

Forecasting Penambahan Kasus Covid-19 Harian di Indonesia dengan Metode LSTM

Benedicta Yesika Meidianti¹, Editha Anggrieniputri Kinbenu², Husna Rahma Yunita³, James Julius⁴, Rafida 'Alaiyya Hayyin⁵, Shofiayumna Felisya Putri⁶

Program Studi Ilmu Aktuaria, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Gadjah Mada, Sekip Utara, Bulaksumur 21, Depok, Sleman, Daerah Istimewa Yogyakarta, 55281, Indonesia

Email Penulis

- a) Penanggung jawab penulis: dicta.ym@mail.ugm.ac.id
- b) edithakinbenu82@mail.ugm.ac.id
- c) husnarahma126@mail.ugm.ac.id
- d) jamesjulius@mail.ugm.ac.id
- e) gulamerahgulaputih@mail.ugm.ac.id
- f) shofiayumna@mail.ugm.ac.id

Abstrak. Persebaran kasus virus Covid-19 terus bertambah belakangan ini. Berbagai upaya sudah dilakukan untuk menghentikan persebaran virus ini. Salah satu upaya yang dilakukan dalam penanganan kasus Covid-19 di Indonesia ini adalah dengan memprediksi pertambahan persebaran virus ini. Maka dari itu dapat dilakukan *forecasting* jumlah kasus Covid-19 di Indonesia. Pada studi yang disajikan dalam *paper* ini, dilakukan uji coba prediksi berbasis teknik *deep learning* yaitu metode *Long Short Term Memory* (LSTM). Untuk menguji coba teknik yang diusulkan digunakan parameter kinerja *Mean Squared Error* (MSE), *Root Mean Squared Error* (RMSE), dan *Mean Absolute Error* (MAE).

Kata kunci : Covid-19, LSTM, *forecasting*, RMSE.

PENDAHULUAN

Virus Covid-19 yang ditemukan sekitar bulan Desember tahun 2019 di Wuhan, China hingga saat ini masih berada di sekeliling kita. Ditambah lagi dengan banyak penemuan adanya varian baru mengenai virus tersebut. Virus Covid-19 ini dapat menyebabkan menimbulkan berbagai komplikasi penyakit terutama gangguan pada saluran pernapasan seperti gagal pernapasan akut, *pneumonia*, *acute respiratory distress syndrome* (ARDS) dan juga komplikasi dan masalah pada organ lain hingga menyebabkan kematian bagi penderitanya (Zendrato, 2020: 242). Virus tersebut jelas menjadi ancaman bagi seluruh dunia terutama Indonesia yang menjadi salah satu negara dengan tingkat pertumbuhan kasus Covid-19 tertinggi. Jumlah pasien yang terinfeksi virus Covid-19 di Indonesia terus bertambah seiring berjalannya waktu meskipun saat ini sudah tidak banyak seperti sebelumnya. Hingga Jumat, 11 November 2020, kasus positif Covid-19 bertambah 423 kasus baru dan secara kumulatif berjumlah 4.249.758 kasus. Sedangkan kasus kematian yang disebabkan oleh virus Covid-19 hingga saat ini mencapai 143.068 (worldometers.info, 11 November 2021).

Untuk menangani kasus Covid-19 yang masih terus bertambah serta membantu tenaga kesehatan dan pemerintah mempersiapkan potensi bertambahnya kasus Covid-19, akan dilakukan *forecasting* jumlah kasus Covid-19 di Indonesia. *Forecasting* atau peramalan adalah seni dan ilmu untuk memperkirakan kejadian di masa depan. Hal ini dapat dilakukan dengan melibatkan pengambilan data historis dan memproyeksikannya ke masa mendatang dengan suatu bentuk model matematis atau prediksi intuisi bersifat subyektif, atau menggunakan kombinasi model matematis yang disesuaikan dengan pertimbangan yang baik (Heizer dan Render, 2009: 167). Penyebaran kasus Covid-19 ini tentu menunjukkan suatu pola tertentu

yang didasarkan pada dinamika penularan virus tersebut. Langkah-langkah dalam pencegahan bertambahnya kasus ini bergantung pada metode yang digunakan. Dalam memprediksi data runtun waktu ada banyak metode *deep learning* yang dapat digunakan, salah satunya adalah *Long Short Term Memory* (LSTM) yang akan digunakan untuk memprediksi pertumbuhan Covid-19 di Indonesia.

Alasan menggunakan *Long Short Term Memory* (LSTM) untuk memprediksi pertumbuhan kasus Covid-19 di Indonesia adalah karena LSTM memiliki *memory cell* yang menggantikan *hidden neuron* pada jaringan saraf tiruan. *Memory cell* pada LSTM bertujuan untuk menyimpan informasi dari data-data sebelumnya, sehingga LSTM memiliki kemampuan yang lebih baik yaitu mengingat dalam jangka panjang (Sen et al., 2020: 35). Kemampuan tersebut dapat menjadikannya sebagai metode yang baik untuk memprediksi data pertumbuhan kasus Covid-19 yang mana datanya bersifat dinamis dan hasilnya seringkali tidak jelas jika hanya menggunakan model statistik biasa. Dengan penelitian ini diharapkan didapatkan hasil performa yang akurat sehingga dapat mempersiapkan tenaga medis, pemerintah, serta masyarakat dengan adanya pertumbuhan kasus Covid-19 di Indonesia.

