EasyChair Preprint

№ 4679

Forecasting, Visualization and Analysis of COVID-19 in India using Time Series Modelling

Afzal Ansari and Sourabh Kumar Burnwal

EasyChair preprints are intended for rapid dissemination of research results and are integrated with the rest of EasyChair.

December 1, 2020

DE GRUYTER

Afzal Ansari and Sourabh Kumar Burnwal

Forecasting, Visualization and Analysis of COVID-19 in India using Time Series Modeling

Abstract: Since the origination of COVID-19 in China and its spread across the globe, humanity has been put at risk and it has set a big alarm till its end across the country. Due to the unprecedented rate of increase in the number of cases and its subsequent pressure on the administration and health professionals globally, it would be highly needed to have a safe future by doing analysis and forecasting the number of new cases using some prediction methods. The current situation in India is getting worsened day-by-day due to which, the economy of this country has been down and unstable. In this paper, we have analyzed, how the numbers of daily infected cases in India could look like, predicting the trend, and investigate what the peak value could hit by now. We have used data-driven estimation methods like Fb-Prophet and long short-term memory (LSTM) as a state-of-the-art method and Deep Learning models respectively for forecasting the number of COVID-19 cases in India a few days ahead. We have proposed a method considering various parameters to predict daily confirmed future cases within a certain range which would be a beneficial tool for administrators and health officials.

Keywords: COVID19, Forecasting, Prediction, Deep learning, LSTM, FbProphet

1. Introduction

World is moving through a very distressing stage by the spread of novel coronavirus (SARS-CoV-2). It is a highly contagious disease and the World Health Organization (WHO) has declared it as a global public health emergence [1]. It is originated in Wuhan, Hubei Province, People's Republic of China (PRC) in late December 2019, when a case of unidentified pneumonia was reported[2]. PRC Centers for Disease Control (CDC) experts declared that pneumonia as novel coronavirus pneumonia (NCP) as caused by a novel coronavirus and WHO officially named the disease COVID-19[2]. In India, the first case of COVID-19 was reported on 30 January 2020 with origin from

Afzal Ansari, Dept of Computer Science and Engg; IIIT Bhubaneswar; Email: b113053@iiit-bh.ac.in

Sourabh Kumar Burnwal, Dept of Data Science and Analytics; Central University of Rajasthan; Email: sourabhkumarburnwal@gmail.com

China[3]. It spreads to the maximum of districts of the country. As a consequence, the number of new cases has increased exponentially since then. As on 5 September 2020 the total cases reported in India are 4023179 with 3107223 recoveries and 69,561 deaths[4]. All countries are trying to save their people lives by implementing measures like travel restrictions, quarantines, event postponements and cancellations, social distancing, testing, hard and soft lockdowns [6]. More than the lives this virus has taken, the economic and social impact is far more disastrous and especially for developing and underdeveloped countries. It is terrifying to imagine the disaster this COVID-19 may cause in India where world's 18% of the population resides [7] with a population density of 32,303 people per square kilometer in cities like Mumbai[8]. The Government of India has proposed multiple lockdowns consecutively between 25 March and 7 July 2020 to prevent the spread of this virus. Initially, in lockdown 1.0 (March 25, 2020, to April 14, 2020), the entire nation was under complete lockdown except for essential services and lockdown 2.0 (April 15, 2020, to May 3, 2020) was implemented with relaxation in areas where the virus was contained and lockdown 3.0 (May 4, 2020, to May 17, 2020) with more relaxations in areas where there were fewer number of coronavirus cases and finally lockdown 4.0 (May 18, 2020, to May 31, 2020) was further extended with low restrictions. Due to these lockdowns, there has been a decrease in the number of cases from 11.8% to 6.3% on a daily basis[9]. While lockdown was observed more rigidly on urban areas, rural areas have been more reluctant. And the government cannot shut the entire nation forever as the economy may fall drastically. Also, it has started unlocking from 1 June 2020 during the growing phase for some obvious economic constraints. So, a practical solution could be to quarantine the very critical zones, so that the people affected by this virus shall remain in that zone only. And this way, the rate of infection has become lower as compared to other countries. However, there is a lot of stress on the part of administration and health officials for accommodating patients with possible symptoms of COVID19. So, for that some prediction tools must be used to know about the number of cases in coming days for making preparations at the administrative level [1]

