

Perbandingan Metode Recurrent Neural Network (RNN) dan Long Short-Term Memory (LSTM) untuk Prediksi Curah Hujan

Taufan Hermawan*, Eri Zuliarso

Fakultas Teknologi Informasi dan Industri, Program Studi Magister Teknologi Informasi, Universitas STIKUBANK, Semarang, Indonesia

Email: ^{1,*}taufanhermawan0028@mhs.unisbank.ac.id, ²eri299@edu.unisbank.ac.id

Email Penulis Korespondensi: taufanhermawan0028@mhs.unisbank.ac.id

Submitted: 25/07/2025; Accepted: 30/09/2025; Published: 30/09/2025

Abstrak—Peningkatan intensitas curah hujan ekstrem akibat perubahan iklim telah menyebabkan Kabupaten Batang menjadi daerah rawan bencana hidrometeorologi. Penelitian ini bertujuan untuk membangun model prediksi curah hujan per harian menggunakan Jaringan Saraf Berulang dan Memori Jangka Panjang Pendek berdasarkan data historis BMKG. Model dievaluasi menggunakan metrik *Mean Absolute Error* (MAE) dan *Root Mean Square Error* (RMSE). Hasil menunjukkan bahwa LSTM memiliki akurasi lebih tinggi dibandingkan RNN, dengan nilai sebagai berikut: LSTM = RMSE: 0.1036 | MAE: 0.0730, sedangkan RNN = RMSE: 0.1035 | MAE: 0.0763. LSTM juga lebih stabil dalam memprediksi variabel suhu, arah, dan kecepatan angin. Temuan ini menunjukkan bahwa LSTM lebih efektif untuk data deret waktu cuaca dan dapat digunakan sebagai dasar untuk mengembangkan sistem peringatan dini bencana berbasis data di daerah lokal. Model berbasis LSTM menunjukkan hasil yang baik dalam memprediksi variabel cuaca seperti suhu, arah, dan kecepatan angin dengan hasil yang lebih stabil dibandingkan RNN. Hal ini terlihat dari angka MAE sebesar 0,0730 dan RMSE sebesar 0,1036 yang lebih rendah. Struktur LSTM mampu menangkap perubahan cuaca yang rumit secara efektif.

Kata Kunci: Curah Hujan; Deret Waktu; LSTM; Prediksi; RNN

Abstract—The increase in extreme rainfall intensity due to climate change has caused Batang Regency to become a hydrometeorological disaster-prone area. This research aims to build a day rainfall prediction model using Recurrent Neural Network (RNN) and *Long Short-Term Memory* (LSTM) based on BMKG historical data. The model is evaluated using *Mean Absolute Error* (MAE) and *Root Mean Square Error* (RMSE) metrics. The results show that LSTM has higher accuracy than RNN, with an RMSE: 0.1036 | MAE: 0.0730. Meanwhile, RNN obtained an RMSE: 0.1035 | MAE: 0.0763. LSTM is also more stable in predicting temperature, direction, and wind speed variables. These findings show that LSTM is more effective for weather time series data and can be used as a basis for developing data-based disaster early warning systems in local areas.

Keywords: Rainfall; Time Series; LSTM; Prediction; RNN

1. PENDAHULUAN

Indonesia menghadapi peningkatan signifikan dalam kejadian bencana hidrometeorologi, seperti banjir dan tanah longsor, akibat perubahan iklim global. Data BNPB (2023) mencatat lebih dari 1.500 kejadian selama tahun 2023, sementara BMKG (2023) melaporkan peningkatan intensitas curah hujan ekstrem sebesar 15% dalam dekade terakhir. Kabupaten Batang di Provinsi Jawa Tengah, dengan topografi yang beragam dan curah hujan tinggi (2.000-2.500 mm/tahun), merupakan salah satu daerah yang sangat rentan terhadap dampak bencana tersebut.

Fokus penelitian ini adalah kebutuhan mendesak untuk mengatasi bencana hidrometeorologi yang semakin meningkat, yang merupakan akibat langsung dari perubahan iklim global, yang berdampak langsung pada kehidupan lokal. Suhu permukaan laut di seluruh dunia meningkat secara bertahap, termasuk perairan di sekitar Indonesia, yang merupakan benua maritim. Karena fenomena ini, laju evaporasi (penguapan) meningkat. Akibatnya, lebih banyak uap air terkumpul di atmosfer. Menurut laporan BMKG (2023) yang mencatat peningkatan sebesar 15% dalam sepuluh tahun terakhir, energi besar yang tersimpan di atmosfer kemudian dilepaskan dalam bentuk curah hujan yang sangat kuat.