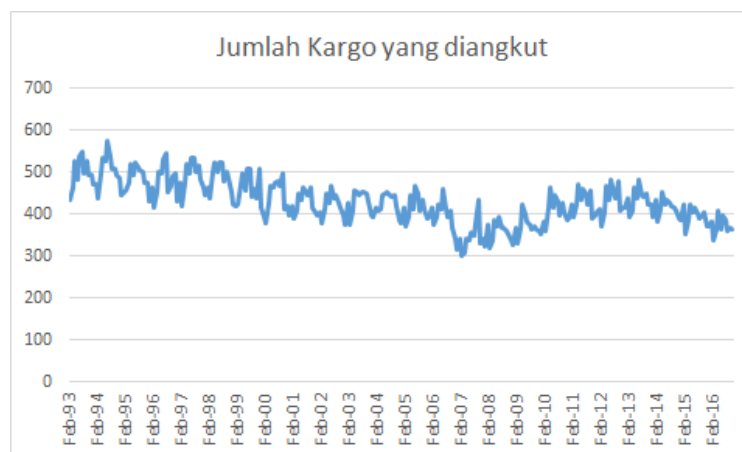
LANDASAN TEORI

Pola Data Runtun Waktu

Dalam pemodelan runtun waktu, pola data menjadi salah satu aspek yang menentukan model mana yang sesuai untuk digunakan. Berdasarkan pola datanya, data runtun waktu dibagi menjadi empat, yaitu :

a) **Pola Stasioner**

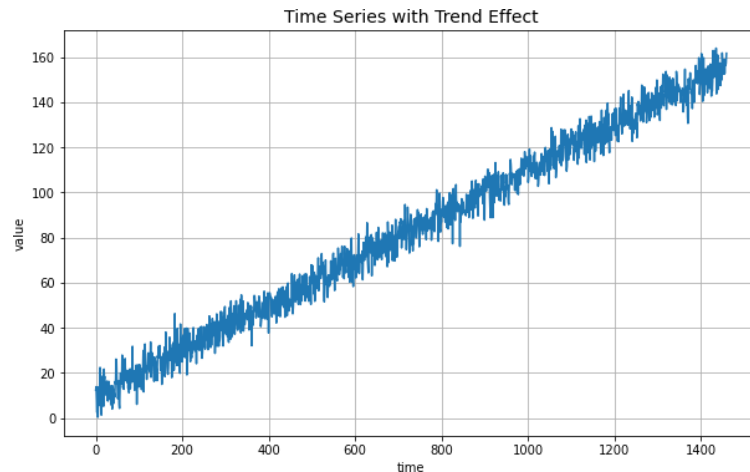
Data dikatakan memiliki pola stasioner apabila pola data yang terbentuk stasioner dalam mean dan/atau variansi. Contoh pola data stasioner :



GAMBAR 1. Pola Data Stasioner

b) **Pola dengan Efek Tren**

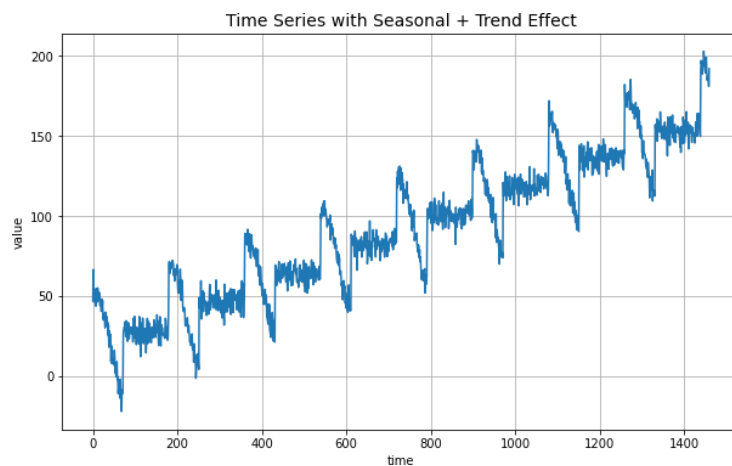
Data dikatakan memiliki pola dengan efek tren apabila pola data yang terbentuk memiliki pola naik atau turun secara terus menerus. Contoh pola data dengan efek tren :



GAMBAR 2. Pola Data Tren

c) Pola dengan efek Musiman

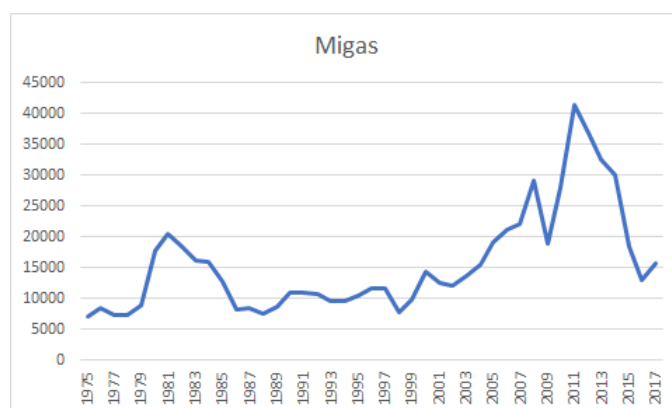
Data dikatakan memiliki pola dengan efek musiman apabila pola data yang terbentuk memiliki pola berulang dengan periode tertentu, misalkan tahunan, bulanan, dan mingguan. Contoh pola data dengan efek musiman :



GAMBAR 3. Pola Data Musiman

d) Pola dengan Efek Tidak Tentu (*Irregular*)

Data dikatakan memiliki pola dengan efek tak tentu apabila pola data yang terbentuk tidak memenuhi sifat dari ketiga pola data diatas. Contoh pola data dengan efek tak tentu :



GAMBAR 4. Pola Data dengan Efek Tak Tentu

Uji Stasioneritas

Uji asumsi stasioneritas adalah hal yang penting dalam pemodelan runtun waktu karena sifat-sifat statistik di masa yang akan datang bisa diramalkan berdasarkan data historis yang telah terjadi apabila model runtun waktu tersebut stasioner (Rosadi, 2011).

Salah satu metode pengujian stasioneritas suatu data runtun waktu adalah dengan uji akar unit atau *unit root test*. *Unit root test* dilakukan dengan mengamati apakah data runtun waktu mengandung *unit root* (terdapat komponen tren berupa jalan acak atau *random walk* dalam data). Salah satu uji yang dapat digunakan adalah uji *Augmented Dickey Fuller* atau ADF.

