# Forecasting Covid-19 Cases in India using Deep CNN LSTM Model

Abhishek Parashar, Yukti Mohan and Neeru Rathee



December 25, 2020

# Forecasting covid-19 cases using Deep CNN LSTM model

Abhishek Parashar[1], Yukti Mohan[1], Neeru Rathee[1]

[a]*Maharaja Surajmal Institute of Technology, C-4, Janakpuri, New Delhi-110058*

**Abstract**

During this worldwide crisis, it is well known that the whole world has been hit by a plenitude of untimely deaths caused during this pandemic. The lockdown in various countries has affected the lives of human beings in many ways. Because of this, it becomes necessary to study the complex interplay of various factors, ranging from macro-scale components such as population density, mortality rate, and recovery rate to singular components such as diabetic patients, smokers, gender, and age. A major concern of higher authorities is the accurate forecasting of Covid-19 cases and the role of various factors in Covid-19 spread to assist the policymakers in understanding the economic situation of the country as well as the factors which affect the current mortality rate. The presented work aims to resolve these concerns by proposing a multivariate hybrid model by taking all the aforementioned factors into account to forecast Covid-19 cases. The proposed model consists of a Convolutional Neural Network (CNN) layer for feature extraction and Long Short Term Memory (LSTM) layers to forecast Covid-19 cases thus exhibiting the inherited advantage of both. The model is trained and tested on the online available dataset acquired from various resources. Experimental results show that the proposed model can forecast the number of cases in the coming month with a mean absolute error equal to 1.78, a training accuracy of 90.63% and validation accuracy of 95.48%.

*Keywords:* Multivariate Analysis, Time series forecasting, CNN, LSTM, Covid-19 Cases, Hybrid Model.

## 1. Introduction

On 31st December 2019, the World Health Organization, China Country Office had registered the first case of pneumonia unknown etiology, which was detected in the Chinese city of Wuhan. This unknown agent was identified as a novel coronavirus on 7th January 2020. Subsequently, the Covid-19 was declared a pandemic on 11th March [25] and the cases have been constantly increasing since then. As of 11th July, four months after the declaration of the pandemic, there are a total of 12,322,395 confirmed cases out of which the total death count has been 556,335 worldwide. Figure 1 displays the total cases reported worldwide. The number of recovered cases is way more than the total number of death cases, signifying the mildness of Coronavirus. India has 820,916 total COVID cases and 22,123 deaths as reported by the World Health Organization [26], making India the third-most country to have the maximum number of cases. Figure 2 shows the Active, Recovered and Death cases of different countries.

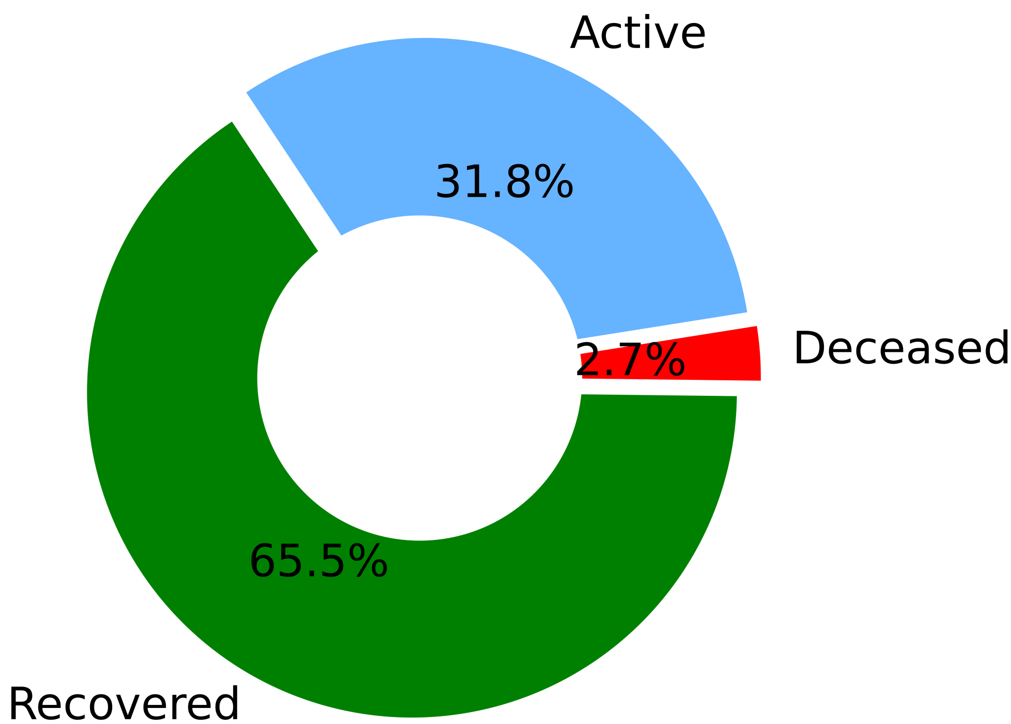


Figure 1: Total cases reported worldwide.

