ANALISIS EFEKTIVITAS TEKNIK IMPUTASI PADA LSTM UNTUK MENINGKATKAN KUALITAS DATA PADA PERAMALAN CURAH HUJAN

Ariyanto Adi Nugroho[1], Muhammad Haris[2]

[12,] Program Studi Ilmu Komputer, Fakultas Teknologi Informasi, Universitas Nusa Mandiri, Jakarta

Nusa Mandiri Tower Jl. Jatiwaringin Raya No. 2, Jakarta Timur 13620
[1] 14210241@nusamandiri.ac.id, [2] muhammad.uhs@nusamandiri.ac.id

Abstract

Climate monitoring data obtained from meteorological stations can have missing values due to various reasons. Data incompleteness occurs because of transmission failure, non-responsive sensors, equipment repairs, and other issues. The problem triggers inconsistent data and noise in climate measurement data. A better solution to handle missing values in weather data proposed on this work. Data imputation method address these issues before further analysis is conducted. This research proposes the application of imputation techniques during the data preparation phase. The research findings indicate that the best imputation method is KNN combined with Bidirectional LSTM. The evaluation metric results are Mean Absolute Error (MAE) 3,3599, Mean Square Error (MSE) 78,4336, Root Mean Squared Error (RMSE) 8,8562 and R-Square 0,5365.

Keywords : *LSTM, LOCF, KNN, NOCB, MICE*

Abstrak

Data pemantauan iklim yang didapatkan dari stasiun meteorologi dapat memiliki *missing value* karena berbagai hal. Ketidaklengkapan data dapat terjadi karena transmisi gagal, sensor tidak merespons, perbaikan perangkat, dan lain-lain. Masalah yang didapati umumnya adalah data tidak konsisten dan adanya *noise* pengukuran data iklim. Diperlukan solusi penanganan *missing values* pada data cuaca agar dapat diatasi sebelum dilakukan analisis lebih lanjut. Penelitian ini mengusulkan penerapan *data imputation* pada fase *data preparation* menyesuaikan karakteristik data. Metode *forecasting* yang diterapkan adalah LSTM dan Bidirectional LSTM yang merupakan turunan dari RNN. Metode ini menghasilkan model dari data *time series* yang lebih baik dibanding RNN. Hasil penelitian menyimpulkan metode imputasi yang memiliki performa terbaik adalah KNN dipadukan dengan metode Bidirectional LSTM. Nilai evaluation metric yang diperoleh adalah Mean Absolute Error (MAE) sebesar 3,3599, Mean Square Error (MSE) sebesar 78,4336, Root Mean Squared Error (RMSE) sebesar 8,8562 dan R-Squared sebesar 0,5365.

Kata kunci : *LSTM, LOCF, KNN, NOCB, MICE*

1. PENDAHULUAN

Data berperan penting dalam menentukan keberhasilan hasil penelitian. Isu utama yang sering muncul dalam kualitas data adalah keberadaan nilai yang hilang atau *'missing values*'. Masalah ini timbul ketika ada bagian dari data yang tidak tersedia atau hilang. Fenomena nilai hilang sering terjadi dalam kumpulan data, yang disebabkan oleh berbagai faktor seperti kerusakan perangkat, kesalahan perhitungan, kegagalan pencatatan data, serta masalah teknis lainnya [1]. Nilai yang hilang ini seringkali menjadi penghalang. Data yang hilang biasanya penting. Hilangnya nilai dapat mengakibatkan proses analisis menjadi tidak akurat, tidak efisien dan menurunkan akurasi [2].

Mekanisme *missing values* atau nilai yang hilang menurut Donald B. Rubin dikelompokkan menjadi tiga [3]. Mekanisme pertama adalah *Missing at Completely Random* (MCAR). MCAR terjadi ketika *missing value* tidak memiliki relasi atau dependensi pada data yang diobservasi, tidak diobservasi maupun missing data itu sendiri [3]. Mekanisme kedua adalah *Missing at Random* (MAR). MAR terjadi ketika *missing value* memiliki relasi dengan nilai yang diobservasi [3]. MCAR mengacu pada situasi di mana kehilangan data terjadi secara sepenuhnya acak, tanpa ada keterkaitan dengan data atau atribut lain. Pada MAR, kehilangan data berkaitan dengan atribut lain atau dengan data yang teramati. Contoh MAR adalah responden perempuan cenderung tidak berkenan menyebutkan umur maupun berat badan. Kehilangan data pada umur dan berat badan berhubungan dengan jenis kelamin. Mekanisme ketiga adalah *Missing Not At Random* (MNAR). MNAR terjadi ketika *missing value* memiliki relasi dengan nilai yang diobservasi. MNAR terjadi ketika kehilangan data berkaitan langsung dengan nilai yang hilang itu sendiri, dimana nilai yang tidak tercatat berhubungan dengan suatu kejadian yang tidak teramati.[1] [4].

Missing value juga sering ditemukan pada data berjenis *time series*. *Time series* merupakan rentetan data yang dikumpulkan secara berkala dan disusun berdasarkan urutan waktu [5]. Metode statistik awalnya digunakan untuk melakukan prediksi *time series*. Pada perkembangannya *deep learning* dengan metode *regression* pada akhirnya mengungguli metode statistik yang sudah diterapkan dari tahun 1970 [6]. Model *deep learning* didapati dapat menganalisis data *time series* secara terukur dan akurat [7]. *Deep learning model* dikenal lazim diterapkan pada data sekuensial atau *time series*. Metode-metode yang termasuk adalah *Convolutional Neural Network* (CNN), *Gated Recurrent Units (GRU), Recurrent Neural Networks* (RNN), *Long Short-Term Memory* (LSTM) [8].