Kabupaten Batang di Provinsi Jawa Tengah adalah episentrum kerentanan. Karena berbatasan langsung dengan Laut Jawa dan memiliki topografi yang beragam mulai dari pesisir hingga perbukitan, tempat ini sangat sensitif terhadap perubahan siklus hidrologi. Sekarang, curah hujan tinggi yang biasanya antara 2.000 dan 2.500 mm/tahun dapat menjadi lebih berbahaya. Oleh karena itu, lebih dari 1.500 kejadian bencana hidrometeorologi yang dilaporkan oleh BNPB (2023) secara nasional bukanlah angka statistik; itu adalah bukti nyata dari ancaman yang dihadapi masyarakat Batang. Pengembangan model prediksi yang akurat dan dapat diandalkan sangat penting karena ada hubungan langsung antara pemanasan global dan peningkatan ancaman banjir dan tanah longsor di Batang. Masalah utama dalam mitigasi bencana adalah ketidakakuratan prediksi cuaca karena pola data yang kompleks dan non-linear. Metode statistik tradisional seperti ARIMA (*Autoregressive Integrated Moving Average*), yang bekerja dengan asumsi linearitas pada data, seringkali gagal menangkap dinamika cuaca yang bersifat stokastik dan non-stasioner [1]. Kegagalan ini mendorong penerapan metode yang lebih canggih. Pendekatan *deep learning*, khususnya arsitektur seperti RNN dan LSTM, telah terbukti unggul karena kemampuannya mempelajari dependensi jangka panjang dan pola non-linear secara efektif dari data deret waktu yang besar [2], [3]. Beberapa penelitian sebelumnya telah menerapkan pembelajaran mendalam untuk prediksi bencana, baik di Jakarta maupun di Jawa Timur, dengan hasil yang menjanjikan. Namun, tidak banyak penelitian yang secara khusus mengkaji wilayah Kabupaten Batang dengan pendekatan ini.

Penerapan model *deep learning* untuk prediksi data deret waktu telah terbukti berhasil di berbagai bidang. Dalam domain hidrometeorologi, sejumlah penelitian telah menunjukkan keunggulan metode seperti *Recurrent Neural Network* (RNN) dan *Long Short-Term Memory* (LSTM) dibandingkan model statistik tradisional. Sebagai contoh, penelitian H. Xue et.al pada tahun 2024 telah berhasil menerapkan model LSTM untuk memprediksi aliran air (*runoff*) dengan akurasi tinggi [4], Sementara Y. Wang, et.al pada tahun 2025 telah mengembangkan varian LSTM yang disebut *Mass-Conserving LSTM* (MC-LSTM) untuk simulasi aliran sungai yang menegaskan kemampuan model dalam menjaga konsistensi [5]. Meskipun demikian, terdapat celah penelitian (*research gap*) yang signifikan karena mayoritas studi yang ada cenderung berfokus pada prediksi skala makro (seperti aliran sungai di DAS) dan masih sangat terbatas penelitian yang secara spesifik membandingkan langsung kinerja RNN dengan LSTM untuk data cuaca lokal di Indonesia, khususnya untuk wilayah rentan seperti Kabupaten Batang.

Oleh karena itu, penelitian ini menjadi penting karena secara langsung mengatasi celah tersebut. Kebaruan (*novelty*) yang ditawarkan adalah penerapan dan evaluasi komparatif dua model fundamental *deep learning* pada data hiperlokal Kabupaten Batang dengan tujuan akhir yang spesifik, yaitu menyediakan dasar untuk sistem peringatan dini bencana hidrometeorologi. Dengan demikian, penelitian ini tidak hanya sekadar menguji metode, tetapi juga mengkontekstualisasikannya sebagai solusi praktis untuk masalah nyata terkait mitigasi bencana di tingkat lokal.

