



JOURNAL CERITA:

Creative Education of Research in Information Technology and Artificial Informatics

Vol. 12 No. 1 (2026) 62-70

e-ISSN: 2655 - 2574

Prediksi Intensitas Curah Hujan di Kota Surabaya Menggunakan Principal Component Analysis (PCA) Dan Long Short-Term Memory (LSTM)

Cintya Juanita Elizabeth Kolimon¹, Muhammad Muharrom Al Haromainy², Afina Lina Nurlaili³

^{1,2,3}Program Studi Informatika Fakultas Ilmu Komputer, UPN "Veteran" Jawa Timur

E-mail: 21081010009@student.upnjatim.ac.id^{*1}; muhammad.muhaarrom.if@upnjatim.ac.id²,
afina.lina.if@upnjatim.ac.id³

Kolimon, C.J.E., Al Haromainy, M.M., & Nurlaili, A.L. (2026). Prediksi Intensitas Curah Hujan di Kota Surabaya Menggunakan Principal Component Analysis (PCA) Dan Long Short-Term Memory (LSTM). *Journal Cerita: Creative Education of Research in Information Technology and Artificial Informatics*, 12(1), 62-70

DOI: <https://doi.org/10.33050/k3j3p763>

ABSTRAK

Kota Surabaya sering mengalami curah hujan tinggi yang berpotensi menyebabkan banjir akibat sistem drainase yang kurang optimal. Oleh karena itu, prediksi curah hujan yang akurat sangat penting untuk mitigasi bencana dan perencanaan kota. Penelitian ini bertujuan untuk mengukur sejauh mana penggunaan Principal Component Analysis (PCA) berpengaruh terhadap peningkatan akurasi model Long Short-Term Memory (LSTM) dalam memprediksi intensitas curah hujan di Surabaya. PCA digunakan untuk mengekstraksi fitur dari variabel meteorologi, seperti tekanan udara, suhu, kelembaban, lama penyinaran matahari, dan arah mata angin, guna mengurangi dimensi data serta meningkatkan efisiensi model. Penelitian ini menguji tiga skenario dengan jumlah variabel PCA yang berbeda, yaitu 5, 6, dan 7 variabel. Hasil menunjukkan bahwa skenario dengan 6 variabel PCA menghasilkan performa terbaik dengan MAE 0.012, RMSE 0.033, dan MSE 0.0010. Skenario ini menunjukkan keseimbangan optimal antara jumlah variabel dan akurasi model. Hasil penelitian ini menegaskan bahwa reduksi dimensi menggunakan PCA dapat meningkatkan efisiensi serta akurasi prediksi model LSTM, namun jumlah variabel yang dipilih harus optimal agar tidak terjadi kehilangan informasi yang signifikan. Model yang dikembangkan diharapkan dapat menjadi referensi dan panduan dalam meningkatkan akurasi prediksi curah hujan.

Kata kunci: Curah hujan; Prediksi, Principal Component Analysis (PCA); Long Short-Term Memory (LSTM); Time Series

ABSTRACT

Surabaya frequently experiences high rainfall, which has the potential to cause flooding due to its suboptimal drainage system. Therefore, accurate rainfall prediction is crucial for disaster mitigation and urban planning. This study aims to measure the extent to which Principal Component Analysis (PCA) influences the accuracy improvement of the Long Short-Term Memory (LSTM) model in predicting rainfall intensity in Surabaya. PCA is used to extract features from meteorological variables such as air pressure, temperature, humidity, sunlight duration, and wind direction to reduce data dimensionality and enhance model efficiency. This study evaluates three scenarios with different numbers of PCA variables: 5, 6, and 7 variables. The results indicate that the scenario with 6 PCA variables achieves the best performance, with an MAE of 0.012, RMSE of 0.033, dan MSE of 0.0010. This scenario demonstrates an optimal balance between the number of variables and model accuracy. The findings confirm that dimensionality reduction using PCA can improve the efficiency and accuracy of LSTM-based rainfall prediction models; however, the number of selected variables must be optimal to avoid significant information loss. The developed model is expected to serve as a reference and guideline for improving rainfall prediction accuracy.

Keywords: Rainfall; Prediction; Principal Component Analysis (PCA); Long Short-Term Memory (LSTM); Time Series.

I. PENDAHULUAN

Kota Surabaya merupakan salah satu kota besar di Indonesia yang sering mengalami curah hujan tinggi, terutama pada musim hujan. Curah hujan yang tidak menentu seringkali menyebabkan genangan air dan banjir di berbagai wilayah perkotaan, mengakibatkan gangguan aktivitas ekonomi dan sosial masyarakat. Kondisi ini diperparah dengan sistem drainase yang kurang optimal, sehingga volume air yang melimpah tidak dapat diatasi secara cepat. Selain itu, perubahan iklim global telah berdampak pada pola cuaca di Indonesia, termasuk di Surabaya, yang menjadikan prediksi cuaca semakin sulit dan penuh ketidakpastian. Oleh karena itu, Prediksi curah hujan menjadi fokus yang sangat penting karena memiliki dampak signifikan pada berbagai aspek kehidupan [14].

