COVID-19 Growth Prediction using Multivariate Long Short Term Memory

Novanto Yudistira

***Abstract*—Coronavirus disease (COVID-19) spread forecasting is an important task to track the growth of the pandemic. Existing predictions are merely based on qualitative analyses and mathematical modeling. The use of available big data with machine learning is still limited in COVID-19 growth prediction even though the availability of data is abundance. To make use of big data in the prediction using deep learning, we use long short-term memory (LSTM) method to learn the correlation of COVID-19 growth over time. The structure of an LSTM layer is searched heuristically until the best validation score is achieved. First, we trained training data containing confirmed cases from around the globe. We achieved favorable performance compared with that of the recurrent neural network (RNN) and vector autoregression (VAR) method with a comparable low validation error. The evaluation is conducted based on graph visualization and root mean squared error (RMSE). We found that it is not easy to achieve the same quantity of confirmed cases over time. However, LSTM provide a similar pattern between the actual cases and prediction. In the future, our proposed prediction can be used for anticipating forthcoming pandemics. The code is provided here: https://github.com/cbasemaster/lstmcorona**

***Index Terms*—COVID-19; deep learning; LSTM; prediction; time-series.**

I. Introduction

THE COVID-19 outbreak first occurred in China and then gradually spread around the world. The factors that cause the outbreak are still in the discussion phase. However, many countries have been anticipating the transmission using social distancing and activity restrictions except Sweden [16]. Since then, not many predictions are available except qualitative and statistical analyses [10][11][12][13]. For single series, Autoregressive Integrated Moving Average (ARIMA) is usually used for forecasting in various applications [22][23]. However, ARIMA is limited by looking at one variable's pattern and more general method is required to capture multivariate data. Auto-regressive model like vector autoregression (VAR) is also popular option to capture multivariate time series pattern by stochastic different equation [17][18][19].

Although long short-term memory (LSTM) has been applied in various and diverse time-series topics, such as stock prediction, weather, signal processing [21] and consumer, findings on the exact manifestations of COVID-19 are still limited. LSTM was used to predict the end of the pandemic in China using a small sample, which only represented local characteristics of the outbreak [7]. Moreover, their training dataset is the 2003 SARS epidemic statistics, which is different from COVID-19 epidemic.

An artificial neural network (ANN) has recently been given attention after deep learning on image classification [14]. In particular, for prediction or forecasting, researchers were re-exploring the old models of ANN for time series prediction, such as a recurrent neural network (RNN) and LSTM. The return of ANN aims to solve the drawback of statistical methods. It performs better than statistical methods in terms of prediction accuracy [1]. Time-series data that contain dynamic information over time are suitable to be captured by the RNN family with which various time-series applications have been adopted [20][24]. One particular property of the RNN family is that every timestamp's activation is stored in the internal state to construct a temporal model [2]. However, the weakness of RNN is dealing with long-sequence data insisting on the inability to handle the vanishing gradient problem during the learning process[3]. To solve this problem, Schmidhuber has proposed the LSTM, which contains the input, output, and forget gate better to capture the correlation of data with long-term dependencies [4]. However, the LSTM parameter needs to be optimized depending on data characteristics by choosing the number of layers or hidden units, especially for highly complex data, non-linear, and long [5].

This paper proposes an LSTM framework that handles the nonlinearity and complexity of COVID-19 time-series data. The LSTM framework contains layers of LSTM cells, followed with sigmoid activation and dropout regularization. Each LSTM layer handles different resolutions of temporal space for specific tasks. Input information is forwarded through layers until the linear layer produces a time-series output. In specific, this framework is run to solve the regression problem. We can gradually add layers and hidden units to increase connections between hidden units horizontally and vertically and improve accuracy depending on the dataset's complexity. It captures temporal dynamics hierarchically and sequentially on complicated and long-sequence data [5].

We have prepared a learning scenario that can train COVID-19 spread over time. We split training and testing data from each country for all samples. Specifically, the sequence of the selected country is split into input training and output training or label. The best LSTM architecture and hyper-parameters are searched heuristically during validation. For evaluation, we also compare the LSTM model with the precedent model of RNN.

The paper is organized as follows. Section 1 presents an introduction to the study. Section 2 elaborates on our motivation for applying LSTM to predict COVID-19 growth. Section 3 describes the methodology used in this research, starting from preprocessing, learning algorithm, training, and validation strategy. Section 4 presents the experimental results. Section 5 provides discussions. Finally, Section 6 displays the conclusion of the study.

Novanto Yudistira is with Intelligent System laboratory, Faculty of Computer Science, Universitas Brawijaya, Indonesia e-mail: (yudistira@ub.ac.id).

II. Related Works and Motivation

COVID-19 growth data include temporal information presenting the dynamic number of confirmed infected people over time. Thus, it is essential to check whether the policy undertaken is effective or not during a pandemic. Studying how to effectively treat pandemics by looking into previous and global patterns can also be performed. Moreover, in real-time, the end of a pandemic can be suitably predicted, given the abundance of available training data. However, by its natural characteristics, COVID-19 time-series data are complex, highly non-linear, long interval (several days and months), and high variance, making it difficult for traditional statistical methods to predict [6]. Furthermore, several hidden layers and nonlinearity are advantageous in terms of graph accuracy by capturing the coarse and fine dynamics of the growth pattern [8].

Multivariate datasets as data sources for training the model are beneficial because many factors influence pandemic growth. The cause of confirmed cases can be seen from several parameters, not only stand-alone variables. In this case, for a preliminary, the number of confirmed cases, death, recovered, latitude, and longitude are used as parameters. There are relationships between geographical parameters like latitude and longitude with the number of confirmed cases in the world based on previous findings [15]. In the future, it can be more beneficial to add new parameters such as the UV index, humidity, and population density.