Meskipun demikian, sebagian besar penelitian tersebut berfokus pada prediksi aliran sungai atau aplikasi di luar meteorologi. Penelitian yang secara spesifik melakukan komparasi kinerja RNN dan LSTM untuk prediksi curah hujan harian sebagai basis mitigasi bencana di wilayah dengan karakteristik topografi unik seperti Kabupaten Batang masih terbatas. Oleh karena itu, penelitian ini bertujuan mengisi celah tersebut dengan mengevaluasi dan membandingkan kedua model menggunakan data cuaca lokal yang spesifik.

Penelitian ini memiliki keterbatasan pada area fokus (Kabupaten Batang) dan parameter cuaca utama (curah hujan), serta hanya menggunakan dua metode: *Jaringan Saraf Berulang* (RNN) dan *Memori Jangka Panjang Jangka Pendek* (LSTM). Rumusan masalah dalam penelitian ini adalah bagaimana membangun model prediksi curah hujan berbasis pembelajaran mendalam yang akurat dan dapat mendukung sistem mitigasi bencana.

Tujuan penelitian ini adalah untuk mengembangkan dan membandingkan model prediksi curah hujan menggunakan RNN dan LSTM dengan data historis dari BMKG. Hasil penelitian ini diharapkan menjadi dasar untuk mengembangkan sistem peringatan dini yang lebih responsif dan mendukung perencanaan kebijakan mitigasi berbasis data. Nilai kebaruan penelitian ini terletak pada penerapan model pembelajaran mendalam pada data lokal Kabupaten Batang dengan fokus khusus pada mitigasi bencana hidrometeorologi.

2. METODOLOGI PENELITIAN

2.1 Analisis Deret Waktu dan ARIMA

Analisis deret waktu adalah sebuah metode statistik yang digunakan untuk mengidentifikasi pola, tren, dan komponen musiman dalam data yang dikumpulkan secara berurutan berdasarkan waktu [6]. Dalam konteks cuaca, analisis ini penting untuk memahami hubungan antara kondisi hidrometeorologi di masa lalu dengan kejadian di masa depan [7]. Data deret waktu cuaca bersifat sangat dinamis dan non-linear, sehingga memerlukan pendekatan yang canggih untuk menghasilkan prediksi yang akurat [8]. Salah satu metode statistik tradisional yang sering digunakan adalah ARIMA (*Autoregressive Integrated Moving Average*). Namun, metode ini memiliki keterbatasan fundamental karena bekerja berdasarkan asumsi linearitas pada data. Akibatnya, model seperti ARIMA seringkali gagal menangkap dinamika dan pola non-linear yang kompleks yang melekat pada data cuaca. Kegagalan ini mendorong penggunaan metode yang lebih maju seperti *deep learning* [9].

2.2 Jaringan Saraf Tiruan Berulang (Recurrent Neural Network - RNN)

Jaringan Saraf Tiruan Berulang telah membuat kemajuan signifikan di bidang-bidang seperti pemrosesan bahasa alami dan pengenalan ucapan [10]. *Deep learning* adalah cabang dari *machine learning* yang efektif untuk menangani data besar dengan karakteristik tidak teratur seperti data deret waktu [11]. Salah satu arsitektur *deep learning* yang dirancang khusus untuk data sekuensial adalah *Recurrent Neural Network* (RNN). RNN telah banyak digunakan dalam berbagai bidang seperti pengenalan tulisan tangan dan pemrosesan bahasa alami karena kemampuannya memproses input secara berurutan. Arsitektur RNN memiliki *feedback loop* atau umpan balik yang memungkinkannya untuk menyimpan dan memanfaatkan informasi dari langkah waktu sebelumnya [12] (memori) untuk memengaruhi output saat ini. Data yang telah diproses awal dan diubah dari masukan dapat disimpan dengan metode ini [13]. Hal ini karena RNN menawarkan kegunaan tertentu ketika karakteristik temporal sinyal memerlukan pertimbangan [14]. Namun, RNN memiliki keterbatasan dalam menangani kondisi jangka panjang, hal ini disebabkan oleh masalah gradien yang menghilang (*vanishing gradient problem*) [15], di mana informasi dari awal hilang seiring waktu. Struktur dasar RNN terdiri dari lapisan input, lapisan tersembunyi (*hidden layer*), dan lapisan output. Proses pada *hidden state* dihitung menggunakan rumus berikut :

$$ht = F(W_{hh}h_{t-1} + W_{xh}x_t + b_h) \quad (1)$$

Dan output dari jaringan pada waktu t dihitung sebagai berikut :