In this novel research, we present the data-driven LSTM method and Fb-Prophet method respectively using Time Series analysis for the prediction of the number of daily cases to be accommodated in the subsequent days based on the data available. Past and current scenario have to be analyzed in order to predict the outbreak further. To forecast the daily cases in near future after learning from the past daily data, first the key features needs to be extracted. After finding the key features, several experiments were conducted to optimize the model that can approximately predict the number of future COVID-19 cases with minimum error, so the administration can make preparations accordingly to accommodate them. This paper has been organized as follows. In Section 1, we have discussed EDA (Exploratory Data Analysis) of COVID 19 cases in India i.e. how it would look like and make trend analysis. In Section 2, the Fb-Prophet method and Deep LSTM technique for the prediction of COVID-19 have been explained in detail. In Sections 3 and 4, the results and conclusions of the work have been presented, respectively.

2. Methods and models

2.1. Dataset

The COVID-19 data used in this paper was collected from the GitHub repository of John Hopkins University [16], which gets updated on daily basis. It contains country-wise information about the number of new cases, cumulative cases, new deaths, and cumulative deaths for each day since the start of the pandemic. Another dataset was collected from the My India Govt. portal [5]. We selected a time period starting from 22nd January to 5th September 2020 for this study when the lockdown phase was released and the unlock phase was started in India.

2.2. Model design

The experiments are conducted on open source libraries such as Numpy, Pandas, Tensorflow (Google) and Keras. Python, as a high-level general-purpose programming language, is used to interact with deep learning libraries as application program interfaces (APIs). The obtained APIs is used to design the current model structure for LSTM network with FbProphet API model.

The models are used to learn the trend behavior present in the data and also map the learning sequence present, to produce future forecasts of the number of confirmed cases present in any particular region. These are provided with region-based historical data of the number of cases appearing daily, and in consideration to the dynamically changing structure of the dataset, we used the historical data ranging from January 22nd, 2020 to September 05th, 2020 for trend analysis as well as for training and testing our prediction model.

2.3. Time series analysis of COVID-19 cases in India with State wise analysis

It's quite normal to encounter temporal data in real-world scenarios. In fact, most of the Stock market data, sensor readings and many others fall into this category. Data collected over regular time intervals is known as time series data, where each data point is equally spaced over time. Its distinctive properties make it very useful in solving numerous unsolved problems with a wide range of applications. Time Series data has three basic components: Trend, Seasonality and Remainder. Trend is said to be observed when the data has a tendency to decrease or increase during a long period of time due to external factors like lockdown of country, mandatory social distancing, quarantines etc. Seasonality refers to repetitive short-term cycle or periodic fluctuations. Like we saw a decrease in the daily new cases during lockdown phase and an increment when people didn't follow the norms. Time Series forecasting of a dataset with temporal features is about using historical values along with the associated patterns

to predict how the future could look like. In many real-world scenarios, either of trend or seasonality are absent. After getting the nature of TS, various forecasting methods have to be applied on given TS. [1]

In this analysis we transform the given data into proper time series data and make various visualization of mainly one feature among all the features i.e. Daily Confirmed Cases in India with state wise.

In Fig. 1 it shows how this trend goes with and in Fig.2 it shows how daily new cases increases up over time. And in Fig. 3 it shows how COVID 19 affects India with state wise analysis during the lockdown and when it is unlocked. In Fig. 2 it combines three features namely Daily Confirmed, Daily Deceased and Daily Recovered Cases where lockdown phase started from last March to last May (represented by L1-L4). Fig. 3 shows the 10 most affected states of India in which the number of positive COVID-19 cases reaches above 6000 on average on daily basis where there are few states like Maharashtra with the highest case then followed by, Tamil Nadu, Andhra Pradesh and Karnataka in which the number of cases is increasing at very high rate and states like Delhi, West Bengal in which cases are increasing linearly and in states like Madhya Pradesh, Telangana and Punjab, linear curve with less growth rate is obtained in Fig. 3. In Fig. 4 it shows number of relative increased cases i.e. calculated using the difference between the daily Confirmed cases and daily Recovered cases each day.