Coronavirus disease 2019 or COVID-19 is a severe acute respiratory syndrome caused by the SARS-CoV-2 pathogen [22]. Mainstream media articles

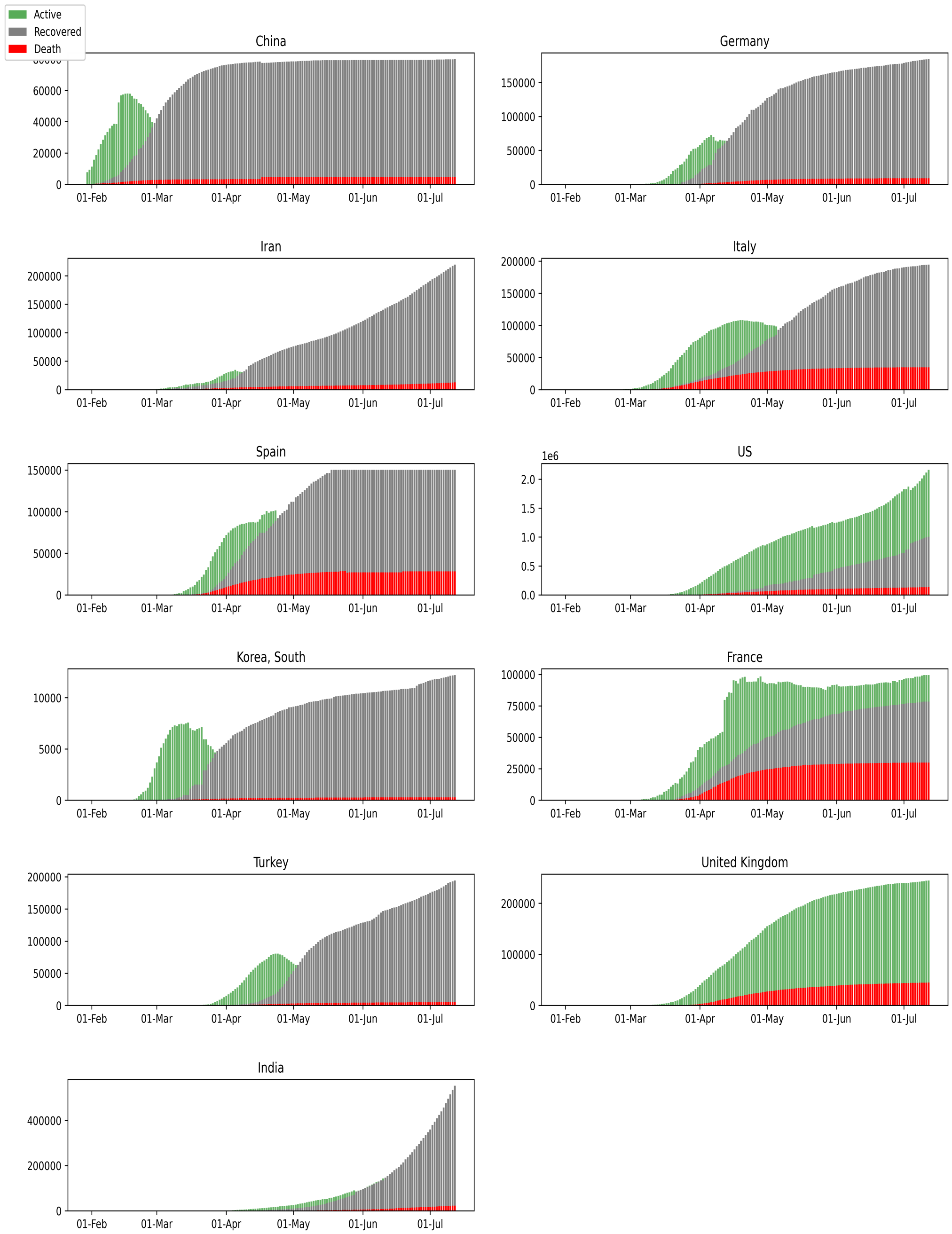


Figure 2: Active, Recovered and Death cases of various countries.

and case reports from various countries [5, 21, 23, 30] indicate that the symptoms of a Coronavirus infected person are like any other common flu. It is also known that it takes around 5-14 days for the symptoms to appear. Thus it becomes extremely difficult to identify an infected person.

The world has experienced other pandemics previously in the past, starting with the 2009 Pandemic Influenza A (H1N1) Virus in the United States [6], Middle East respiratory syndrome coronavirus (MERS-CoV) in 2013 [7, 8], Ebola Virus Disease in West Africa in 2014 [39] and the Brazilian Zika virus in 2016 [12]. It has already been witnessed the inordinate loss of life and the devastation caused by these pandemics. It becomes essential to study the effect of multivariate features associated with the Coronavirus to deal with it presently as well as for similar situations in the future.

Previous works [27, 28, 33, 46] of various authors have provided us with the inspiration to perform multivariate analysis for COVID-19 and thus forecast the future values based on the findings. The presented work displays the insights that the authors have obtained from the analysis of the data available. We have studied the effect on the susceptibility of a person to get infected by Coronovirus because of various factors such as population density, mortality rate, recovery rate, diabetes, smoking tendency, age, and gender.