$$y_t = \sigma(W_{hy}h_t + b_y) \quad (2)$$

Dalam model jaringan RNN, keadaan tersembunyi pada waktu t (h_t) berfungsi menyimpan informasi dari langkah sebelumnya sehingga jaringan dapat memahami urutan data. Nilai ini diperbarui berdasarkan input pada waktu t (x_t) melalui fungsi aktivasi non-linear F , seperti tanh atau ReLU, yang membantu menangkap pola kompleks. Output jaringan pada waktu t (y_t) dihasilkan melalui fungsi aktivasi σ , misalnya softmax atau sigmoid, yang menyesuaikan bentuk keluaran sesuai kebutuhan pemodelan. Proses perhitungan ini dipengaruhi oleh matriks bobot W_{hh} , W_{xh} , dan W_{hy} yang menghubungkan berbagai lapisan, serta bias b_h dan b_y yang memastikan fleksibilitas dalam pembelajaran.

2.3 Long Short-Term Memory (LSTM)

Untuk mengatasi kelemahan pada RNN, dikembangkanlah arsitektur *Long Short-Term Memory* (LSTM). LSTM adalah versi canggih dari RNN yang secara eksplisit dirancang untuk menghindari masalah dependensi jangka panjang. Keunggulan utama LSTM terletak pada unit memorinya yang disebut *cell*, yang diatur oleh tiga mekanisme gerbang (*gate*), yaitu *forget gate*, *input gate*, dan *output gate* [16], [17]. Mekanisme gerbang ini memungkinkan LSTM untuk menyimpan, menghapus, atau meneruskan informasi secara selektif, sehingga mampu menangkap pola temporal dan hubungan jangka panjang dalam data deret waktu secara efektif [18], [19], [20]. Berikut adalah persamaan matematis yang mengatur operasi di dalam sel LSTM :

Forget Gate (f_t) : Menentukan informasi apa yang akan dibuang dari *cell state* sebelumnya.

$$f_t = \sigma(W_{if}x_t + b_{if} + W_{fh}h_{t-1} + b_{fh}) \quad (3)$$

Input Gate (i_t): Menentukan informasi baru apa yang akan disimpan di *cell state*.

$$i_t = \sigma(W_{ix}x_t + b_{ix} + W_{ih}h_{t-1} + b_{ih}) \quad (4)$$

Cell Candidate (g_t): Membuat kandidat nilai baru yang akan ditambahkan ke *cell state*.

$$g_t = \tanh(W_{ig}x_t + b_{ig} + W_{hg}h_{t-1} + b_{hg}) \quad (5)$$

Cell State Update (c_t): Memperbarui *cell state* lama (c_{t-1}) menjadi *cell state* baru (c_t).

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \quad (6)$$

Output Gate (o_t): Menentukan output berdasarkan *cell state* yang telah diperbarui.

$$o_t = \sigma(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho}) \quad (7)$$

Hidden State Update (h_t): Menghasilkan *hidden state* baru.

$$h_t = o_t \odot \tanh(c_t) \quad (8)$$

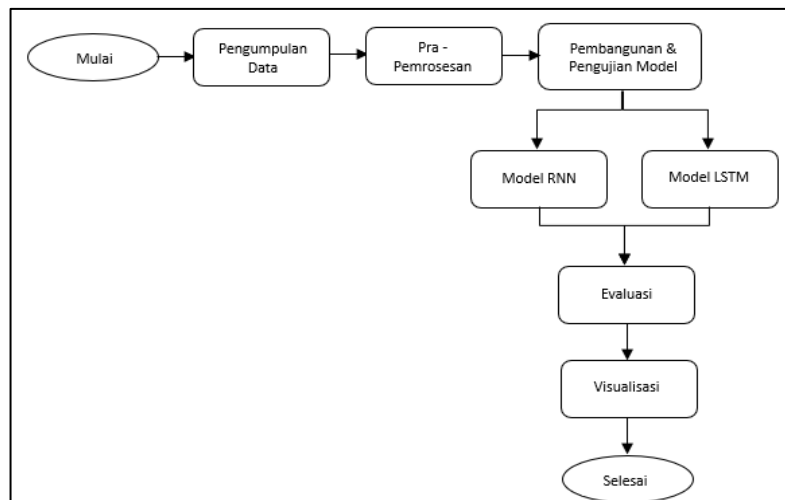
Berkat arsitekturnya yang kompleks ini, LSTM terbukti mampu menangani non-linearitas yang tidak dapat ditangani oleh model linier tradisional, menjadikannya pilihan yang sangat baik untuk data deret waktu yang memiliki dependensi jangka panjang.