The use of LSTM to overcome the drawback such as non-linearity, long series, and heterogeneous properties, basically starts from the RNN problem. The main drawback of RNN is vanishing gradients, which can be handled by LSTM. Due to the use of the hyperbolic tangent as the activation function, the derivative of the function inside RNN cell is in the range of 0 to 1. If the gradient is minimal, then there is no effect on the update.

III. Methods

A. Data Preprocessing

We use a MinMaxScaler to normalize data because LSTM is very sensitive to normalization, especially capturing time-series data. First, we transform data into the same scale and thus avoid bias during training and validation. The scaling function is defined as

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{1}$$

where X is input training dataset and X_{scaled} is output of normalized training dataset.

B. Model

LSTM is an extension of RNN that uses a forgetting mechanism to handle long-sequence inputs. In the LSTM cell, the memory cell is divided into memory cell c_t and working cell h_t. Memory cells are responsible for the retention of the sequence controlled with forgetting gate f_t. The working memory h_t is used as the output of each memory cell, and output gate o_t controls the portion of c_t to be remembered. The input gate i_t controls the portion the former state h_{t-1} and the current input x_t to be remembered in the memory cell. The former state h_{t-1} and current input x_t are jointly fed to the non-linear activation function $tanh$ and thus not static even after a linear combination. The previously described LSTM cell shown in Figure 3 is elaborated as follows:

$$\begin{aligned}
f_t &= \sigma(w_f[h_{t-1}, x_t] + b) \\
i_t &= \sigma((w_i[h_{t-1}, x_t] + b_i) \\
C_t &= tanh(w_c \times [h_{t-1}, x_t] + b_c)) \\
c_t &= f_t \times c_{t-1} + i_t \times c_t \\
o_t &= \sigma(w_o[h_{t-1}, x_t] + b) \\
h_t &= o_t \times tanh(c_t)
\end{aligned} \tag{2}$$

Our architecture contains one to four hidden layers with a hidden unit of 1-30 each. An example of an LSTM architecture with two hidden layers is shown in Figure 2. The last layer is a linear layer that outputs 100-sequence prediction. The linear layer's output is fed to the activation function of sigmoid to guarantee a range of 0 to 1. We use a dropout of 0.1 to avoid overfitting.

As shown in Figure 1, the framework of learning and evaluation consists of an input, fed into the model, and output. The input and output are both in the form of daily cases. The result of daily cases is then finally accumulated to show the growth curve over time. In the training phase, a 100-sequence input is split into the 1^{st} to 67^{th}-day as input graph and 68^{th} to 100^{th}-day as the label. Input is normalized before processing using the normalization. The validation and testing data are normalized using the scaling factors obtained from training data before feeding them into the trained model.

TABLE I: Composition of training and testing data. *) has more than one province or state

Training data	Validation Data
'China*','Germany','Australia*','Brazil', 'US','Belgium','Spain','Italy','France*', 'Malaysia','Vietnam','Iran','UEA', 'Singapore','Thailand','Korea, South', 'Japan','Iran', 'Netherlands*','Russia','Chile', 'India','Greece','Mexico', 'Mongolia','Philippines','New Zealand', 'South Africa','Botswana','Uruguay', 'Paraguay','Madagascar','Peru', 'Portugal', 'Denmark*','Hungary','Kenya','Ireland','Israel', 'Norway','Mauritius','Rwanda','Iceland', 'Kazakhstan','Switzerland','Cyprus','Zimbabwe'	'Indonesia','Sweden', 'Saudi Arabia','Argentina'

C. Training Data

We use 100 regions (countries/provinces/states) as the training data and four countries as validation data. The composition training and validation of selected countries are shown in Table I. The parameters of the dataset are shown in Table II. To provide the input and label, a sequence for each sample is divided into two parts, of which the first part is from January 22, 2020 to March 29, 2020 as the input training and from March 30, 2020 to May 1, 2020 as output label.

D. Prediction Accuracy Measurement

To measure the loss function and prediction performance of the trained model, the mean squared error and root mean squared error (RMSE), respectively, are employed. Booth

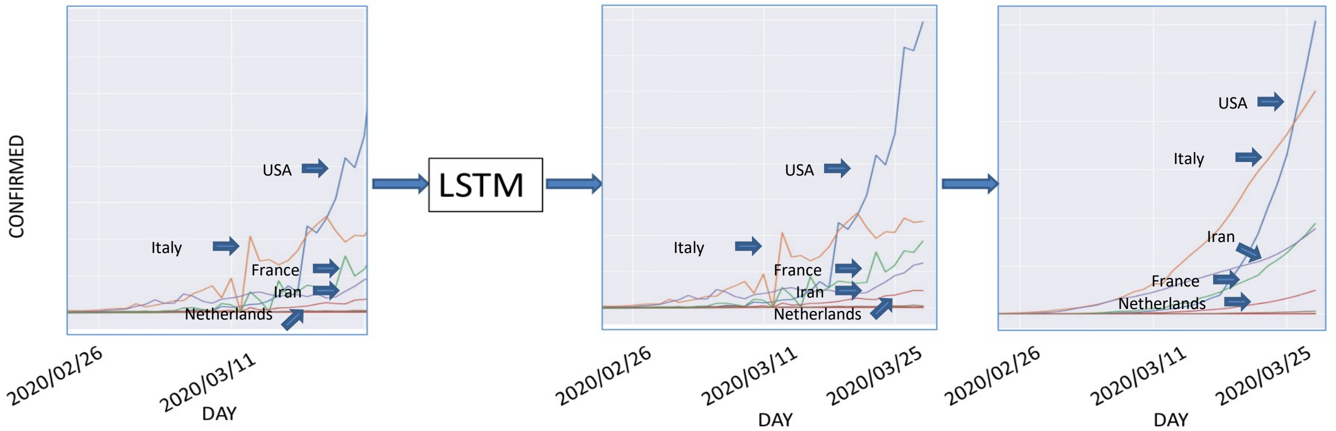

Fig. 1. Framework of training and testing. The input is composed of 67 timestamps and the output is composed of 100 timestamps of daily cases. The final output is then accumulated sequentially.