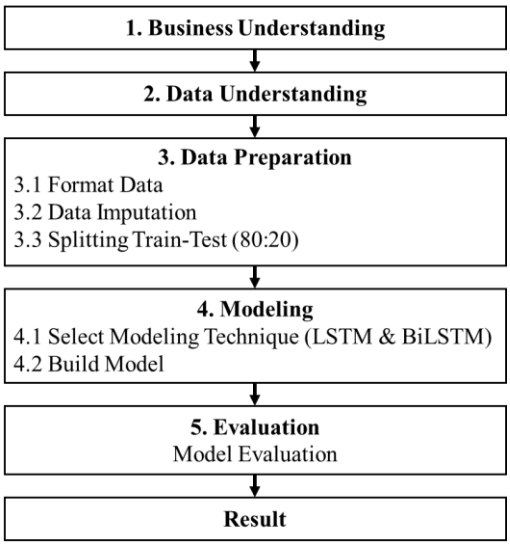
LSTM didesain untuk memecahkan isu *vanishing gradient* pada RNN [9]. Isu ini terjadi ketika nilai *gradient* terlalu kecil sehingga model berhenti melakukan *training* [10]. Model LSTM memiliki kemampuan yang baik untuk mengingat dependensi jangka panjang karena memiliki *constructed cell* untuk menyimpan informasi. Prinsip dari *cell* ini untuk mengelola pembaruan memori jangka sehingga informasi dan *gradient* dapat mengalir tanpa ada perubahan selama iterasi berlangsung [8].

Data pemantauan iklim yang didapatkan dari stasiun meteorologi dapat memiliki *missing value* karena berbagai hal. Ketidaklengkapan data dapat terjadi karena transmisi gagal, sensor tidak merespons, perbaikan perangkat, dan lain-lain. Masalah yang didapati umumnya adalah data tidak konsisten dan adanya *noise* pengukuran data iklim. Diperlukan solusi penanganan *missing values* pada data cuaca agar dapat diatasi sebelum dilakukan analisis lebih lanjut [11]. Hal ini penting karena *dataset* yang lengkap dapat mempengaruhi pengambilan keputusan yang memanfaatkan data tersebut. Data dengan kualitas yang rendah dapat menyebabkan analisis yang tidak akurat yang berujung kesalahan pengambilan kebijakan [3].

Berbeda dengan penelitian sebelumnya, kontribusi penelitian ini terletak pada perbandingan komprehensif antara variasi metode imputasi data dan algoritma LSTM serta Bi LSTM, yang belum pernah dilakukan dengan tingkat kedalaman dan variasi yang sama dalam penelitian terdahulu, sehingga menghasilkan performa yang terbaik menggunakan LSTM. Penelitian ini membuktikan bahwa imputasi data mampu meningkatkan kualitas data cuaca yang berdampak kepada performa model. Pada hasil eksperimen, performa terbaik diperoleh dengan menggunakan BiLSTM dan KNN sebagai metode data imputasi. Evaluasi performa model dilakukan melalui berbagai metrik evaluasi seperti MAE, MSE, RMSE, dan R-Squared, untuk menilai keakuratan dan efektivitas model. Hasil prediksi dibandingkan dengan data asli dan divisualisasikan untuk memberikan gambaran jelas mengenai peningkatan kualitas data yang dicapai melalui pendekatan ini.

2. METODOLOGI PENELITIAN

Penelitian mengadopsi metode Cross Industry Standard Process for Data Mining (CRISP-DM) dari tahapan Business Understanding hingga Evaluation. Metodologi penelitian yang digunakan dan diterapkan diilustrasikan pada Gambar 1 dibawah.



Gambar 1 Metodologi Penelitian

1. *Business Understanding*

Fase *business understanding* digunakan tujuan dan sasaran dari penelitian dipahami dan didefinisikan. Penelitian dilakukan untuk menentukan metode data imputation terbaik untuk meningkatkan *forecast* data iklim dari Stasiun Meteorologi Kelas III Kemayoran Jakarta Pusat selama 3.530 hari dari Januari 2011 hingga Agustus 2020. Penelitian dilakukan *multivariate* pada parameter curah hujan dengan nama *field* Tavg, RH_avg dan RR.

2. Data Understanding

Dataset yang diunduh berasal dari Stasiun Meteorologi Kemayoran adalah Unit Pelaksana Teknis (UPT) yang berkedudukan pada kantor pusat BMKG.

Data stasiun kemayoran terdiri atas 11 *field* yang terdiri atas Tanggal, Tn (suhu minimum dalam derajat Celcius), Tx (suhu maksimum dalam derajat Celcius), Tavg (suhu rata-rata dalam derajat Celcius), RH_avg (pengukuran kelembaban rata-rata dengan satuan %), RR (curah hujan dengan satuan mm), ss (lamanya durasi penyinaran matahari dengan satuan jam), ff_x (pengukuran kecepatan angin maksimum dengan satuan m/s), ddd_x (arah angin terdeteksi saat kecepatan maksimum dengan satuan °) , ff_avg (pengukuran kecepatan angin rata-rata dengan kecepatan m/s), dan ddd_car (arah angin terbanyak yang tercatat dengan satuan °).

Eksperimen dilakukan secara multivariate dengan target field Tavg, RH_avg dan RR. Field dipilih karena Tavg menunjukkan variabilitas yang sesuai. Kelembaban relatif (RH_avg) memiliki rata-rata 75,73% dengan standar deviasi 6,27% yang menunjukkan variabilitas tetapi masih dalam kisaran yang umum untuk iklim Indonesia yang cenderung lembab. Curah hujan (RR) memiliki rentang yang luas dari 0 hingga 277,5 mm dengan rata-rata 6,29 mm.

3. Data Preparation

Dataset diunduh pada rentang Januari 2011 sampai Agustus 2020 dari aplikasi pada URL https://dataonline.bmkg.go.id/home. Aplikasi ini disediakan oleh BMKG untuk menyediakan layanan data bagi kalangan internal maupun eksternal yang terdiri dari perguruan tinggi, institusi kementerian, lembaga, swasta dan masyarakat. Penulis tidak melakukan pengukuran dengan sensor mandiri seperti pada tulisan [12] dan Handayani [13].

