COSC 6342 Machine Learning

Swish-LSTM – Improving the forecasting on the COVID-19 transmission rate with a custom loss function

Sai Pavani Gandla -2090151
Krithikashree Lakshminarayanan - 2072237
Jahanavi Sanda -2092208
Code link

1 Introduction

The COVID-19 pandemic has led to a global health crisis by leaving an enormous impact on people's lives all around the world. It affected all the segments of the world population irrespective of their age factor. Machine Learning has a profound role in all areas, it has shown a significant amount of progress in the healthcare industry with the findings drawn. Based on several instances, Machine learning was used to develop solutions to diagnose and result in treatment for diseases. It has the proficiency of identifying the diseases in earlier stages as uses the Data in sequences that are gathered throughout time and use it to provide diagnosis so that it can be addressed in earlier stages.
Similarly, Machine Learning can be used to automate the task of predicting COVID-19 infection and help in forecasting future cases based on the existing data.For our project,a trial was done with different algorithms, the Traditional approach, the Relu approach, and the Swish approach. The objective of the project is to Improve the forecasting on the COVID-19 transmission rate with a custom loss function.

2 Survey section

2.1 What is the core focus of the project?

The area we want to explore and experiment with our project is Neural Network,Neural network has shown significant improvement in the field of Machine Learning.As Machine Learning has a prominent impact in the Medical Domain,we chose the reoccurring problem arising in the Medical field, i.e, Covid-19.

2.2 Do we have enough data to achieve this?

It is a well-known fact that humongous number of people were affected due to Covid-19, as it spread at an exponential rate causing death all across the world. The COVID-19 data that we used for our model is a result of the research conducted by John Hopkins University. Wherein, it recorded country-wise data about vaccination, deaths, and an increase in the number of cases. Analysis of the data set with such diverse data helped us expose advanced trends which will help in forecasting the prediction for the number of cases. The data is considered on a day-to-day basis, i.e, the data is time-series data.Firstly,This could easily lead to over fitting of the data plausibly.In addition to this,We can understand that the data is not in the standardized (feature,response) input format that we feed a supervised machine learning model,we look forward to transform the data to get started with.

2.3 How was the problem approached?

Commonly,Viruses uses a pattern for transmission. So, for building the forecasting algorithm we were keen on understanding the pattern and in the meanwhile looking for other factors that influenced the patterns. Where in this case, the lockdown, and vaccination were some of the factors. Initially, as far as the time-series data set is concerned, we started to work with the Recurrent Neural Network (RNN), which is popular to work with time-series data. But we encountered the vanishing gradient problem.

We went ahead with the LSTM (Long short-term memory) which is an upgrade over the traditional RNN. Generally, LSTM uses a sigmoid activation function which ranges their output from 0 to 1. We tend to increase the input value and the output value to be around the maximum threshold, which leads to a possibility to converge near the optimum (the vanishing gradient problem). This is the same problem that we faced while using the Tanh activation function.

Both the activation functions resulted in a saturation problem, we went ahead with Relu, one of the advanced activation functions.With the scope of research,we looked for other activation functions that will perform in a similar fashion to Relu or better and encountered Swish function.Interestingly,we saw that there were many theoretical findings about the Swish function but none of them had a practical approach to it.Therefore,we were keen on exploring the outcomes of the Swish functions.On implementing the Swish function aim improve the model's prediction accuracy.As a matter of curiosity,we also want to tweak the architecture of the model and look for any findings there.

2.4 The business value of research?

The intent of the project is to forecast the prediction of Covid-19 cases, which would serve as a moderate help to the medical and research domain.We have used a custom loss function to evaluate our model, that describes the error between predicted and actual COVID cases.The error factor we have used will help in easy readability,which can be interpreted by a layman without an expert intervention.

Through the process of research,we have encountered countries that were under-reporting. Therefore, we added an increased penalty weight on case count based on the country.This factor would make our model generalized,capture the trend accurately and help majorly in the Medical Domain.

2.5 How are we planning to evaluate the success of the project?

Taking the impact of Covid-19 into consideration, the research and the results of projects like these would be a helping and act as a supporting factor in the area of research and medical resources. Initially, As we are using multiple module evaluation, we expect that the end product would be an effective forecasting model. As there is an existing pool of models and the model that would stand out among all of these is what we are looking forward to. Secondly, as the novel factor is concerned, we are aiming to build the custom loss function as to how this would impact the performance of the models and results. To add on, we would see how the penalty weight factor would make a difference when the under-reported countries are taken into consideration.