Prediksi curah hujan yang akurat memiliki peran penting dalam perencanaan kota dan mitigasi risiko bencana, terutama di daerah padat penduduk yang rentan terkena banjir. Informasi yang tepat mengenai curah hujan dapat membantu mengurangi dampak buruk dari banjir dan memberikan panduan bagi pengambilan keputusan yang lebih baik. Dalam upaya meningkatkan akurasi prediksi, berbagai metode canggih mulai diterapkan, termasuk teknologi deep learning. Salah satu metode deep learning yang banyak digunakan untuk analisis data cuaca adalah Long Short-Term Memory (LSTM).

Long Short-Term Memory (LSTM) merupakan pengembangan dari RNN yang memiliki kemampuan bagus dalam menangani interaksi nonlinear yang tinggi dan dapat mengatasi masalah gradien terutama pada kasus prediksi curah hujan [19]. Long Short-Term Memory adalah salah satu algoritma perhitungan data time series dengan data yang digunakan seperti curah hujan, algoritma ini cukup terbukti handal dalam memprediksi beberapa tahun terakhir [1]. Pemilihan menggunakan metode atau algoritma LSTM (Long Short Term Memory) merujuk kepada penelitian terdahulu yang telah melakukan perbandingan atau keakurasian dengan mengunakan metode Long Short-Term Memory (LSTM).

Muhammad Rizki dalam penelitiannya berhasil memprediksi curah hujan di Kota Malang dengan akurasi di atas 85%, menunjukkan bahwa metode Long Short Term Memory (LSTM) mampu memberikan prediksi yang cepat dan akurat. Keberhasilan metode ini dalam menangkap pola data historis curah hujan membuktikan potensinya sebagai acuan dalam pengembangan model prediksi cuaca yang lebih efektif. Efisiensi ini sangat relevan, terutama untuk kota atau daerah dengan karakteristik cuaca yang serupa. Temuan ini menjadi kontribusi penting dalam meningkatkan kualitas prediksi cuaca [13].

Riza Farikhul Firdaus dalam penelitiannya berhasil memprediksi curah hujan di Kota Bandung menggunakan metode Long Short Term Memory (LSTM). Penelitian tersebut

menghasilkan nilai akurasi tertinggi dengan Train Score RMSE sebesar 12,24 dan Test Score RMSE sebesar 8,86. Nilai Test Score sebesar 8,86 menunjukkan bahwa hasil prediksi memiliki tingkat akurasi yang baik. Semakin kecil nilai Test Score, semakin tinggi akurasi prediksi yang dihasilkan [3].

M. Devid Alam Carnegie dalam penelitiannya menganalisis dan memprediksi parameter cuaca dengan membandingkan dua algoritma, yaitu Long Short Term Memory (LSTM) dan Gated Recurrent Unit (GRU). Penelitian tersebut menunjukkan bahwa model LSTM 1 dengan pembagian dataset 7:3 memiliki performa terbaik dalam memprediksi curah hujan. Hasil evaluasi model mencatat nilai RMSE sebesar 16,81, MSE sebesar 282,55, dan MAD sebesar 10,43. Temuan ini menegaskan keunggulan LSTM dalam menghasilkan prediksi yang lebih akurat dibandingkan dengan GRU [2].

Selain itu, LSTM dinilai cocok untuk melakukan prediksi karena dapat melakukan proses dengan bentuk data timeseries dengan melibatkan variabel variabel yang menjadi faktor yang dapat mempengaruhi terjadinya hujan. Namun, faktor yang mempengaruhi curah hujan sangat kompleks, sehingga reduksi dimensi sangat diperlukan agar model bekerja secara optimal [19], salah satunya dengan melibatkan Principal Component Analysis (PCA).

Principal Component Analysis (PCA) merupakan metode feature extraction yang digunakan untuk mereduksi dengan mengompres atribut yang jumlahnya banyak menjadi lebih sedikit [9]. Dalam penelitian yang dilakukan oleh Bashir, PCA digunakan untuk mengidentifikasi variabel-variabel yang paling berpengaruh terhadap kerentanan Akuifer Varamin di Iran. Dari 14 variabel yang dianalisis, PCA berhasil menyeleksi 7 variabel utama [11]. Selain itu, penelitian oleh Tripathi juga memanfaatkan PCA dalam pemilihan parameter kualitas air untuk Sungai Ganga di India, menghasilkan 9 parameter utama dari total 28 parameter yang dianalisis [17].

