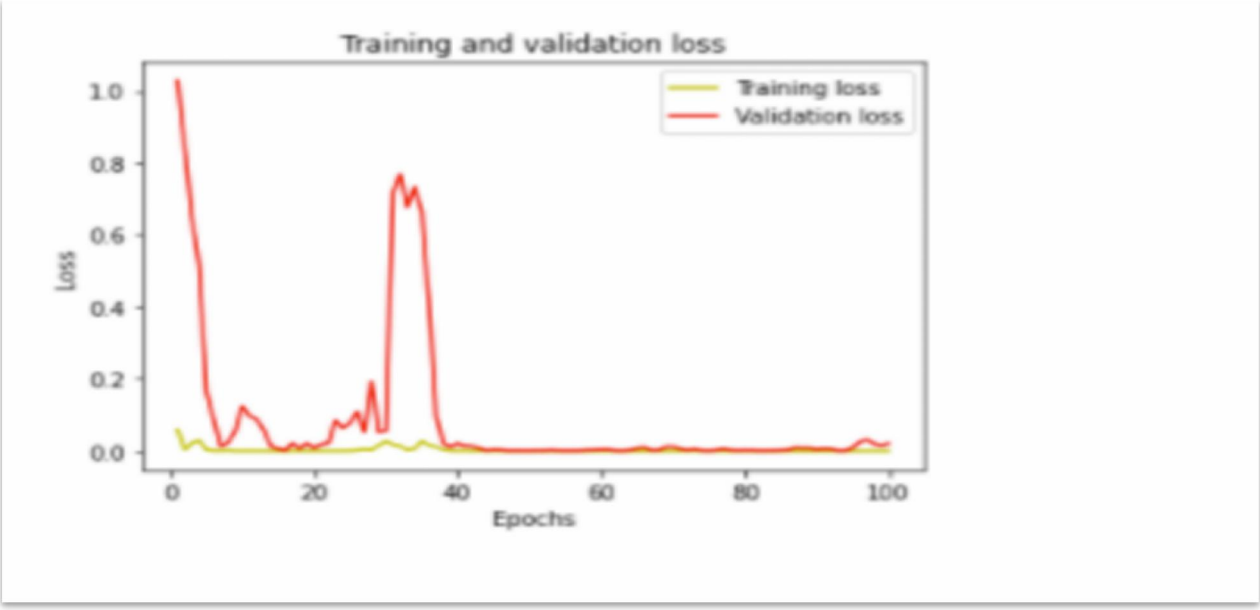
Results of specific countries from 5 continents:

❖ Asia:

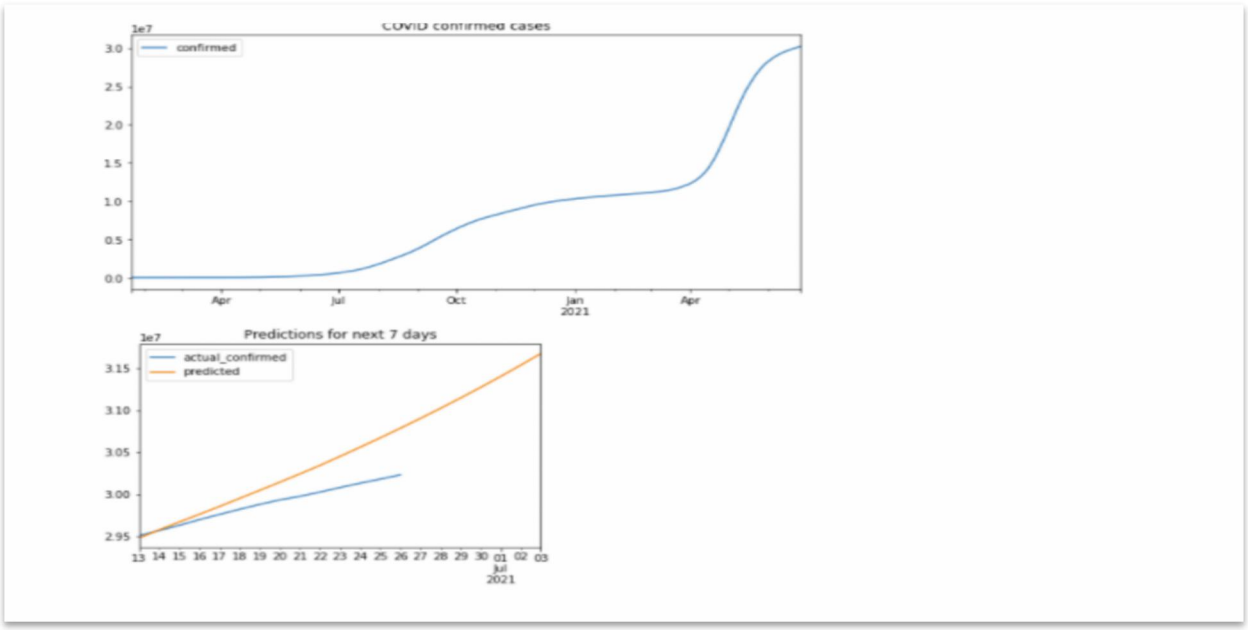
- **Vietnam**



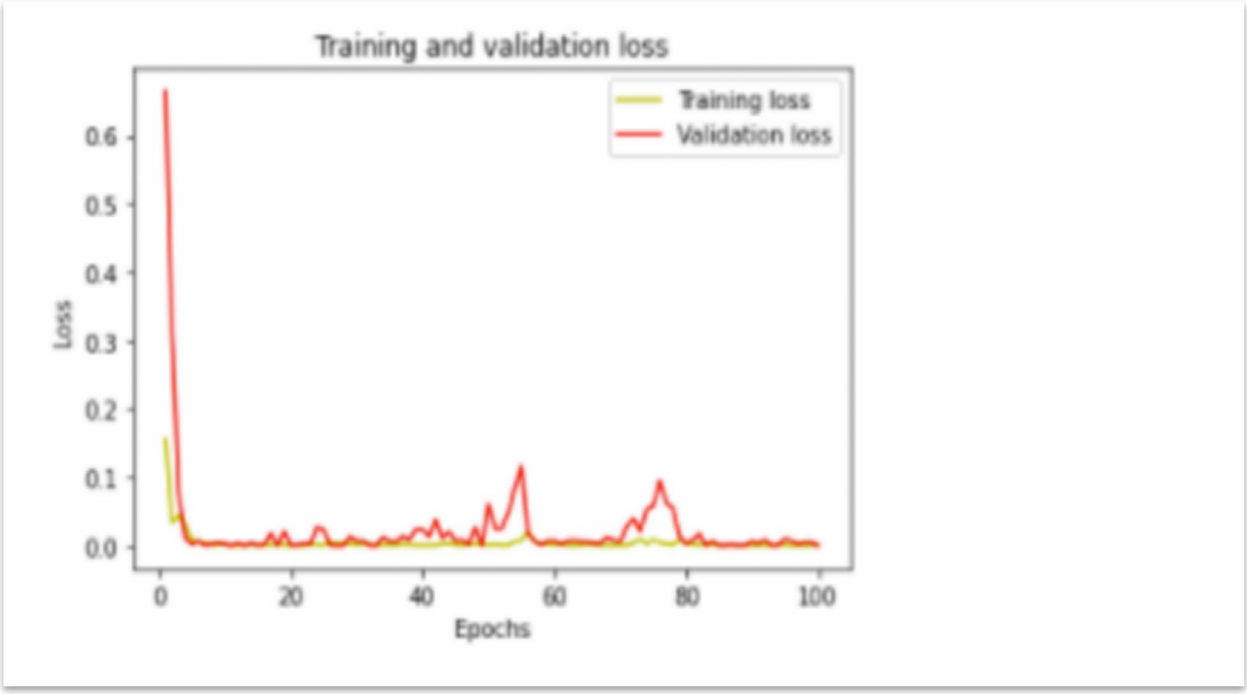
As you can see in the Vietnam predicted cases, the model reached about 20000 which is relatively close to the number 20261 cases of our country on July 3.
But in this graph the validation loss is much higher than the training loss at first, it's a sign that the model learns "superstitions" patterns that accidentally happened to be true in the training data aren't true in the validation data.