Uji ADF melihat apakah terdapat akar unit dalam model atau tidak dengan menguji hipotesis null $H_0: \rho = 0$ (terdapat akar unit atau data tidak stasioner) dalam persamaan regresi

$$Y_t = \alpha + \delta t + \rho Y_{t-1} + \sum_{j=1}^k \phi Y_{t-j} + \epsilon_t$$

Hipotesis null ditolak jika nilai statistic uji ADF memiliki nilai lebih rendah dibandingkan dengan nilai daerah kritis. Jika hipotesis null tersebut ditolak, didapat kesimpulan bahwa data bersifat stasioner.

Proses Wide-Sense Stasioner

Definisi Proses runtun waktu $X_t, t \in T$ dengan $T = \mathbb{Z} = \{0, \pm 1, \pm 2, \dots\}$ disebut proses (W-S) stasioner jika :

1. $E(|X_t|^2) < \infty, \forall t \in \mathbb{Z}$
2. $E(X_t) = \text{konstanta, independen dengan } t, \forall t \in \mathbb{Z}$
3. $\gamma_X(t, s) = \gamma_X(t + h, s + h), \forall t, s, h \in \mathbb{Z}$

Proses White Noise

Proses White Noise $\{X_t\}$ adalah barisan variabel random tidak berkorelasi dengan mean μ (sering diasumsikan bernilai nol) dan variansi σ^2 yaitu

$$\text{cov}(X_t, X_s) = \begin{cases} \sigma^2, & \text{jika } t = s \\ 0, & \text{jika } t \neq s \end{cases}$$

Perhatikan dari definisi di atas, diperoleh bahwa

$$\begin{aligned} \text{cov}(X_{t+h}, X_t) &= \begin{cases} \sigma^2, & \text{jika } h = 0 \\ 0, & \text{jika } h \neq 0 \end{cases} \\ \text{cor}(X_{t+h}, X_t) &= \begin{cases} 1, & \text{jika } h = 0 \\ 0, & \text{jika } h \neq 0 \end{cases} \end{aligned}$$

Maka, proses *White Noise* bersifat stasioner. Selanjutnya, proses *White Noise* pada $\{X_t\}$ dapat ditulis sebagai $X_t \sim WN(0, \sigma^2)$

Proses Autoregressive (AR)

Proses $\{X_t\}$ adalah proses *autoregressive* orde p (AR(p)) didefinisikan dengan persamaan

$$X_t = a_1 X_{t-1} + a_2 X_{t-2} + \dots + a_p X_{t-p} + \epsilon_t, t \in \mathbb{Z}$$

Dengan $a_1, a_2, \dots, a_p \in \mathbb{R}, \epsilon_t \sim WN(0, \sigma^2)$. Dengan mendefinisikan operator *backward-shift*

(operator lag) untuk proses $\{X_t\}$ sebagai

$$(B^j X)_t = X_{t-j}, j, t \in \mathbb{Z}$$

Maka proses AR(p) dapat dituliskan sebagai berikut :

$$\begin{aligned} X_t - a_1 X_{t-1} - a_2 X_{t-2} - \dots - a_p X_{t-p} &= \epsilon_t \\ X_t - a_1 (BX)_t - a_2 (B^2 X)_t - \dots - a_p (B^p X)_t &= \epsilon_t \\ (1 - a_1 B - a_2 B^2 - \dots - a_p B^p) X_t &= \epsilon_t \\ D(B) X_t &= \epsilon_t \end{aligned}$$

Dengan polinomial $D(z) = (1 - a_1 z - a_2 z^2 - \dots - a_p z^p)$. Jika $D(z) \neq 0, |z| < 1, z \in \mathbb{C}$, maka proses AR(p) bersifat stasioner.

Fungsi Autokorelasi Parsial (PACF)

Fungsi Autokorelasi Parsial (PACF) pada lag- k adalah korelasi antara X_t dan X_{t-k} setelah dependensi linear antara X_t dan X_{t-k} dengan variabel antara $X_{t+1}, X_{t+2}, \dots, X_{t+k-1}$ dihapus (Rosadi, 2014). Misalkan $\{X_t\}$ adalah suatu proses stasioner dengan mean nol dan $\{X_{t+k}\}$ ditulis sebagai model linear

$$X_{t+k} = a_{k1} X_{t+k-1} + a_{k2} X_{t+k-2} + \dots + a_{kk} X_t + \epsilon_{t+k}$$

dengan a_{ki} adalah parameter ke- i dari persamaan regresi, dan ϵ_{t+k} adalah komponen error yang tidak berkorelasi dengan $\{X_{t+k-j}\}$ untuk $j \geq 1$. Mengalikan kedua sisi pada persamaan di atas dengan $\{X_{t+k-j}\}$ dan mengambil nilai ekspektasinya, dapat membuat diperolehnya

$$\gamma(j) = a_{k1} \gamma(j-1) + a_{k2} \gamma(j-2) + \dots + a_{kk} \gamma(j-k)$$

Dengan membagi kedua sisi dengan $\gamma(0)$, diperoleh

$$\rho(j) = a_{k1} \rho(j-1) + a_{k2} \rho(j-2) + \dots + a_{kk} \rho(j-k)$$

untuk $j = 1, 2, 3, \dots, k$. Ketika menggunakan metode Cramer, dapat diperoleh nilai-nilai fungsi autokorelasi parsial a_{kk} untuk lag $k = 1, 2, 3, \dots$ sebagai berikut :