Dataset diunduh pada rentang Januari 2011 sampai Agustus 2020 dari aplikasi pada URL https://dataonline.bmkg.go.id/home. Aplikasi ini disediakan oleh BMKG untuk menyediakan layanan data bagi kalangan internal maupun eksternal yang terdiri dari perguruan tinggi, institusi kementerian, lembaga, swasta dan masyarakat. Langkah untuk akuisisi data adalah sebagai berikut:

1. Cek ketersediaan data iklim pada menu “Ketersediaan Data”
2. Bagi pengguna baru, klik menu Registrasi dan isi formulir registrasi dengan benar. Aktivasi akun dengan mengakses link verifikasi pada alamat e-mail yang didaftarkan
3. Bagi pengguna dengan akun aktif, dapat login dengan email dan password yang teraktivasi. Isi kode Captcha untuk menyelesaikan proses login
4. Pilih menu Data Iklim, lalu pilih menu Data Harian
5. Pilih Jenis Stasiun, Parameter, Provinsi, Kabupaten, No/Nama Stasiun dan Rentang Waktu. Isian yang digunakan adalah sebagai berikut:
   - Jenis Stasiun: UPT
   - Parameter: arah angin saat kecepatan maksimum (ddd_x), arah angin terbanyak (ddd_car), curah hujan (RR), kecepatan angin maksimum (ff_x), kecepatan angin rata-rata (ff_avg), kelembaban rata-rata (RH_avg), lamanya penyinaran matahari (ss), tempratur maksimum

(Tx), tempratur minimum (Tn), tempratur rata-rata (Tavg)
- Provinsi: DKI Jakarta
- Kabupaten: Kota Adm. Jakarta Pusat
- No/Nama Stasiun: 96745 Stasiun Meteorologi Kemayoran
- Rentang Waktu: hari pertama bulan dan hari terakhir bulan yang akan diunduh

6. Klik tombol Proses. Aplikasi kemudian menampilkan kolom penilaian pelayanan. Isi penilaian pelayanan dan data akan terunduh.
7. Klik nama Profil (misalnya Ariyanto Adi Nugroho) dan klik Logout untuk keluar dari aplikasi

Data diunduh dalam format Microsoft Excel dengan ekstensi .xlsx. Satu file Microsoft Excel berisi data pengukuran selama 1 bulan. Terdapat 116 file yang diunduh untuk mendapatkan data pengukuran dari Januari 2011 hingga Agustus 2020.

Data diformat untuk mempersiapkannya dengan beberapa tahapan, antara lain skiprows, drop, to_datetime, replace, dan append. Seluruh parameter menggunakan library Pandas. Pandas adalah library bahasa pemrograman Python untuk melakukan manipulasi dan analisis terhadap string, numeric, datetime dan time series data [29]. Penjelasannya adalah sebagai berikut:

a. skiprows

Parameter pada function pandas.read_excel(). Parameter ini dapat digunakan untuk melewati sejumlah baris pada saat awal pembacaan file untuk dapat diubah menjadi DataFrame. Tipe isian data yang diterima dapat berupa integer. Penggunaan parameter ini dalam rangka mengabaikan header dari pada file dataset.

b. drop

Function pada DataFrame pandas. Function digunakan untuk menghilangkan sejumlah baris atau kolom pada DataFrame. Tipe isian data yang diterima dapat berupa string atau daftar beberapa string. Penggunaan parameter ini dalam rangka menghilangkan data yang tidak relevan agar peneliti dapat fokus pada data yang relevan.

c. to_datetime

Function pada pandas untuk melakukan konversi format data menjadi datetime. Tipe isian data yang diterima dapat berupa string, angka maupun list berisi tanggal dan waktu pada parameter 'arg', format tanggal waktu pada 'format', dan konstanta ignore, raise, dan coerce pada parameter 'errors'. Penggunaan parameter ini dalam rangka standarisasi format tanggal dan waktu.

d. replace

Function pada pandas untuk melakukan penggantian nilai pada DataFrame. Penggunaan parameter ini dalam rangka membersihkan, mempersiapkan data, dan mengganti nilai yang tidak valid seperti 8888 dan 9999 yang ditemui pada dataset.

e. append

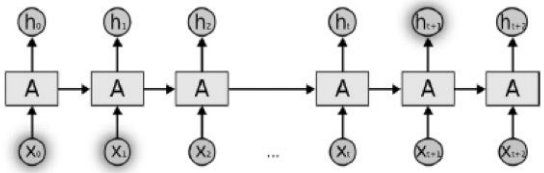
Function pada pandas untuk menambahkan baris pada DataFrame. Penggunaan parameter ini dalam rangka menggabungkan dataset pengukuran data iklim selama 116 bulan dari Januari 2011 sampai Agustus 2020.

Langkah terakhir dilakukan data imputation dengan metode yang sesuai. Data kemudian dipisah menjadi 80% testing data dan 20% training data.

4. Modeling

Fase modeling dilakukan dengan melakukan penerapan LSTM dan Bi LSTM untuk melakukan training menggunakan TensorFlow dan Keras.

Long Short-Term Memory (LSTM) adalah pengembangan dari RNN. LSTM memiliki struktur yang lebih canggih dan memiliki hidden layer atau lapisan tersembunyi [14]. Model ini dikenal efektif dalam menganalisis data berurutan atau sequential, termasuk data *time series*. [15].