2.6 Expected hurdles we may encounter

- The data that we have considered is a time-series data,We can understand that the data is not in the standardized (feature,response) input format that we feed our LSTM model.
- Transforming and Normalizing the sequential data would be one of the major hurdle that we might come across.
- Maintaining the trade-off between the accuracy and increase in number of nodes.
- Deciding the number of hidden layers for our LSTM model and weight initialization.

3 Experiment section

3.1 Previous work

COVID-19 firstly emerged in December 2019 and within a short span of time, it got spread rapidly and became a global pandemic [1]. Numerous statistical and mathematical models[2, 3] have been put forth to simulate the COVID-19 epidemic's current transmission patterns.

Based on information about the previous transmission, we attempted to forecast the COVID-19 outbreak. The currently available dataset is a time series dataset. Every time we must perform pattern-based forecasting for a time series collection, Recurrent Neural Network (RNN) comes to mind when we have a time series data, but it has some issues with vanishing gradient. So, to overcome this issue we use Long short-term memory (LSTM) which uses memory cells in the hidden layer, with improved abilities to remember the long-term dependencies.

In this paper [4] the authors worked on forecasting the spread of COVID-19 with the help of LSTM network and made a model prediction based on the current situation. With the use of LSTM networks, Bodapati et al. [5] predicted the COVID-19 daily cases, deaths caused, and recovered cases for the entire world. We searched for different activation functions to get better results since the RELU results are not that good. Then Prajit Ramachandran, Barret Zoph, Quoc V. Le in the paper [3] proposed a new activation function (Swish) that performs better than RELU activation function and sigmoid activation functions. The main contribution of our work is to implement a custom loss function that describes the error between predicted and actual COVID cases. And finally compares the performance of the model with three different activation functions and evaluated using a custom loss function.

3.2 Proposed architecture

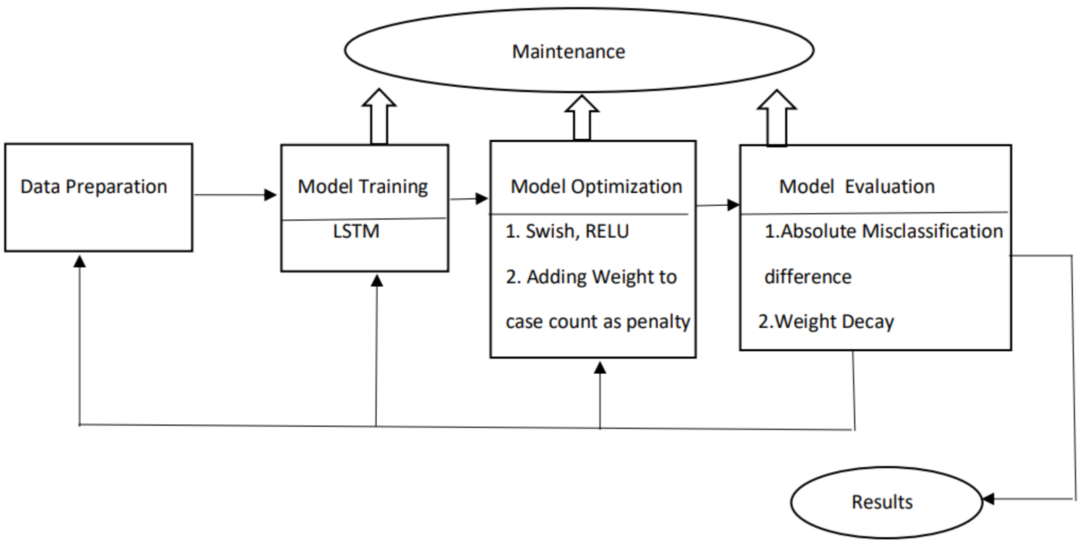

Figure 1: Proposed Architecture

The diagram 1 shows the proposed architecture. The following are the steps that are followed to get accurate results. In the data preparation step, we prepared our data by performing data preprocessing, and since it is time series data, we normalized the series. After all these transformations our input data is ready.