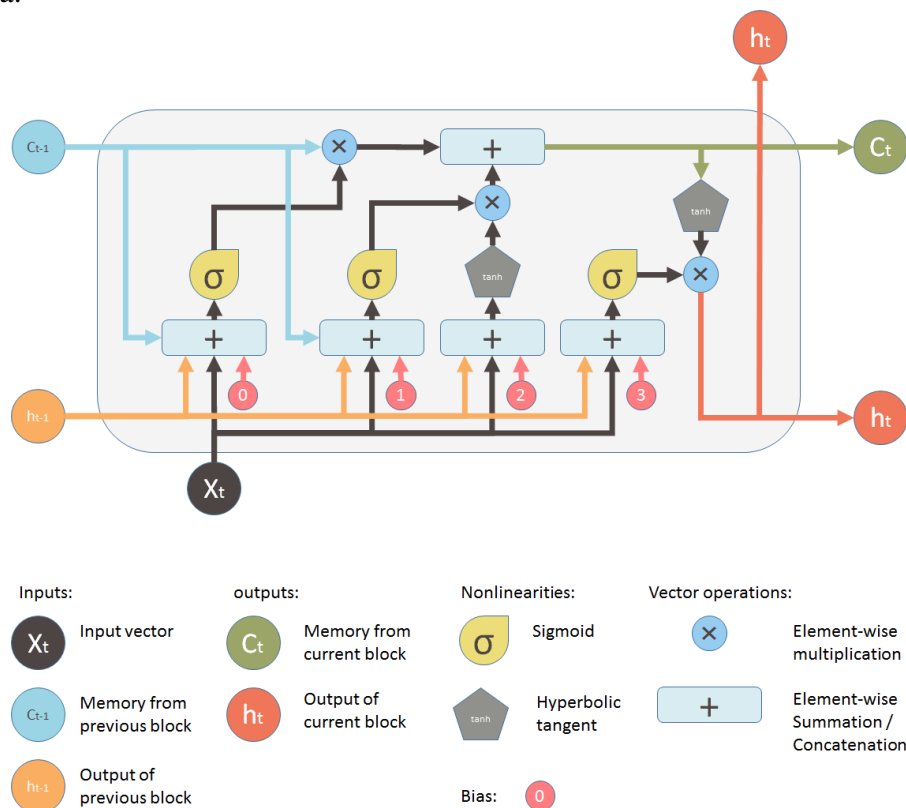
$$\begin{aligned} a_{11} &= \rho(1), \\ a_{22} &= \frac{\begin{vmatrix} 1 & \rho(1) \\ \rho(1) & \rho(2) \end{vmatrix}}{\begin{vmatrix} 1 & \rho(1) \\ \rho(1) & 1 \end{vmatrix}}, \\ a_{33} &= \frac{\begin{vmatrix} 1 & \rho(1) & \rho(2) \\ \rho(1) & 1 & \rho(3) \\ \rho(2) & \rho(1) & \rho(3) \end{vmatrix}}{\begin{vmatrix} 1 & \rho(1) & \rho(2) \\ \rho(1) & 1 & \rho(1) \\ \rho(2) & \rho(1) & \rho(1) \end{vmatrix}}, \dots \end{aligned}$$

$$a_{kk} = \frac{\begin{vmatrix} 1 & \rho(1) & \rho(2) & \cdots & \rho(k-2) & \rho(1) \\ \rho(1) & 1 & \rho(1) & \cdots & \rho(k-3) & \rho(2) \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \rho(k-1) & \rho(k-2) & \rho(k-3) & \cdots & \rho(1) & \rho(k) \end{vmatrix}}{\begin{vmatrix} 1 & \rho(1) & \rho(2) & \cdots & \rho(k-2) & \rho(k-1) \\ \rho(1) & 1 & \rho(1) & \cdots & \rho(k-3) & \rho(k-2) \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \rho(k-1) & \rho(k-2) & \rho(k-3) & \cdots & \rho(1) & 1 \end{vmatrix}}$$

Metode LSTM

Long Short Term Memory (LSTM) adalah model *deep learning* yang diperoleh dari pengembangan Recurrent Neural Network (RNN) yang juga bisa digunakan untuk memprediksi data runtun waktu. RNN bisa merekam informasi yang telah digunakan sebelumnya. Namun, RNN mempunyai kelemahan yaitu permasalahan luruhnya efektivitas gradien sehingga prediksi infeksi virus COVID-19 di Indonesia lebih baik menggunakan metode LSTM.

LSTM mempunyai kelebihan dapat memanggil urutan data jangka panjang, dapat menggunakan ukuran data yang besar, dan dapat menggunakan semua informasi sebagai input ke sistem. LSTM memiliki arsitektur yang berupa input, output, dan *hidden layers* yang terdiri dari sel memori atau *memory cells*. Satu sel memori mempunyai tiga pintu/*gate*: *input gate*, *forget gate*, dan *output gate*. Input dari setiap unit LSTM adalah output sebelumnya dari LSTM, yang berarti dalam kasus ini prediksi kasus Covid-19 saat ini bergantung pada kasus di hari sebelumnya.



GAMBAR 5. Ilustrasi Metode LSTM

Fungsi Aktivasi ReLU

Fungsi aktivasi ini digunakan pada *input* dan *hidden layer* baik dalam pemodelan klasifikasi maupun regresi. Ketika input bernilai negatif atau nol, maka fungsi aktivasi ReLU akan selalu menghasilkan output bernilai nol, hal ini dapat mempersulit proses *back-propagation* pada jaringan saraf. Fungsi aktivasi ReLU memiliki persamaan :

$$f(x) = \max\{0, x\}$$

Dengan $f(x) \in [0, \infty)$

Model Evaluation

Mean Absolute Error (MAE)

MAE adalah perbedaan absolut antara nilai prediksi (*forecast*) dengan nilai aktual dari sampel tes dimana semua perbedaan individual mempunyai bobot yang sama. MAE dapat menghitung besaran rata-rata dari error pada rangkaian *forecast*. MAE mempunyai rumus:

$$MAE(y, y') = \frac{1}{n} \sum_{i=1}^n |y_i - y'_i|$$

dengan

y'_i : nilai prediksi (*predicted value*)

y_i : nilai actual (*actual value*)

n : banyaknya observasi

Mean Squared Error (MSE)