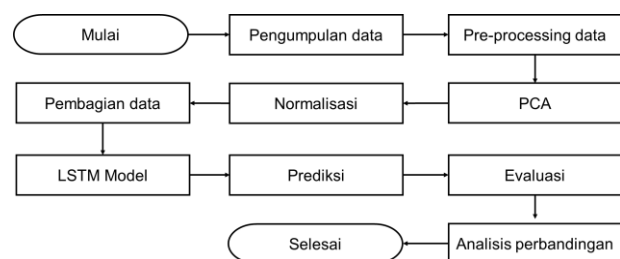
Dari beberapa penelitian serta analisis yang dilakukan, dapat disimpulkan bahwa LSTM dengan kemampuannya sangat cocok untuk memprediksi intensitas curah hujan pada data time series dan PCA dapat mempercepat pelatihan neural network, sehingga penulis memutuskan untuk meneliti terkait “Prediksi Intensitas Curah Hujan di Kota Surabaya

Menggunakan Principal Component Analysis (PCA) dan Long Short-Term Memory (LSTM)”. Data yang akan digunakan untuk penelitian ini berasal dari BMKG (Badan Meteorologi, Klimatologi, dan Geofisika), yang menyediakan variabel metodologi diantaranya tekanan, evaporasi, suhu maksimal, kelembaban rata-rata, dan lama penyinaran matahari untuk memprediksi curah hujan di Kota Surabaya. Variabel tersebut akan dilakukan ekstraksi fitur menggunakan PCA. Hasil PCA akan dijadikan input dalam memprediksi curah hujan harian menggunakan metode LSTM, dan hasil prediksi akan dievaluasi menggunakan MSE, MAE, dan RSME. Harapan penulis, Dengan menggunakan metode ini, prediksi curah hujan di Surabaya dapat dilakukan lebih adaptif terhadap perubahan pola cuaca yang dinamis dan sulit diprediksi.

II. METODE PENELITIAN

A. Tahap Penelitian

Dalam penelitian ini, peneliti menggunakan metode PCA dan LSTM untuk memprediksi curah hujan menggunakan data curah hujan di Kota Surabaya. Data yang digunakan mencakup tekanan, evaporasi, suhu maksimal, kelembaban rata-rata, dan lama penyinaran matahari. Secara garis besar, penelitian ini dibagi menjadi tujuh tahap, yaitu: (1) Pengumpulan data, (2) Pre-processing data, (3) Pembagian Data, (4) LSTM Model, (5) Prediksi, (6) Analisis perbandingan kasus, dan (7) Analisis Perbandingan.



Gambar 1 Tahap Penelitian

Proses penelitian ini diawali dengan pengumpulan data curah hujan dan variabel terkait dari BMKG. Data kemudian melalui tahap pre-processing untuk menangani nilai tidak valid dan menyamakan tipe data. Selanjutnya, dilakukan Principal Component Analysis (PCA) untuk menentukan komponen utama dengan informasi paling signifikan. Data hasil PCA dibagi ke dalam tiga skenario berdasarkan jumlah PCA tertinggi (5, 6, dan 7) sebelum

dinormalisasi dan diolah menggunakan model Long Short-Term Memory (LSTM). Prediksi yang dihasilkan dievaluasi menggunakan metrik MSE, MAE, dan RMSE, kemudian dibandingkan untuk menentukan jumlah PCA optimal dalam meningkatkan akurasi prediksi curah hujan di Kota Surabaya.

B. Pre-Processing

Data curah hujan Kota Surabaya tahun 2018-2024 diperoleh dari BMKG Perak, dicatat harian dalam milimeter (mm). Selain curah hujan, data lain yang dikumpulkan meliputi suhu, kelembapan, tekanan, kecepatan dan arah angin, serta penyinaran matahari. Data ini digunakan untuk meningkatkan akurasi model PCA-LSTM dalam prediksi curah hujan serta dianalisis secara statistik dan grafis.

C. Principal Component Analysis

Principal Component Analysis (PCA) merupakan teknik statistik multivariate yang sangat kuat dalam menganalisis dataset dimensi tinggi. PCA berguna untuk mengekstraksi fitur untuk memperoleh variabel yang paling berpengaruh sehingga mempercepat pelatihan neural network [8]. PCA terdiri dari tiga tahap utama: standarisasi data, pembentukan matriks kovarian, dan perhitungan nilai serta vektor eigen. Standarisasi menyamakan skala variabel, matriks kovarian mengukur hubungan antar-variabel, dan nilai eigen menentukan komponen utama untuk mengurangi dimensi tanpa kehilangan informasi penting.