2.4 Tahapan Penelitian

Dalam penelitian ini, data adalah komponen utama yang menjadi dasar pengembangan model prediksi cuaca berbasis pembelajaran mendalam. Oleh karena itu, pemilihan dan pemrosesan data dilakukan dengan hati-hati agar sesuai dengan kebutuhan analisis deret waktu yang kompleks dan dinamis. Data yang digunakan diperoleh dari Badan Meteorologi, Klimatologi, dan Geofisika (BMKG), yang menyediakan data cuaca dengan resolusi waktu yang cukup tinggi.

Secara spesifik, data yang digunakan dalam penelitian ini adalah data pengamatan cuaca setiap 10 menit selama 4 tahun terakhir (2021-2024), yang mencakup berbagai parameter seperti curah hujan, suhu udara, arah angin, dan kecepatan angin. Resolusi tinggi dari data ini memungkinkan para peneliti untuk menangkap fluktuasi cuaca pada skala waktu yang lebih rinci dan akurat. Ini penting mengingat karakteristik cuaca yang sangat dipengaruhi oleh dinamika jangka pendek.

Data mentah yang diperoleh kemudian melalui beberapa tahap pra-pemrosesan, seperti agregasi data ke dalam unit waktu per hari, imputasi nilai yang hilang, dan normalisasi skala data untuk memfasilitasi proses pelatihan model. Tahapan-tahapan ini dilakukan agar struktur data yang dihasilkan sesuai dengan kebutuhan input model deep learning, terutama *Recurrent Neural Network* (RNN) dan *Long Short-Term Memory* (LSTM). Dengan pendekatan ini, model yang dibangun diharapkan mampu menangkap pola cuaca secara lebih representatif dan menghasilkan prediksi yang lebih akurat. Tahapan penelitian ini ditunjukkan pada Gambar 2.



Gambar 2. Diagram Alur Penelitian

Gambar 2 menunjukkan tahapan sistematis yang dilakukan dalam proses membangun model prediksi cuaca harian menggunakan pendekatan pembelajaran mendalam berbasis *Jaringan Saraf Berulang* (RNN) dan *Memori Jangka Pendek Panjang* (LSTM). Proses dimulai dari tahap Mulai, yang menandai awal dari seluruh rangkaian pemrosesan data hingga evaluasi model.

2.4.1 Pengumpulan Data

Langkah pertama adalah mengumpulkan data, yaitu proses pemuatan data cuaca harian AWS Kandeman Kabupaten Batang yang mencakup beberapa parameter seperti Data Curah Hujan, Angin, Suhu dan Kelembaban, dengan rentang waktu data selama 4 tahun terakhir (tahun 2021 – 2024), dalam hal ini data historis BMKG.

2.4.2 Pra Pemrosesan Data dan Pemilihan Fitur

Pra pemrosesan data bertujuan untuk mengubah data mentah sehingga dapat meningkatkan kualitas data yang akan di analisis. Setelah data dikumpulkan, dilakukan Imputasi dan Normalisasi, yaitu mengisi nilai yang hilang (imputasi) dan normalisasi agar semua data berada pada skala yang seragam, sehingga model dapat lebih efisien. Selanjutnya, data akan dipisahkan menjadi dua bagian, yaitu data pelatihan (training set) untuk melatih model dan data pengujian (testing set) untuk mengukur keakuratan model yang telah dilatih.

Tahap pra-pemrosesan data adalah langkah krusial untuk memastikan kualitas dan kesiapan data sebelum dimasukkan ke dalam model *deep learning*. Pada tahap ini, dilakukan justifikasi pemilihan fitur (parameter cuaca), penanganan data yang hilang (imputasi), dan penyeragaman skala data (normalisasi).