MSE adalah rata-rata kesalahan kuadrat antara nilai aktual dan nilai ramalan. MSE biasanya digunakan untuk mengecek estimasi besarnya nilai kesalahan pada peramalan. Nilai MSE yang rendah atau mendekati nol menunjukkan bahwa data hasil ramalan sesuai dengan data aktual, artinya model bisa digunakan untuk perhitungan ramalan di periode selanjutnya. Persamaan dari MSE ditunjukkan sebagai berikut :

$$MSE(y, y') = \frac{1}{n} \sum_{i=1}^n (y_i - y'_i)^2$$

Dengan y'_i adalah nilai ramalan, y_i adalah nilai aktual, dan n adalah banyaknya observasi.

Root Mean Squared Error (RMSE)

RMSE adalah akar dari MSE yang juga mengukur besarnya rata-rata dari kesalahan. Biasanya RMSE digunakan untuk mengevaluasi metrik dan *loss function*. Persamaan dari RMSE ditunjukkan sebagai berikut :

$$RMSE(y, y') = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - y'_i)^2}$$

Dengan y'_i adalah nilai ramalan, y_i adalah nilai aktual, dan n adalah banyaknya observasi.

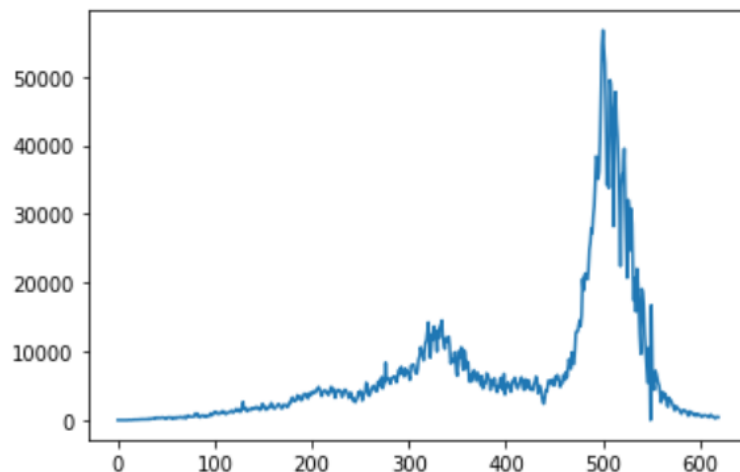
PEMBAHASAN DAN ANALISIS HASIL PENELITIAN

Dataset Kasus Harian Covid-19 di Indonesia

TABEL 1. Dataset

Date	New Case
02-03-2020	2
03-03-2020	0
04-03-2020	0
05-03-2020	0
06-03-2020	2
....
07-11-2021	444
08-11-2021	244
09-11-2021	434
10-11-2021	480
11-11-2021	435

Data yang digunakan di penelitian ini data penambahan kasus harian Covid-19 di Indonesia yang diambil secara online melalui laman *worldometer* dan dimulai dari 2 Maret 2020 hingga 11 November 2021. Bentuk data yang digunakan adalah data *time series*. Data yang terkumpul adalah sebanyak 620. Setiap data mengandung 2 atribut, yaitu *Date* dan *New Case*. *Date* adalah tanggal terjadinya penambahan kasus dan *New Case* adalah kasus penambahan harian Covid-19 di Indonesia.

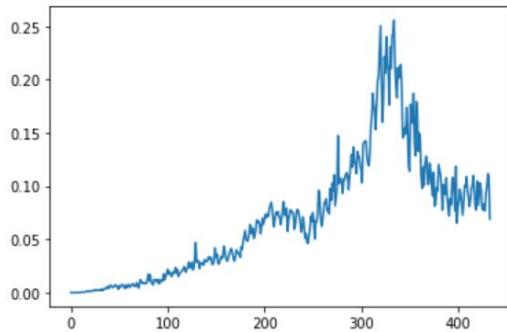


GAMBAR 6. Plot dari Data

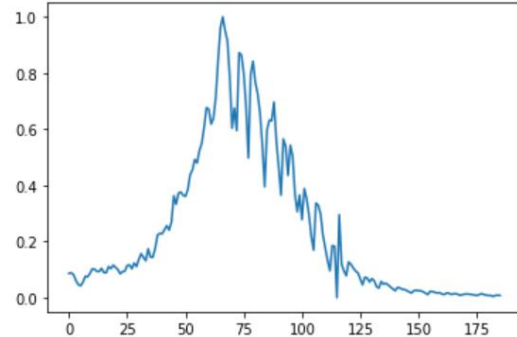
Data yang sudah dikumpulkan dipilah menjadi dua bagian, yaitu data training dan data testing. Skenario penggunaan dataset untuk penelitian disiapkan berdasarkan komposisi 70% data training dan 30% data testing dan didapatkan jumlah data untuk masing-masingnya adalah seperti Tabel 2. Sedangkan untuk grafiknya dapat dilihat seperti Gambar 7.

