

Prediksi Visibility Pada Penerbangan Menggunakan Metode Gate Recurrent Unit Berbasis Seleksi Fitur Cross Correlation Function

Rahmat Tri Wahyu Widodo ¹⁾, Nurissaiddah Ulinnuha ^{2)*}, Dian Candra Rini Novitasari ³⁾

Universitas Islam Negeri Sunan Ampel Surabaya, Indonesia ^{1),2),3)}

rahmattriwahyul@gmail.com ¹⁾, nuris.ulinnuha@uinsa.ac.id ^{2)*}, diancrini@uinsa.ac.id ³⁾

Abstrak

Visibility yang rendah menyebabkan terganggunya penerbangan terutama pada keterlambatan atau pembatalan jadwal penerbangan. Keadaan tersebut sulit diprediksi karena kondisinya yang relatif berubah-ubah tiap jamnya. Salah satu cara untuk mengetahui rendahnya visibility adalah dengan melihat jarak pandang aman di udara > 5 -kilometer sesuai aturan Dinas Perhubungan. Oleh karena itu, tujuan dari penelitian ini adalah membangun model prediksi visibility untuk mengantisipasi rendahnya visibility dengan menggunakan metode deep learning yaitu Gate Recurrent Unit (GRU). Variabel independen atau fitur yang digunakan dalam penelitian ini adalah suhu titik embun, suhu bola kering, kelembapan, arah dan kecepatan angin, serta curah hujan mulai awal hingga akhir tahun 2023. Hasil Cross Correlation Function (CCF) menunjukkan korelasi fitur yang memiliki nilai tinggi digunakan untuk masuk model pelatihan adalah kecepatan angin, suhu bola kering, dan kelembapan. Beberapa uji coba parameter dilakukan terbukti bahwa model memiliki performa yang baik dalam memprediksi visibility. Hasil model yang memiliki tingkat kesalahan prediksi menggunakan MAPE sebesar 12.75%.

Kata kunci: *Prediksi, Visibility, Cross Correlation Function, Gate Recurrent Unit*

Abstract

[Visibility Prediction in Flight Using Gate Recurrent Unit Method Based on Cross Correlation Function Feature Selection] Low visibility disrupts flights, especially delays or cancellations of flight schedules. This situation is difficult to predict because the conditions change relatively every hour. One way to find out the low visibility is to see the safe visibility in the air > 5 -kilometer according to the rules of the Department of Transportation. Therefore, the purpose of this research is to build a visibility prediction model to anticipate low visibility using the deep learning Gate Recurrent Unit (GRU) method. The independent variables or features used in this study are dew point temperature, dry bulb temperature, humidity, wind direction and speed, and rainfall from the beginning to the end of 2023. The Cross Correlation Function (CCF) results show that the feature correlations that have high values used to enter the training model are wind speed, dry bulb temperature, and humidity. Several parameter tests were conducted to prove that the model has good performance in predicting visibility. Model results that have a prediction error rate using MAPE of 12.75%.

Keywords: *Prediction, Visibility, Cross-Correlation Function, Gate Recurrent Unit*

1. PENDAHULUAN

Visibility atau biasa disebut jarak pandang pada penerbangan adalah jarak maksimum suatu objek pada area mendatar yang dapat dilihat dan diidentifikasi berdasarkan keadaan yang lebih cerah [1]. Rendahnya visibility secara umum disebabkan oleh kabut, asap, dan hujan dengan intensitas tinggi [2]. Kabut yang sering kali menyelimuti bandara di pagi hari atau setelah hujan semakin memperburuk visibility. Asap dari kebakaran hutan pada sekitar penerbangan di musim kemarau juga menjadi ancaman menutupi langit dan mengaburkan pandangan menurunkan visibility [3]. Angin kencang dan angin puting beliung selain menimbulkan bahaya bagi kestabilan penerbangan juga dapat membawa debu dan

partikel lain ke udara yang menyebabkan berkurangnya visibility di landasan [4].

Berdasarkan data Badan Meteorologi, Klimatologi, dan Geofisika (BMKG), unsur-unsur cuaca seperti curah hujan, suhu, kelembapan, dan angin yang dapat berdampak atau berpengaruh signifikan terhadap visibility di bandara. Hujan yang secara ilmiah dapat mengurangi jarak pandang, terutama saat terjadi badai atau hujan deras yang dapat mempengaruhi tingkat visibility. Suhu memiliki dampak langsung pada visibility dalam penerbangan. Saat suhu tinggi menyebabkan pemuaian udara sehingga terjadi pembentukan asap yang mengakibatkan rendahnya visibility. Kelembapan yang tinggi akan menghasilkan pembentukan kabut mengakibatkan berkurangnya visibility [5]. Keadaan tersebut sulit diprediksi karena

kondisinya yang relatif berubah-ubah tiap jamnya. Salah satu cara untuk mengetahui rendahnya *visibility* adalah dengan melihat jarak pandang aman di udara > 5 kilometer menurut aturan Dinas Perhubungan [6].

Sebelum melakukan prediksi diperlukan korelasi fitur menggunakan *Cross-Correlation Function* (CCF) untuk mengetahui faktor-faktor cuaca apa saja yang paling berpengaruh terhadap *visibility* dan mengetahui korelasi antar fitur yang maksimal. CCF memungkinkan untuk menganalisis hubungan antara dua variabel yaitu fitur dan target dalam waktu yang sama [7]. Dengan penambahan metode CCF ini dapat meningkatkan akurasi prediksi *visibility* [8].