The results obtained from the study have been verified using previous works and reports as well. We have also predicted the number of cases for 15 days using a combined model of CNN and LSTM with an MAE score equal to 1.78. Figure 3 represents the procedure followed in the presented approach to forecast the number of Covid-19 cases using a hybrid model consisting of CNN and LSTM layers.

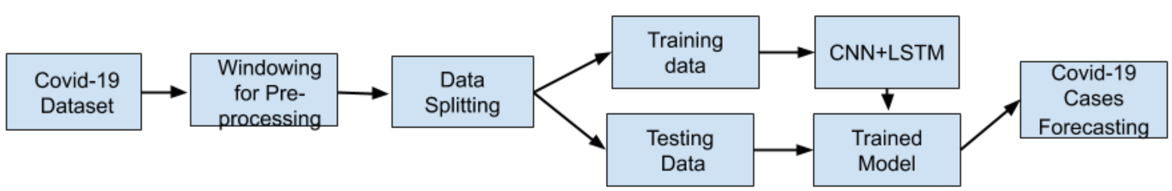


Figure 3: Block diagram representation of CNN and LSTM based Covid-19 forecasting

The rest of the paper is organized as follows: Section 2 explains the preprocessing of data using the Windowing technique. The proposed model consisting of CNN and LSTM layers is explained in Section 3. Section 4 gives a brief description of the dataset used. Section 5 gives a detailed discussion of experimental results of the proposed model. The validation of myth and verity is presented in section 6. Finally, the conclusion has been drawn in section 7.

## 2. Windowing

The time-series data is generally preprocessed before applying to any deep learning model to increase the efficiency of the model. In signal processing and statistics, windowing has been well explored for detecting transient events and time averaging [37]. In the case of time series data, windowing has been used to determine the seasonality and trend of the data to comment on the stationary nature of the data. If the trend and seasonality of data are changing it is said to be non-stationary time-series data [11].

In general, windowing function (also known as apodization function) is zero-valued outside some chosen interval, normally symmetric around the median value thus following Gaussian distribution. The product of any other function with window function is zero for values outside the window interval. Thus, a large dataset can be easily broken into smaller chunks of data to ease the analysis. This property of windowing has been used in the presented work.

In the proposed work, sliding window method [44] is used, wherein the data is divided into smaller parts known as windows, windows are further divided into buffers. This helps in reducing computational complexity and extraction of Important Spatio-temporal features from small chunks using the 1D CNN layer.

## 3. Basics of CNN and LSTM

As mentioned in the introduction, the proposed model is a hybrid model comprising of CNN and LSTM to perform Spatio-temporal analysis. The efficacy of combining the Spatio-temporal features have been well explored by many researchers for time series forecasting [19, 31, 42, 48**?** ]. CNNs are capable enough to extract the spatial components of the data while LSTM networks can extract the temporal components. Thus the presented work makes use of a Hybrid CNN-LSTM model for time series forecasting. In this section, the basics of both of these along with the proposed model are explained.

### *3.1. CNN Layer*

The convolutional neural networks are well known for the state of the art performance in the field of image classification [18, 38, 41], object detection [3, 4, 32], face recognition [16, 29] and other similar tasks [24, 47]. Hu et. al. [16] excellently explains the working of CNNs, their structure, and the way they handle images as inputs. Recently, [40] and [2] present the application of CNNs on the prediction of Covid-19. CNN may be used for feature extraction, classification, and both. In the presented approach, CNN has been used to compute the feature map. The feature map of $i^{th}$ image is computed using the kernel $k$ using the following equation,

$$M[x, y] = (i * k)[x, y] = \sum_p \sum_q k[p, q] i[x - p, y - q] \tag{1}$$

where $x$ represents the row index while $y$ represents the column index of the resultant matrix.

The output of a CNN is given by:

$$\begin{aligned}[m, m, m_c] * [f, f, m_c] = [[&\frac{m + 2p - f}{s} - 1], \\ &[\frac{m + 2p - f}{s} - 1], m_f]\end{aligned} \tag{2}$$

where:
$m$ represents number of rows and columns in input data, $m_c$ is the number of channels, $f$ is the filter size (number of rows and columns is same), $p$ is the padding used, $s$ is the stride used and $n_f$ is the number of filters applied to extract the feature map.