2.4.2.1 Justifikasi Pemilihan Parameter Cuaca

Pemilihan parameter atau fitur input dilakukan berdasarkan relevansi dan pengaruhnya terhadap variabel target, yaitu curah hujan. Parameter yang digunakan dalam penelitian ini curah hujan (historis), suhu udara, kelembaban, serta arah dan kecepatan angin merupakan variabel-variabel meteorologi fundamental yang telah terbukti memiliki pengaruh signifikan dalam pemodelan prediksi curah hujan pada berbagai studi [21]., dengan justifikasi sebagai berikut :

- Curah Hujan : Dipilih sebagai variabel target utama (*target variable*) yang akan diprediksi oleh model.
- Suhu Udara : Merupakan faktor kunci dalam siklus hidrologi. Suhu yang lebih tinggi dapat meningkatkan laju evaporasi (penguapan air) dan kapasitas atmosfer untuk menahan uap air, yang merupakan prekursor utama terjadinya hujan.
- Kelembaban : Mengindikasikan jumlah uap air di atmosfer. Tingkat kelembaban yang tinggi adalah syarat penting untuk pembentukan awan dan presipitasi (hujan). Oleh karena itu, kelembaban adalah prediktor yang sangat kuat untuk curah hujan.
- Arah dan Kecepatan Angin : Berperan penting dalam transpor massa udara lembab. Perubahan pada pola arah dan kecepatan angin dapat menandakan kedatangan sistem cuaca (misalnya, front tekanan rendah) yang membawa awan hujan ke suatu wilayah.

2.4.2.2 Metode Imputasi Data Hilang

Data cuaca yang diperoleh dari sensor seringkali mengalami masalah data hilang (*missing values*) akibat kegagalan teknis atau gangguan transmisi, sebuah tantangan umum dalam analisis data meteorologi. Untuk menangani hal ini, penelitian ini menggunakan metode imputasi dengan interpolasi linear. Metode ini dipilih karena data cuaca merupakan data deret waktu di mana nilai pada satu titik waktu umumnya memiliki korelasi kuat dengan titik waktu terdekat sebelumnya dan sesudahnya (autokorelasi tinggi). Interpolasi linear efektif untuk kasus seperti ini karena mengisi nilai yang hilang dengan membuat garis lurus antara dua titik data yang diketahui, sehingga menghasilkan

estimasi yang logis dan menjaga kontinuitas tren data tanpa memasukkan bias yang signifikan, sebuah pendekatan yang umum digunakan dan terbukti efektif untuk data hidrometeorologi.

2.4.2.3 Teknik Normalisasi Data

Setiap parameter cuaca memiliki rentang nilai dan satuan yang berbeda (misalnya, Suhu dalam °C, Curah Hujan dalam mm). Perbedaan skala ini dapat menyebabkan model, khususnya *neural network*, memberikan bobot yang tidak proporsional pada fitur dengan nilai numerik yang lebih besar, sehingga memperlambat konvergensi dan menurunkan performa model [22]. Untuk mengatasi masalah ini, digunakan teknik Normalisasi Min-Max (*Min-Max Scaling*), sebuah metode penskalaan fitur yang umum digunakan dalam *deep learning* [23]. Teknik ini mentransformasikan semua nilai fitur ke dalam rentang [0, 1]. Normalisasi Min-Max dipilih karena sangat efektif untuk algoritma *neural network*, terutama yang menggunakan fungsi aktivasi seperti *Sigmoid* atau *Tanh* yang sensitif terhadap rentang input dan bekerja optimal pada rentang tersebut. Rumus yang digunakan adalah sebagai berikut :

$$X_{\text{norm}} = \frac{X - X_{\text{min}}}{X_{\text{max}} - X_{\text{min}}} \quad (9)$$

X_{norm} menggambarkan nilai yang sudah dinormalisasi, yang diperoleh dari nilai asli data (X) melalui proses penskalaan menggunakan batas minimum (X_{min}) dan maksimum (X_{max}) dari fitur tersebut. Normalisasi ini memastikan setiap nilai berada dalam rentang yang sama sehingga perbedaan skala antar variabel tidak memengaruhi proses analisis atau pemodelan.

Metode ini memastikan bahwa semua fitur memiliki kontribusi yang setara pada model, mencegah fitur dengan rentang nilai yang besar mendominasi fitur dengan rentang nilai yang kecil. Dengan mengubah skala data, normalisasi Min-Max dapat meningkatkan efisiensi dan performa algoritma pembelajaran mesin. Meskipun demikian, metode ini sensitif terhadap *outlier* atau data pencilan, karena nilai ekstrem dapat memengaruhi secara signifikan nilai minimum dan maksimum, yang pada akhirnya dapat mengompresi sebagian besar data ke dalam rentang yang sangat kecil. Oleh karena itu, penting untuk memeriksa dan menangani *outlier* sebelum menerapkan normalisasi ini.