1. Standarisasi

Tujuan utama PCA adalah untuk meminimalkan kehilangan informasi saat mereduksi dimensi data. Proses PCA dimulai dengan standarisasi data, yang dinyatakan dengan rumus:

$$Z_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j} \quad (1)$$

Standar normalisasi atau standarisasi Z-score. Nilai $Z_{(ij)}$ diperoleh dengan mengurangi nilai asli dengan rata-rata dari variabel tersebut, kemudian membaginya dengan standar deviasi $\sigma_{(j)}$. Normalisasi ini bertujuan untuk menyamakan skala data dengan mean 0 dan standar deviasi 1, sehingga setiap variabel memiliki kontribusi yang seimbang dalam analisis.

2. Matriks Kovarian

Setelah standarisasi, matriks kovarian dihitung untuk menganalisis hubungan antar variabel. Selanjutnya, nilai eigen dan vektor eigen diperoleh untuk menentukan komponen utama yang paling berpengaruh. Komponen utama dengan kontribusi terbesar dipilih sebagai representasi data untuk diproses lebih lanjut.

$$Cov(x_j, x_k) = \frac{\sum_{i=1}^n (x_{ij} - \bar{x}_j)(x_{ik} - \bar{x}_k)}{n - 1} \quad (2)$$

Rumus di atas menghitung kovarian antara dua variabel, yang menunjukkan seberapa kuat hubungan linier antara keduanya. Kovarian diperoleh dengan menjumlahkan hasil kali selisih setiap nilai data terhadap rata-ratanya, kemudian dibagi dengan $n-1$ untuk mendapatkan estimasi yang tidak bias. Jika hasilnya positif, kedua variabel cenderung bergerak searah; jika negatif, bergerak berlawanan arah.

3. Eigenvalues & Eigenvector

$$\det(A - \lambda I) = 0 \quad (3)$$

$$(A - \lambda I)v = 0 \quad (4)$$

Eigenvalues dan eigenvectors diperoleh dari matriks kovarian dengan mencari nilai yang mewakili besarnya variasi data serta arah dari variasi tersebut. Pertama, eigenvalues dihitung untuk mengetahui seberapa besar kontribusi setiap komponen dalam menjelaskan data. Setelah itu, eigenvectors ditentukan untuk mengidentifikasi arah utama dari distribusi data, yang nantinya digunakan dalam reduksi dimensi.

D. Normalisasi

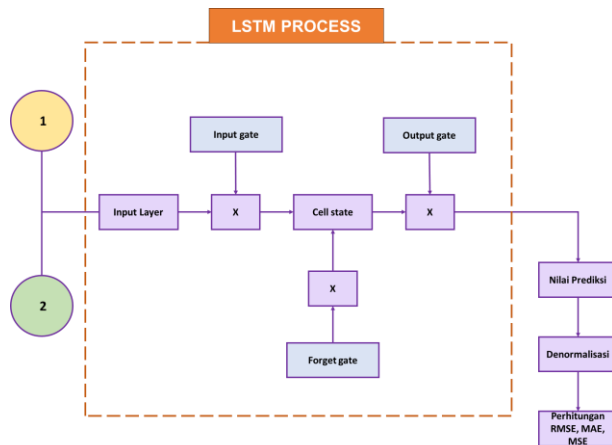
Langkah selanjutnya adalah melakukan normalisasi. Dalam penelitian ini, normalisasi dilakukan menggunakan metode MinMaxScaler. Tujuan dari normalisasi ini adalah untuk mempercepat proses perhitungan selama pelatihan model. Rumus yang digunakan untuk normalisasi dengan MinMaxScaler adalah sebagai berikut:

$$MinMax = \frac{X - \min(x)}{\max(x) - \min(x)} \quad (5)$$

Dalam rumus ini, X adalah nilai data asli, $\min(x)$ adalah nilai minimum dalam dataset, dan $\max(x)$ adalah nilai maksimum dalam dataset. Normalisasi ini membantu menyamakan skala data agar lebih mudah diproses dalam model pembelajaran mesin.

E. LSTM

Long Short-Term Memory (LSTM) merupakan salah satu metode dalam deep learning, yang dirancang untuk meniru cara kerja otak manusia dalam memproses data yang kompleks dan non-linear. Struktur LSTM terdiri dari blok memori yang dilengkapi dengan tiga gerbang utama, yaitu memory gate, forgetting gate, dan output gate. Adapun proses dapat dilihat pada gambar 2.2



Gambar 2 LSTM Model

Input Gate menentukan informasi baru apa saja yang akan ditambahkan ke dalam sel memori LSTM. Persamaan 6 yang merupakan persamaan menghitung input gate (i_t), yang menentukan seberapa banyak informasi baru akan ditambahkan ke cell state. Nilai ini diperoleh dengan menerapkan fungsi aktivasi sigmoid (σ) pada kombinasi linear dari hidden state sebelumnya (h_{t-1}) dan input saat ini (x_t), ditambah bias (b_i). Berlanjut Forget Gate memutuskan informasi apa yang harus dilupakan dari sel memori sebelumnya. Ini memungkinkan model menghapus informasi yang dianggap tidak relevan dari memori yang dapat dilihat pada persamaan 7.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (6)$$