In the proposed network only one layer of CNN network is used. During forward propagation, two major operations are performed on the input data. The first operation is convolution of input data $A$ with the weights $W$ of the layer along with addition of bias $b$ to result in an intermediate value, $Z$ as shown in the following equation

$$Z^{[l]} = W^{[l]}.A^{[l]} + b^{[l]} \tag{3}$$

The second operation is to calculate the output of the layer by the application of an activation function called $a$ as shown in Equation **4**.

$$A^{[l]} = a^{[l]}(Z^{[l]}) \tag{4}$$

In backward propagation, derivatives of each of the output variables is calculated and used to update the values of the parameters, using a process called gradient descent.

$$dZ^l = \frac{\delta L}{\delta Z[l]}, dW^l = \frac{\delta W}{\delta A[l]}, db^l = \frac{\delta L}{\delta b[l]}, dA^l = \frac{\delta L}{\delta A[l]} \tag{5}$$

*3.2. LSTM*

Long Short Term Memory neural networks are used for various sequential processing tasks such as language modeling [17, 20, 36], machine translation [1, 13], speech recognition [14, 15, 35, 49] and others [9, 34, 45]. Recently LSTMs have been well explored for covid-19 cases forecasting. Inspired from the promising results for handling of time-series data by LSTM, authors have also explored the same.

A single LSTM unit consists of three gates namely, Input gate, Forget gate and Output gate. Mathematically they are represented as follows:

- Input Gate:
$$i_x = sigmoid(W_i[A_{x-1}, t_x] + b_i) \tag{6}$$

- Forget Gate:
$$f_x = sigmoid(W_f[A_{x-1}, t_x] + b_f) \tag{7}$$

- Output Gate:
$$o_x = sigmoid(W_o[A_{x-1}, t_x] + b_o) \tag{8}$$

where,
$i_x$, $f_x$, $o_x$ represent the Input, Forget and Output gate respectively, $W_t$ represents the vector of weights for the respective gates where $t \in i, f, o$, $A_{x-1}$ represents the output of previous LSTM block (at time stamp x-1), $t_x$ represents the input at current timestamp, $b_t$ represents the bias for the respective gates where $t \in i, f, o$.

The output of the LSTM unit is given by:

1. Candidate cell at timestamp x:

$$\hat{c}_x = tanh(W_c[A_{x-1}] + b_c) \tag{9}$$

2. Cell memory at timestamp x:

$$c_x = f_x * c_{x-1} + i_x * \hat{c}_x \tag{10}$$

3. Output at timestamp x:

$$A_x = o_x * tanh(c^x) \tag{11}$$

### *3.3. The Proposed Model*

The proposed model has a 1-dimensional Convolutional layer as the input layer which is being used to simply stride on the data provided and extract important features. These features are then fed to the LSTM network, which recursively learns about the data, thus becoming aware of the information contained by the data. This mixed-model of CNN and LSTM layers is lightweight, accurate and has lesser computation complexity.

| Layer Type | Output shape | Param# |
|---|---|---|
| conv1d (Conv1D) | (None, None, 32) | 192 |
| lstm (LSTM) | (None, None, 64) | 24832 |
| lstm_1 (LSTM) | (None, None, 64) | 33024 |
| dense (Dense) | (None, None, 30) | 1950 |
| dense_1 (Dense) | (None, None, 10) | 310 |
| dense_2 (Dense) | (None, None, 1) | 11 |
| lambda (Lambda) | (None, None, 1) | 0 |

Table 1: Model structure of of the proposed model.

The parameters of interest such as the total number of trainable parameters, the loss (mean absolute error) incurred by model, and the time elapsed (in seconds) in training the model have been calculated for the proposed model.

The architecture of the model is shown in figure 4. It comprises of 1 CNN layer to gather important features and patterns present in the data. The next 2 LSTM layers play the most important role in learning the patterns and sequences to make the prediction. The 2 dense layers after that fulfill the same purpose. They are used since the complexity of a dense layer is less than that of an LSTM layer. The last dense layer is the output layer which gives out the final prediction.