During our research for this project, we studied the complete flow and architecture of LSTM and came to know that applications of LSTM are high in demand. A cell, an input gate, an output gate, and a forget gate make up a typical LSTM unit. Three gates regulate the information flow into and out of the cell, and the cell retains values for arbitrary time periods. As it was already established, the vanishing gradient problem was explicitly addressed by LSTM. Therefore, their key characteristic is basically their capacity to retain information for a long time. This makes learning for LSTM quite simple. Now we are using the prepared data and fitting the data on the model to train the LSTM model.

In model optimization, we are going to change or update some parameters to improve the performance of the model. Here we are going to define a custom loss function (novel feature) and predict the difference between the actual result and the predicted result of the model. In our case, we are going to add weights to the case count as a penalty and now observe the performance of three activation functions (sigmoid, RELU, and Swish).

In the model evaluation phase, we are going to calculate absolute misclassification error (abs (predicted – true)*penalty) which is easy to interpret than MSE, to observe how many observations are incorrectly predicted. With this analysis, we can make some changes to improve the prediction. And penalizing underreporting countries a.k.a weight decay, improves the generalization performance of the model.

In the maintenance of this complete flow, we are performed hyperparameter tuning ,added dropouts etc for improving model stability and to avoid overfitting .

3.3 Methodology

In contemporary society like ours, no decision is made without understanding the future implication. Especially in studies like covid 19 transmission rate, where it can mean life or death of millions of people, building a good forecasting model would prove useful to government to make proper decisions. In this section of the report, we provide a detailed explanation of our unique LSTM network-based model for projecting the transmission rate for the upcoming two weeks while keeping in mind the problem's business significance.

Let's first discuss the setup, requirements, and end objective of our research before delving deeply into the technique we followed. The dataset at hand is a time series data that includes daily confirmed cases worldwide and broken down by nation. The dataset considered for our research is maintained by John Hopkins University. Since the information for this is taken from WHO (world health organization), the data is legitimate. Finding a precise count of confirmed instances is the desired result. To accomplish this, we loaded the default essential packages like NumPy, pandas, etc. on the working platform of Collab. After loading the dataset, the required transformation was done. The subsequent procedures we took to construct the model are listed below.

Step 1: Data transformations

Setup

As the dataset is loaded and is bought in as a data frame already, we took a deeper look at what piece of information will serve useful to us. The skeleton of the data available to us is shown in 2

Before transformation

	Province/State	Country/Region	Lat	Long	1/22/20	1/23/20	1/24/20	1/25/20
0	NaN	Afghanistan	33.939110	67.709953	0	0	0	0
1	NaN	Albania	41.153300	20.168300	0	0	0	0
2	NaN	Algeria	28.033900	1.659600	0	0	0	0
3	NaN	Andorra	42.506300	1.521800	0	0	0	0
4	NaN	Angola	-11.202700	17.873900	0	0	0	0
...	...	...	...	...	...	...	...	...
284	NaN	West Bank and Gaza	31.952200	35.233200	0	0	0	0
285	NaN	Winter Olympics 2022	39.904200	116.407400	0	0	0	0
286	NaN	Yemen	15.552727	48.516388	0	0	0	0
287	NaN	Zambia	-13.133897	27.849332	0	0	0	0
288	NaN	Zimbabwe	-19.015438	29.154857	0	0	0	0

289 rows × 1051 columns

After transformation

	confirmed
2020-01-22	557
2020-01-23	657
2020-01-24	944
2020-01-25	1437
2020-01-26	2120
...	...
2022-11-29	642749423
2022-11-30	643274699
2022-12-01	644001063
2022-12-02	644733502
2022-12-03	645032173

Figure 2: Transformed data

We are trying to analyze only the count for each day we removed all other columns but for date. Then we did a transpose of these dates by calculating the total count for each day worldwide.

Stationary data and normalizing

Since the data in hand is a time series dataset, it is important to check if it is stationary or not. This is a crucial step that has influence on how the data is perceived and predicted. To ensure the independence between data point we performed ADF test and based on the p-value we concluded that the series does not have a unit root and it is called stationary series. The next important transformation to do is normalizing the series. With the help of

MinMaxScaler from sklearn package we transformed the input. The graph 3 shows the global trend of covid 19 transmission from Jan 10 2020.