Beberapa teknologi *machine learning* mampu untuk memprediksi dan mengantisipasi terjadinya cuaca ekstrem terutama pada rendahnya *visibility*. Metode *machine learning* digunakan untuk memprediksi faktor cuaca yang bersifat *time series* dan dapat memberikan informasi yang tepat waktu kepada pihak-pihak terkait mengenai kemungkinan gangguan *visibility*, sehingga langkah-langkah pencegahan dapat dilakukan terlebih dahulu.

Penelitian tentang prediksi *visibility* ini telah dilakukan beberapa kali, diantaranya yaitu, menggunakan metode ANN memperoleh nilai akurasi 0.9743, perbandingan metode *Long Short-Term Memory* (LSTM) dan *Multi-Layer Perceptron* (MLP) didapatkan nilai RMSE terendah yaitu sebesar 0.250 dan dikatakan LSTM metode yang lebih efektif dalam memprediksi *visibility* [9][10][11].

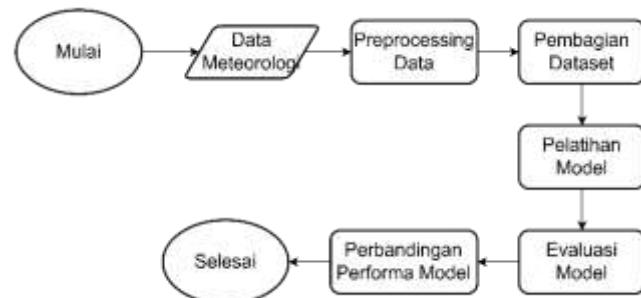
Selain itu, bagian dari *machine learning* yang dapat memahami sebuah pola dan mampu melakukan prediksi faktor cuaca dengan akurasi yang tinggi adalah *deep learning* dan salah satu algoritmanya yaitu *Gate Recurrent Unit* (GRU) [11]. GRU sebagai salah satu jenis jaringan saraf tiruan, mampu memproses data deret waktu dan menangkap pola yang mungkin tidak terdeteksi oleh metode konvensional [12]. GRU memiliki arsitektur lebih sederhana dibandingkan dengan model LSTM dengan menggabungkan *input gate* dan *forget gate* LSTM menjadi *update gate*

tunggal sehingga menghasilkan arsitektur yang lebih ramping [13]. Keunggulan metode GRU ditekankan pada perbandingan GRU dengan metode lainnya, yaitu dibuktikan dengan nilai dari uji coba GRU paling rendah [14][15][16].

Berdasarkan perbandingan penelitian sebelumnya, metode GRU mempunyai efisiensi yang baik dalam konteks prediksi dengan nilai akurasi yang tinggi dibandingkan dengan metode lain. Tujuan dari penelitian ini adalah untuk membuat sebuah model GRU untuk meramalkan *visibility* pada periode waktu tertentu. Model GRU yang dibuat terbukti dapat memberikan nilai eror yang rendah sehingga dianggap mampu meramalkan *visibility* dengan baik.

2. BAHAN DAN METODE

Alur Penelitian ini dapat terlihat pada Gambar 1:



Gambar 1. Alur Penelitian

2.1 Dataset Penelitian

Dataset yang digunakan dalam penelitian ini adalah sampel data cuaca setiap jam yang didapatkan dari website resmi Stamet kelas 1 BMKG Juanda Sidoarjo. Periode data yang digunakan yaitu pada tanggal 01 Januari - 31 Desember 2023. Selama periode tersebut, data yang terkumpul mencakup 8760 baris dan 8 kolom, yang merepresentasikan berbagai parameter cuaca secara terperinci untuk analisis lebih lanjut. Jumlah data dari beberapa parameter cuaca terdapat pada Tabel 1.

Tabel 1. Data Cuaca Januari – Desember 2023

Tanggal (WIB)	Suhu (°C)		Angin		Curah Hujan	Kelembapan	Visibility (km)
	Titik embun	Bola keriting	Kecepatan (kt)	Arah (°)	(mm/jam)	(%)	
01/01/2023 00.00	23.6	25.8	4.5	262.0	0.0	92.24	10
01/01/2023 01.00	23.7	26.9	8.7	243.0	0.0	89.18	10
01/01/2023 02.00	24.1	28.6	11.6	265.0	0.0	85.72	10
:	:	:	:	:	:	:	:
31/12/2023 22.00	25.7	26.2	6.8	16.0	0.0	98.28	6
31/12/2023 23.00	25.8	26.6	9.5	22.0	0.0	97.29	6

Tabel 1. di atas dapat diketahui unsur-unsur cuaca yang digunakan pada prediksi *visibility*, antara

lain suhu titik embun (°C), suhu bola kering (°C), curah hujan (mm/jam), kelembapan (%), arah (°) dan kecepatan angin (kt) yang digunakan sebagai fitur, serta *visibility* (km) sebagai target yang akan diprediksi.

2.2 Preprocessing Data

Sebelum melakukan pembentukan model, terlebih dahulu melakukan *preprocessing* data seperti data *cleaning*, statistik deskriptif, korelasi data dan normalisasi data. *Preprocessing* dilakukan dengan menyiapkan data dengan membersihkan data dari *noise* atau mengubah struktur data. Hal ini dilakukan karena sering ditemui data mentah yang tidak lengkap atau formatnya tidak sesuai dengan kebutuhan penelitian [17].