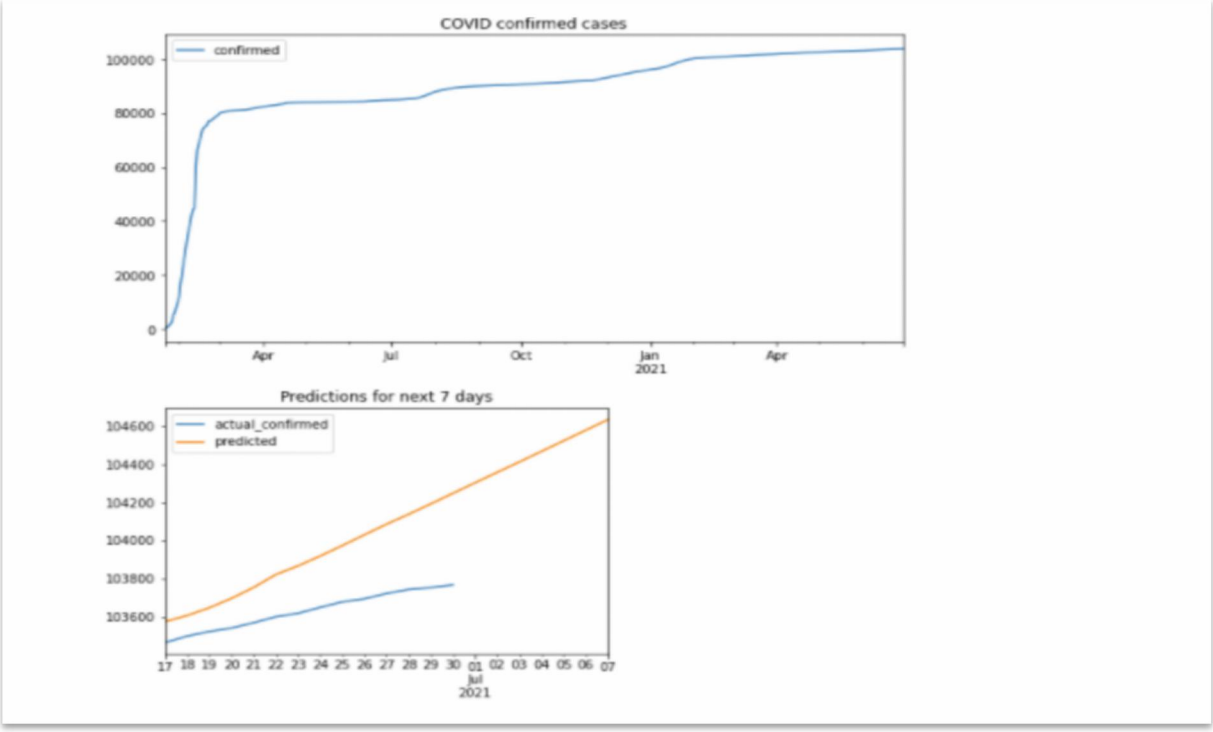
- **India**



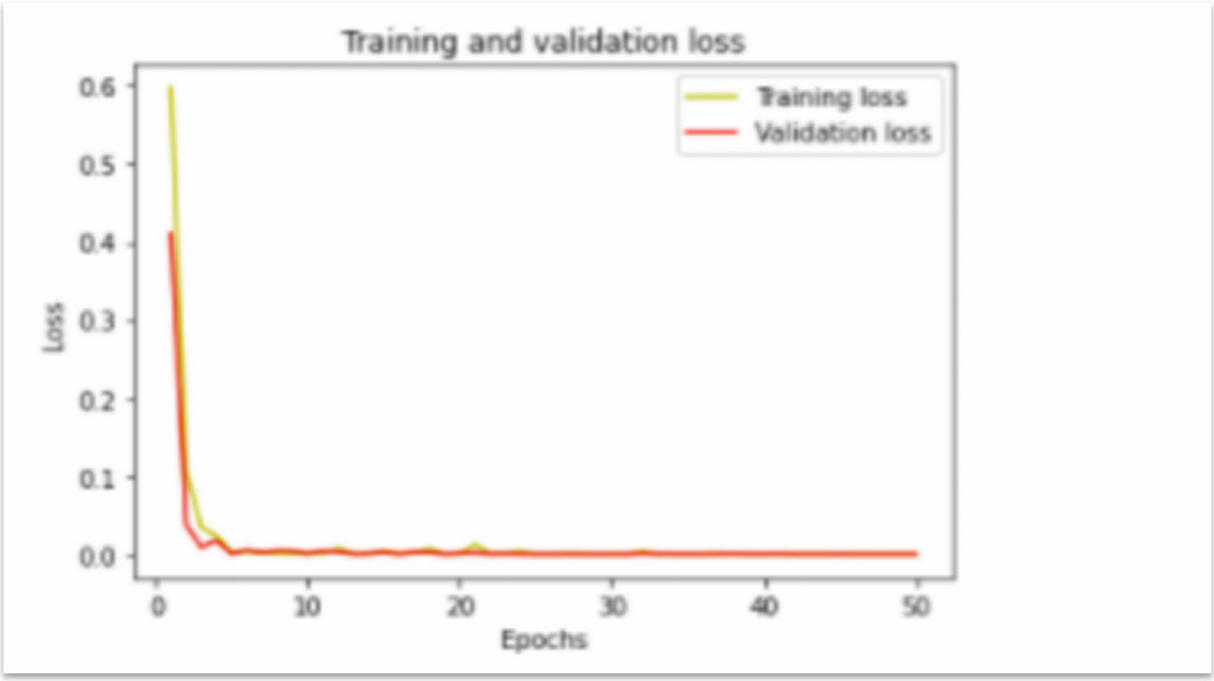
Another example of the model being overfitting, as you can see the predicted cases are much higher than the actual records. However, the influencing factors (mentioned below) across different countries also affect the results the model produces. The cases predicted are over 31M where in real life, India reached about 30M on July 3.