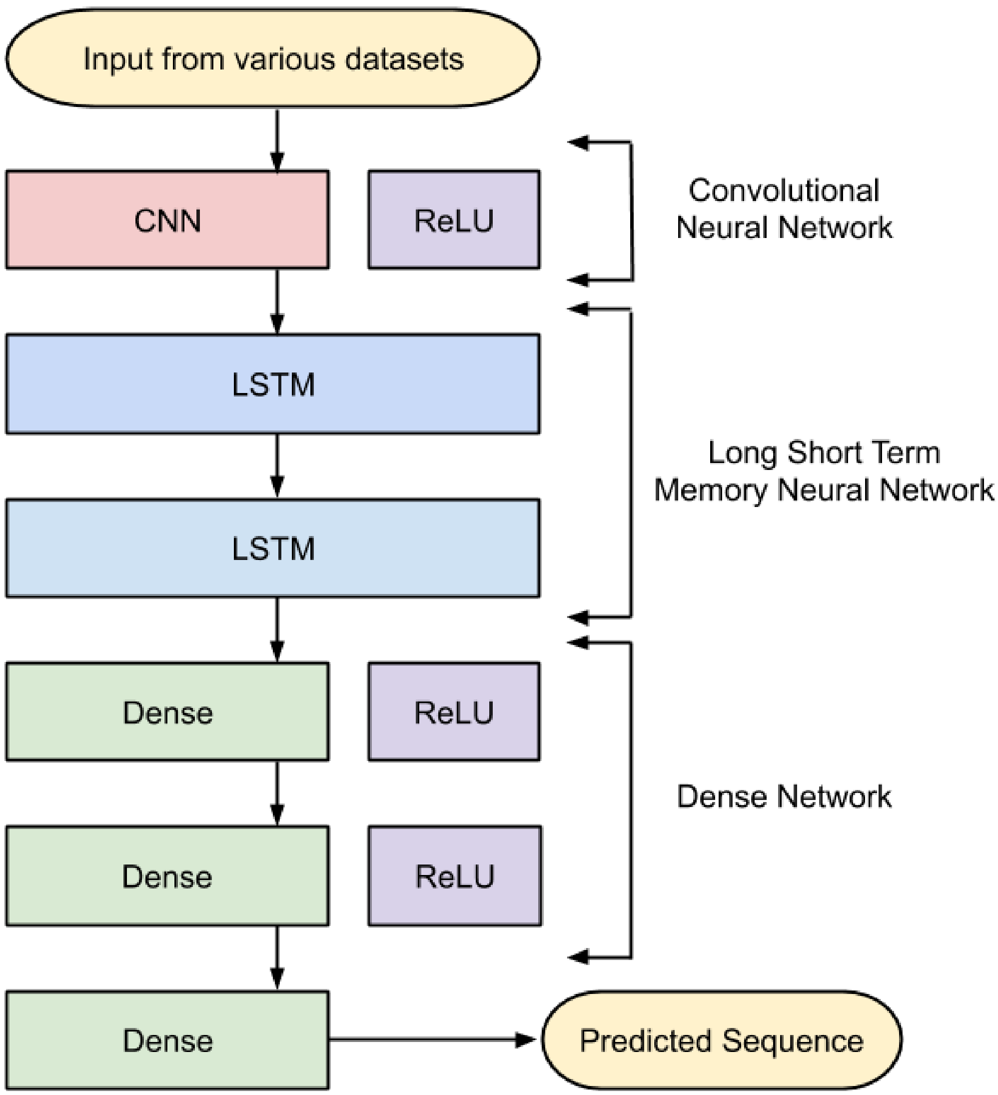


Figure 4: The Proposed Model Architecture.

## 4. Dataset Description

The proposed network has been trained using the database available online.

**1. Ministry of Health and Family Welfare Government of India:** The details regarding day-wise total cases, confirmed cases, active as well as recovered cases has been taken from the site of the Ministry of Health and Family Welfare, Government of India. This site also provides the data regarding daily testing done by the Indian Council of Medical research.

**2. Dataset regarding state-wise data:** The dataset to analyse the trend of the Covid-19 data in various states is taken from the website with link https://www.covid19india.org/.

**3. Country-wise data:** The dataset to analyse myths regarding the Covid-19 pandemic has been taken from the link https://www.worldometers.info/.

**4. Novel Corona Virus 2019 Dataset** This dataset is available on Kaggle for the research community. In the presented work, the details regarding age group and state-wise health facilities have been taken from this dataset.

**5. Our World in Data:** This data has many variables of potential interest, out of which data regarding diabetic patients, smoking, and non-smoking persons, etc. has been used in the presented approach.

## 5. Experimental Results and Discussion

Several predictions have been done by researchers in the past to predict Covid-19 cases by analyzing the confirmed cases to date. The present work not only focuses on the confirmed cases but also considers various other susceptible features, which might affect corona spread and predicted Covid-19 cases using a hybrid CNN-LSTM model. The research aims to assist the government in decision-making and policy-making to ease the life of citizens at the time of this pandemic spread.

### *5.1. Covid-19 forecasting*

The proposed model is trained on various datasets of 171 days (starting from $30^{th}$ January to $21^{st}$ July) available online. The data is split among training, testing, and validation set. Out of 275 days, 165 days has been used for training ($30^{th}$ January to $12^{th}$ July), 82 days ($13^{th}$ July to $2^{nd}$ October) for testing, and 28 days ($3^{rd}$ October to $30^{th}$ October) for validation, which accounts for 60% training data, 30% testing data and 10% validation data, respectively. The data is first preprocessed using the windowing technique with a window size of 30, batch size of 32, and shuffle buffer size of 1000. The filtered data is then fed to the hybrid CNN-LSTM model for training.

As mentioned earlier, the number of layers selection is the major challenge faced by researchers. To address this challenge in an unbiased manner, the loss for all the models is computed at various learning rates and different number of epochs. The loss characteristics for all the five versions are shown in figure 5 for a similar learning rate and a similar number of epochs. The left side of the figure shows the loss in terms of mean absolute error (MAE) for various learning rates [10] [43].