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (7)$$

Output gate (O_t) pada persamaan 8, yang menentukan seberapa banyak informasi dari cell state saat ini yang akan dikeluarkan sebagai hidden state baru. Nilai ini diperoleh dengan menerapkan fungsi aktivasi sigmoid pada kombinasi linear dari hidden state sebelumnya (h_{t-1}) dan input saat ini (x_t), ditambah bias (b_o). Persamaan 9 menghitung hidden state baru, yang merupakan hasil dari output gate dikalikan dengan nilai cell state saat ini yang telah diaktivasi menggunakan fungsi tanh. Hidden state ini akan digunakan dalam langkah waktu berikutnya.

$$O_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (8)$$

$$h_t = O_t \cdot \tanh(C_t) \quad (9)$$

F. Evaluasi

Pada tahap akhir penelitian, evaluasi performa model dilakukan dengan menghitung tiga metrik error utama, yaitu Root Mean Square Error (RMSE), Mean Absolute Error (MAE), dan Mean Squared Error (MSE). Metrik-metrik ini digunakan untuk menilai seberapa akurat prediksi curah hujan yang dihasilkan oleh model, dengan mengukur seberapa jauh hasil prediksi dibandingkan dengan nilai actual.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (10)$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (11)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (12)$$

MSE mengukur rata-rata kuadrat dari perbedaan antara nilai prediksi \hat{y}_i dan nilai aktual y_i . MAE mengukur rata-rata dari nilai absolut perbedaan antara nilai prediksi dan nilai aktual. RMSE adalah akar dari MSE, dan sering digunakan karena memberikan penalti lebih besar pada kesalahan yang lebih besar. RMSE memberikan informasi mengenai deviasi standar dari prediksi, dengan nilai yang lebih rendah menunjukkan performa prediksi yang lebih baik.

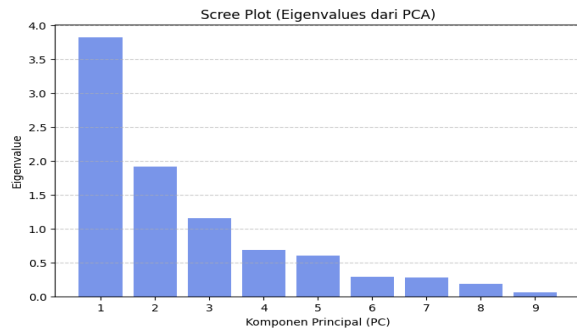
III. HASIL DAN PEMBAHASAN

A. Hasil Preprocessing dan PCA

Data diolah dengan mengubah nilai 8888 atau data yang tidak terukur menjadi 0. Selanjutnya, nilai 0 tersebut digantikan dengan median dari data yang tersedia untuk menjaga konsistensi. Setelah itu, dilakukan perhitungan

```
Eigenvalues (nilai eigen) untuk setiap PC:
PC1: 3.8453
PC2: 1.9110
PC3: 1.1612
PC4: 0.6878
PC5: 0.6073
PC6: 0.2880
PC7: 0.2725
PC8: 0.1708
PC9: 0.0597
```

Gambar 3 Eigenvalue



Gambar 4 Scree plot

Nilai eigenvalues menunjukkan seberapa besar variasi data yang dapat dijelaskan oleh masing-masing Principal Component (PC) dalam analisis PCA. PC1 memiliki nilai eigenvalue tertinggi (3.8453), yang berarti komponen ini

menjelaskan proporsi terbesar dari total variansi dalam dataset. Seiring bertambahnya jumlah komponen, nilai eigenvalues semakin kecil, menunjukkan bahwa PC berikutnya menjelaskan variasi yang lebih sedikit.

```
Eigenvectors (koefisien utama untuk setiap variabel):
suhu_rata suhu_max suhu_min kelembapan tekanan kecepatan_angin \
PC1 -0.388114 -0.381642 -0.087209 0.446358 -0.203766 -0.347933
PC2 0.436261 0.344878 0.616319 0.098558 -0.321641 -0.274511
PC3 -0.001287 -0.086825 0.225222 0.194055 -0.484584 0.480314
PC4 -0.021764 0.234054 -0.207450 -0.082300 0.122404 -0.028315
PC5 0.091805 -0.136330 0.440958 0.147209 0.764506 0.106446
PC6 -0.214069 0.379427 0.026610 0.593929 0.122589 -0.366431
PC7 0.007086 -0.541344 0.287202 0.251191 0.002952 0.200098
PC8 -0.156112 0.434947 -0.120323 0.390336 0.064962 0.621558
PC9 0.761508 -0.168244 -0.476099 0.395902 0.038677 -0.010343

kecepatan_angin_max penyinaran arah_mata_angin
PC1 -0.264485 -0.415909 0.302267
PC2 -0.311120 -0.052080 0.151180
PC3 0.609930 -0.168257 0.195526
PC4 0.063070 0.322529 0.878151
PC5 0.149653 -0.333710 0.165264
PC6 0.398443 0.317618 -0.218660
PC7 -0.232049 0.683640 0.007467
PC8 -0.469174 -0.087922 -0.034663
PC9 0.046215 -0.068073 0.004306
```