2.1.1 Interpolasi Linier

Missing value merupakan kekosongan dari beberapa data yang ada pada data penelitian [18]. Terdapat salah satu metode untuk menangani data kosong pada data dengan jenis deret waktu yakni interpolasi jenis *linear* [19]. *Interpolasi linear* adalah metode yang bertujuan untuk memperoleh nilai antara dua rentan nilai yang didapatkan dari fungsi persamaan [20]. Dengan kata lain, *interpolasi linear* menggunakan garis lurus untuk menghubungkan dua titik dan memperkirakan nilai yang berada di antara keduanya. Berikut adalah rumus *interpolasi linear* pada Persamaan 1.

$$y = y_a + (y_b - y_a) \frac{x - x_a}{x_b - x_a} \quad (1)$$

dimana y adalah data hasil *interpolasi*, x data yang berada antara x_a dan x_b , y_a dan x_a merupakan sampel data $t - 1$, y_b dan x_b merupakan sampel data $t + 1$.

2.1.2 Normalisasi Data

Metode *MinMaxScaler* untuk normalisasi data melakukan transformasi linier pada data asli dengan *outputnya* [21]. Tujuan dari dilakukannya normalisasi adalah untuk memetakan nilai setiap variabel dalam interval yang sama, yaitu 0 hingga 1, sehingga pada saat menghitung nilai kemiripan, setiap variabel yang diuji memberikan pengaruh yang sama [22]. Berikut ini perhitungan normalisasi ditunjukkan pada Persamaan 2 [23].

$$\hat{x} = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \quad (2)$$

dimana \hat{x} data hasil normalisasi, x_i data per variabel, x_{\min} nilai minimal per variabel, dan x_{\max} nilai maksimal per variabel.

2.1.3 Cross-Correlation Function (CCF)

Cross-Correlation Function (CCF) merupakan salah satu metode statistik yang dapat digunakan untuk korelasi atau menentukan kesamaan antar variabel fitur dengan target sebagai fungsi dari pergeseran waktu yang diterapkan pada target [7]. CCF membantu dalam mengidentifikasi antara variabel fitur dan target berkorelasi serta sejauh mana korelasinya pada berbagai pergeseran waktu X_t dan Y_t pada lag (k). Penggunaan CCF dalam analisis data *time series* hanya digunakan untuk mengestimasi korelasi antara variabel-variabel dalam analisis data *time series* [8].

Berikut rumus CCF ditunjukkan pada Persamaan 3 [24].

$$CCF(k) = \frac{\sum_{t=1}^{N-k} (X_t - \bar{X})(Y_{t+k} - \bar{Y})}{\sqrt{(\sum_{t=1}^{N-k} (X_t - \bar{X})^2)(\sum_{t=1}^{N-k} (Y_{t+k} - \bar{Y})^2)}} \quad (3)$$

dimana $CCF(k)$ merupakan hasil *cross-correlation function* pada lag k , X_t dan Y_{t+k} merupakan nilai dari X dan Y pada waktu t dan $t + k$ masing-masing, \bar{X} dan \bar{Y} merupakan rata-rata dari X dan Y , N panjang dari deret waktu, dan k pergeseran waktu yang diterapkan pada deret waktu Y . Ketentuan *threshold* sangat penting untuk mempertimbangkan nilai r dalam CCF, dimana korelasi minimum dianggap signifikan jika $|r| > 0.5$.

2.3 Pembagian Data

Setelah proses *preprocessing* data selesai dilakukan, langkah selanjutnya adalah proses data split atau pembagian data. Proses ini dilakukan dengan membagi dataset menjadi dua bagian, yaitu data latih dan data uji. Data latih digunakan untuk melatih model agar mampu mengenali pola dalam dataset, sedangkan data uji digunakan untuk mengevaluasi performa model pada data yang belum pernah dilihat sebelumnya. Data tersebut dibagi dengan rasio 90% untuk data latih dan 10% untuk data uji.

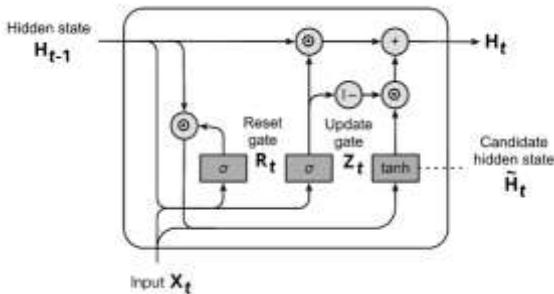
Tabel 2. Pembagian Data Latih dan Data Uji

Presentase	Data Latih	Data Uji
90%:10%	7884	876

Tabel 2. Menunjukkan pembagian dataset menjadi data latih dan uji dengan data latih (90%) sebanyak 7884 data digunakan untuk melatih model prediksi *visibility*. Data ini mencakup mayoritas dataset yang bertujuan untuk membantu model memahami pola dan hubungan antara variable input dengan target yang akan diprediksi. Data uji (10%) sebanyak 876 data digunakan untuk menguji kinerja model yang telah dilatih. Data ini digunakan untuk mengevaluasi kemampuan model dalam membuat prediksi pada data baru selama proses pelatihan.