TABEL 2. Komposisi Dataset

Data Training	Data Testing
434	186

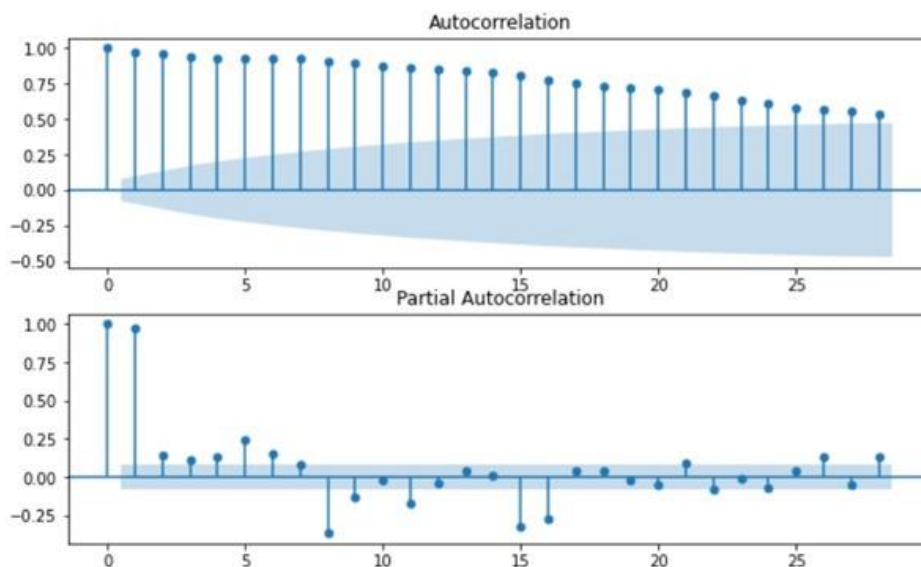


(a)



(b)

GAMBAR 7. Data Split untuk Data Training (a) dan Data Testing (b)



GAMBAR 8. Correlogram Data Penambahan Kasus Covid-19 Harian di Indonesia

Pada kasus ini akan diperhatikan 5 lag pertama pada PACF. Berdasarkan correlogram di atas, diperoleh nilai lag ke-5 merupakan lag yang terakhir keluar sehingga input adalah data dari period ke $t-5$ sampai $t-1$, dan output period ke- t .

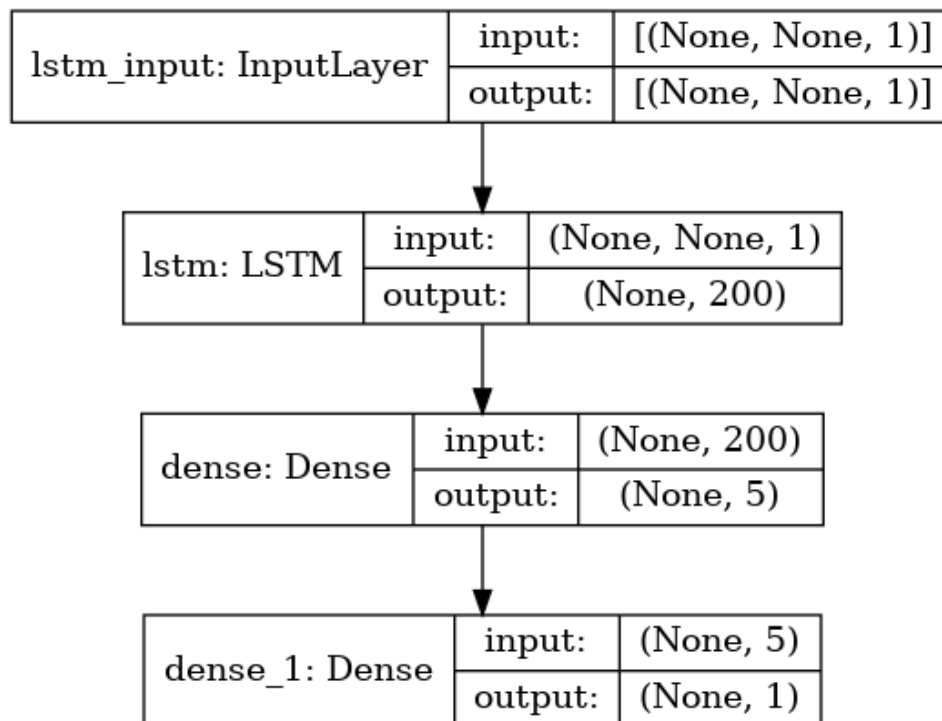
Berdasarkan correlogram, kemudian dibentuk data dengan dimensi yang sesuai dengan input dari LSTM, dengan jumlah data yang digunakan sebanyak 1, yaitu data kasus Covid-19, jumlah neuron 200, dan dense lag ke 5 dengan fungsi aktivasi ReLU. Diperoleh model LSTM dibentuk dengan arsitektur sebagai berikut :

```

Model: "sequential"
-----
Layer (type)                Output Shape                Param #
-----
lstm (LSTM)                  (None, 200)                 161600
-----
dense (Dense)                (None, 5)                   1005
-----
dense_1 (Dense)              (None, 1)                   6
-----
Total params: 162,611
Trainable params: 162,611
Non-trainable params: 0
-----

```

GAMBAR 9. Hasil Model LSTM



GAMBAR 10. Arsitektur Model LSTM

Pemilihan Epoch, Batch Size, dan Neuron

Dataset COVID19 akan dilatih menggunakan model LSTM. Dalam melatih dataset ini, akan dipilih banyak epoch, ukuran batch, dan jumlah neuron yang meminimumkan nilai RMSE. Selain meminimumkan nilai RMSE, akan dilihat nilai loss, MAE, dan MSE. Namun, pada tabel berikut hanya disediakan nilai loss, MAE, dan MSE terbesar sehingga untuk menentukan banyak epoch, ukuran batch, dan jumlah neuron akan dilihat nilai RMSE nya saja. Percobaan dilakukan dengan `set_seed(123)`.

Pemilihan Epoch

Dalam menentukan banyak epoch dilakukan pelatihan model LSTM dengan ukuran batch 64 dan jumlah neuron 100. Diperoleh nilai RMSE, loss, MAE, dan MSE untuk masing-masing epoch yang bernilai 25, 50, 100, dan 200 sebagai berikut.