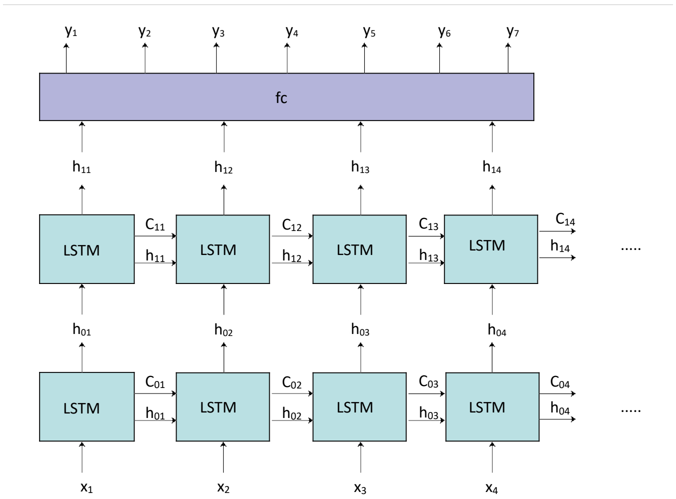

Fig. 2. LSTM model architecture with two hidden layers and fully connected layer (fc).

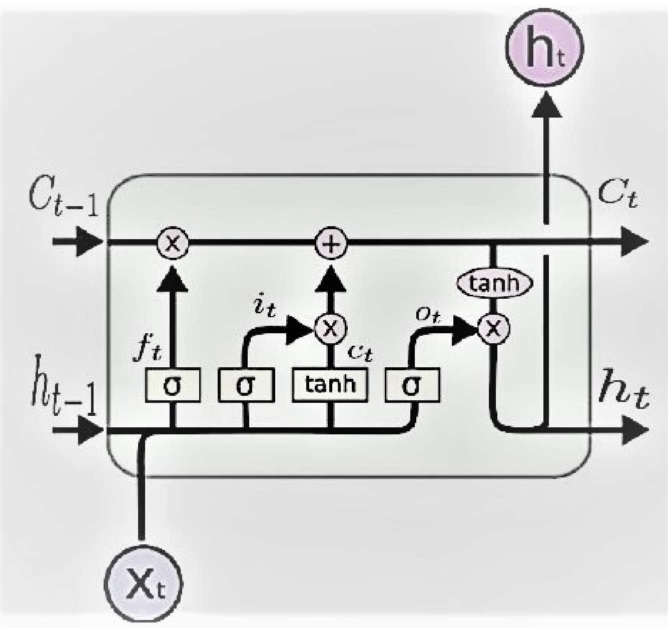

Fig. 3. Memory Cell.

TABLE II: Parameters

Dataset Parameter	Unit
Confirmed cases	number of people
Death cases	number of people
Recovered cases	number of people
Latitude	degree
Longitude	degree

mean squared error and RMSE are basically the measurement of the difference between the actual cases and prediction. The RMSE is given by

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(P-A)^2} \tag{3}$$

Note that P isthe prediction sequence and A is the actual or ground-truth sequence.

IV. Experimental Results

We used Adam optimizer for the hyper-parameters with a learning rate of 0.001 and iteration number of 10,000. These settings gave satisfying results. We have prepared several experimental setups. First, training and testing are performed once to predict a long growth curve starting from January 22, 2020 to May 1, 2020. Second, training and testing are performed five times to reduce bias due to random initialization. In this way, we can also gather the mean and interval from several curve predictions. Meanwhile, quantitative evaluation is done using RMSE. RMSE evaluates each country in five trials. We also evaluate how the number of hidden states influences RMSE. Furthermore, an evaluation of the optimum number of hidden layers was conducted, in which we used a fixed number of 30 hidden states. Finally, we test to foresee when the number of daily confirmed cases is decreasing. To realize this, we predict the future growth of confirmed cases given the last 67 days of known time-series input. The best number of hidden states and layers revealed from the validation are set for testing, and a sample country of Indonesia is used as the testing input.

A. Validation Results

Figure 4 shows the validation results of Indonesia. The prediction curve has an exponentially similar pattern to the

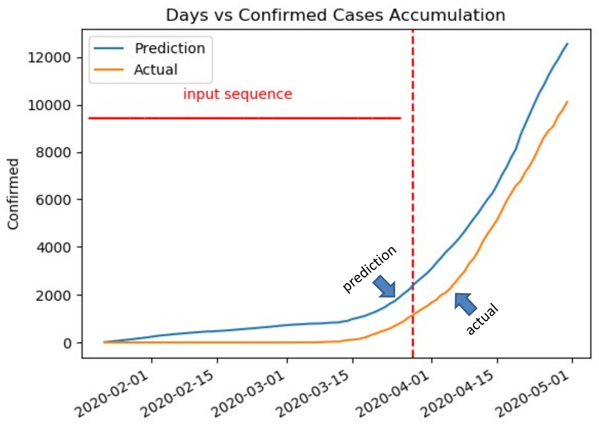

Fig. 4. Sample of prediction of Indonesia's confirmed COVID-19 cases from January 22, 2020 to May 1, 2020.