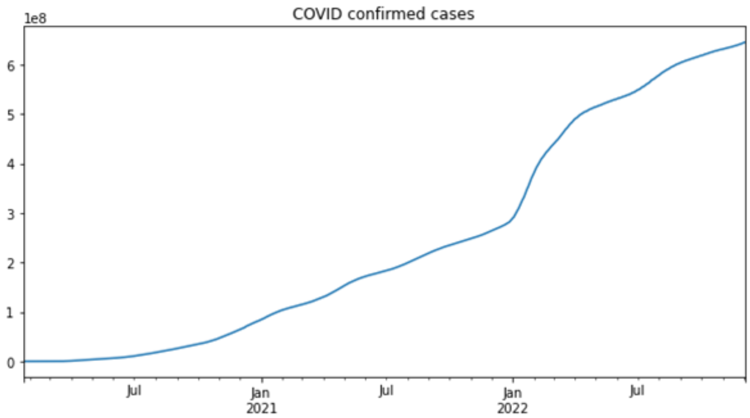

Figure 3: Covid 19 transmission trend

An important point we have to mention here is while analyzing the global trend we ignored each country specific factors like lockdown, restrictions specific to each country, vaccination rate, government initiative etc. We would like to analyze country trend to global trend but due to increases complexity and time constraint we were not able to accomplish this.

Train test split

We split the dataset into train and test sets. We ignored the last 14 days and build the training set and included only the last 14 days in the test set.

Time series generation

Time series data must be transformed into input-output structure for it to be feed into the supervised machine learning model. To make things easier we used TimeseriesGenerator from keras to structure the data in a desired format. For our case we took the last 7 days(length=7) for forecasting the count for 8th day(Batch size=1). We did this for both train and test set. Our input data to the model will look like the figure 4

Input days							Output day
1	2	3	4	5	6	7	8
2	3	4	5	6	7	8	9
3	4	5	6	7	8	9	10

Figure 4: Input formatted with Timeseriesgenerator

As we have already mentioned the issue with comparing local restriction with global trend, we are planning to build our own time series generator. We would generate 2 series one for nation wise and the other for global trend and analyze the results.

Step 2: Atypical methodology we used

During the course of the research, we experimented with the internal architecture of the LSTM model. Based on concrete research (highlighted in previous work) we found evidence justifying swish activation consistently outperforms Relu. So, we were curious to test this hypothesis. We evaluated our model with a custom loss function. Here we considered the fact that there are a few countries that do not report correct case count. So we added a increased penalty weight on case count based on the country. A penalized absolute misclassification count serves as our unique loss function.

The LSTM model is build using keras. We build a sequential model and added LSTM and dense layers. From keras.backend we loaded sigmoid function. We build swish function from scratch. We then updated swish activation function into the keras custom objects. The loss function is built using tensorflow.

Step 3: Justification for the methodology choices

The available prediction model's primary flaw was that it did not take underreporting into account when it was being built. Major misclassification results from this. The proposed penalization term is intended to boost the model's generalization capabilities. We determined the absolute difference in misclassification after applying the additional weight. This would provide the precise difference count and make it simple for even a layperson to comprehend the findings of our model.

4 Comparison and Results

The methodology used for this study is entirely data driven. In no prior study is the approach employed explained. To establish a baseline, we conducted in-depth study and evaluated a number of theories. We will have trained our model until November 25, 2022. The forecast's outcomes are as follows 5.

The plots 5 depict the forecast for the next few weeks. The blue line represents the actual count, while the orange line represents our predictions. Although all approaches produced good results, the traditional LSTM architecture produced the most incorrect predictions.