- **China**



The number of cases predicted for China is far greater than the real life records on that day (104600 > 91,869). We can see the huge impacts of the influencing factors on the prediction. Though both training and validation loss have the perfect correlation, the model is still far from being exactly correct.

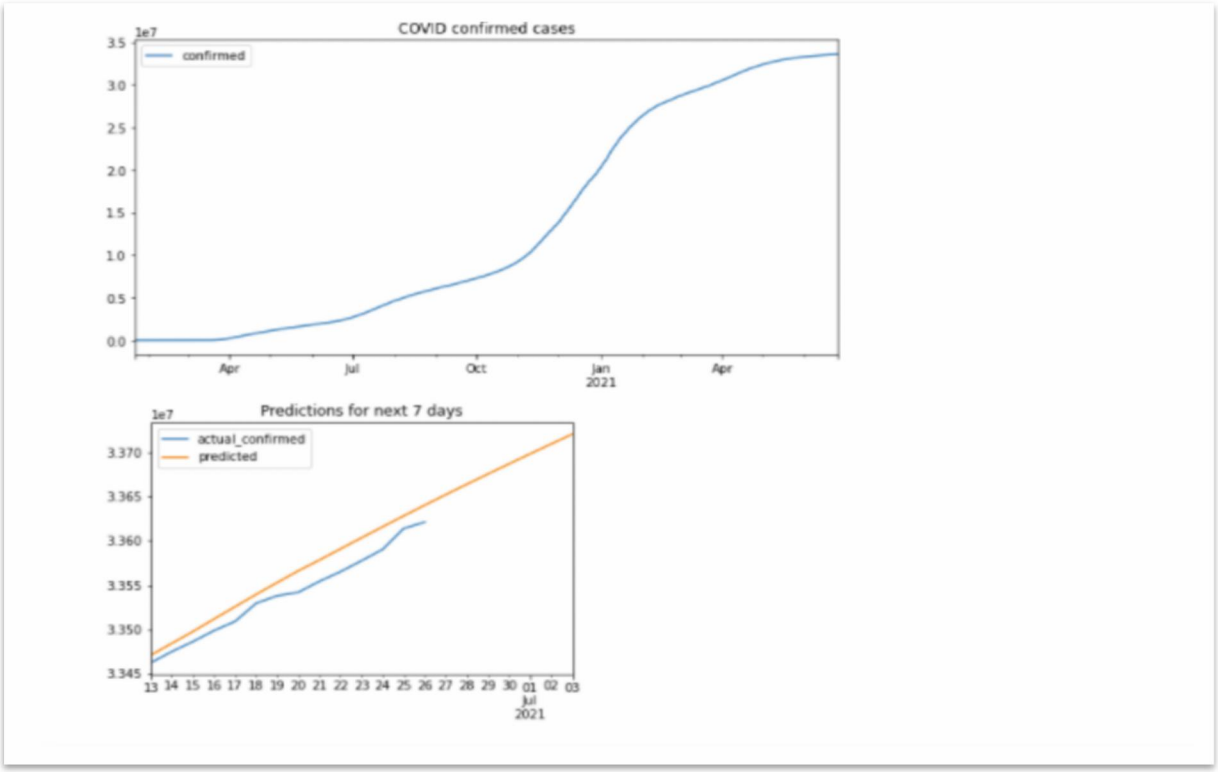


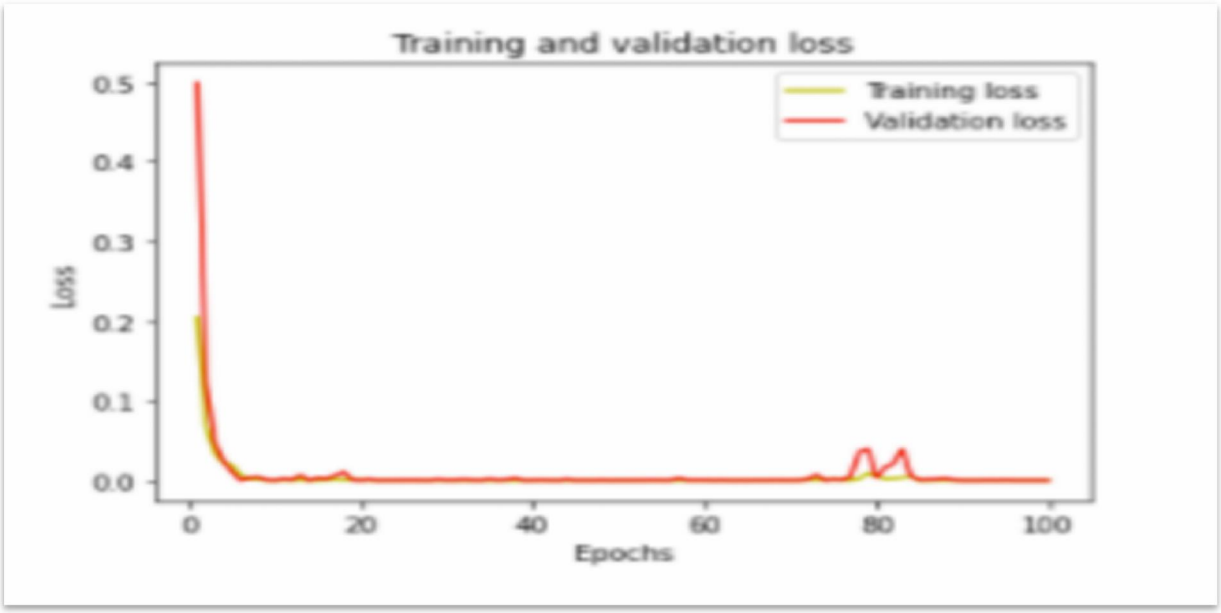
❖ Americas:

The prediction on these 2 countries is relatively accurate.

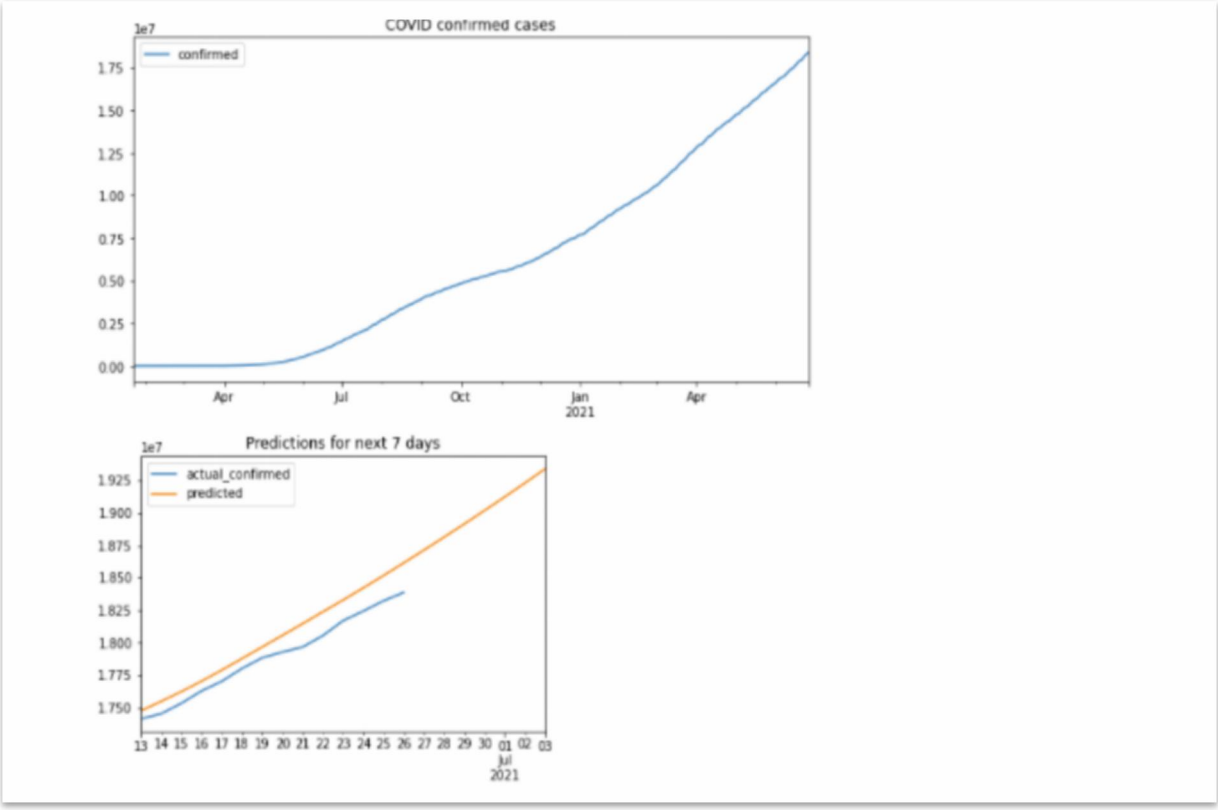
The performance of the model, according to the loss graph, is great as both training and validation loss stabilize at the end.