2.4 Gated Recurrent Unit (GRU)

Gated Recurrent Unit atau biasa disingkat GRU adalah varian lain dari arsitektur *Recurrent Neural Network* (RNN) yang mengatasi masalah memori jangka pendek dan menawarkan struktur yang lebih sederhana dibandingkan dengan LSTM [25]. GRU menggabungkan *input gate* dan *forget gate* LSTM menjadi *update gate* tunggal, menghasilkan arsitektur yang lebih ramping. GRU tidak seperti LSTM karena tidak menyertakan *cell state* yang terpisah [13]. Unit GRU terdiri dari tiga komponen utama yaitu *update gate*, *reset gate*, dan *hidden gate* terbaru. Berikut Gambar 1. menunjukkan arsitektur GRU [16].



Gambar 1. Arsitektur GRU [4]

Update gate (z_t) menentukan berapa banyak informasi masa lalu harus dipertahankan dan digabungkan dengan input saat ini pada langkah waktu tertentu [26]. *Gate* ini dihitung berdasarkan pada penggabungan *hidden state* sebelumnya (h_{t-1}) dengan input saat ini (x_t) diikuti dengan transformasi linear dimana (W) adalah bobot metriks dan (b) adalah bias pada setiap *gate* serta (σ) sebagai fungsi aktivasi sigmoid. Berikut perhitungan *update gate* diuraikan pada Persamaan 4.

$$z_t = \sigma(W_z[h_{t-1}, x_t] + b_z) \quad (4)$$

Reset gate (r_t) menentukan berapa banyak informasi masa lalu harus dilupakan. Hal ini dihitung dengan cara yang sama dengan cara yang sama dengan *update gate* menggunakan penggabungan dari *hidden state* sebelumnya dan *input* saat ini. Berikut perhitungan *reset gate* diuraikan pada Persamaan 5.

$$r_t = \sigma(W_r[h_{t-1}, x_t] + b_r) \quad (5)$$

Kandidat *hidden state* (\tilde{h}_t) menentukan memori baru yang akan disimpan dalam *hidden state* saat ini. Perhitungan *reset gate* dan penggabungan dari transformasi *hidden state* sebelumnya (h_{t-1}) bersamaan dengan input saat ini. Hasilnya adalah melewati fungsi *tanh* untuk menghasilkan aktivasi kandidat. Berikut perhitungan kandidat *hidden state* diuraikan pada Persamaan 6.

$$\tilde{h}_t = \tanh(W_h[r_t \cdot h_{t-1}, x_t]) \quad (6)$$

Proses terakhir yaitu jaringan akan menghitung *output* akhir dari unit saat ini dan meneruskannya ke *time step* berikutnya sebagai *hidden state* sebelumnya. *Hidden state* saat ini (h_t) ditentukan oleh kombinasi dari \tilde{h}_t dan aktivasi kandidat pada Gambar 1. *Update gate* menentukan keseimbangan antara \tilde{h}_t dan aktivasi kandidat. Berikut perhitungan *hidden state* diuraikan pada Persamaan 7.

$$h_t = z_t \cdot h_{t-1} + (1 - z_t) \cdot \tilde{h}_t \quad (7)$$

$$\hat{y}_t = W_o \cdot h_t + b_o$$

Setiap model dilatih selama 50 *epoch* dengan menggunakan fungsi *loss mean squared error* dan *optimizer Adam*. Pengaturan *hyperparameter* terlihat seperti pada Tabel 3.

Tabel 3. Uji coba Parameter

Hyperparameter	Nilai
Bath size	16, 32, 64 [27]
Dropout	0.1, 0.2, 0.3 [28]
Learning rate	0.01, 0.001 [28]

2.5 Denormalisasi

Denormalisasi merupakan suatu proses pengembalian data setelah dinormalisasi menuju data aslinya [29]. Berikut perhitungan denormalisasi ditunjukkan pada Persamaan 8 [30].

$$\hat{y}_i = \hat{y}_i(y_{maks} - y_{min}) + y_{min} \quad (8)$$

dimana \hat{y}_i merupakan hasil prediksi setelah di denormalisasi, \hat{y}_i adalah hasil prediksi sebelum di denormalisasi, y_{maks} adalah nilai maksimal data target, dan y_{min} adalah nilai minimal data target.

2.6 Mean Absolute Percentage Error (MAPE)

Mean Absolute Percentage Error atau biasa disingkat MAPE adalah salah satu ukuran yang paling umum digunakan untuk mengevaluasi akurasi model peramalan [31]. MAPE dihitung dengan mengambil rata-rata dari persentase kesalahan absolut antara nilai aktual dan nilai prakiraan atau prediksi [32]. Hasil MAPE menunjukkan seberapa jauh rata-rata nilai prakiraan menyimpang dari nilai aktual dalam persentase. Berikut perhitungan MAPE ditunjukkan pada Persamaan 9 [33].

$$MAPE = \frac{\sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right|}{n} \times 100\% \quad (9)$$

dimana \hat{y}_i adalah hasil prediksi, y_i adalah data aktual, dan n adalah jumlah data. Semakin rendah nilai MAPE yang diperoleh dari hasil prediksi maka semakin optimal prediksi yang dihasilkan. Berikut kriteria nilai MAPE pada Tabel 4 [36].