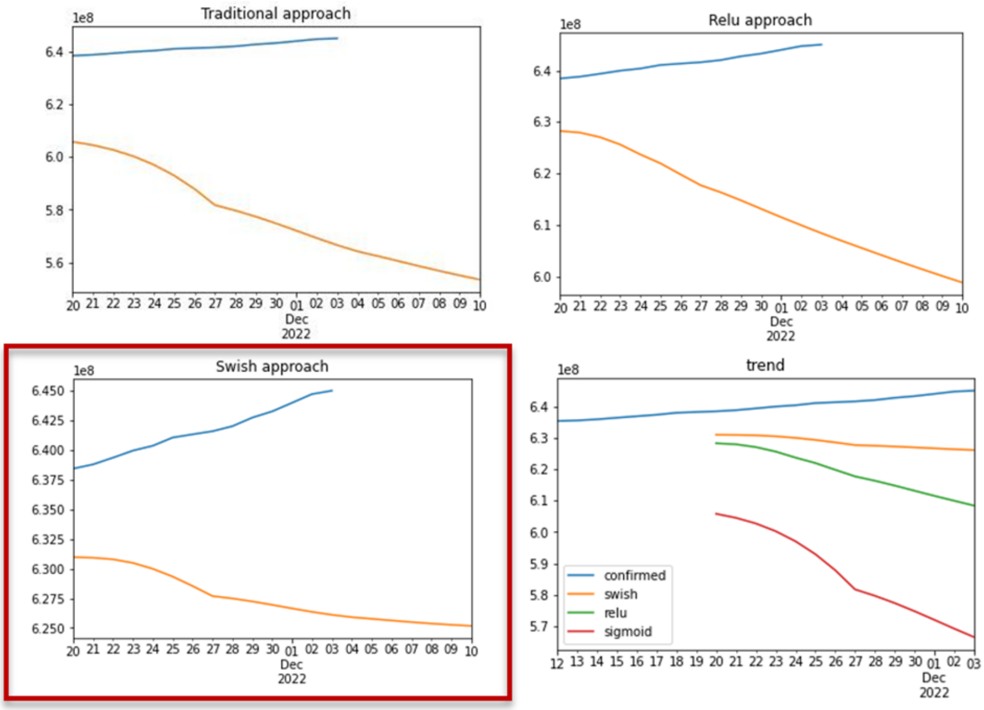

Figure 5: Results

In situations like these, where people's lives are at stake, we must make the most accurate predictions possible. As we can see, the swish performs better than expected. However, we discovered that swish did not consistently outperform relu. The findings we discovered contradicted the findings of the paper [5].

In addition to this the penalty weight to the underreporting countries did improve the models generalization performance. The plot 6 shows there is significant improvement in the forecast. The green line is the weighted input, and the orange line is original forecast.

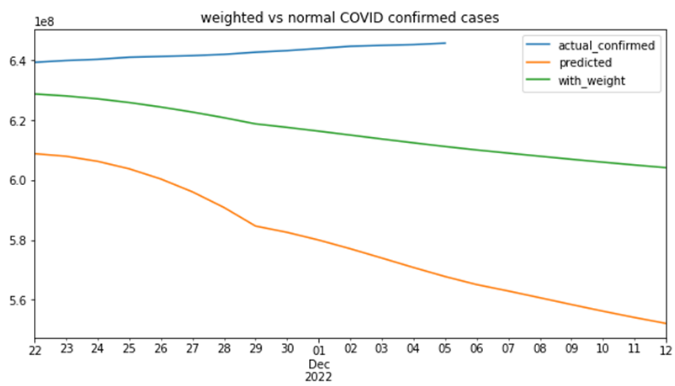

Figure 6: Weighted penality improving forecast

5 Conclusion

The spread of covid 19 disease is increasing exponentially and even developed countries have a tough time in planning for medical resources. Therefore, studies like this will be helpful. Using LSTM, we developed an effective forecasting model throughout the project. Then, we conducted a thorough investigation to enhance the forecasts. We experimented with various activation functions and discovered that relu and swish were good choices. We disproved the claim that "swish consistently outperforms relu" through our testing. Additionally, we created a unique loss function that measures the absolute weighted difference between the actual and anticipated case counts. With the help of this penalty factor, the model's generalization and forecasting accuracy were both enhanced.

References

[1] Cucinotta D, Vanelli M. WHO declares COVID-19 a pandemic. Acta Biomed. 2020;91(1):157–60.

[2] Forecasting COVID-19 confirmed cases

[3] Searching for Activation functions

[4] Spread of COVID-19 with the help of LSTM network

[5] S. Bodapati, H. Bandarupally, and M. Trupthi, "COVID-19 time series forecasting of daily cases, deaths caused and recovered cases using long short term memory networks," in 2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA). IEEE, 2020, pp. 525–530.

[6] Time series forecasting of COVID-19 transmission in Canada using LSTM network

[7] Time series forecasting of Covid-19 using deep learning models: India-USA comparative case study