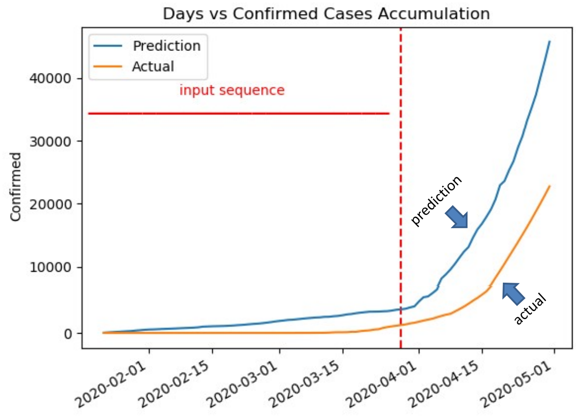

Fig. 6. Sample of prediction of Saudi Arabia's confirmed COVID-19 cases from January 22, 2020 to May 1, 2020.

actual growth. The prediction is ahead of several days than the actual. The prediction on May 1, 2020 shows the number of confirmed cases, more than 12,000. However, in actual growth, the number of confirmed cases is still more than 10,000. This small gap is not considered significant, and it can be revealed that the daily reported cases are still on track with the reported cases worldwide. There are various COVID-19 human test sampling in the training data that has been performed by several countries. In the US, the test sampling has already been above 1,000,000, whereas in several other countries is still below 1,000 [9].

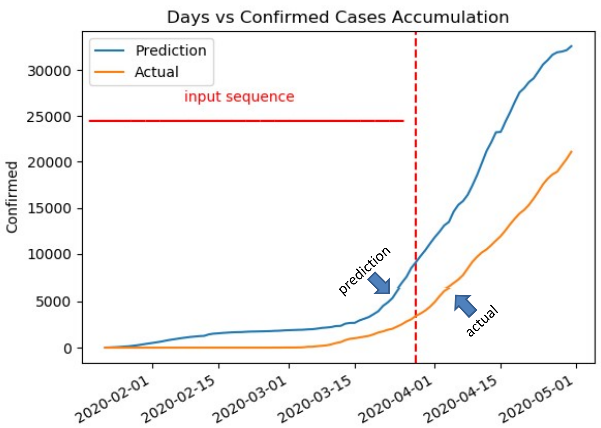

Fig. 5. Sample of prediction of Sweden's confirmed COVID-19 cases from January 22, 2020 to May 1, 2020.

Figure 5 shows the validation results of Sweden. The country is a northern subtropical country and is also well known for implementing light restrictions during the COVID-19 pandemic by identifying the aged people as the at-risk group. The prediction shows that the number of cases grows exponentially higher than the actual cases. However, it has quite a similar slope with the actual prediction. The prediction for May 1, 2020 shows that the confirmed cases reach around 30,000, greater than the actual cases, which still reaches is more than 20,000 [9].

Figure 6 shows the validation results of Saudi Arabia. Saudi Arabia is a tropical country similar to Indonesia, but it has higher confirmed cases than Indonesia. The prediction is quite similar to the actual prediction in terms of the exponential curve. However, there is a significant gap in terms of quantity where the prediction reaches more than 40,000, and the actual growth is still more than 20,000.

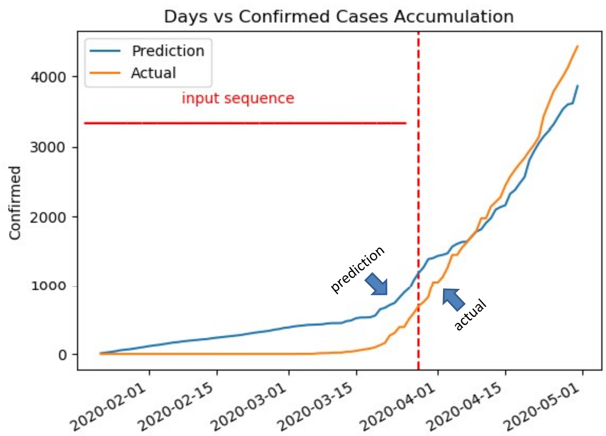

Fig. 7. Sample of prediction of Argentina's confirmed COVID-19 cases from January 22, 2020 to May 1, 2020.

Figure 7 shows validation results of Argentina. Argentina is a southern subtropical country. The prediction curve interchanges with the actual growth over time and grow exponentially. The prediction and actual growth reach around 4,000 on May 2, 2020.

B. *Interval and Mean Validation*

We also investigate the interval and mean validation of the interval and mean validation of the training and testing data five times. This evaluation is set due to the initial weight's randomness, making it advantageous to output several possibilities of the prediction curve. The output validation can be categorized into best, normal, and worst-case depending on the final accumulation of confirmed cases. The normal case is an average of five times the training and validations. The best case is a graph that achieves the lowest number of accumulations of confirmed cases on June 2, 2020 and vice versa.

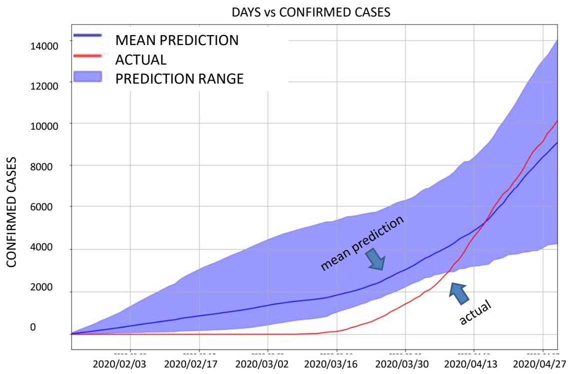

Fig. 8. Sample of the mean prediction of Indonesia's confirmed COVID-19 cases from January 22, 2020 to May 1, 2020.

Figure 8 shows the mean validation results of Indonesia. The actual prediction starts from the lower part of the prediction curve and gradually passes the prediction curve. The final actual growth is still within the range of prediction area. The evaluation result shows the mean RMSE is ,1,111.52, as shown in Table III.

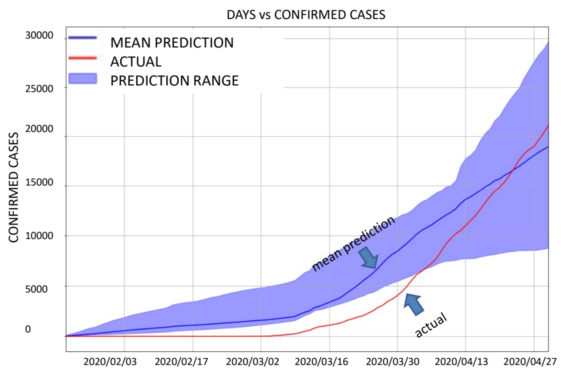

Fig. 9. Sample of the mean prediction of Sweden's confirmed COVID-19 cases from January 22, 2020 to May 1, 2020.