Tabel 4. Kriteria Nilai MAPE

Kemampuan Prediksi	Nilai (%)
Sangat Baik	< 10
Baik	10-20
Cukup	20-50
Tidak Akurat	> 50

Tabel 4. diatas menjelaskan klasifikasi kemampuan prediksi suatu model berdasarkan nilai presentase error yang dihasilkan. Klasifikasi ini dibagi menjadi 4 kategori, yaitu sangat baik, baik, cukup, dan tidak akurat yang menunjukkan seberapa akurat model dalam memprediksi nilai yang diinginkan. Dengan klasifikasi tersebut, evaluasi model dapat dilakukan untuk melihat seberapa baik kinerja prediksi model berdasarkan besarnya *error*.

3. HASIL DAN PEMBAHASAN

Pada penelitian ini menggunakan bahasa pemrograman *Python*. *Python* digunakan karena menyediakan banyak pustaka untuk melakukan pemrosesan *Deep Learning* seperti GRU. Data cuaca setiap jam dari mulai awal tanggal 01 Januari- akhir 31

Desember 2023 seperti pada Tabel 1. merupakan

Suhu (°C)		Angin		Curah Hujan	Kelembaban	Visibility
Titik embun	Bola kering	Kecepatan (kt)	Arah (°)	(mm/jam)	(%)	(km)
0.44	0.31	0.17	0.73	0.0	0.82	1.00
0.45	0.38	0.34	0.67	0.0	0.74	1.00
0.47	0.50	0.45	0.73	0.0	0.66	1.00
:	:	:	:	:	:	:
0.57	0.33	0.26	0.04	0.0	0.96	0.56
0.58	0.36	0.37	0.06	0.0	0.94	0.56

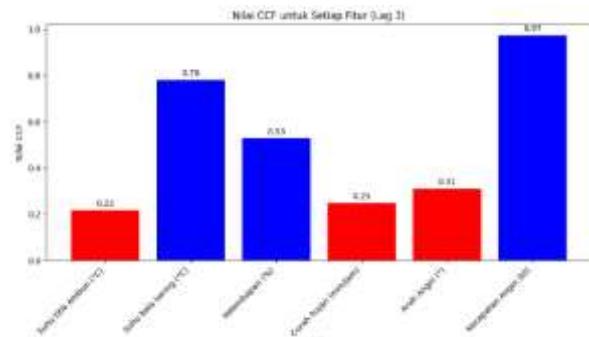
parameter-parameter digunakan sebagai prediksi.

Sebelum masuk pada tahap pelatihan model dilakukan *preprocessing* data terlebih dahulu. Pada parameter arah dan kecepatan angin terdapat data yang hilang, oleh karena itu, langkah pertama untuk pengisian data yang hilang dapat menggunakan interpolasi linier menggunakan Persamaan 1.

Kemudian untuk langkah kedua setiap parameter dinormalisasi dengan *MinMaxScaler* menggunakan Persamaan 2. Analisis GRU memerlukan normalisasi untuk mengurangi nilai kesalahan. Oleh karena itu, normalisasi dilakukan karena jaringan GRU menggunakan fungsi sigmoid dengan *range* [0,1]. Berikut adalah hasil normalisasi untuk setiap parameter ditunjukkan pada Tabel 5.

Tabel 5. Normalisasi Data Cuaca

Pada langkah ketiga yaitu perhitungan CCF untuk menentukan korelasi antar fitur-fitur yang akan digunakan pada model prediksi terhadap target yaitu *visibility* menggunakan Persamaan 3. Setelah dilakukan perhitungan sampai lag $k = 3$ didapatkan hasil pada Gambar 2.



Gambar 2. Nilai Korelasi Setiap Fitur

Grafik pada Gambar 2. memiliki nilai hasil korelasi untuk setiap fitur terhadap *visibility* yaitu suhu titik embun sebesar 0.22, suhu bola kering sebesar 0.78, kecepatan angin sebesar 0.97, arah angin sebesar 0.31, curah hujan sebesar 0.25, dan kelembaban sebesar 0.53. Oleh karena itu, fitur yang paling berkorelasi terhadap *visibility* dengan nilai lebih dari 0.5 adalah kecepatan angin, suhu bola kering, dan kelembaban yang akan digunakan sebagai parameter prediksi.

Kemudian tahapan dari prediksi adalah membagi data menjadi dua bagian. Pembagian data

yang digunakan adalah 90% untuk data untuk data latih dan 10% untuk data uji. Data pertama hasil dari perhitungan sebelumnya akan digunakan untuk pembentukan model prediksi. Kemudian data kedua akan digunakan dalam pengujian.

Setelah itu, membangun 18 pelatihan model dengan beberapa parameter yang berbeda untuk mengetahui model yang memberikan performa terbaik untuk peramalan. Pada penelitian ini, pelatihan model dihitung menggunakan Persamaan 4 hingga Persamaan 7 secara berurutan untuk mengetahui seberapa bagus model yang dibuat dan MAPE digunakan sebagai evaluasi model. Parameter - parameter yang berbeda pada pelatihan model dapat dilihat pada Tabel 6.