Gambar 4 Halaman Input Transaksi

Matriks eigenvectors menunjukkan kontribusi setiap variabel terhadap Principal Component (PC) dalam PCA. Nilai dalam tabel mencerminkan bobot variabel asli terhadap komponen utama, di mana nilai absolut yang lebih besar menunjukkan pengaruh lebih

signifikan. Misalnya, pada PC1, kelembapan dengan nilai 0.4464 memiliki kontribusi dominan. Matriks ini membantu memahami hubungan variabel dan mengidentifikasi faktor utama dalam reduksi dimensi data. Selanjutnya pemilahan sesuai skenario seperti pada gambar 5,6 dan 7.

Variabel yang terpilih berdasarkan PCA dan bobotnya:
 kelembapan: 0.4476
 penyinaran: -0.4172
 suhu_rata: -0.3893
 suhu_max: -0.3830
 kecepatan_angin: -0.3411

Gambar 5 Skenario 1 (5 variabel)

Variabel yang terpilih berdasarkan PCA dan bobotnya:
 kelembapan: 0.4476
 penyinaran: -0.4172
 suhu_rata: -0.3893
 suhu_max: -0.3830
 kecepatan_angin: -0.3411
 arah_mata_angin: 0.3034

Gambar 6 Skenario 2 (6 Variabel)

Variabel yang terpilih berdasarkan PCA dan bobotnya:
 kelembapan: 0.4476
 penyinaran: -0.4172
 suhu_rata: -0.3893
 suhu_max: -0.3830
 kecepatan_angin: -0.3411
 arah_mata_angin: 0.3034
 kecepatan_angin_max: -0.2633

Gambar 7 Skenario 3 (variabel 7)

B. LSTM

1. Skenario 1

Tabel 1 Hasil 5 variabel

5 Variabel	Epoch	Batch Size	MAE	RMSE	MSE
	50	32	0.015	0.035	0.0012
	100	64	0.016	0.036	0.0013
	150	128	0.013	0.035	0.0012

Hasil eksperimen dengan 5 variabel input dan tiga jenis neuron untuk melatih model, tabel 4.3 menunjukkan bahwa epoch 150 dengan batch size 128 dapat menghasilkan kinerja terbaik dengan MSE = 0.0012, RMSE = 0.035, dan MAE = 0.013, mengungguli epoch 50 dan 100. Ini menunjukkan bahwa hasil PCA 5 variabel menghasilkan hasil yang lebih efektif ketika menggunakan model latihan 150 epoch dengan 128 batch size.

2. Skenario 2

Tabel 2 Hasil 6 Variabel

6 Variabel	Epoch	Batch Size	MAE	RMSE	MSE
	50	32	0.014	0.035	0.0012
	100	64	0.013	0.033	0.0011
	150	128	0.012	0.033	0.0010

Hasil eksperimen dengan 6 variabel input menunjukkan bahwa model latihan dengan epoch 150 tetap menjadi jumlah terbaik, dengan MSE = 0.0010, RMSE = 0.033, dan MAE = 0.012, lebih baik dibandingkan skenario pertama. Secara keseluruhan, perubahan jumlah variabel input membantu meningkatkan akurasi model, yang

menunjukkan penurunan MSE, RMSE, dan MAE dibandingkan saat menggunakan 6 variabel.

3. Skenario 3

Tabel 3 Hasil 7 variabel

7 Variabel	Epoch	Batch Size	MAE	RMSE	MSE
	50	32	0.012	0.034	0.0012
	100	64	0.011	0.034	0.0012
	150	128	0.010	0.034	0.0011

Hasil eksperimen dengan 7 variabel input menunjukkan bahwa model latihan 150 epoch adalah jumlah terbaik, dengan MSE = 0.0011, RMSE = 0.034, dan MAE = 0.010. Hasil ini menunjukkan bahwa semakin banyaknya variabel tidak selalu meningkatkan performa model.