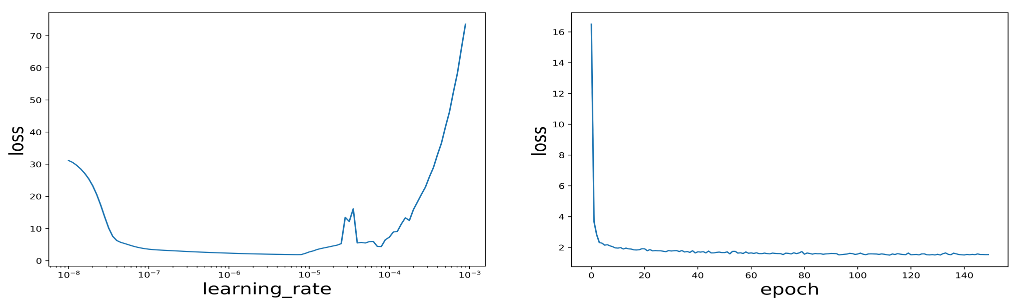


Figure 5: Loss characteristics of the proposed model.

To have a clearer picture of the proposed model, a graph is plotted taking three different features of the model: number of parameters, loss (MAE), and time taken (figure **??**). It may be observed that the proposed model takes 700 seconds to execute on $60,319$ parameters accounting for 1.78 MAE loss. It has a reasonable amount of parameters to train and executes quickly with decent loss. The specifications of the device are as follows: i7 processor, 8GB RAM, 4GB Nvidia 1050ti graphics card.

In order to predict future cases, the authors have performed a multivariate analysis of the following factors: recovery rate, mortality rate, diabetic patients, gender, smoking tendency, and population density. The forecasting performed by the presented model is shown in figure 6.

The training and validation accuracy for this model are shown in figure 7.

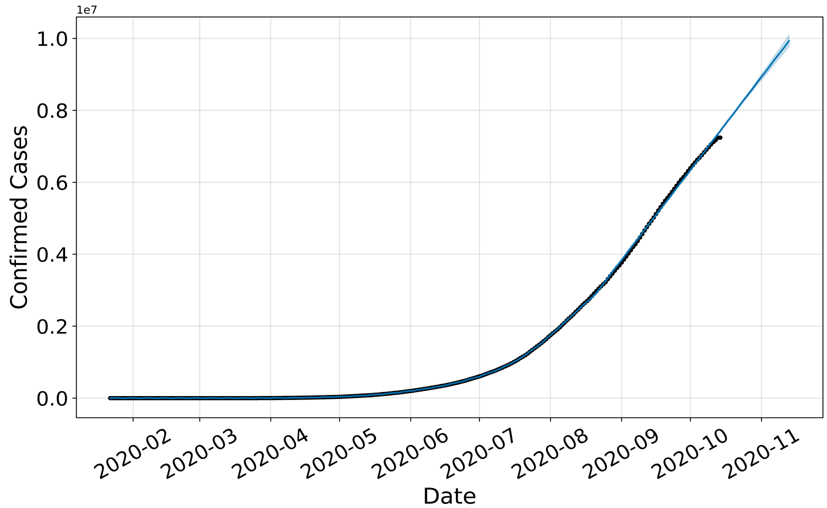


Figure 6: Comparison of Confirmed Vs Predicted Cases.

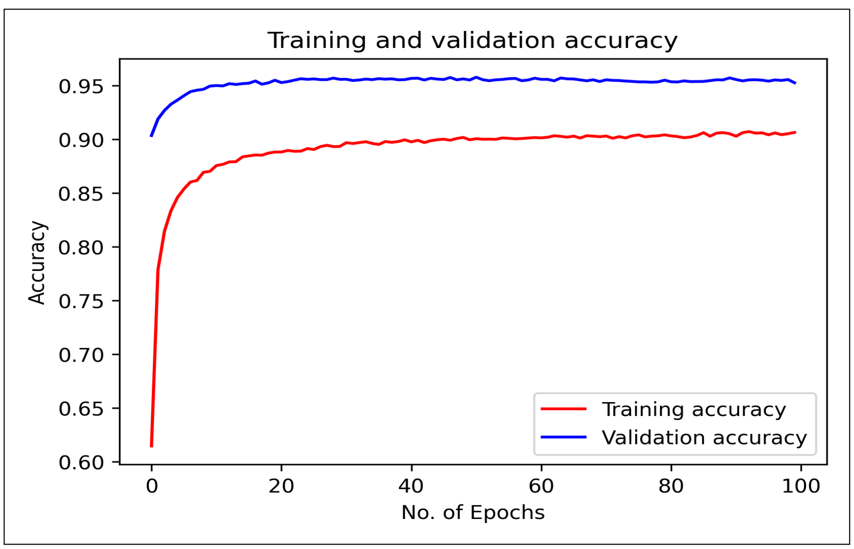


Figure 7: Training and Validation accuracy for the presented model.

## 6. Conclusion

We have presented a multivariate analysis based on a Hybrid CNN-LSTM model, with a training accuracy and validation accuracy equal to 90.63 and 95.48, respectively. The accuracy plot is shown in figure 7. The mean absolute error (mAE) equal to 1.78 and the time taken to compute the predictions using the given model is 700 seconds which proves the efficiency of the model in terms of time.

## References


[1] B. Athiwaratkun and J. W. Stokes. Malware classification with lstm and gru language models and a character-level cnn. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2482–2486. IEEE, 2017.