Dengan menerapkan teknik ini, semua fitur akan memiliki skala yang seragam, sehingga proses pelatihan model menjadi lebih stabil dan konvergen lebih cepat.

2.4.3 Pembagian Data Pelatihan dan Pengujian

Untuk mengevaluasi kinerja model secara objektif, dataset dibagi menjadi dua bagian terpisah : data pelatihan (*training set*) dan data pengujian (*testing set*). Mengingat data yang digunakan adalah data deret waktu (*time series*), pembagian tidak dilakukan secara acak untuk menghindari kebocoran data (*data leakage*), di mana model secara tidak sengaja "melihat" data dari masa depan saat proses pelatihan.

Oleh karena itu, penelitian ini menerapkan pembagian data secara kronologis dengan rasio 80% untuk data pelatihan dan 20% untuk data pengujian.

- Data Pelatihan (80%): Terdiri dari data historis paling awal yang digunakan untuk melatih model dalam mengenali pola-pola cuaca. Data yang digunakan mencakup periode dari 1 Januari 2021 hingga sekitar pertengahan tahun 2024.
- Data Pengujian (20%): Merupakan data yang lebih baru dan sama sekali tidak terlihat oleh model selama pelatihan. Data ini digunakan untuk menguji kemampuan generalisasi model dalam memprediksi data baru. Data ini mencakup periode dari pertengahan tahun 2024 hingga akhir Desember 2024.

Pendekatan ini mensimulasikan skenario dunia nyata di mana model yang telah dilatih dengan data masa lalu digunakan untuk memprediksi kejadian di masa depan.

2.4.4 Tuning Hiperparameter dan Pencegahan Overfitting

Untuk memastikan model RNN dan LSTM dapat memberikan kinerja yang optimal dan tidak mengalami *overfitting*, dilakukan proses tuning hiperparameter serta penerapan teknik regularisasi. *Overfitting* adalah kondisi di mana model bekerja sangat baik pada data pelatihan tetapi gagal menggeneralisasi polanya pada data pengujian yang baru [24].

- Tuning Hiperparameter :

Pemilihan hiperparameter yang tepat sangat memengaruhi kinerja model. Beberapa hiperparameter kunci seperti *learning rate*, jumlah *unit neuron* pada lapisan tersembunyi, *batch size* adalah 32, dan *jumlah epoch* adalah 100, tidak dipilih secara acak. Nilai-nilai ini ditentukan melalui proses eksperimen iteratif. Model dilatih dengan beberapa kombinasi nilai yang berbeda, dan kinerja setiap kombinasi dievaluasi berdasarkan metrik *validation loss*. Kombinasi hiperparameter yang menghasilkan *validation loss* terendah tanpa menunjukkan tanda-tanda *overfitting* kemudian dipilih untuk digunakan dalam model final.

- Teknik Regularisasi :

Untuk mencegah *overfitting* pada model RNN yang rentan terhadap masalah tersebut, dua teknik regularisasi utama diterapkan: Dropout dan Early Stopping. Dropout bekerja dengan menambahkan lapisan yang menonaktifkan sebagian neuron secara acak (misalnya, dengan rate 0.2) selama pelatihan, memaksa model untuk mempelajari fitur yang lebih robust dan meningkatkan kemampuan generalisasinya dengan mengurangi ketergantungan pada neuron tertentu. Sementara itu, Early Stopping memantau metrik *validation loss* di setiap epoch, menghentikan pelatihan secara otomatis jika tidak ada penurunan *validation loss* setelah sejumlah epoch.

berturut-turut (patience), dan menyimpan model dari epoch dengan validation loss terendah. Kombinasi teknik ini efektif mencegah overfitting, seperti yang terlihat pada grafik riwayat pelatihan RNN, sehingga model dapat berhenti tepat waktu sebelum memasuki fase overfitting.

2.4.5 Pembangunan Model

Proses pembangunan model bertujuan untuk menghasilkan model yang mampu memprediksi curah hujan yang lebih responsif dan mendukung perencanaan kebijakan mitigasi berbasis data, dengan menggunakan metode RNN dan LSTM, di mana setiap model akan dilatih secara terpisah dengan arsitektur yang sesuai.