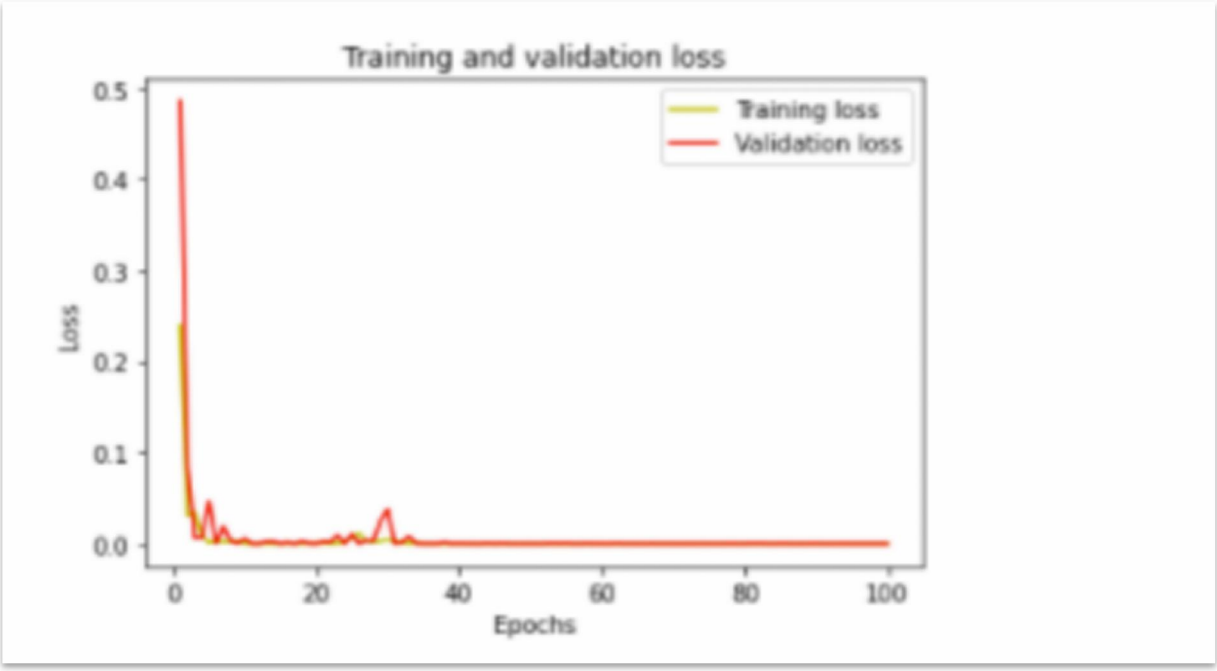
- **U.S**





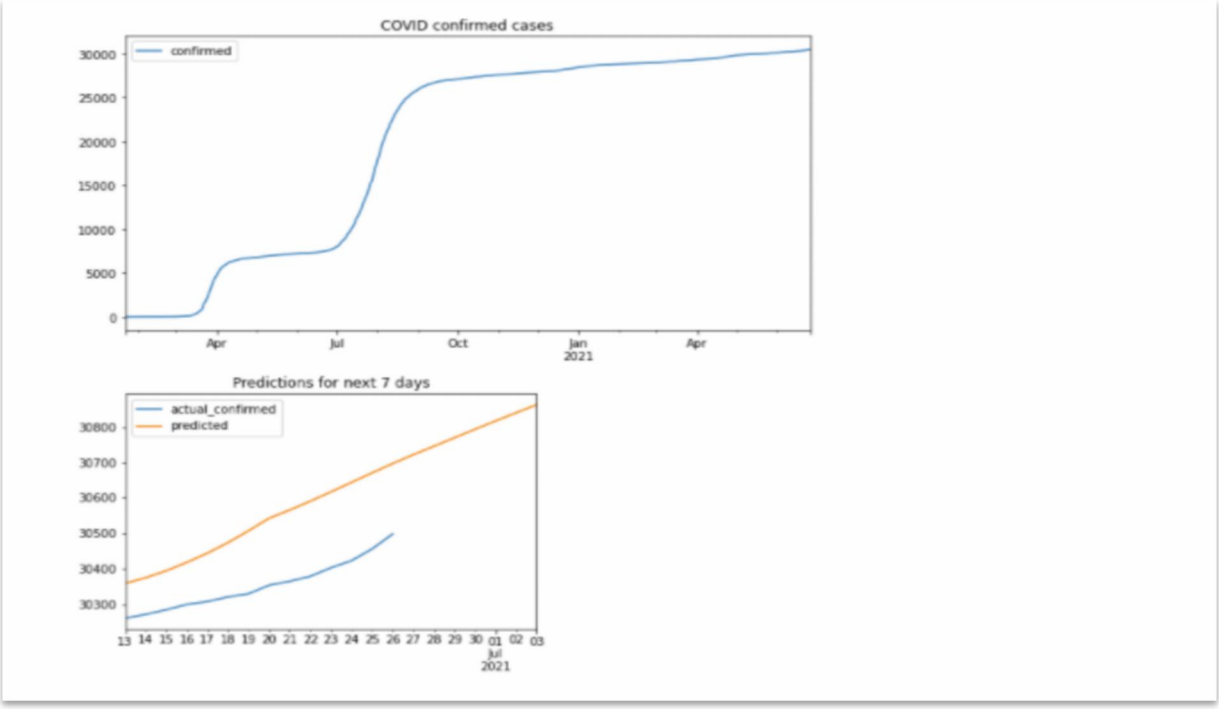
- **Brazil**


COVID confirmed cases
1e7
confirmed
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
Apr
Jul
Oct
Jan
2021
Apr
Predictions for next 7 days
1e7
actual_confirmed
predicted
1.925
1.900
1.875
1.850
1.825
1.800
1.775
1.750
13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 01 02 03
Jul
2021