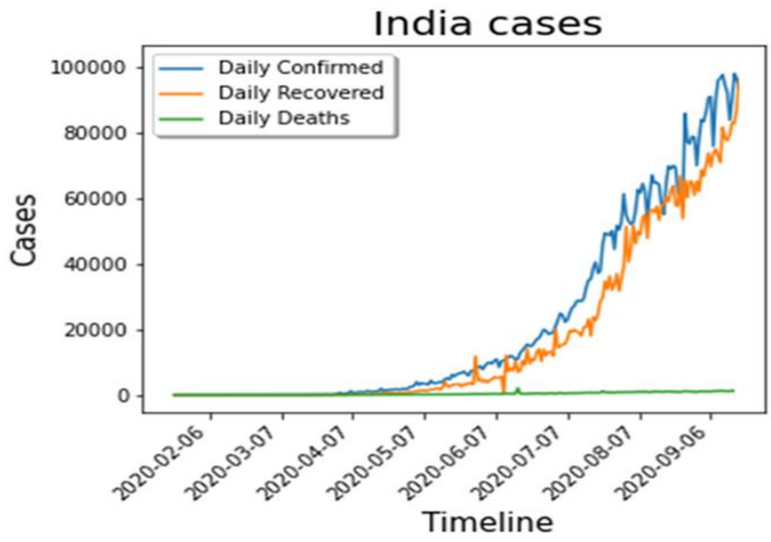

Fig. 1: Number of COVID-19 cases in India till date

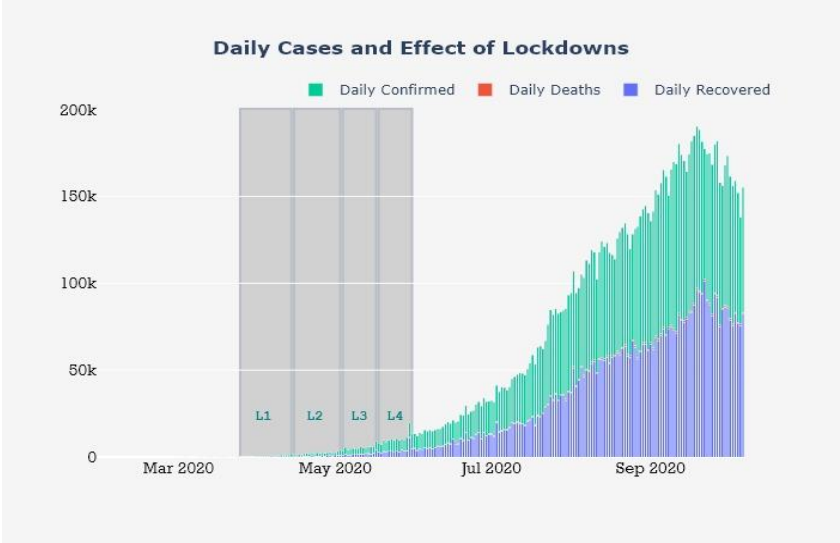

Fig. 2: Number of Daily Confirmed, Daily Deaths, Daily Recovered Cases in India

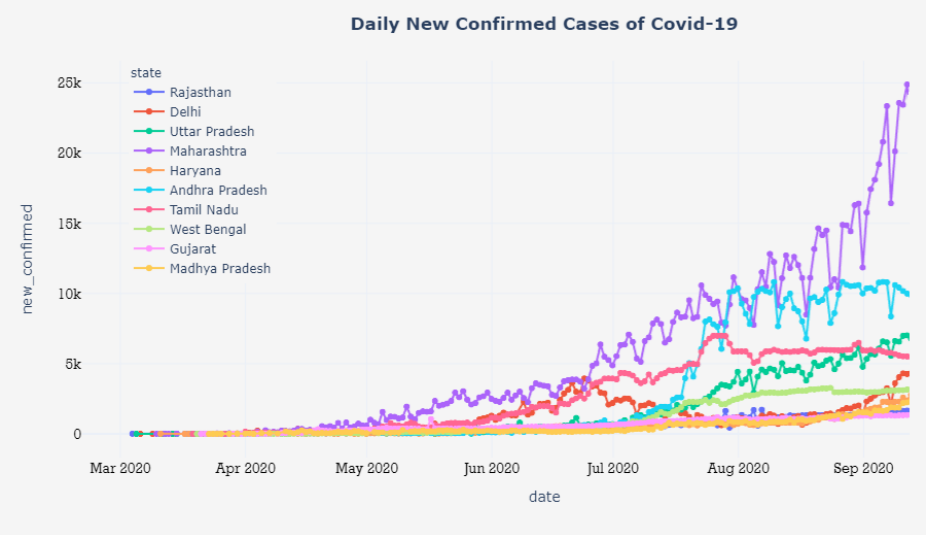

Fig. 3: Number of Daily New Confirmed cases of COVID-19 state-wise in India

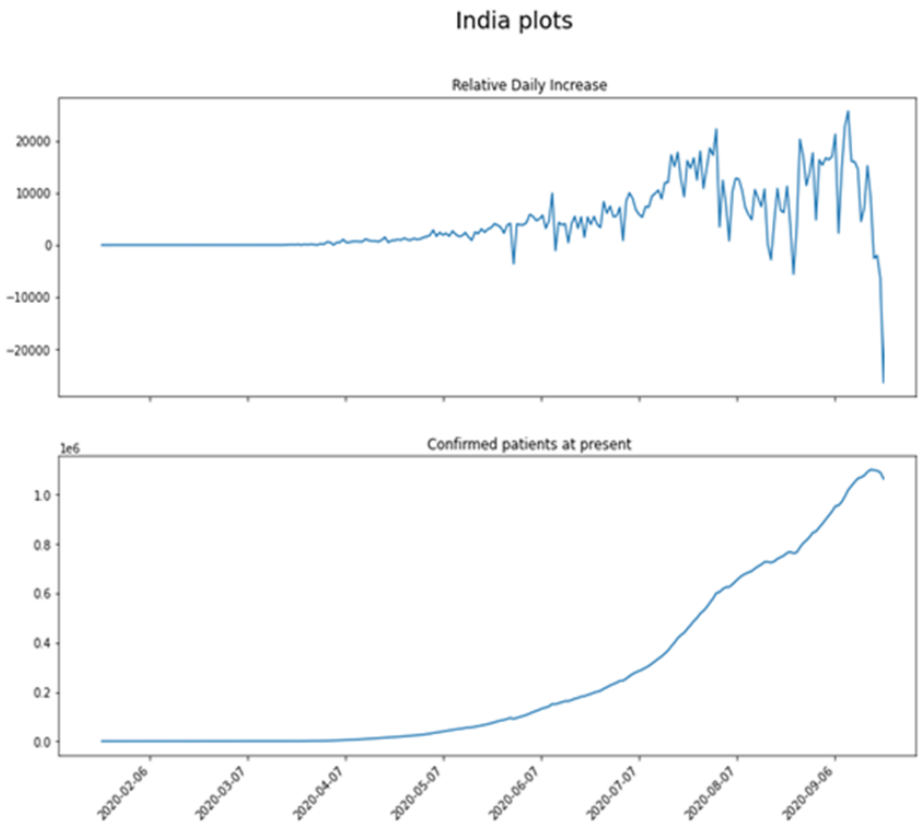

Fig. 4: Number of relative daily increased cases compared to daily confirmed cases in India

2.4. Fb-Prophet as a state-of-the-art method and LSTM method for the prediction of COVID-19

Fb Prophet is an open source framework of Facebook for time series forecasting based on additive model which is opened up to the public in 2017 [12]. It is a forecasting procedure implemented in R and Python. It is fast and provides completely automated forecasts that can be tuned by hand by data scientists and analysts. The idea of the method is to choose a suitable training model according to the characteristics of historical data and use it to predict the future observation results. The non-linear trends of Prophet are fitted with yearly, weekly, and daily seasonality, plus holiday effects. The perfect Prophet function can not only predict the future, but also fill in missing values and detect anomalies. In Prophet, the prediction model consists of superposition $y(t) = g(t) + s(t) + h(t) + t$, where $g(t)$ is a trend function used to analyze the non-periodic changes of time series. $s(t)$ a periodic term, reflecting the periodic change, such as the periodicity of a week or a year. $h(t)$ is the influence of an occasional day or days, such as a holiday. t is an error term, on behalf of the failed to consider the effect of the error of the model.

We apply this technique to COVID-19 forecasting in the country India. The model can be used to predict more with necessary changes. We create an instance of the Prophet class and then call its fit and predict methods. The input to Prophet is always a time series with two features: date ds and value y. Here in our study, ds are the dates of day, and y is the accumulated infected cases in India.

2.4.1. Background: Recurrent neural networks (RNN)

Deep learning methods like recurrent neural networks (RNN) proved to be effective for prediction [6] due to automatic extracting relevant features from the training samples, feeding the activation from the previous time step as input for the current time step and networks self-connections. RNN is good at processing data and exhibiting great potential in time-series prediction [7] through storing large historical information in its internal state.