Gambar 2 Ilustrasi *Memory* pada RNN

Gambar 2 mengilustrasikan kekurangan RNN. Dapat diamati pada gambar tersebut terdapat input *X, X1*. Keduanya memiliki rentang informasi besar dengan *Xt, X1+1*, sehingga ketika output *Ht+1* memerlukan nilai input yang sesuai dengan *X1, X0* model ini tidak dapat melakukan pembelajaran untuk menyesuaikan informasi karena sudah tergantikan. Memori lama ini tergantikan oleh data yang terdapat pada baris time series selanjutnya. Masalah dapat diatasi dengan model LSTM. Metode turunan ini dapat mengatur memori terhadap setiap inputannya. LSTM menerapkan memory cells dan gate units dengan nama Input Gate, Forget Gate, dan Output Gate [15].

a. Forget Gate
Forget gate menentukan informasi mana dari cell state sebelumnya yang harus dipertahankan atau dibuang. Rumus Forget Gate adalah:

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right)$$

b. Input Gate
Input gate memutuskan nilai mana dari input yang akan diperbarui dalam cell state. Rumus Input Gate adalah:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

c. Candidate Cell State
Vektor candidate cell state berisi informasi baru yang bisa ditambahkan ke cell state. Rumus Candidate Cell State adalah:

$$\widetilde{C_t} = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

d. Cell State
Cell state diperbarui dengan mengombinasikan cell state sebelumnya dan kandidat cell state, yang diatur oleh forget dan input gate. Rumus Cell State adalah:

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \widetilde{C_t}$$

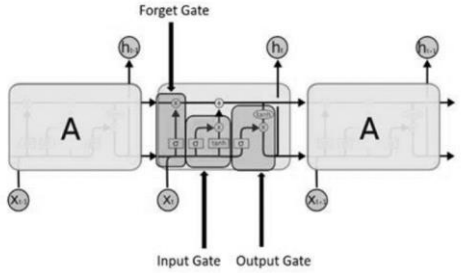
e. Output Gate
Output gate menentukan keluaran dari unit LSTM pada waktu tertentu. Rumus Output Gate adalah:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

f. Hidden State
Hidden state adalah keluaran akhir dari unit LSTM pada waktu yang digunakan pada langkah waktu berikutnya. Rumus Hidden State adalah:

$$h_t = o_t \cdot \tanh(C_t)$$

Gate-gate yang ada diilustrasikan pada Gambar 3.



Gambar 3 Ilustrasi *Memory Cells* dan *Gate Units* LSTM

Bidirectional Long Short-Term Memory (Bi LSTM) adalah ekstensi dari LSTM network tradisional dan implementasi bi-directional recurrent neural network. Terdapat dua hidden layer yang berlawanan arah yang saling terhubung kepada output layer yang sama. Karena koneksi tambahan ini, maka output layer mendapat keuntungan dari backward dan future state secara simultan [16].

Pada LSTM tradisional informasi bergerak ke depan melalui jaringan. Setiap node hanya menerima informasi dari node sebelumnya. Pada Bi LSTM informasi dapat mengalir dalam dua arah karena terdapat dua lapisan terpisah dalam jaringan. Lapisan pertama menyampaikan informasi dari belakang ke depan (forward pass). Lapisan kedua mengirimkan informasi dari depan ke belakang (backward pass). Pendekatan ini memungkinkan jaringan memiliki konteks dari masa lalu dan masa depan dan dibaca dari dua arah [17]. Hasilnya kemampuan jaringan dalam memahami konteks data meningkat ketika diterapkan pada Natural Language Processing dan analisis time series.

Percobaan pertama diterapkan menggunakan metode LSTM. Percobaan kedua diterapkan dengan Bidirectional LSTM.

5. Evaluation

Pada fase evaluation performa model diukur. Pengukuran performa menggunakan Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) dan R-Square.

a. Mean Absolute Error (MAE)
Mean Absolute Error (MAE) merupakan nilai pengukuran error antara observasi yang disandingkan yang memiliki fenomena yang sama. MAE mengukur kesalahan rata-rata absolut dalam prediksi statistik atau machine learning. Metrik pengukuran MAE menghitung rata-rata dari selisih absolut antara nilai yang diprediksi dan nilai sebenarnya [30]. Makin kecil nilai MAE menunjukkan perbedaan antara nilai prediksi dan nilai aktual minimal. Dapat disimpulkan model lebih akurat dalam melakukan prediksi [31]. Rumus perhitungan MAE adalah sebagai berikut:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

Nilai n adalah jumlah sampel, yi adalah nilai sebenarnya dan $\hat{y}_i$ adalah nilai prediksi [32].

b. Mean Squared Error (MSE)

Mean Squared Error (MSE) adalah metrik pengukuran kualitas estimator, yakni rata-rata dari kuadrat kesalahan atau perbedaan antara nilai yang diestimasi dan nilai sebenarnya [32]. Secara matematis, MSE dihitung sebagai rata-rata dari kuadrat perbedaan antara nilai prediksi yang dihasilkan oleh model dan nilai aktual. Nilai MSE yang rendah menunjukkan bahwa model memiliki kesalahan prediksi yang kecil. Kesalahan prediksi yang kecil maknanya kinerjanya baik dalam memprediksi hasil. Rumus perhitungan MSE adalah sebagai berikut:

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

Nilai n adalah jumlah sampel, *yi* adalah nilai sebenarnya untuk sampel ke-i dan $\hat{y}_i$ adalah nilai prediksi untuk sampel ke-i [32].

c. Root Mean Squared Error (RMSE)
Root Mean Squared Error (RMSE) merupakan metrik evaluasi yang digunakan untuk mengukur rata-rata kesalahan kuadrat antara nilai-nilai yang diobservasi (atau nilai sebenarnya) dan nilai-nilai yang diprediksi oleh model. RMSE merupakan akar kuadrat dari Mean Squared Error (MSE). RMSE memberikan estimasi ukuran kesalahan model dalam unit yang sama dengan variabel yang diprediksi. Rumus perhitungan RMSE adalah sebagai berikut:

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

Nilai n adalah jumlah sampel, *yi* adalah nilai sebenarnya untuk sampel ke-i dan $\hat{y}_i$ adalah nilai prediksi untuk sampel ke-i [33].