TABEL 3. Hasil Eksperimen dari Efek Jumlah Epoch

Epochs	RMSE	loss	MAE	MSE
25	0.086596	0.007	0.07	0.007
50	0.08744	0.007	0.075	0.007
100	0.089255	0.0072	0.075	0.0072
200	0.101844	0.0072	0.075	0.0072

Berdasarkan tabel di atas, diperoleh RMSE terkecil dengan epoch sebanyak 25. Dengan demikian, untuk model LSTM ini akan digunakan banyak epoch sebesar 25.

Pemilihan Batch Size

Dari percobaan pemilihan epoch, diperoleh nilai epoch sebanyak 25 sehingga dalam menentukan ukuran batch size dilakukan pelatihan model LSTM dengan banyak epoch 25 dan jumlah neuron 100. Diperoleh nilai RMSE, loss, MAE, dan MSE untuk masing-masing batch size yang bernilai 2, 4, 8, 16, 32, dan 64 sebagai berikut.

TABEL 4. Hasil Eksperimen dari Efek Ukuran *Batch*

Batch Size	RMSE	loss	MAE	MSE
2	0.108915	0.002	0.032	0.002
4	0.086785	0.0027	0.04	0.0027
8	0.096487	0.0038	0.047	0.0038
16	0.089547	0.005	0.05	0.005
32	0.089295	0.006	0.055	0.006
64	0.086596	0.007	0.07	0.007

Berdasarkan tabel di atas, diperoleh RMSE terkecil dengan batch size sebesar 64. Dengan demikian, untuk model LSTM ini akan digunakan ukuran batch size sebesar 64.

Pemilihan Neuron

Dari percobaan pemilihan epoch dan batch size, diperoleh nilai epoch sebanyak 25 dan

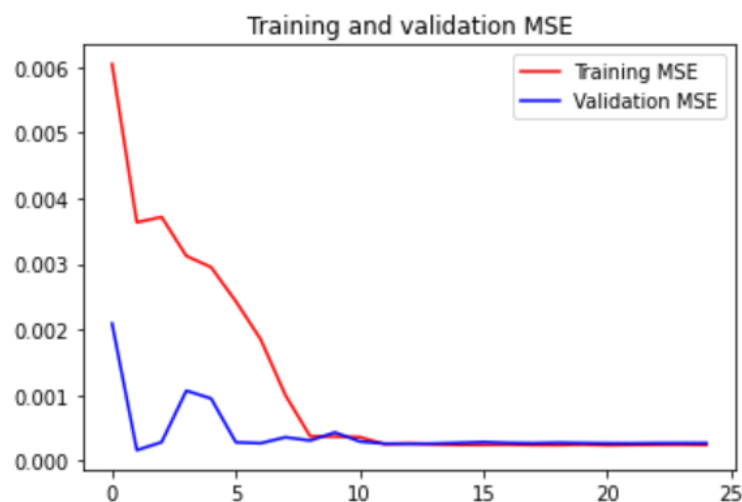
batch size sebesar 64 sehingga dalam menentukan jumlah neuron dilakukan pelatihan model LSTM dengan banyak epoch 25 dan ukuran batch size 64. Diperoleh nilai RMSE, loss, MAE, dan MSE untuk masing-masing neuron yang berjumlah 10, 25, 50, 100, dan 200 sebagai berikut.

TABEL 5. Hasil Eksperimen dari Efek Jumlah Neuron

Neuron	RMSE	loss	MAE	MSE
10	0.193807	0.0075	0.08	0.0075
25	0.088729	0.0072	0.075	0.0072
50	0.090305	0.0072	0.075	0.0072
100	0.086596	0.007	0.07	0.007
200	0.085917	0.007	0.07	0.007

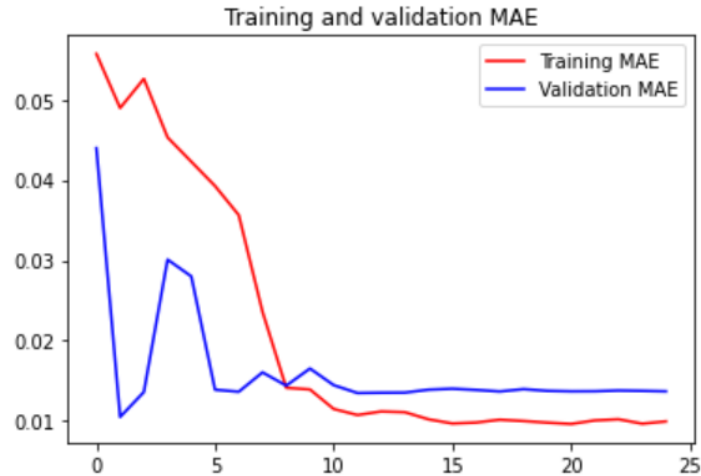
Berdasarkan tabel, diperoleh RMSE terkecil dengan jumlah neuron sebesar 200. Dengan demikian, untuk model LSTM ini akan digunakan jumlah neuron sebesar 200.

Dengan demikian, pada pelatihan dataset COVID19 dengan model LSTM ini akan digunakan nilai epoch sebanyak 25, ukuran batch sebesar 64, dan jumlah neuron sebesar 200. Untuk plot MSE, MAE, dan *loss* pada data *training* dan data *validation*, dapat dilihat pada grafik berikut.



GAMBAR 11. Nilai MSE Selama Proses Pelatihan LSTM

Berdasarkan plot diatas, dapat dilihat bahwa nilai dari *training* dan *validation* MSE menurun, dan menghasilkan nilai MSE yang cukup kecil serta mendekati 0, artinya model yang dihasilkan oleh model LSTM pada data uji menghasilkan model yang baik.