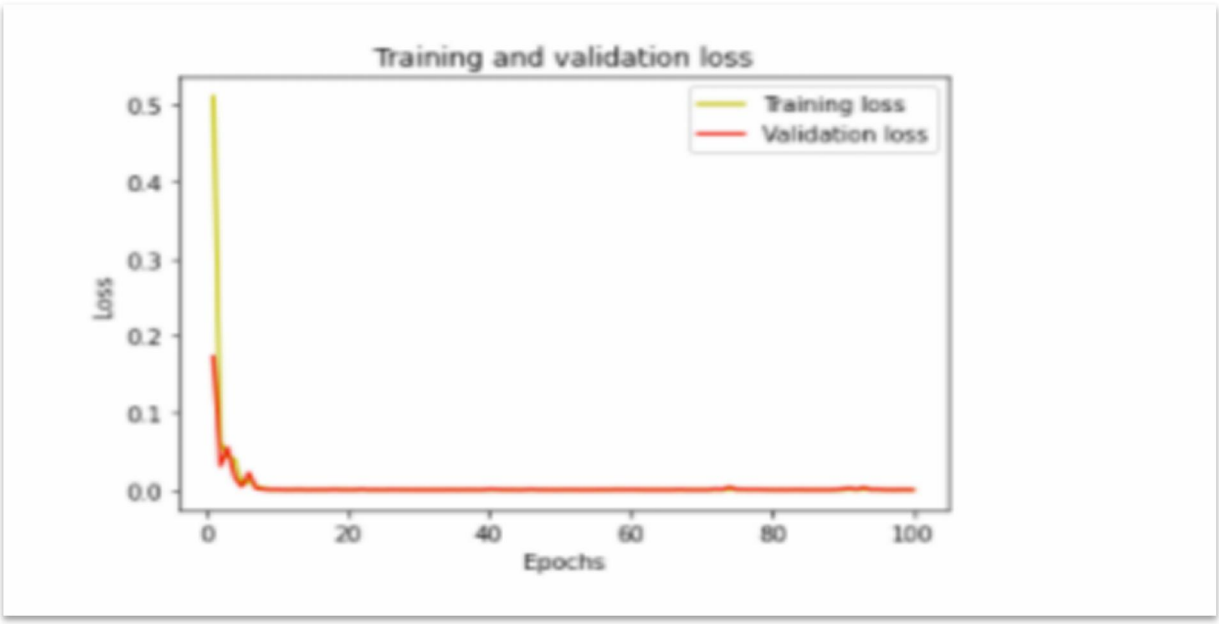

Training and validation loss
Training loss
Validation loss
Loss
0.5
0.4
0.3
0.2
0.1
0.0
0
20
40
60
80
100
Epochs

❖ Oceania:

- **Australia**



This is the same case as the China one, the loss graph has the perfect trend but still the destination of the predicted line surpasses the Australia cases that day.



Conclusion

As you can see, in most of our Training and Validation loss graphs there is clearly a health correlation between training loss and the validation loss. They both seem to reduce and stay at a constant value. This means that the model is well trained and is equally good on the training data as well as the hidden data.

And the performance of this model cross-country is also good according to the accuracy of the trend inferred from prediction graphs on every country and the precise correlation between the training and validation loss graphs. However, the predicted cases are not exactly the same as the records as there will be different influencing factors across different countries

Influencing factors such as:

- Methods of treatment
- Awareness of the citizens
- Vaccine
- The ability to track and zone affected cases or areas
- Environment factors
- More…

These factors which the model does not cover can either reduce or increase the trend of the pandemic. However, in general, it can predict the trend of a country and can be used as a method to evaluate whether the cases will be increasing or decreasing under certain circumstances.

4. Further model analysis:

A. Linear Regression

Linear Regression is a machine learning algorithm based on supervised learning. It performs a regression task. Regression models are target prediction value based on independent variables. It is mostly used for finding out the relationship between variables and forecasting. Please refer Linear Regression for complete reference.

Advantages

- Linear Regression performs well when the dataset is linearly separable. We can use it to find the nature of the relationship among the variables.
- Linear Regression is easier to implement, interpret and very efficient to train.
- Linear Regression is prone to overfitting but it can be easily avoided using some dimensional reduction techniques, regularization (L1 and L2) techniques and cross-validation.

Disadvantages

- Assumption of linearity between the dependent variable and the independent variables. In the real world, the data is rarely linearly separable. It assumes that there is a straight-line relationship between the dependent and independent variables which is incorrect many times.
- Prone to noise and overfitting
- Prone to outliers - linear regression is very sensitive to outliers. So, outliers should be analyzed and removed before applying Linear Regression to the dataset.

In summary, Linear Regression is great tool to analyze the relationships among the variables but it isn't recommended for most practical applications because it over-simplifies real world problems by assuming linear relationship among the variables

B. Random Forest Regression

The word 'Forest' in the term suggests that it will contain a lot of trees. The algorithm contains a bundle of decision trees to make a classification and it is also considered a saving technique when it comes to overfitting of a decision tree model.

Advantages

- It reduces overfitting in decision trees and helps to improve the accuracy
- It is flexible to both classification and regression problems
- It works well with both categorical and continuous values
- It automates missing values present in the data
- Normalizing of data is not required as it uses a rule-based approach.

Disadvantages

- It requires much computational power as well as resources as it builds numerous trees to combine their outputs.
- It also requires much time for training as it combines a lot of decision trees to determine the class.
- Due to the ensemble of decision trees, it also suffers interpretability and fails to determine the significance of each variable.

It's fairly good for training even small samples and can be easily parallelized in R/Python/other software. However, it fails when there are rare outcomes or rare predictors, as the algorithm is based on bootstrap sampling. This makes it non-ideal if you're working with rare personality traits, high segmented customer behavior, or rare variants in genomics research.

CONTENTS