Figure 9 shows the mean validation results of Sweden. The actual prediction starts from the lower part of the prediction curve and gradually passes the prediction curve. The final actual growth is still within the range of the prediction area, with a mean RMSE of 1,756.58 (Table III).

Figure 10 shows the mean validation results of Saudi Arabia. The actual curve starts from the lower part of prediction curve and finally achieves the same number of accumulated confirmed cases with the prediction. The final prediction is still within the range of the prediction areas with a mean RMSE of 2,795.88 (Table III).

Figure 11 shows the mean prediction results of Argentina. The actual prediction starts from the lower part of the prediction curve, and the gap becomes wider over time. The final prediction is still outside the range of the prediction areas with a mean RMSE of 3,691.23 (Table III). This result regards the importance of the initial weight until

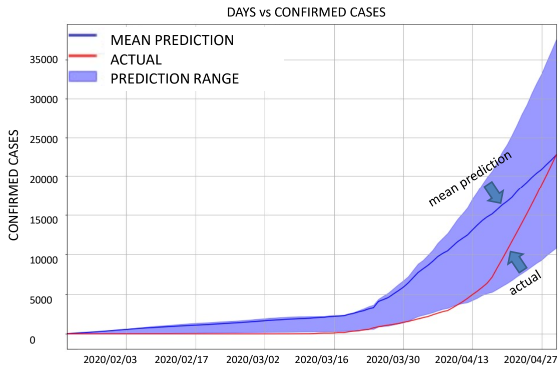

Fig. 10. Sample of the mean prediction of Saudi Arabia's confirmed COVID-19 cases from January 22, 2020 to May 1, 2020.

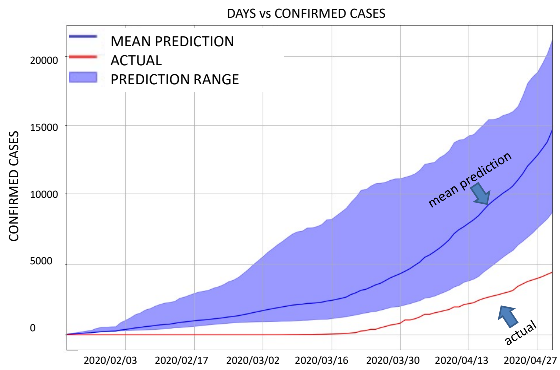

Fig. 11. Sample of the mean prediction of Argentina's confirmed COVID-19 cases from January 22, 2020 to May 1, 2020.

achieving the best validation results. Another factor is the sample imbalance, where the number of southern subtropical countries is less than that of northern subtropical and tropical countries.

TABLE III: Accuracy result of each country

Country	RMSE
Indonesia	1111.52
Sweden	1756.58
Saudi Arabia	2795.88
Argentina	3691.23

TABLE IV: Accuracy results given with various numbers of hidden states using four LSTM layers

Number of hidden states	RMSE
1	4517.87
5	1284.94
10	924.89
30	889.44

C. Effect of different Architectures

We try on a different number of hidden states in each LSTM layer. The higher the number of hidden states, the

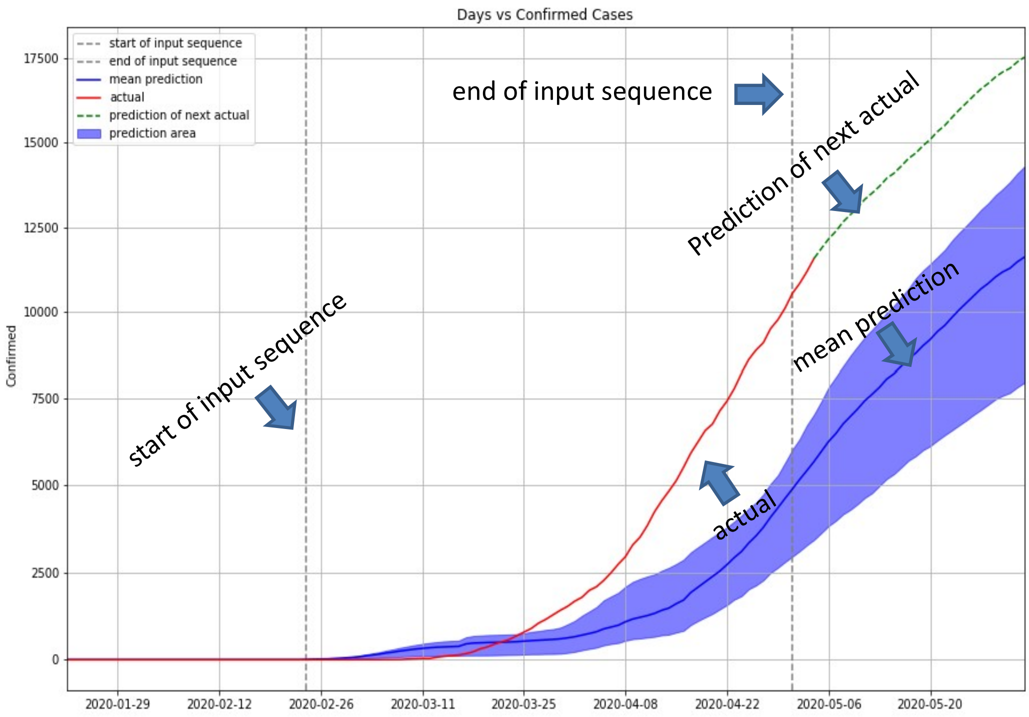

Fig. 12. Sample of prediction of accumulation of confirmed COVID-19 cases in Indonesia from April 22, 2020 to June 2, 2020.