[2] S. U. K. Bukhari, S. S. K. Bukhari, A. Syed, and S. S. H. SHAH. The diagnostic evaluation of convolutional neural network (cnn) for the assessment of chest x-ray of patients infected with covid-19. *medRxiv*, 2020.

[3] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos. A unified multi-scale deep convolutional neural network for fast object detection. In *European conference on computer vision*, pages 354–370. Springer, 2016.

[4] Z. Cai and N. Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018.

[5] M. Cascella, M. Rajnik, A. Cuomo, S. C. Dulebohn, and R. Di Napoli. Features, evaluation and treatment coronavirus (covid-19). In *Statpearls [internet]*. StatPearls Publishing, 2020.

[6] S. Cauchemez, C. A. Donnelly, C. Reed, A. C. Ghani, C. Fraser, C. K. Kent, L. Finelli, and N. M. Ferguson. Household transmission of 2009 pandemic influenza a (h1n1) virus in the united states. *New England Journal of Medicine*, 361(27):2619–2627, 2009.

[7] S. Cauchemez, C. Fraser, M. D. Van Kerkhove, C. A. Donnelly, S. Riley, A. Rambaut, V. Enouf, S. van der Werf, and N. M. Ferguson. Middle east

respiratory syndrome coronavirus: quantification of the extent of the epidemic, surveillance biases, and transmissibility. *The Lancet infectious diseases*, 14(1):50–56, 2014.

[8] S. Cauchemez, P. Nouvellet, A. Cori, T. Jombart, T. Garske, H. Clapham, S. Moore, H. L. Mills, H. Salje, C. Collins, et al. Unraveling the drivers of mers-cov transmission. *Proceedings of the national academy of sciences*, 113(32):9081–9086, 2016.

[9] G. S. Chadha, A. Panambilly, A. Schwung, and S. X. Ding. Bidirectional deep recurrent neural networks for process fault classification. *ISA transactions*, 2020.

[10] T. Chai and R. R. Draxler. Root mean square error (rmse) or mean absolute error (mae)?–arguments against avoiding rmse in the literature. *Geoscientific model development*, 7(3):1247–1250, 2014.

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

[12] F. R. Cugola, I. R. Fernandes, F. B. Russo, B. C. Freitas, J. L. Dias, K. P. Guimarães, C. Benazzato, N. Almeida, G. C. Pignatari, S. Romero, et al. The brazilian zika virus strain causes birth defects in experimental models. *Nature*, 534(7606):267–271, 2016.

[13] J. Gehring, M. Auli, D. Grangier, and Y. N. Dauphin. A convolutional encoder model for neural machine translation. *arXiv preprint arXiv:1611.02344*, 2016.

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

[15] S. Han, J. Kang, H. Mao, Y. Hu, X. Li, Y. Li, D. Xie, H. Luo, S. Yao, Y. Wang, et al. Ese: Efficient speech recognition engine with sparse lstm on fpga. In *Proceedings of the 2017 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*, pages 75–84, 2017.

[16] G. Hu, Y. Yang, D. Yi, J. Kittler, W. Christmas, S. Z. Li, and T. Hospedales. When face recognition meets with deep learning: an

evaluation of convolutional neural networks for face recognition. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 142–150, 2015.

[17] M. À. India Massana, J. A. Rodríguez Fonollosa, and F. J. Hernando Pericás. Lstm neural network-based speaker segmentation using acoustic and language modelling. In *INTERSPEECH 2017: 20-24 August 2017: Stockholm*, pages 2834–2838. International Speech Communication Association (ISCA), 2017.

[18] N. Jmour, S. Zayen, and A. Abdelkrim. Convolutional neural networks for image classification. In *2018 International Conference on Advanced Systems and Electric Technologies (IC_ASET)*, pages 397–402. IEEE, 2018.

[19] M. Khodayar and J. Wang. Spatio-temporal graph deep neural network for short-term wind speed forecasting. *IEEE Transactions on Sustainable Energy*, 10(2):670–681, 2018.

[20] B. Krause, L. Lu, I. Murray, and S. Renals. Multiplicative lstm for sequence modelling. *arXiv preprint arXiv:1609.07959*, 2016.

[21] T. G. Ksiazek, D. Erdman, C. S. Goldsmith, S. R. Zaki, T. Peret, S. Emery, S. Tong, C. Urbani, J. A. Comer, W. Lim, et al. A novel coronavirus associated with severe acute respiratory syndrome. *New England journal of medicine*, 348(20):1953–1966, 2003.

[22] C.-C. Lai, T.-P. Shih, W.-C. Ko, H.-J. Tang, and P.-R. Hsueh. Severe acute respiratory syndrome coronavirus 2 (sars-cov-2) and corona virus disease-2019 (covid-19): the epidemic and the challenges. *International journal of antimicrobial agents*, page 105924, 2020.

[23] Y. Liu, L.-M. Yan, L. Wan, T.-X. Xiang, A. Le, J.-M. Liu, M. Peiris, L. L. Poon, and W. Zhang. Viral dynamics in mild and severe cases of covid-19. *The Lancet Infectious Diseases*, 2020.