d. R-Squared
R-squared atau koefisien determinasi, adalah metrik statistik yang digunakan untuk mengukur proporsi variansi dalam variabel dependen yang dapat dijelaskan oleh variabel independen dalam model regresi. R-squared adalah ukuran seberapa baik prediksi model sesuai dengan data aktual. Nilai R-Squared dinyatakan pada rentang 0 hingga 1, di mana nilai yang lebih tinggi menunjukkan bahwa model dapat menjelaskan proporsi variansi yang lebih besar dalam variabel dependen.
Nilai R-Square dihitung dengan rumus nilai 1 dikurangi hasil bagi dari SSres dan SStot. SSres adalah jumlah kuadrat residu (sum of squared residuals) yang mengukur variasi antara nilai yang diamati dan nilai yang diprediksi oleh model. SStot adalah jumlah kuadrat total (total sum of squares) yang mengukur variasi total dari nilai yang diamati [34]. Perhitungan R-Square dapat dituliskan sebagai persamaan sebagai berikut:

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

6. Deployment
Penelitian ini tidak menerapkan fase Deployment dari CRISP-DM.

3. HASIL DAN PEMBAHASAN

Eksperimen dilakukan secara *multivariate* dengan *target field* Tavg, RH_avg dan RR. *Field* dipilih karena Tavg menunjukkan variabilitas yang sesuai. Kelembaban relatif (RH_avg) memiliki rata-rata 75,73% dengan standar deviasi 6,27% yang menunjukkan variabilitas tetapi masih dalam kisaran yang diharapkan untuk iklim Indonesia yang cenderung lembab. Curah hujan (RR) memiliki rentang yang luas dari 0 hingga 277,5 mm dengan rata-rata 6,29 mm.

Dataset yang dipublikasikan berformat Microsoft Excel. Satu file Microsoft Excel berisi data pengukuran selama 1 bulan. Terdapat 116 file yang diunduh untuk mendapatkan data pengukuran dari Januari 2011 hingga Agustus 2020.

Dataset yang telah diperoleh kemudian melalui tahap *data preparation* (penghapusan teks yang tidak diperlukan, penghapusan *noise*, dan penerapan *data imputation*).

Eksperimen dilakukan sebanyak 16 kali yang dibagi menjadi 2 kelompok. Kelompok pertama menerapkan LSTM dan kelompok kedua menerapkan Bidirectional LSTM. Skema pembagiannya dijelaskan pada Tabel 1.

TABEL I. METODE *DATA IMPUTATION* DAN METODE *FORECASTING*

Eksperimen	Metode *Data Imputation*	Metode *Forecasting*
LSTM.1	-	LSTM
LSTM.2	Mean	
LSTM.3	Median	
LSTM.4	Mode	
LSTM.5	LOCF	
LSTM.6	NOCB	
LSTM.7	kNN	
LSTM.8	MICE	
BiLSTM.9	-	Bi LSTM
BiLSTM.10	Mean	
BiLSTM.11	Median	
BiLSTM.12	Mode	

Eksperimen	Metode *Data Imputation*	Metode *Forecasting*
BiLSTM.13	LOCF	
BiLSTM.14	NOCB	
BiLSTM.15	kNN	
BiLSTM.16	MICE	

TABEL II. PARAMETER DAN NILAI STANDAR

Nama Parameter	Nilai Standar pada Eksperimen
Window Size	240
Dense Units	25 neurons
Dropout Rate	0,5 (50%)
Optimizer	Adam optimizer
Number of Neurons in Dense Layer	25
Epochs	50 epoch
Batch Size	500
Loss Function	Mean Squared Error

Hyperparameter pada 16 eksperimen pada Tabel 1 didaftar pada Tabel 2. Penjelasan masing-masing *hyperparameter* adalah sebagai berikut:

a. *Window Size* ditetapkan pada 240, yang berarti model akan melihat 240 langkah waktu sebelumnya untuk memprediksi nilai saat ini. Pemilihan angka ini didasarkan pada panjang data sekuensial yang digunakan dalam pelatihan, yaitu data cuaca yang diukur selama 3.530 hari. Ukuran *window* ini dianggap optimal setelah beberapa skenario uji coba karena memberikan keseimbangan yang baik antara menangkap informasi temporal yang relevan dan menghindari *overfitting*. Ukuran *window* yang terlalu kecil mungkin tidak cukup menangkap pola jangka panjang, sementara ukuran yang terlalu besar bisa menambah kompleksitas dan menyebabkan model *overfitting*.
b. *Dense Units* diatur pada 25 neuron untuk lapisan LSTM dan Bi-LSTM. Jumlah neuron ini dipilih setelah beberapa uji coba yang menunjukkan bahwa dengan 25 neuron, model mampu menangkap informasi penting tanpa menjadi terlalu kompleks. Jika jumlah neuron terlalu sedikit, model mungkin gagal menangkap pola yang relevan dalam data. Namun, jika jumlah neuron terlalu banyak, model bisa menjadi terlalu kompleks, memakan waktu komputasi yang lebih lama, dan berisiko overfitting. Optimizer adalah algoritma optimasi yang digunakan untuk meminimalkan *loss fuction*.
c. *Dropout Rate* ditetapkan pada 0,5 (50%). *Dropout* adalah teknik regulasi yang digunakan untuk mengurangi *overfitting* dengan secara acak menonaktifkan setengah dari neuron selama pelatihan. Nilai ini dipilih karena hasil uji coba menunjukkan bahwa *dropout* sebesar 50% memberikan keseimbangan yang optimal antara mengurangi *overfitting* dan mempertahankan kapasitas model. Dropout yang lebih rendah mungkin tidak cukup untuk mencegah *overfitting*, sementara *dropout* yang lebih tinggi bisa mengurangi terlalu banyak informasi yang diperlukan untuk pelatihan yang efektif.
d. *Optimizer* yang digunakan adalah Adam optimizer. Adam dipilih karena kemampuan adaptifnya yang secara otomatis menyesuaikan learning rate selama pelatihan, yang membuatnya sangat efektif untuk berbagai jenis data dan model. Uji coba dengan optimizer lain, seperti SGD atau RMSprop, menunjukkan bahwa Adam memberikan hasil yang lebih stabil dan cepat dalam konvergensi dengan loss function yang lebih rendah.
e. *Number of Neurons in Dense Layer* juga ditetapkan pada 25 neuron. Angka ini dipilih karena model dengan jumlah neuron ini memberikan hasil yang optimal dalam uji coba, terutama setelah lapisan LSTM dan Bi-LSTM. Jika jumlah neuron terlalu kecil, kemampuan model untuk memproses informasi yang diekstraksi dari lapisan LSTM bisa berkurang. Namun, jika terlalu banyak neuron, itu bisa menyebabkan model menjadi terlalu kompleks dan meningkatkan risiko *overfitting*.
f. *Epochs* ditetapkan pada 50. Jumlah *epoch* ini dipilih setelah uji coba menunjukkan bahwa model mulai mencapai konvergensi sekitar *epoch* ke-40 hingga ke-50, dengan tidak ada peningkatan signifikan pada performa setelah itu. Penggunaan lebih dari 50 *epoch* dapat menyebabkan *overfitting*, sementara lebih sedikit *epoch* mungkin tidak cukup untuk model belajar dari data secara efektif.
g. *Batch Size* diatur pada 500. Nilai ini dipilih untuk mencapai keseimbangan antara efisiensi komputasi dan keakuratan model. *Batch size* yang lebih kecil dari 500 menunjukkan hasil yang lebih fluktuatif dan membutuhkan waktu komputasi lebih lama, sementara *batch size* yang lebih besar dari 500 bisa meningkatkan waktu komputasi dan memori yang dibutuhkan tanpa peningkatan performa yang signifikan.