TABLE V: Accuracy results given various numbers of hidden layer using 30 hidden states

Number of hidden layers	RMSE
1	3004.28
2	654.22
3	641.93
4	568.35

TABLE VI: Accuracy results of VAR, RNN and LSTM

Architecture	RMSE
VAR	25395.77
RNN	1520.61
LSTM	1238.66

higher parameter will be available inside the model. Over-parameterization will waste computational power and is thus inefficient for applications. Under-parameterization will reduce the prediction accuracy than it should be. The optimum number of the parameter is somewhat more desirable, and thus we heuristically add the number of hidden states and layers to examine the effect on prediction accuracy. As shown in Table IV, with four hidden layers, as the number of hidden states increases, the RMSE decreased. Thus, with 30 hidden states, the LSTM model still produces significant accuracy. In Table V, with 30 hidden states, as the number of layers increases, the performance of the LSTM model increases.

D. Comparison with VAR and RNN

We compare our LSTM with the previous version of the time-series prediction model of RNN. As shown in Table VI, using one-layer LSTM or RNN, LSTM outperformed VAR and RNN by 24,157.11 and 281.95, respectively. This finding confirms the importance of data-driven models with machine learning and the ability of LSTM to recognize a long series by minimizing vanishing gradients.

E. Effect of different time lags

Historical data given different time lags contain information for future predictions. It is useful for evaluating the robustness of the method against prediction in the arbitrary days that is not necessarily in a consecutive way between historical and prediction. Therefore, we arrange experiments on different lags for VAR, RNN, and LSTM methods. For RNN and LSTM, four hidden layers and five hidden units are used in model architecture, and trainings are done until 5000 epochs. The training and testing are repeated three times, and the final accuracies are averaged due to the uncertainty of RNN and LSTM. In this investigation, a 10-day prediction is used for all methods and experiments. In each training, the historical and prediction's sequence is 99-day long with the time lags used in this experiment are 4, 6, 8, 12, and 16. Each model is trained separately to predict COVID-19 growth using several time lags.

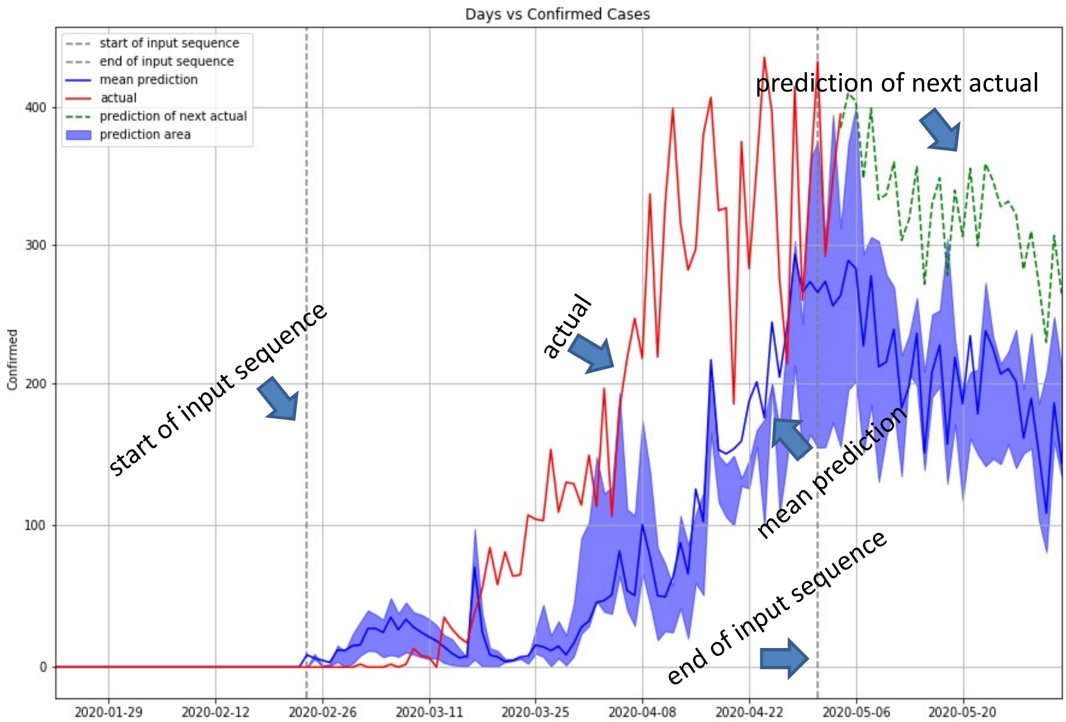

Fig. 13. Sample of prediction of daily confirmed COVID-19 cases in Indonesia from April 22, 2020 to June 2, 2020.

TABLE VII: Effect of different time lags

Time lag	Method	RMSE	MAE
4	VAR	25448.04	14050.89
	RNN	889.47	623.69
	LSTM	893.34	721.31
6	VAR	34448.10	18100.03
	RNN	1017.34	770.25
	LSTM	1592.71	1255.07
8	VAR	24440.44	12955.38
	RNN	1569.96	1314.55
	LSTM	2288.64	1572.83
12	VAR	1.99×10^9	5.96×10^8
	RNN	2882.82	2234.98
	LSTM	2802.63	1825.42
16	VAR	3.01×10^{13}	6.98×10^{12}
	RNN	4937.15	3400.68
	LSTM	3707.79	2504.51

Given different time lags, we investigate the effect on methods indicated by RMSE and MAE. As shown in table VII, VAR, RNN, and LSTM achieved the best performance when time lag is 4 in RMSE and MAE. The historical and prediction have autocorrelation relationship where the highest correlation can be in particular lag. Our results show that the best predictions are obtained in a small lag, indicating that the correlation between historical and prediction is best obtained. The higher the number of lags, the more complex the growth pattern will be, and the more challenging network will memorize the pattern of growth and thus achieved more error rates. In this case, LSTM with a long term memory model is robust to the number of lag indicated by the smallest RSME and MAE compared to RNN and VAE when the number of time lags is greater than 4.