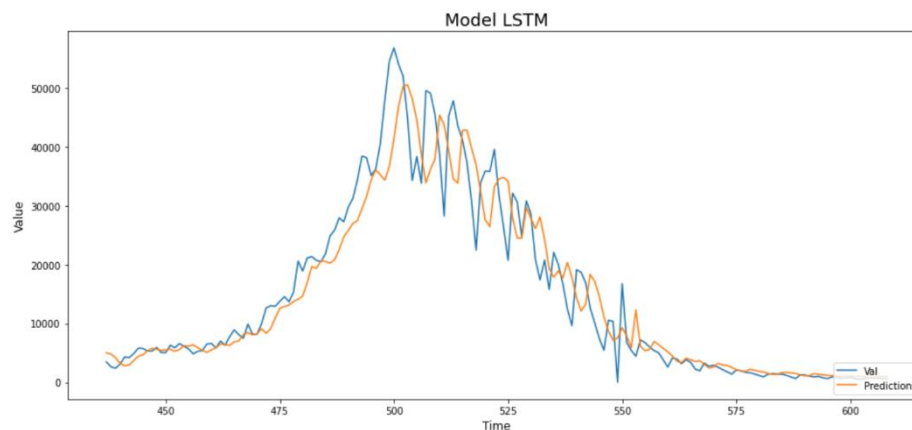
GAMBAR 12. Nilai MAE Selama Proses Pelatihan LSTM

Berdasarkan plot diatas, dapat dilihat bahwa nilai dari *training* dan *validation* MAE menurun, dan menghasilkan nilai MAE yang cukup kecil serta mendekati 0, artinya model yang dihasilkan oleh model LSTM pada data uji menghasilkan model yang baik.



GAMBAR 13. Nilai *Loss* Selama Proses Pelatihan LSTM

Berdasarkan plot diatas, dapat dilihat bahwa nilai dari *training* dan *validation* loss menurun, dan menghasilkan nilai loss yang cukup kecil serta mendekati 0, artinya model yang dihasilkan oleh model LSTM pada data uji menghasilkan model yang baik.



GAMBAR 14. Grafik Historis dari Penambahan Kasus Covid-19 dan Prediksi dari Penambahan Kasus Covid-19

Berdasarkan pengujian data didapatkan hasil terbaik dengan model LSTM menggunakan jumlah epoch sebanyak 25, batch size sebanyak 64, dan jumlah neuron sebanyak 200. Plot diatas menunjukkan bahwa hasil peramalan yang dihasilkan oleh model LSTM pada data uji menghasilkan pola data yang cukup mirip dengan data uji asli. Sehingga, secara eksploratif dapat dikatakan bahwa model LSTM ini menghasilkan performa yang baik pada data uji.

KESIMPULAN

Paper ini memberikan gambaran tentang prediksi kasus Covid-19 di Indonesia. Jumlah kasus Covid-19 di Indonesia sempat melonjak tinggi hingga ribuan per hari dengan data yang fluktuatif. Banyak penelitian sebelumnya yang menggunakan teknik *deep learning*, namun nilainya masih memberikan hasil error pada pemodelannya. Studi ini menggunakan uji coba prediksi berbasis teknik *deep learning* metode *Long Short Term Memory* (LSTM) untuk membentuk sebuah model. Dengan metode ini dapat diminimalkan nilai kesalahan pada model yang dihasilkan. Berdasarkan pada percobaan, nilai error minimum adalah RMSE 0,085917. Berdasarkan hasil prediksi dapat dilihat pada gambar Model LSTM, grafik prediksi yang ditampilkan oleh Model LSTM pada data uji sudah cukup mirip dengan data uji asli. Maka dari itu, *paper* ini disusun untuk membantu memberikan gambaran mengenai metode yang baik untuk prediksi kasus Covid-19 yaitu dengan menggunakan metode LSTM.

REFERENSI

1. Alzahrani, S.I., Aljamaan, I.A., & Al-Fakih, E.A. (2020). Forecasting the Spread of the COVID-19 Pandemic in Saudi Arabia Using ARIMA Prediction Model Under Current Public Health Interventions. *Journal of Infection and Public Health*, Vol. 13(7), pp. 914–919. <https://doi.org/10.1016/j.jiph.2020.06.001>.
2. Heizer, B., 2009. Jay dan Render, Manajemen Operasi, Buku 1 Edisi 9. *Jakarta: Salemba Empat*.
3. Rosadi, D. (2011). Analisis Ekonometrika dan Runtun Waktu Terapan dengan R. Yogyakarta: Andi Offset.
4. Rosadi, D. (2014). Analisis Runtun Waktu dan Aplikasinya dengan R. Yogyakarta: UGM Press.
5. Rustam, F., Reshi, A.A., Mehmood, A., Ullah, S., On, B.W., Aslam, W. & Choi, G.S. (2020). COVID-19 Future Forecasting Using Supervised Machine Learning Models. *IEEE Access*, Vol. 8, pp. 101489–101499. <https://doi.org/10.1109/ACCESS.2020.2997311>.
6. Sen, S., Sugiarto, D. and Rochman, A., 2020. Komparasi Metode Multilayer Perceptron (MLP) dan Long Short Term Memory (LSTM) dalam Peramalan Harga Beras. *target*, 4, p.1.
7. Vinayakumar, R., Soman, K.P. & Poornachandran, P. (2017). Long Short-Term Memory Based Operation Log Anomaly Detection. 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI) 2017, pp. 236–242. <https://doi.org/10.1109/ICACCI.2017.8125846>.

8. Worldometer, Indonesia Population (LIVE), Worldometer.info, 2021. [Online]. Available: <https://www.worldometers.info/world-population/indonesia-population/>. [Diakses tanggal 11-Nov-2021]
9. Zandrato, W., 2020. Gerakan Mencegah Daripada Mengobati terhadap Pandemi Covid-19. *Jurnal Education and Development*, 8(2), pp.242-242.
10. Zhang, Y., Hutchinson, P., Lieven, N.A.J. & Nunez-Yanez, J. (2020). Remaining Useful Life Estimation Using Long Short-Term Memory Neural Networks and Deep Fusion. *IEEE Access*, Vol. 8, pp. 19033-19045. <https://doi.org/10.1109/ACCESS.2020.2966827>.