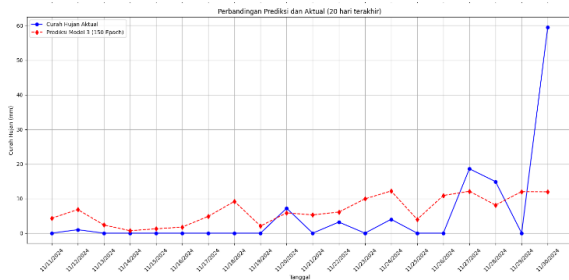
C. Analisis Perbandingan

Setelah menguji model dengan jumlah variabel utama yang berbeda (5, 6, dan 7 variabel), hasil evaluasi menunjukkan variasi dalam performa berdasarkan nilai MAE, MSE, dan RMSE. Perbandingan ini bertujuan untuk menentukan konfigurasi terbaik serta mengevaluasi efektivitas penggunaan PCA dalam memprediksi curah hujan.

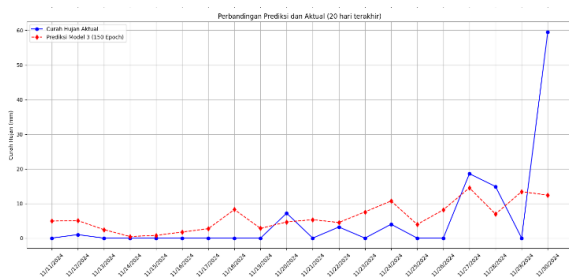
Tabel 4 Hasil perbandingan

PCA	Hasil terbaik dari tiga jenis epoch		
	MAE	RMSE	MSE
5 Variabel	0.013	0.035	0.0012
6 Variabel	0.012	0.033	0.0010
7 Variabel	0.010	0.034	0.0011

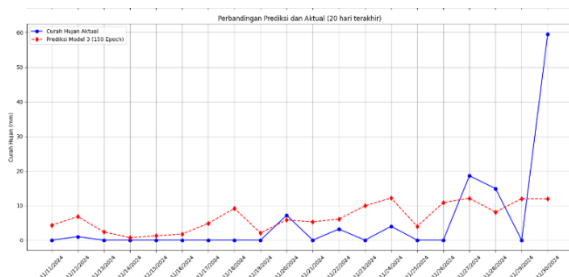
Berdasarkan tabel, model dengan 6 variabel menunjukkan performa terbaik karena memiliki nilai MAE, RMSE, dan MSE terendah dibandingkan dengan 5 dan 7 variabel. Nilai MAE (0.012), RMSE (0.033), dan MSE (0.0010) yang lebih kecil menunjukkan bahwa model dengan 6 variabel menghasilkan prediksi yang lebih akurat dan memiliki tingkat kesalahan yang lebih rendah.



Gambar 8 Grafik Prediksi dan Aktual 14 Hari terakhir (5 Variabel)



Gambar 9 Grafik Prediksi dan Aktual 14 Hari terakhir (6 Variabel)



Gambar 10 Grafik Prediksi dan Aktual 14 Hari terakhir (7 Variabel)

Hal ini juga menunjukkan bahwa PCA berpengaruh terhadap akurasi model, di mana jumlah variabel yang lebih banyak belum tentu menghasilkan performa terbaik. Model dengan 6 variabel menunjukkan hasil evaluasi error yang lebih rendah dan lebih stabil dibandingkan 5 atau 7 variabel, menegaskan bahwa pemilihan jumlah variabel yang optimal lebih penting daripada sekadar menambah variabel utama.

IV. KESIMPULAN

Penelitian ini mengevaluasi pengaruh PCA terhadap LSTM dalam memprediksi curah hujan. Berdasarkan hasil yang diperoleh, penggunaan 6 variabel PCA menghasilkan error terkecil dengan MAE 0.012, RMSE 0.033, dan MSE 0.0010. Hasil ini menunjukkan bahwa reduksi dimensi menggunakan PCA dapat meningkatkan akurasi prediksi dengan LSTM, dibandingkan dengan penggunaan 5 atau 7 variabel. Pemilihan jumlah variabel yang optimal berperan penting dalam meningkatkan performa model.

Model yang digunakan dalam penelitian ini belum sepenuhnya memiliki ketepatan yang tinggi terhadap data asli, yang dapat disebabkan oleh berbagai faktor seperti keterbatasan jumlah data, pola curah hujan yang tidak menentu, atau pemilihan parameter yang belum optimal. Untuk meningkatkan akurasi prediksi, beberapa langkah yang dapat dilakukan antara lain: melakukan hyperparameter tuning menggunakan teknik seperti Grid Search atau Bayesian Optimization agar model lebih optimal, menambah jumlah dan variasi data pelatihan agar pola curah hujan lebih terwakili, serta menerapkan metode optimasi tambahan untuk meningkatkan kinerja model. Selain itu, eksplorasi arsitektur model lain seperti GRU atau Transformer, serta pendekatan ensemble learning, juga dapat menjadi alternatif yang berpotensi meningkatkan ketepatan prediksi.