F. *Effect of different number of day on prediction*

The n-day prediction is held to understand the effect of the length of day prediction on errors. The results are illustrated in figure 14 where predictions of 5-day, 20-day, and 40-day future series have RMSE of 472.57, 1779.56, and 3050.82, respectively, and MAE of 380.53, 1511.50, 2633.20, respectively. It is shown that the longer the prediction is, the lower error will be, indicating that the problem becomes complex.

We separately train each model in each n-day prediction for 5000 epochs. Each experiment is done five times and then averaged to obtain overall performance due to the uncertainty of LSTM and RNN. A fixed architecture with four LSTM/RNN hidden layers, five hidden units, and a fully connected layer with sigmoid activation are used throughout experiments. Table VIII shows that as the number of days to be predicted is increased, the accuracy decreases, as indicated by RMSE and MAE. Except for 1-day prediction, LSTM produces better performance than RNN and VAR due to its long-term model, making it suitable for long series of COVID-19 growth predictions.

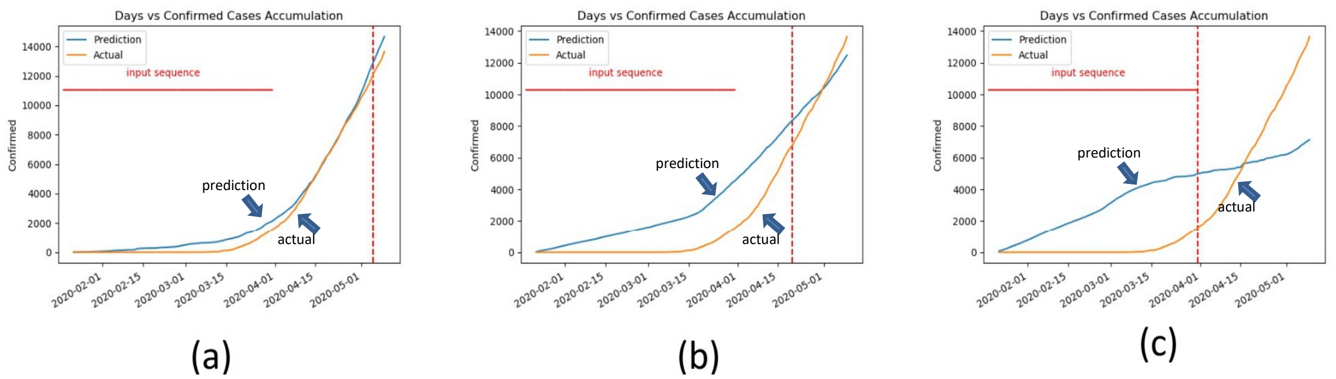

Fig. 14. The LSTM prediction and actual of COVID-19 growth on (a) 5-day prediction (b) 20-day prediction (c) 40-day predcition.

TABLE VIII: Effect of different number of day on prediction

Task	Method	RMSE	MAE
1-day prediction	VAR	13625.15	7929.89
	RNN	250.89	224.53
	LSTM	451.43	369.94
5-day prediction	VAR	33108.22	16550.63
	RNN	998.90	924.07
	LSTM	870.90	712.04
10-day prediction	VAR	1.07×10^5	52461.99
	RNN	1955.94	1422.18
	LSTM	1703.20	1299.44
20-day prediction	VAR	35227.58	19295.41
	RNN	2938.18	2286.85
	LSTM	2118.23	1818.09
30-day prediction	VAR	3.49×10^5	1.55×10^5
	RNN	2461.68	1881.03
	LSTM	1737.91	1482.41
40-day prediction	VAR	7.46×10^5	3.18×10^5
	RNN	3894.55	3346.18
	LSTM	2231.12	1753.52

G. Testing to predict the future growth of COVID-19 cases until June 2, 2020

We also arrange a real prediction using the LSTM model mentioned above on training data from January 22, 2020 to May 1, 2020. It is then tested on the input sequence with a duration from February 29, 2020 to May 1, 2020. As shown in Figure 12, the predicted and actual curve grow with significantly different quantities, but the prediction pattern still follows the same exponential curve. We confirm this assumption by looking into the daily confirmed cases (Figure 13). The blue one has the same growth pattern with the daily cases, presented by the red lines (actual), but with different quantities. This finding shows the ability of LSTM to capture growth pattern more than the quantity. We suggest that more data should be included in the training phase for more precise results. To predict the continuation of the actual graph (red), the portion of the blue graph (mean prediction) is cut starting from the end of the actual graph (May 2, 2020). Its cut series is then uniformly augmented, such that it is at the same level as the actual graph. The final continuation prediction shows the decreasing trend in May with the range of cases between 400 to 300 and below 300 cases after May 20, 2020. This cut and augmentation method can be performed daily to update the prediction.

V. Discussions

LSTM is a model that captures the correlation of time series dynamics. This research verifies the ability of LSTM to predict the COVID-19 growth curve, given enough training data. The results will be convincingly better if we add more variety of data with large quantities (big data). Our approach is better than the traditional statistical approach or qualitative modeling because it is trained to represent temporal data structure. The samples used to train LSTM are divided to 67 days after January 22, 2020 as the input and 33 days before May 1, 2020 as the output with the entire sequence of 100 days. In testing, the length of input and output are 67-sequence and 100 days, respectively. Due to its long-range time-series, long-term COVID-19 growth prediction is a challenging problem prediction (many-to-many).

Regarding the parameters employed in this research, the latitude and longitude represent the confirmed cases well. They show that northern subtropical countries tend to have a steeper growth slope than the tropical and southern ones. This conclusion is drawn quantitatively from the RMSE results in the validation phase. The additional variables might be beneficial for accuracy improvements from predictors.