[24] S. Modi, J. Bhattacharya, and P. Basak. Estimation of energy consumption of electric vehicles using deep convolutional neural network to reduce driver's range anxiety. *ISA transactions*, 98:454–470, 2020.

[25] W. H. Organization et al. Who director-general's opening remarks at the media briefing on covid-19-11 march 2020, 2020.

[26] W. H. Organization, W. health organization, et al. Coronavirus disease (covid-2019) situation reports, 2020.

[27] H. Panwar, P. Gupta, M. K. Siddiqui, R. Morales-Menendez, and V. Singh. Application of deep learning for fast detection of covid-19 in x-rays using ncovnet. *Chaos, Solitons & Fractals*, page 109944, 2020.

[28] D. Parbat et al. A python based support vector regression model for prediction of covid19 cases in india. *Chaos, Solitons & Fractals*, page 109942, 2020.

[29] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. 2015.

[30] J. Peiris, S. Lai, L. Poon, Y. Guan, L. Yam, W. Lim, J. Nicholls, W. Yee, W. Yan, M. Cheung, et al. Coronavirus as a possible cause of severe acute respiratory syndrome. *The Lancet*, 361(9366):1319–1325, 2003.

[31] H. Qiao, T. Wang, P. Wang, S. Qiao, and L. Zhang. A time-distributed spatiotemporal feature learning method for machine health monitoring with multi-sensor time series. *Sensors*, 18(9):2932, 2018.

[32] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[33] B. K. Sahoo and B. K. Sapra. A data driven epidemic model to analyse the lockdown effect and predict the course of covid-19 progress in india. *Chaos, Solitons & Fractals*, page 110034, 2020.

[34] M. Sayah, D. Guebli, Z. Al Masry, and N. Zerhouni. Robustness testing framework for rul prediction deep lstm networks. *ISA transactions*, 2020.

[35] H. Soltau, H. Liao, and H. Sak. Neural speech recognizer: Acoustic-to-word lstm model for large vocabulary speech recognition. *arXiv preprint arXiv:1610.09975*, 2016.

[36] D. Soutner and L. Müller. Application of lstm neural networks in language modelling. In *International Conference on Text, Speech and Dialogue*, pages 105–112. Springer, 2013.

[37] L. Stanković, T. Alieva, and M. J. Bastiaans. Time–frequency signal analysis based on the windowed fractional fourier transform. *Signal processing*, 83(11):2459–2468, 2003.

[38] A. O. Tarasenko, Y. V. Yakimov, and V. N. Soloviev. Convolutional neural networks for image classification. In *Computer Science & Software Engineering: Proceedings of the 2nd Student Workshop (CS&SE@ SW 2019), Kryvyi Rih, Ukraine, November 29, 2019*, number 2546, pages 101–114. CEUR Workshop Proceedings, 2019.

[39] W. E. R. Team. Ebola virus disease in west africa—the first 9 months of the epidemic and forward projections. *New England Journal of Medicine*, 371(16):1481–1495, 2014.

[40] S. Wang, B. Kang, J. Ma, X. Zeng, M. Xiao, J. Guo, M. Cai, J. Yang, Y. Li, X. Meng, et al. A deep learning algorithm using ct images to screen for corona virus disease (covid-19). *MedRxiv*, 2020.

[41] Y. Wei, W. Xia, M. Lin, J. Huang, B. Ni, J. Dong, Y. Zhao, and S. Yan. Hcp: A flexible cnn framework for multi-label image classification. *IEEE transactions on pattern analysis and machine intelligence*, 38(9):1901–1907, 2015.

[42] C. Wen, S. Liu, X. Yao, L. Peng, X. Li, Y. Hu, and T. Chi. A novel spatiotemporal convolutional long short-term neural network for air pollution prediction. *Science of the total environment*, 654:1091–1099, 2019.

[43] C. J. Willmott and K. Matsuura. Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate research*, 30(1):79–82, 2005.

[44] A. D. Wyner and J. Ziv. The sliding-window lempel-ziv algorithm is asymptotically optimal. *Proceedings of the IEEE*, 82(6):872–877, 1994.

[45] S. Xiang, Y. Qin, C. Zhu, Y. Wang, and H. Chen. Lstm networks based on attention ordered neurons for gear remaining life prediction. *ISA transactions*, 2020.

[46] M. Yadav, M. Perumal, and M. Srinivas. Analysis on novel coronavirus (covid-19) using machine learning methods. *Chaos, Solitons & Fractals*, page 110050, 2020.

[47] Y. Yang, H. Zheng, Y. Li, M. Xu, and Y. Chen. A fault diagnosis scheme for rotating machinery using hierarchical symbolic analysis and convolutional neural network. *ISA transactions*, 91:235–252, 2019.

[48] B. Yu, H. Yin, and Z. Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. *arXiv preprint arXiv:1709.04875*, 2017.

[49] A. Zeyer, P. Doetsch, P. Voigtlaender, R. Schlüter, and H. Ney. A comprehensive study of deep bidirectional lstm rnns for acoustic modeling in speech recognition. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2462–2466. IEEE, 2017.