2.4.2. Long-short-term memory (LSTM)

For prediction tasks, LSTMs are considered to be among the most feasible solutions, and they anticipate the future forecasts dependent on various highlighted features present in the dataset. With LSTMs, the data moves through components known as cell states. LSTMs can accurately recollect or overlook things. Information gathered over progressive time frames are portrayed as time series data and to produce forecasts with these data values generally LSTMs are proposed to be a stable methodology. In this sort of design, the model passes the past shrouded state to the subsequent stage of the arrangement. Since RNNs can store only limited amount of information, for long term memory storage long short-term memory cells (LSTM) [13] are used along with RNNs. LSTMs overcome the issues of vanishing gradient and exploding gradient [14], which plagues RNN. LSTM cells are similar to RNN with hidden units replaceable with memory blocks.

Recurrent LSTM networks has also the potential to deal with the constraints of traditional time series forecasting techniques by reorganizing nonlinearities of given COVID-19 dataset and can give output state of the art results on temporal data. Every block of LSTM works at various time step and sends its output to another block until the final LSTM block gives the sequential output.

The basic component of LSTM networks is memory block, which was developed to handle vanishing gradients by storing network parameters for long durations in the memory. Memory block in LSTM network is same as the differential storage systems of a digital systems. Gates in LSTM help in computing the data with the help of activation function (sigmoid) and give the result in between 0 and 1. The sigmoid activation function is used because it is needed to pass only positive values to the next gates to obtain a clear result. The three gates of LSTM architecture are represented with the following equations below:

$$Jt = sigmoid(wJ[ht-1, kt] + bJ) \quad (1)$$

$$Gt = sigmoid(wG[ht-1, kt] + bG) \quad (2)$$

$$Pt = sigmoid(wP[ht-1, kt] + bP) \quad (3)$$

Where: Jt = function of input gate

Gt = function of forget gate

Pt = function of output gate

Wx = coefficients of neurons at gate (x)

$Ht-1$ = result from previous time step

kt = input to the current function at time-step t

bx = bias of neurons at gate (x)

In the first equation, input gate provides the information that requires to be saved in the cell state. Second equation sends the information depending on the forget gate activation output. The third equation of output gate connects the information from the cell state with the output of forget gate at time step t for giving the result.

The purpose of initiating self-loops is to build a path so that gradients or weights can be shared for long durations. Specifically, this is needful while creating deep networks model where vanishing gradient is a frequent problem to face. We can control the time scale to detect the dynamically changing parameters by regulating weights as self-looped gates. Using the above methods, LSTMs are able to give the state-of-the-art results in [15].

Stacked LSTM [13], also called Deep LSTM which is the extension of standard LSTM which we have used in our case. In stacked LSTM, there are multiple hidden layers with multiple memory cells. Stacking multiple layers increases the depth of the neural networks where every layer is having some information and passes it on to another. Top LSTM layer provides sequence data to the preceding layer and so on. Also, this model uses the ReLu activation function to prevail over the most commonly existing issue in recurrent neural networks as vanishing gradient problem as described above.

This works on a two-layer deep LSTM setup with each of the layer containing 100 hidden neurons units. The input shape in the model is taken to be the lag structure with number of steps as 7 and number of features to be 1.

We explore two different models independently i.e. Fb-Prophet and Deep LSTM models on the dataset for the same time-period for predicting the number of daily infected cases in India. A multivariate time series data has more than one observation for each time step. Many researchers have advocated that multivariate analysis gives better performance in forecasting than by studying just one variable. Tomar, A., & Gupta, N. [9] have given data-driven estimation using long short-term memory (LSTM) and predicted the number of COVID-19 cases in India. It also analyzed effect of preventive measures like social isolation and lockdown on the spread of the disease.

The training set for this work contains cases' data is up to 5th Sept 2020, and the prediction was made for next few days. Hyper-parameter tuning of this model is done rigorously and selection procedure of these parameters.

Hence, considering various parameters, LSTM model gives pure statistical prediction solely based on the pattern of daily cases. However, each model gives an idea of how far the number of daily cases can rise in an epidemic model in unconstraint environment. The regression model first learns from the growth and decay pattern of daily cases of different countries using features like daily confirmed cases, daily death cases etc. and then make prediction on the number of cases for India. Finally, Fb-Prophet and Deep LSTM models are trained with these predicted time-series data to match with the actual data. Then both the trained model make forecasting on the number of future cases in India.