h. *Loss Function* yang digunakan adalah *Mean Squared Error* (MSE). MSE dipilih karena memberikan ukuran yang jelas dan kuantitatif dari kesalahan prediksi model. MSE dihitung sebagai rata-rata dari kuadrat perbedaan antara nilai yang diprediksi dan nilai sebenarnya. Fungsi kerugian ini efektif untuk regresi dan sangat sensitif terhadap outlier, yang sesuai dengan tujuan untuk meminimalkan kesalahan prediksi pada data cuaca. Uji coba dengan *loss function lain*, seperti Mean Absolute Error (MAE), menunjukkan bahwa MSE memberikan performa yang lebih baik dalam hal konvergensi dan stabilitas model.

1. Eksperimen Menggunakan Metode LSTM

Kelompok eksperimen dilakukan pertama kali terhadap dataset menggunakan metode LSTM dengan hasil pada Tabel 3

Nilai rujukan yang digunakan adalah MAE 11,3 yang dihasilkan oleh penelitian "Prediksi Curah Hujan Harian di Stasiun Meteorologi Kemayoran menggunakan Artificial Neural Network (ANN)" yang ditulis oleh Richard Mahendra Putra dan Nurhastuti Anjar Rani. Paper tersebut menggunakan rentang dataset yang sama yaitu Januari 2011 hingga Agustus 2020 [18]. Tabel 3 memaparkan 8 eksperimen yang menggunakan data imputation dan menerapkan LSTM. Hasil yang diperoleh secara umum memiliki performa MAE dibawah nilai 11,3.

TABEL III. HASIL EKSPERIMEN METODE LSTM

Eksperimen	Metode Forecasting	Metode Data Imputation	Evaluation Metrics			
			MAE	MSE	RMSE	R-Square
Rujukan	ANN	-	11,3	-	-	-
LSTM.1	LSTM	-	3,9079	131,0827	11,4491	0,3272
LSTM.2	LSTM	Mean	3,6772	115,2757	10,7366	0,3422
LSTM.3	LSTM	Median	3,6715	119,4415	10,9289	0,3306
LSTM.4	LSTM	Mode	3,5651	116,8508	10,8097	0,3503
LSTM.5	LSTM	LOCF	3,7328	124,4447	11,1554	0,3311
LSTM.6	LSTM	NOCB	3,8043	125,7679	11,2146	0,3441
LSTM.7	LSTM	KNN	3,7243	121,9033	11,0409	0,3262
LSTM.8	**LSTM**	**MICE**	**3,6319**	**116,3358**	**10,7859**	**0,3582**

Eksperimen menghasilkan nilai MAE dibawah 11,3. Nilai MAE terendah didapat pada Eksperimen LSTM.4 dengan nilai 3,5651. Nilai MSE dan RMSE terbaik didapat pada Eksperimen 2 yang mampu menghasilkan prediksi yang mendekati actual value dan memiliki tingkat error yang rendah. Nilai R-Squared terbaik diperoleh pada Eksperimen LSTM.8 dengan nilai 0,358 yang berarti memiliki proporsi varians tertinggi pada variabel dependen yang dapat diprediksi dari variabel independen.

Eksperimen LSTM.8 dapat disimpulkan sebagai eksperimen terbaik pada kelompok ini karena nilai yang didapat merepresentasikan keseimbangan akurasi dan kemampuan model menjelaskan varian (*explain the variance*).

2. Eksperimen Menggunakan Metode Bidirectional LSTM (Bi LSTM)

Kelompok eksperimen kedua dilakukan terhadap dataset dengan menggunakan Bi LSTM dengan hasil pada Tabel 4 menjabarkan 8 eksperimen yang menerapkan data imputation dan menggunakan metode Bi LSTM.