DAFTAR PUSTAKA

- Badriyah, J., Fariza, A., & Harsono, T. (2022). Prediksi Curah Hujan Menggunakan Long Short Term Memory . *JURNAL MEDIA INFORMATIKA BUDIDARMA*, 1297 - 1303.
- Carnegie, M. D., & Chairani. (2023). Perbandingan Long Short Term Memory (LSTM) dan Gated Recurrent Unit (GRU) Untuk Memprediksi Curah Hujan. *JURNAL MEDIA INFORMATIKA BUDIDARMA*, 1022-1032.
- Firdaus, R. F., & Papatungan, I. V. (2022). Prediksi Curah Hujan di Kota Bandung Menggunakan Metode Long Short Term Memory. *Jurnal Penelitian Inovatif* , 453 - 460.
- Gunadi, I., Khuriati, A., Maulana, M. F., Putranto, A. B., Suseno, J. E., & Hersaputri, M. (2021). Penentuan Curah Hujan Berdasarkan Input Cuaca Menggunakan Metode Logika Fuzzy

- Mamdani. *Jurnal Pengabdian Vokasi*, 155 - 159.
- Lattifia, T. (2022). Model Prediksi Cuaca Menggunakan Metode LSTM. *Jurnal Ilmiah Teknologi dan Komputer* .
- Lubis, N., Siambaton, M. Z., & Aulia, R. (2024). Implementasi Algoritma Deep Learning pada Aplikasi Speech to Text Online dengan Metode Recurrent Neural Network (RNN). *Jurnal Teknik Informatika*.
- Muhammad, T. A., & Irawan, M. I. (2023). Implementasi Long Short-Term Memory (LSTM) untuk Prediksi Intensitas Curah Hujan (Studi Kasus: Kabupaten Malang). *JURNAL SAINS DAN SENI ITS* .
- Musfiroh, Novitasari, D. C., Intan, P. K., & Wisnawa, G. G. (2023). Penerapan Metode Principal Component Analysis (PCA) dan Long Short-Term Memory (LSTM) dalam Memprediksi Curah Hujan Harian. *Building of Informatics, Technology and Science (BITS)* , 1 - 11.
- Pujianto, R., Adiwijaya, & Rohmawati, A. A. (2019). Analisis Ekstraksi Fitur Principle Component Analysis Pada Klasifikasi Microarray Data Menggunakan Classification And Regression Trees. *eProceedings of Engineering*, 2368 - 2379.
- Rachmawati, R. N. (2020). Estimation of Extreme Rainfall Patterns Using Generalized Linear Mixed Model for Spatio-temporal data in West Java, Indonesia. *Procedia Computer Science*, 330-336.
- Rahmani, B., Javadi, S., & Shahdany, S. M. (2019). Evaluation of aquifer vulnerability using PCA technique and various clustering methods. *Geocarto International*, 2117-2140.
- Rizki, A. M., Yuliasuti, G. E., Nurlaili, A. L., & Aditiawan, F. P. (2022). Forecasting the Inflation Rate in Indonesia Using Backpropagation Artificial Neural Network. *IEEE*, 297-300.
- Rizki, M., Basuki, S., & Azhar, Y. (2020). Implementasi Deep Learning Menggunakan Arsitektur Long Short Term Memory (LSTM) Untuk Prediksi Curah Hujan Kota Malang. *Jurnal Repositor*, 331.
- Rohman, A., Hastono, T., & Tyaka, A. P. (2024). Penerapan Metode Fuzzy Mamdani Untuk Prediksi Curah Hujan di Kota Yogyakarta. *Jurnal Publikasi Ilmu Komputer dan Multimedia (JUPIKOM)*, 47-59.
- Sari, A. P., Prasetya, D. A., Yasuno, T., Sihananto, A. N., Haromainy, M. M., & Saputra, W. S. (2022). Forecasting Model of Wind Speed and Direction by Convolutional Neural Network - Deep Convolutional Long Short Term Memory. *IEEE*, 200-205.
- Soekondro, C. A. (2021). PREDIKSI CURAH HUJAN DI KAB.BANDUNG DENGAN ANALISIS TIME SERIES, MENGGUNAKAN MODEL SARIMA (Seasonal Autoregressive Integrated Moving Average). 2865-2875.
- Tripathi, M., & Singal, S. K. (2019). Use of Principal Component Analysis for parameter selection for development of a novel Water Quality Index: A case study of river Ganga India. *Ecological Indicators*, 430 - 436.
- Yusuf, M., Setyanto, A., & Aryasa, K. (2022). Analisis Prediksi Curah Hujan Bulanan Wilayah Kota Sorong Menggunakan Metode Multiple Regression. *Jurnal Sains Komputer & Informatika*, 405 - 417.
- Zhang, J., Chen, X., Khan, A., Zhang, Y.-k., Kuang, X., Liang, X., . . . Nuttall, J. (2021). Daily runoff forecasting by deep recursive neural network. *Journal of Hydrology* .