3. Results and Analysis

In this section, we study the spread of COVID-19 in India after lockdown released and unlock phase started by the Government as there are more than lacs of cases reported each day. For validation and analysis of the proposed model, data pertaining to India from (data maintained by Johns Hopkins University) and MyGov Portal has been used with the Python environment.

3.1 Data-driven methods to predict COVID-19

The data has been used from 22th January 2020 (when the first case of COVID-19 was reported in India) to 5th September 2020. The resulting plots showing the total number of confirmed cases are shown in Fig. 5(a) and 5(b) using Time Series Analysis respectively. In Fig. 5(a) we use state-of-the-art method i.e. Fb-Prophet method to forecast the necessary feature i.e. number of daily confirmed positive cases.

In Fig. 5(b), we use Stacked LSTM model to predict the outcome where we take 90% of data is used for training and rest 10% for forecasting and validation purposes. And official data (blue line) indicates the official data available and forecasted data (red line) indicates the forecast of a total number of confirmed cases. From both these graphs, it is observed that the

forecasted number of total confirmed positive cases closely matches with the available official data.

For these data-driven estimations, the data has been taken up to 5th of September. The comparison has also been made for the total positive reported cases and daily re- ported cases with estimated cases (by data driven model) from 1st to 5th September 2020 where percentage error is also observed as shown in Table. 1.

In the Fig. 5(b), we have represented LSTM with 100 neurons in the first hidden layer and 1 neuron in the output layer for predicting number of infected cases in future. The input shape will be 7 time-steps with 1 feature. From this figure, it is observed that our proposed method has worked significantly well with the very low error values between "0.8 to 0.9" for total positive confirmed cases and "-2.01 to 6.68" for daily reported positive cases.

We use the Mean Absolute Error (MAE) loss function and the efficient Adam version of stochastic gradient descent. The model is fit for 80 training epochs with a batch size of 8.

From Table 2, one can conclude that, compared with the state-of-the-art detection methods, the proposed algorithm produces nearly better results in case of predictive daily confirmed cases where Estimation 1 and 2 contain the values of LSTM model predictive values and Fb-Prophet model predictive values respectively.

The experimental results show that the proposed approaches not only show the better predictive analysis, but also gives the better results.

Hence, we apply our LSTM model in forecasting relative increased cases day by day as shown in the fig. 6 and find the prediction of number of future cases as result shown in Table 3. It shows that the Relative daily increase is below the zero value with some variance for many days in a row. Initially, it was above the zero and positive. That means, in the start of the pandemic the number of cases coming each day was higher than the number of people getting recovered. And currently the situation is getting better with each day. The LSTM is predicting the decreased values of Relative increased cases that will lead to a significant amount of drop in the Active cases on each day.

To measure the extent of the spread and to define the direction of the pandemic, we defined some more features.

1. **Daily Confirmed = Total Confirmed – Total Confirmed (Shifted by 1 day)**

Since the original data was cumulative, cases for each particular day was calculated using this formula. Similarly, **Daily Deaths** and **Daily Recoveries** could be calculated.

2. **Relative Increased = Daily Confirmed – Daily Recoveries**

We have defined this feature to measure the direction of the pandemic situation. If the slope of the curve of Relative increase is getting steep with each day with a negative value, this indicates that the number of recoveries for each day is increasing comparative to the number of confirmed cases. Please note that we haven't considered Daily Deaths feature while calculating Relative increase since, this feature is only a representation of the relative increment in recoveries in comparison to the increment in the confirmed cases.

3. Confirmed cases at present = Confirmed – (Recoveries + Deaths)

This feature denotes the actual number of active patients on any particular day. If the number of deaths is almost constant throughout a period of time and in that interval Relative increase in cases feature has negative slope, it'll cause the Confirmed cases at present feature to go down as well. Hence, this is a representation of the betterment of pandemic situation in that region.