TABEL IV. HASIL EKSPERIMEN METODE BIDIRECTIONAL LSTM (BI LSTM)

Eksperimen	Metode *Forecasting*	Metode *Data Imputation*	Evaluation Metrics			
			MAE	MSE	RMSE	R-Square
Rujukan	ANN	-	11,3	-	-	-
BiLSTM.1	LSTM	-	4,8895	179,8081	13,4092	0,1284
BiLSTM.2	LSTM	Mean	4,0619	135,2290	11,6288	0,2570
BiLSTM.3	LSTM	Median	4,1676	135,5840	11,6440	0,2660

Eksperimen	Metode *Forecasting*	Metode *Data Imputation*	Evaluation Metrics			
			MAE	MSE	RMSE	R-Square
BiLSTM.4	LSTM	Mode	4,0849	132,0755	11,4924	0,2481
BiLSTM.5	LSTM	LOCF	3,4059	85,2731	9,2343	0,5282
BiLSTM.6	LSTM	NOCB	3,3998	81,6515	9,0361	0,5430
BiLSTM.7	**LSTM**	**KNN**	**3,3599**	**78,4336**	**8,8562**	**0,5365**
BiLSTM.8	LSTM	MICE	3,4103	80,0259	8,9457	0,5271

Penerapan metode *data imputation* mean, median, dan mode menampilkan hasil yang tidak jauh berbeda dengan eksperimen menggunakan metode LSTM. Hasil eksperimen menampilkan hasil yang bervariasi. Nilai MAE terendah didapat pada Eksperimen BiLSTM.7 dengan nilai 3,3599. Nilai MSE dan RMSE terbaik didapat pada Eksperimen BiLSTM.7 dengan MSE 78.4336 dan RMSE 8.8562. Eksperimen BiLSTM.7 mampu menghasilkan prediksi yang mendekati actual value dan memiliki tingkat error yang rendah. Nilai R-Squared terbaik diperoleh pada Eksperimen BiLSTM.6 dengan nilai 0.543. Nilai R-Square ini menu berarti model memiliki kekuatan penjelasan terbaik, mampu menjelaskan proporsi yang lebih besar dari varians pada variabel dependen.

Pada eksperimen-eksperimen diatas, terlihat bahwa terdapat variasi yang signifikan dalam kinerja model antara delapan eksperimen berbeda. MSE, RMSE, dan MAE memberikan ukuran tentang kesalahan prediksi model. Ketiga metrik tersebut ketika menghasilkan nilai lebih rendah maka menunjukkan kinerja yang lebih baik.

Nilai R-Squared dibagi menjadi tiga klasifikasi: kuat, moderat, dan lemah, dengan nilai 0,75 dianggap kuat, 0,50 sebagai moderat, dan 0,25 sebagai lemah. R-Squared adalah evaluation metric yang melengkapi MSE untuk mengukur performa model yang berasal dari data timeseries [19]. R-Square yang diperoleh 0,5365 yang masuk pada klasifikasi moderat.

Dari hasil yang diperoleh, dapat disimpulkan Eksperimen BiLSTM.7 yang menggunakan metode *data imputation* KNN dan metode BiLSTM merupakan eksperimen dengan performa terbaik. *Evaluation metric* yang diperoleh adalah MAE 3,3599, MSE 78,4336, RMSE 8,8562, dan R-Square 0,5365. Hasil eksperimen BiLSTM.7 juga mengungguli Eksperimen LSTM.8 sebagai perbandingan.

Keunggulan hasil eksperimen BiLSTM.7 ditunjang penerapan *data imputation* KNN yang menerapkan *similarity-based imputation*. Metode KNN bekerja dengan menemukan tetangga terdekat dari titik data yang hilang dan memasukkan nilai berdasarkan kesamaan antar titik data. Mengingat sifat data cuaca yang kontinu dan dapat diprediksi, KNN dapat secara efektif memanfaatkan korelasi antara titik waktu terdekat atau kondisi serupa untuk memperhitungkan nilai yang hilang secara akurat.

Dataset cuaca memiliki *temporal correlation* atau korelasi temporal. Data cuaca pada dasarnya bersifat temporal dan kondisi dari satu titik waktu sering kali berkaitan erat dengan kondisi di titik waktu terdekat. KNN dapat memanfaatkan korelasi temporal ini, terutama jika kumpulan data disusun dalam *timeseries*, untuk menemukan pola imputasi yang serupa.

Pendekatan multivariat yang menargetkan field Tavg, RH_avg dan RR. Field ini saling terkait membuat KNN dapat menangani data multivariat dengan baik. KNN mempertimbangkan ruang multidimensi yang diciptakan oleh variabel-variabel ini, sehingga memungkinkan imputasi yang lebih akurat dengan mempertimbangkan hubungan antara berbagai variabel meteorologi.

4. KESIMPULAN DAN SARAN

Pada penelitian ini, metode terbaik yang ditemukan menggunakan BiLSTM dan KNN sebagai teknik data imputasi dengan MAE 3.35, MSE 78.43, RMSE 8.86, R-square 0.54. Metode yang diajukan berhasil mengungguli performa model tanpa teknik data imputasi sebesar 45% jika dibandingkan pada metrik MAE.

Eksperimen BiLSTM.7 berhasil menunjukkan kinerja yang superior dibandingkan dengan eksperimen lainnya, dengan nilai MAE, MSE, dan RMSE terendah, serta nilai R-Squared yang paling mendekati 1. Ini menandakan bahwa model BiLSTM.7 memiliki kesesuaian terbaik dengan data dan kesalahan prediksi paling kecil. Hasil ini juga menunjukkan bahwa metode data imputation menggunakan KNN efektif dalam meningkatkan kualitas dataset dengan missing values, karena metode ini memanfaatkan kesamaan antara data untuk memperkirakan nilai yang hilang, yang penting untuk dataset temporal seperti curah hujan.

Selain itu, eksperimen lain yang dilakukan dengan variasi parameter dan metode menunjukkan kinerja yang bervariasi, dengan

nilai R-Squared yang lebih rendah dan kesalahan prediksi yang lebih tinggi. Ini mengindikasikan bahwa model tersebut kurang optimal dalam menyesuaikan diri dengan data yang memiliki missing values. Dalam beberapa kasus, penggunaan metode imputasi terbukti lebih efektif daripada menghapus data dengan missing values atau menggantinya dengan fitur lain, yang menunjukkan bahwa pendekatan ini dapat mempertahankan integritas dan keterkaitan temporal dalam dataset.