Tabel 6. Perbandingan Parameter

<i>Batch Size</i>	<i>Learning rate</i>	<i>Dropout</i>	<i>MAPE</i>
16	0.01	0.1	23.76
		0.2	17.43
		0.3	20.80
	0.001	0.1	14.71
		0.2	13.51
		0.3	16.05
32	0.01	0.1	12.75
		0.2	18.71
		0.3	15.56
	0.001	0.1	14.98
		0.2	15.82
		0.3	18.14
64	0.01	0.1	14.95
		0.2	14.15
		0.3	14.81
	0.001	0.1	15.43
		0.2	15.26
		0.3	16.83

Tabel 6. Menunjukkan bahwa dengan melakukan uji coba parameter mampu meningkatkan akurasi model. Pada penggunaan *batch size* sebanyak 16 menghasilkan nilai MAPE terkecil 17.43% dengan *learning rate* 0.01 dan *dropout* 0.2. Sedangkan nilai MAPE terbesar 23.76% dengan *learning rate* 0.01 dan *dropout* 0.1. Pada penggunaan *batch size* sebanyak 32 menghasilkan nilai MAPE terkecil 12.75% dengan *learning rate* 0.01 dan *dropout* 0.1. Sedangkan nilai MAPE terbesar 18.71% dengan *learning rate* 0.01 dan *dropout* 0.2. Pada penggunaan *batch size* sebanyak 64 menghasilkan nilai MAPE terkecil 14.15% dengan *learning rate* 0.01 dan *dropout* 0.1. Sedangkan nilai MAPE terbesar 16.83% dengan *learning rate* 0.001 dan *dropout* 0.3. Oleh karena itu, untuk menghitung hasil prediksi menggunakan model pelatihan *batch size* 32, *learning rate* 0.01 dan *dropout* 0.1 dengan nilai MAPE terendah yaitu 12.75%. Setelah pelatihan selesai file model pelatihan terbaik disimpan dalam format .h5 untuk dapat digunakan sebagai peramalan.

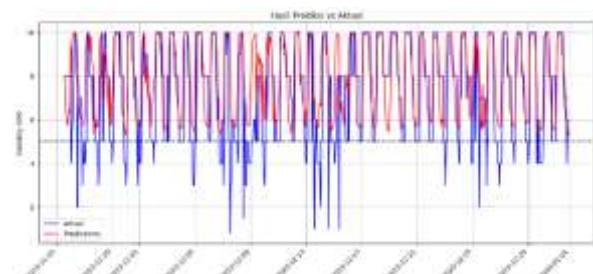
Tahap selanjutnya yaitu pengujian model. Pada tahap ini hasil peramalan didapatkan dari pengujian model terbaik menggunakan data uji yang telah dibagi sebanyak 876 data yaitu 10% dari data keseluruhan. Setelah itu, hasil prediksi didenormalisasi menggunakan Persamaan 8 dengan tujuan mengembalikan nilai prediksi seperti data asli sebelum

dinormalisasi. Berikut hasil peramalan pengujian menggunakan model terbaik setelah didenormalisasi dapat dilihat pada Tabel 7.

Tabel 7. Denormalisasi Data Uji dan Hasil Prediksi

Tanggal (WIB)	Data Uji	Hasil Prediksi	
		Fitur	Tanpa Fitur
25/11/2023 13.00	8.00	7.98	8.47
25/11/2023 13.00	8.00	7.61	8.25
25/11/2023 13.00	8.00	6.69	8.03
:	:	:	:
31/12/2023 22.00	6.00	5.32	6.49
31/12/2023 22.00	6.00	5.49	6.16

Gambar 3 menunjukkan hasil prediksi data uji dari tanggal 25 November hingga 31 Desember 2023. Hasil prediksi menggunakan pola input data dengan fitur kecepatan angin, suhu bola kering, dan kelembapan. Data hasil prediksi memiliki pola yang mirip dengan data aktual, namun di beberapa titik, masih terdapat perbedaan antara data aktual dengan data prediksi. Meskipun masih ada beberapa kesalahan prediksi, namun model secara umum mampu menangkap tren dan pola data target.



Gambar 3. Hasil Prediksi Dengan Fitur Terpilih

4. KESIMPULAN

Hasil perhitungan *Cross Correlation Function* (CCF) untuk menentukan parameter cuaca yang paling tinggi korelasi terhadap *visibility* dengan nilai $|r| > 0.5$ diperoleh nilai korelasi yaitu: suhu titik embun ($^{\circ}\text{C}$) sebesar 0.217, suhu bola kering ($^{\circ}\text{C}$) sebesar 0.780, kelembapan (%) sebesar 0.527, curah hujan (mm/jam) sebesar 0.247, arah angin ($^{\circ}$) sebesar 0.309, dan kecepatan angin (kt) sebesar 0.972. Diperoleh kecepatan angin (kt), suhu bola kering ($^{\circ}\text{C}$), dan kelembapan (%) yang digunakan sebagai parameter pada prediksi *visibility*. Penerapan algoritma *Gate Recurrent Unit* (GRU) dalam memprediksi *visibility* dengan parameter terpilih dengan menggunakan pola data 12 *timestep* untuk setiap parameternya. Model yang didapat dari proses pelatihan selanjutnya digunakan dalam proses pengujian. Pada proses pengujian menghasilkan nilai *error* terkecil dengan menggunakan parameter terbaik 32 *batch size*, 0.1 *dropout*, 0.01 *learning rate* yang menghasilkan nilai MAPE sebesar 12.75%.