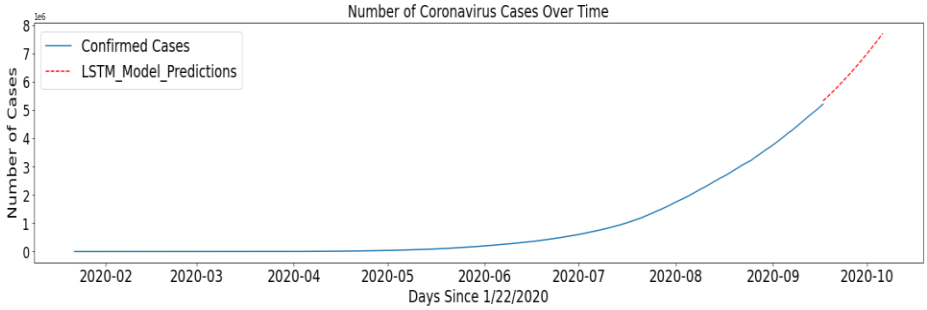

Fig. 5(a): Daily number of positive cases forecasting by LSTM model

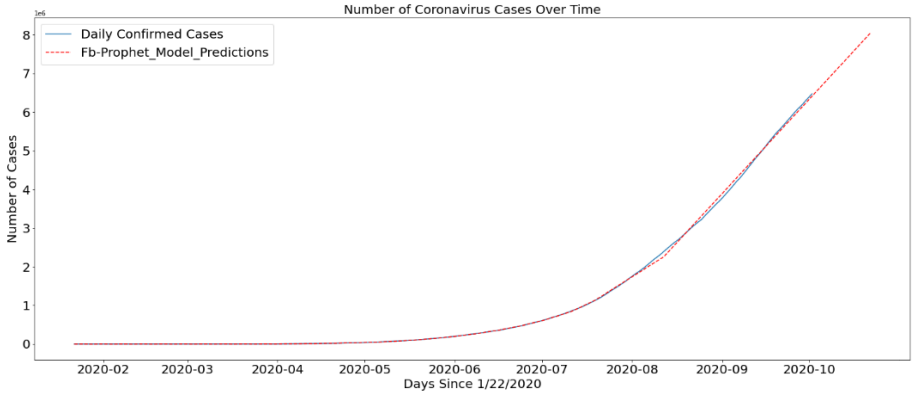

Fig. 5(b): Daily number of positive cases forecasting by Fb-Prophet model

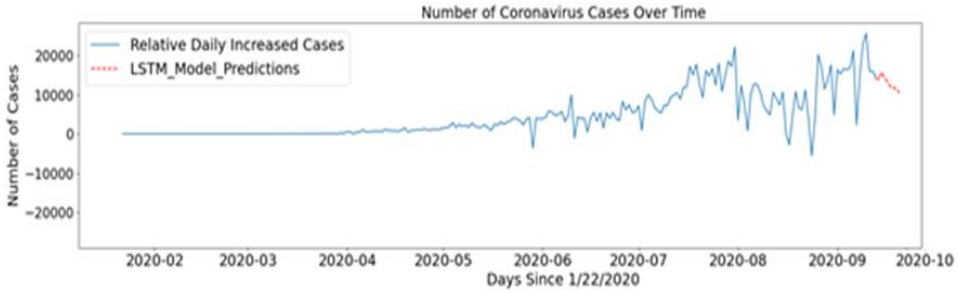

Fig. 6: Number of relative daily increased cases forecasted by LSTM model

Table 1: Comparison for total positive confirmed cases and for daily reported positive cases with estimated cases.

Day	Date	Official data	Estimation	Error percentage (%)
218th	1st September 2020	3769523	3735096	0.91
219th	2nd September 2020	3853406	3820672	0.849
220th	3rd September 2020	3936747	3905083	0.804
221st	4th September 2020	4023179	3989125	0.846
222nd	5th September 2020	4113811	4073695	0.975
218th	1st September 2020	83522	85724	2.56
219th	2nd September 2020	83883	85576	-2.01
220th	3rd September 2020	83341	84411	1.28
221st	4th September 2020	86432	84042	2.76
222nd	5th September 2020	90632	84570	6.68

Table 2: Prediction of confirmed cases in next 10 days where Estimation 1 and 2 are done by LSTM and FbProphet respectively

Day	Date	Estimation 1	Estimation 2
228th	6th September 2020	4160791	4296142
229th	7th September 2020	4250541	4374849
230th	8th September 2020	4343430	4455513
231st	9th September 2020	4438136	4538676
232nd	10th September 2020	4533657	4620950
233rd	11th September 2020	4629022	4703358
234th	12th September 2020	4724805	4784414
235th	13th September 2020	4823705	4865883
236th	14th September 2020	4925053	4944590
237th	15th September 2020	5029179	5025254

Table 3: Prediction of relative increased cases in next 10 days where Estimation is done by LSTM

Day	Date	Estimation
228th	6th September 2020	16414
229th	7th September 2020	16024
230th	8th September 2020	14021
231st	9th September 2020	14112
232nd	10th September 2020	16045
233rd	11th September 2020	17231
234th	12th September 2020	15456
235th	13th September 2020	15241
236th	14th September 2020	14412
237th	15th September 2020	14011