Ada beberapa hal yang dapat dianalisis lebih lanjut untuk meningkatkan kegunaan dari hasil penelitian:

1. Penelitian selanjutnya dapat melakukan modifikasi *hyperparameter* agar menghasilkan model yang lebih sesuai
2. Pada penelitian selanjutnya dapat menerapkan *cross-validation* sehingga dapat memperoleh konfigurasi *hyperparameter* yang lebih optimal dibanding *default value* yang telah diterapkan.

Daftar Pustaka

[1] S. Nikfalazar, C.-H. Yeh, S. Bedingfield, and H. A. Khorshidi, "Missing data imputation using decision trees and fuzzy clustering with iterative learning," *Knowl. Inf. Syst.*, vol. 62, pp. 2419–2437, 2020.

[2] W. Lan, X. Chen, T. Zou, and C.-L. Tsai, "Imputations for high missing rate data in covariates via semi-supervised learning approach," *J. Bus. \& Econ. Stat.*, vol. 40, no. 3, pp. 1282–1290, 2022.

[3] M. Alabadla *et al.*, "Systematic Review of Using Machine Learning in Imputing Missing Values," *IEEE Access*, vol. 10, pp. 44483–44502, 2022, doi: 10.1109/ACCESS.2022.3160841.

[4] M. Alabadla *et al.*, "Systematic Review of Using Machine Learning in Imputing Missing Values," *IEEE Access*, vol. 10, pp. 44483–44502, 2022, doi: 10.1109/ACCESS.2022.3160841.

[5] C. Chatfield, *The analysis of time series: An introduction*. Chapman & Hall/CRC, 2003.

[6] J. F. Torres, D. Hadjout, A. Sebaa, F. Martínez-Álvarez, and A. Troncoso, "Deep Learning for Time Series Forecasting: A Survey," *Big Data*, vol. 9, no. 1, pp. 3–21, 2021, doi: 10.1089/big.2020.0159.

[7] J. F. Torres, A. Galicia, A. Troncoso, and F. Martínez-Álvarez, "A scalable approach based on deep learning for big data time series forecasting," *Integr. Comput. Aided. Eng.*, vol. 25, no. 4, pp. 335–348, 2018, doi: 10.3233/ICA-180580.

[8] N. H. A. Rahman, M. Z. Hussin, S. I. Sulaiman, M. A. Hairuddin, and E. H. M. Saat, "Univariate and multivariate short-term solar power forecasting of 25MWac Pasir Gudang utility-scale photovoltaic system using LSTM approach," *Energy Reports*, vol. 9, no. S11, pp. 387–393, 2023, doi: 10.1016/j.egyr.2023.09.018.

[9] F. Wang, Z. Xuan, Z. Zhen, K. Li, T. Wang, and M. Shi, "A day-ahead PV power forecasting method based on LSTM-RNN model and time correlation modification under partial daily pattern prediction framework," *Energy Convers. Manag.*, vol. 212, p. 112766, 2020, doi: https://doi.org/10.1016/j.enconman.2020.112766.

[10] W. Yu, G. Liu, L. Zhu, and W. Yu, "Convolutional neural network with feature reconstruction for monitoring mismatched photovoltaic systems," *Sol. Energy*, vol. 212, pp. 169–177, 2020, doi: https://doi.org/10.1016/j.solener.2020.09.026.

[11] E. Afrifa-Yamoah, U. A. Mueller, S. M. Taylor, and A. J. Fisher, "Missing data imputation of high-resolution temporal climate time series data," *Meteorol. Appl.*, vol. 27, no. 1, p. e1873, 2020.

[12] K. Auliasari and M. Kertaningtyas, "ANALISIS KUALITAS UDARA MENGGUNAKAN ALGORITMA K-MEANS," *J. Inform. dan Rekayasa Elektron.*, vol. 4, no. 2, pp. 95–105, 2021.

[13] A. S. Handayani, S. Soim, T. E. Agusdi, R. Rumiasih, and A. Nurdin, "Klasifikasi Kualitas Udara Dengan Metode Support Vector Machine," *J. Inform. dan Rekayasa Elektron.*, vol. 3, no. 2, pp. 187–199, 2020.

[14] A. S. Temur, M. Akgün, and G. Temur, "Predicting housing sales in Turkey using ARIMA, LSTM and hybrid models," 2019.

[15] M. A. Ridla, N. Azise, and M. Rahman, "Perbandingan Model Time Series Forecasting Dalam Memprediksi Jumlah Kedatangan Wisatawan Dan Penumpang Airport," *Simkom*, vol. 8, no. 1, pp. 1–14, 2023, doi: 10.51717/simkom.v8i1.103.

[16] P. Nath, P. Saha, A. I. Middya, and S. Roy, "Long-term time-series pollution forecast using statistical and deep learning methods," *Neural Comput. Appl.*, vol. 33, no. 19, pp. 12551–12570, 2021, doi: 10.1007/s00521-021-05901-2.

[17] D. R. Alghifari, M. Edi, and L. Firmansyah, "Implementasi Bidirectional LSTM untuk Analisis Sentimen Terhadap Layanan Grab Indonesia," *J. Manaj. Inform.*, vol. 12, no. 2, pp. 89–99, 2022.

[18] R. M. Putra and N. Anjar Rani, "Prediksi Curah Hujan Harian di Stasiun Meteorologi Kemayoran Menggunakan Artificial Neural Network (ANN)," *Bul. GAW Bariri*, vol. 1, no. 2, pp. 101–108, 2020, doi: 10.31172/bgb.v1i2.35.

[19] I. Ghozali, "Aplikasi Analisis Multivariete dengan Program IBM SPSS 23," 2016.