5. DAFTAR PUSTAKA

- [1] A. Rizki Ramadhan, D. Deddy Kurniawan, and T. Sutabri, "Penerapan Machine Learning Dalam Pelayanan UPT BMKG Palembang di

- Bandara SMBII Palembang," *J. Sistim Inf. dan Teknol.*, vol. 5, no. 2, pp. 203–207, 2023, doi: 10.60083/jsisfotek.v5i2.255.
- [2] M. Kamal and A. Finawan, "Rancang Bangun Sistem Pendekripsi Suhu Dan Kecepatan Angin Untuk Informasi Data Di Bmkg Lhokseumawe Berbasis Koneksi Wifi," vol. 3, no. 1, pp. 58–63, 2019.
- [3] F. Giantara, N. Yanti, S. Handayani, and Y. Anis, "Pola Pendidikan Keluarga Saat Bencana Kabut Asap di Kota Pekanbaru," *J. Obs. J. Pendidik. Anak Usia Dini*, vol. 4, no. 2, p. 778, 2020, doi: 10.31004/obsesi.v4i2.446.
- [4] M. F. Syam, A. Qashlim, R. Tamin, U. Al, and A. Mandar, "Sistem Monitoring Kecepatan Dan Arah Angin Berbasis Internet of Things (Iot) Sebagai Peringatan Dini Bencana Alam," vol. 10, no. 1, pp. 58–66, 2024, [Online]. Available: <http://ejournal.fikom-unasman.ac.id>
- [5] A. T. Sutanto, E. A. M. A. Setiawan, and E. Purnomo, "Rancang Bangun Portable Automated Weather Observing System (AWOS) Kategori 1 Berbasis Antarmuka Website Dan Android," *J. Instrumentasi dan Teknol. Kebumian*, vol. 1, no. 1, pp. 1–10, 2022.
- [6] H. Ismanto, H. Hartono, and M. A. Marfai, "Spatiotemporal Visibility Characteristics Impacted By Forest And Land Fire Over Airports In Sumatera And Borneo Island, Indonesia," *Quaest. Geogr.*, vol. 38, no. 3, pp. 5–16, 2019, doi: 10.2478/quageo-2019-0024.
- [7] U. Simola, X. Dumusque, and J. Cisewski-Kehe, "Measuring precise radial velocities and cross-correlation function line-profile variations using a Skew Normal density," *Astron. Astrophys.*, vol. 622, 2019, doi: 10.1051/0004-6361/201833895.
- [8] S. V. Kurniasari and M. Prastuti, "Peramalan Nilai Ekspor Non Migas Berdasarkan Nilai Tukar Rupiah terhadap US Dollar di Provinsi Jawa Timur Menggunakan Fungsi Transfer," *J. Sains dan Seni ITS*, vol. 11, no. 1, 2022, doi: 10.12962/j23373520.v11i1.63215.
- [9] A. Shankar, B. C. Sahana, and S. P. Singh, "Prediction of Low-Visibility Events by Integrating the Potential of Persistence and Machine Learning for Aviation Services," *Mausam*, vol. 75, no. 4, pp. 977–992, 2024, doi: 10.54302/mausam.v75i4.6624.
- [10] F. M. Madjid and T. B. Kurniawan, "Prediksi Visibility Menggunakan Lstm Dan Mlp Di Bandara Sultan Mahmud Badaruddin II Palembang," *J. Sist. Inf.*, vol. 14, no. 1, pp. 2497–2507, 2022, [Online]. Available: <http://ejournal.unsri.ac.id/index.php/jsi/index>
- [11] Parung, L. J., S. S., Amelia., Prayogo, and D. N, "Penggunaan Teknologi Blokchain Internet of Things dan Artificial Intelligence untuk Mendukung Kota Cerdas," *J. Tek. Inform.*, p.