LSTM gives very accurate results (error less than 3%) for short-term prediction (2-4 days) [Table 1]. In the long term this error seems increasing since there are a few subjective factors which also affect the situation of this pandemic at a particular period of time. The factors include awareness about this highly contagious virus. How are people following government's suggestions for prevention? Citizens following social distancing norms and wearing recommended masks properly can lead to a discrepancy in our results as well. And, also if ongoing research and trials on potential vaccines achieve fruitful results. The actual data might be higher than the one being reported because of the inconsistency in tests and their results. Hence, it's recommended to conduct a thorough survey to collect real world data to be more precise about the situation.

4. Conclusion

In this paper, a data-driven forecasting/estimation method has been used to estimate the possible number of positive cases of COVID-19 in India for the next few days. We conducted experiments for current pandemic situation in India to forecast the epidemic peak. The number of positive cases, recovered cases and deceased cases have also been explored by using Exploratory Data analysis. With this time series analysis, we can find how this number of cases can go with predictive trend and how health officials can monitor unlock phases in India. Our estimation Table 1 also shows that our proposed predictive model predicts quite well with very low percentage errors which is better than estimation done by other proposed methods in [9]. Since, the death rate is almost constant in India [Fig 1], the downwards direction of the curve of Confirmed patients at present [Fig 6] represents the Coronavirus pandemic is getting under control in India. If this trend continues for a while, we will surely witness a significant amount of drop in the active cases and this could lead to the end of coronavirus pandemic in India.

In real world, due to various involvement of the government and different public support, there may be the chance of some small peak during the pandemic. In addition, when we forecast the epidemic, the effects of input cases and spatial influence among public cooperation are not taken into account.

5. References

[1] Wang, L., Li, J., Guo, S., Xie, N., Yao, L., Cao, Y., Day, W., Howard, C., Graff, J.C., Gu, T., fu Ji, J., Gu, W., Sun, D., 2020a. Real-time estimation and prediction of mortality caused by covid-19 with patient information based algorithm. Sci. Total Environ.

[2] Huang, C., Wang, Y., Li, X., Ren, L., Zhao, J., Hu, Y., Zhang, L., Fan, G., Xu, J., Gu, X., et al., 2020.

[3] Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China.

[4] PIB, 2020. https://pib.gov.in/pressreleaseiframepage.aspx?prid=1601095.

[5] *Covid-19 India*. (2020). Retrieved September 18, 2020, from MyGov India: https://www.mygov.in/covid-19/?cbps=1

[6] Tobías, A., 2020. Evaluation of the lockdowns for the SARS-CoV-2 epidemic in Italy and Spain after one month follow up. Sci. Total Environ.

[7] Jiang, W., Schotten, H.D., 2020. Deep learning for fading channel prediction. IEEE Open J. Commun. Soc.

[8] Connor, J.T., Martin, R.D., Atlas, L.E., 1994. Recurrent neural networks and robust time series prediction. IEEE Trans. Neural Netw.

[9] Tomar, A., & Gupta, N. (2020). Prediction for the spread of COVID-19 in India and effectiveness of preventive measures. Science of The Total Environment.

[10] Jin, X.; Yu, X.; Wang, X.; Bai, Y.; Su, T.; Kong, J. Prediction for Time Series with CNN and LSTM. In Proceedings of the 11th International Conference on Modelling, Identification and Control (ICMIC2019), Tianjin, China, 13–15 July 2019; Springer.

[11] Chimmula V.K.R., Zhang L. (2020). Time series forecasting of COVID-19 transmission in Canada using LSTM networks. Chaos, Solitons and Fractals

[12] Prophet: automatic forecasting procedure, ([EB/OL]).https://facebook.github.io/prophet/docs/ or https://github.com/facebook/prophet

[13] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput 1997;9(8):1735–80.

[14] Bengio Y, Simard P, Frasconi P. Learning long-term dependencies with gradient descent is difficult. IEEE Trans Neural Netw 1994;5(2):157–66.

[15] Karim F, Majumdar S, Darabi H. Insights into lstm fully convolutional networks for time series classification. IEEE Access 2019;7:67718–25.

[16] Dong E, Du H, Gardner L. An interactive web-based dashboard to track COVID-19 in real time. Lancet Inf Dis. 20(5):533-534. doi: 10.1016/S1473-3099(20)30120-1