VI. Conclusion

We have developed an LSTM based prediction model to foresee the COVID-19 pandemic growth over countries. The accumulated number of confirmed COVID-19 cases is monotonically increasing over time until it arrives at a particular converged peak curve. Given extensive training data, LSTM captures the dynamic growth pattern of graphs with a minimum RMSE compared to RNN. The results suggest that LSTM is a promising tool to predict the COVID-19 pandemics by learning from big data and can potentially predict future outbreaks. Future work should increase the training data by either adding new data or a data augmentation strategy.

Acknowledgment

The authors of this paper would like to express their thanks and gratitude to Sutiman Bambang Sumitro from the Department of Biology, Faculty of Mathematics and Natural Science, Brawijaya University for his support and guidance in this COVID-19 research.

REFERENCES

[1] Zhang, Guoqiang, B. Eddy Patuwo, and Michael Y. Hu. ”Forecasting with artificial neural networks:: The state of the art.” International journal of forecasting 14.1 (1998): 35-62.
[2] Bayer, Justin Simon. Learning Sequence Representations. Diss. Technische Universität München, 2015.
[3] Pascanu, Razvan, Tomas Mikolov, and Yoshua Bengio. ”On the difficulty of training recurrent neural networks.” International conference on machine learning. 2013.
[4] Hochreiter, Sepp, and Jürgen Schmidhuber. ”Long short-term memory.” Neural computation 9.8 (1997): 1735-1780.
[5] Längkvist, Martin, Lars Karlsson, and Amy Loutfi. ”A review of unsupervised feature learning and deep learning for time-series modeling.” Pattern Recognition Letters 42 (2014): 11-24.
[6] Taieb, Souhaib Ben, et al. ”A review and comparison of strategies for multi-step ahead time series forecasting based on the NN5 forecasting competition.” Expert systems with applications 39.8 (2012): 7067-7083.
[7] Yang, Zifeng, et al. ”Modified SEIR and AI prediction of the epidemics trend of COVID-19 in China under public health interventions.” Journal of Thoracic Disease 12.3 (2020): 165.
[8] LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton. ”Deep learning.” nature 521.7553 (2015): 436-444.
[9] https://www.worldometers.info/coronavirus/
[10] Peng, Liangrong, et al. ”Epidemic analysis of COVID-19 in China by dynamical modeling.” arXiv preprint arXiv:2002.06563 (2020).
[11] Roda, Weston C., et al. ”Why is it difficult to accurately predict the COVID-19 epidemic?.” Infectious Disease Modelling (2020).
[12] Sajadi, Mohammad M., et al. ”Temperature and latitude analysis to predict potential spread and seasonality for COVID-19.” Available at SSRN 3550308 (2020).
[13] Benvenuto, Domenico, et al. ”Application of the ARIMA model on the COVID-2019 epidemic dataset.” Data in brief (2020): 105340.
[14] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. ”Imagenet classification with deep convolutional neural networks.” Advances in neural information processing systems. 2012.
[15] Yudistira, Novanto, et al. ”UV light influences covid-19 activity through big data: trade offs between northern subtropical, tropical, and southern subtropical countries.” medRxiv (2020).
[16] Paterlini, M. ”’Closing borders is ridiculous’: the epidemiologist behind Sweden’s controversial coronavirus strategy.” Nature (2020).
[17] Abrigo, Michael RM, and Inessa Love. ”Estimation of panel vector autoregression in Stata.” The Stata Journal 16.3 (2016): 778-804.
[18] Phillips, Peter CB. ”Fully modified least squares and vector autoregression.” Econometrica: Journal of the Econometric Society (1995): 1023-1078.
[19] Ang, Andrew, and Monika Piazzesi. ”A no-arbitrage vector autoregression of term structure dynamics with macroeconomic and latent variables.” Journal of Monetary economics 50.4 (2003): 745-787.
[20] Youji Ochi, ”Development of Crutch-Motion Recognition System Using RNN,” Lecture Notes in Engineering and Computer Science: Proceedings of The International MultiConference of Engineers and Computer Scientists 2019, 13-15 March, 2019, Hong Kong, pp51-54
[21] Kishida, Kazuya, Kei Hasegawa, and Kiyotaka Kamata, ”Construction of Myoelectric Signal Classifier Using LSTM and Efficacy of Prior Learning and Relearning Processes,” Engineering Letters, vol. 28, no. 3, pp668-675, 2020
[22] Teresita L. Todelo, and Chris Jordan G. Aliac, ”Predictability of Earthquake Occurrence Using Auto Regressive Integrated Moving Average (ARIMA) Model,” Lecture Notes in Engineering and Computer Science: Proceedings of The International MultiConference of Engineers and Computer Scientists 2019, 13-15 March, 2019, Hong Kong, pp252-256
[23] Kavasseri, Rajesh G., and Krithika Seetharaman. ”Day-ahead wind speed forecasting using f-ARIMA models.” Renewable Energy 34.5 (2009): 1388-1393.
[24] Ma, Jing, et al. ”Detecting rumors from microblogs with recurrent neural networks.” (2016): 3818.

Novanto Yudistira is currently lecturer and researcher at Brawijaya University, Indonesia. He received his BS in informatics engineering from the Institut Teknologi Sepuluh in November in 2007, his MS in computer science from Universiti Teknologi Malaysia in 2011, and his Dr. Eng. from information engineering, Hiroshima University, Japan in 2018. In 2016, he involved in research collaboration with Mathematical Neuroinformatics Group, National Institute of Advanced Industrial Science and Technology (AIST), Japan. In 2018, he continued Postdoctoral fellow in informatics and data science analytic for 2 years working with the Japanese large scientific research institute, RIKEN and Osaka university. His current research interests include deep learning, multi modal computer vision, medical informatics and big data analysis.