- 1131, 2021.
- [12] A. Kurniawan Purnomo, "Konsep Bandara Cerdas (Smart Airport) Dalam Kemajuan Teknologi Penerbangan," *Bul. Yasau*, no. January, pp. 44–49, 2023, [Online]. Available: <https://www.researchgate.net/publication/377839130>
- [13] A. I. Cianiago, W. Kaswidjanti, and J. Juwairiah, "Recurrent Neural Network With Gate Recurrent Unit For Stock Price Prediction," *Telematika*, vol. 18, no. 3, p. 345, 2021, doi: 10.31315/telematika.v18i3.6650.
- [14] K. Sofi, A. S. Sunge, S. R. Riady, and A. Z. Kamalia, "Perbandingan Algoritma Linear Regression , Lstm , Dan Gru Dalam Memprediksi Harga Saham Dengan Model Time Series Stock Price Prediction Using Time Series Model By Comparing Linear Regression , Lstm , And Gru Algorithms," *SEMINASTIKA*, 2021, doi: 10.47002/seminastika.v3i1.275.
- [15] W. Surta, K. T. Basuki, E. S. Negara, and Y. N. Kunang, "Rainfall Prediction in Palembang City Using the GRU and LSTM methods," *J. Data Sci.*, vol. 4, no. 2, 2023, [Online]. Available: <http://eprints.intimal.edu.my/1730/>
- [16] Z. Zainuddin, E. A. P. Akhir, and M. H. Hasan, "Predicting Machine Failure Using Recurrent Neural Network-Gated Recurrent Unit (RNN-GRU) Through Time Series Data," *Bull. Electr. Eng. Informatics*, vol. 10, no. 2, pp. 870–878, 2021, doi: 10.11591/eei.v10i2.2036.
- [17] P. Mishra, A. Biancolillo, J. M. Roger, F. Marini, and D. N. Rutledge, "New data preprocessing trends based on ensemble of multiple preprocessing techniques," *TrAC - Trends Anal. Chem.*, vol. 132, p. 116045, 2020, doi: 10.1016/j.trac.2020.116045.
- [18] A. Ilham, "Hybrid Metode Bootstrap Dan Teknik Imputasi Pada Metode C4-5 Untuk Prediksi Penyakit Ginjal Kronis," *Statistika*, vol. 8, no. 1, pp. 43–51, 2020.
- [19] B. Sencer, Y. Kakinuma, and Y. Yamada, "Linear Interpolation Of Machining Tool-Paths With Robust Vibration Avoidance And Contouring Error Control, Precision Engineering," vol. 66, no. 0141–6359, pp. 269–281, 2020.
- [20] A. Andiojaya and H. Demirhan, "A Bagging Algorithm For The Imputation Of Missing Values In Time Series," *Expert Syst. Appl.*, vol. 129, no. 0957–4174, pp. 10–26, 2019, [Online]. Available: <https://doi.org/10.1016/j.eswa.2019.03.044.%0A> (<https://www.sciencedirect.com/science/article/pii/S0957417419302180>)
- [21] C. Fadlan, A. P. Windarto, and I. S. Damanik, "Penerapan Metode MOORA pada Sistem Pemilihan Bibit Cabai (Kasus: Desa Bandar Siantar Kecamatan Gunung Malela)," *J. Appl. Informatics Comput.*, vol. 3, no. 2, pp. 42–46, 2019, doi: 10.30871/jaic.v3i2.1324.
- [22] L. Friedman and O. V. Komogortsev, "Assessment of the Effectiveness of Seven Biometric Feature Normalization Techniques," *IEEE Trans. Inf. Forensics Secur.*, vol. 14, no. 10, pp. 2528–2536, 2019, doi: 10.1109/TIFS.2019.2904844.
- [23] L. Munkhdalai, T. Munkhdalai, K. H. Park, H. G. Lee, M. Li, and K. H. Ryu, "Mixture of Activation Functions With Extended Min-Max Normalization for Forex Market Prediction," *IEEE Access*, vol. 7, no. 10.1109/ACCESS.2019.2959789, pp. 183680–183691, 2019.
- [24] M. R. Mahmoudi and A. R. Zarei, "Modified Version of the Cross-Correlation Function to Measure Drought Occurrence Time-Delay Correlation," *J. Water Clim. Chang.*, vol. 14, no. 2, pp. 454–476, 2023, doi: 10.2166/wcc.2023.323.
- [25] Zikri Wahyuzi, "Analisis dan Prediksi Konsumsi Listrik Smart Office Berbasis IoT Terhadap Faktor Internal dan Eksternal Menggunakan Deep Learning," *Univ. Islam Indones.*, pp. 1–77, 2024.
- [26] A. Alkesaiberi *et al.*, "Water Level Prediction Model Applying a Long Short-Term," *Energies*, vol. 15, no. 7, pp. 3021–3039, 2022, [Online]. Available: http://repository.radenintan.ac.id/11375/1/PERPUS_PUSAT.pdf %0A <http://business-law.binus.ac.id/2015/10/08/pariwisatasyariah/> %0A <https://www.ptonline.com/articles/how-to-get-better-mfi-results> %0A <https://journal.uir.ac.id/index.php/kiat/article/view/8839> %0A
- [27] A. P. Meriani and A. Rahmatulloh, "Perbandingan Gated Recurrent Unit (GRU) Dan Algoritma Long Short Term Memory (LSTM) Linear Refression Dalam Prediksi Harga Emas Menggunakan Model Time Series," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 1, 2024, doi: 10.23960/jitet.v12i1.3808.
- [28] M. A. Rohman, Suhartono, and T. Chamidy, "Bidirectional GRU dengan Attention Mechanism pada Analisis Sentimen PLN Mobile Bidirectional GRU with Attention Mechanism on Sentiment Analysis of PLN Mobile," *Techno.com*, vol. 22, no. 2, pp. 358–372, 2023.
- [29] S. Kumar Chandar, "Grey Wolf Optimization-Elman Neural Network Model For Stock Price Prediction," *Soft Comput.*, vol. 25, no. 1, pp. 649–658, 2021, doi: 10.1007/s00500-020-05174-2.
- [30] M. Y. Alang, K. Suryowati, and F. Astuti, "Prediksi Indeks Pembangunan Manusia Pada Kabupaten/Kota Di Provinsi Nusa Tenggara Timur Menggunakan Backpropagation Dengan Kombinasi Learning Rate Dan Epoch," *J. Stat.*

- Ind. dan Komputasi*, vol. 9, no. 1, pp. 10–20, 2024, doi: 10.34151/statistika.v9i1.4821.
- [31] N. M. Siregar, N. Wisna, and Asniar, “Qris Dengan Efektivitas Transaksi Digital Mahasiswa Kota Bandung Melalui Regresi Linier Berganda,” *Edunomika*, vol. 08, no. 02, pp. 1–9, 2024.
- [32] I. Amansyah, J. Indra, E. Nurlaelasari, and A. R. Juwita, “Prediksi Penjualan Kendaraan Menggunakan Regresi Linear: Studi Kasus pada Industri Otomotif di Indonesia,” vol. 4, pp. 1199–1216, 2024.
- [33] K. Jepriadi, “Kalibrasi dan Validasi Model Vissim untuk Mikrosimulasi Lalu Lintas pada Ruas Jalan Tol dengan Lajur Khusus Angkutan Umum (LKAU),” *J. Keselam. Transp. Jalan (Indonesian J. Road Safety)*, vol. 9, no. 2, pp. 110–118, 2022, doi: 10.46447/ktj.v9i2.439.