2.4.6 Pelatihan dan Pengujian Data

Setelah pelatihan model selesai, maka data diuji menggunakan data uji untuk mengukur keakuratan data dalam memprediksi curah hujan. Pada proses pelatihan data dengan RNN dan LSTM, Model dilatih dengan 100 epoch dan batch size 32. Setelah model selesai dilatih, model diuji pada data training dan testing untuk mengambil kelas dengan probabilitas tertinggi sebagai prediksi.

2.4.7 Evaluasi

Setelah model dilatih, proses Evaluasi dilakukan, yaitu pengukuran kinerja model menggunakan metrik evaluasi seperti MAE dan RMSE. Hasil evaluasi ini akan digunakan untuk menilai tingkat akurasi setiap model dalam membuat prediksi. Kemudian, hasil prediksi divisualisasikan pada tahap Visualisasi untuk membandingkan pola antara data asli dan hasil prediksi kedua model. Proses ini diakhiri dengan langkah Selesai, yang mengakhiri seluruh rangkaian pengembangan dan analisis model prediksi.

2.5 Metode Penyelesaian

2.5.3 Bencana Hidrometeorologi

Hidrometeorologi adalah cabang ilmu yang mempelajari hubungan antara proses atmosfer (*meteorologi*) dan siklus *hidrologi*. Faktor *hidrometeorologi* umumnya dikenal karena non-linearitasnya yang tinggi dan interaksi yang kompleks dengan atmosfer. Faktor meteorologi yang memengaruhi kondisi hidrologi seperti suhu udara, kelembapan relatif, kecepatan angin, dan radiasi matahari. Pemahaman tentang hidrometeorologi diperlukan untuk mengetahui pola atau siklus curah hujan. Kurangnya pemahaman ini akan memicu bencana hidrometeorologi dalam bentuk fenomena cuaca ekstrem, seperti banjir, tanah longsor, badai, dan kekeringan. Parameter cuaca yang relevan meliputi curah hujan (intensitas dan durasi), suhu udara (pengaruh terhadap penguapan), serta arah dan kecepatan angin (faktor dalam penyebaran awan hujan).

Hujan intensitas tinggi dapat menyebabkan air di tanah meluap dan ini pasti akan mengganggu ekosistem dan keselamatan manusia. Selain itu, intensitas hujan juga mengganggu perambatan gelombang elektromagnetik pada frekuensi tinggi karena menyebabkan atenuasi sinyal. Jika intensitas curah hujan tinggi berlangsung lama, akan mengakibatkan bencana seperti banjir dan tanah longsor. Menurut BMKG (2023), curah hujan ekstrem di atas 100 mm/hari sering memicu banjir dan tanah longsor di Indonesia.

2.5.4 Analisis Deret Waktu

Analisis deret waktu adalah metode untuk mengidentifikasi pola, tren, dan musiman dalam data yang dikumpulkan secara berurutan. Hasil analisis ini akan lebih akurat jika metrik yang saling bergantung ditangani bersama. Dalam konteks cuaca, analisis ini dilakukan untuk memahami hubungan tertunda antara hidrometeorologi sebelumnya dan kejadian infeksi. Data deret waktu umumnya bersifat dinamis, dengan komponen seperti tren jangka panjang, yang memerlukan pendekatan prediksi yang lebih canggih seperti pembelajaran mesin untuk meningkatkan akurasi prediksi. Saat ini, peramalan deret waktu telah berkembang pesat dengan model pembelajaran mesin, menggantikan model statistik tradisional karena akurasi prediksinya yang tinggi. Penerapan analisis deret waktu yang tepat sangat penting dalam mendukung sistem peringatan dini, pengelolaan sumber daya air, dan mitigasi bencana hidrometeorologi.

2.5.5 Pembelajaran Mendalam

Pembelajaran mendalam adalah bagian dari pembelajaran mesin yang lebih fokus pada penyelesaian data besar dengan karakteristik tidak teratur, seperti deret waktu. Kemampuan pembelajaran mendalam ini telah diadopsi dalam metrologi optik untuk mengatasi tantangan pengukuran fisik, seperti analisis tepi data, kebisingan, dan karakterisasi struktur spesimen. Dalam bidang pembelajaran mendalam ini, metode yang digunakan meliputi model hibrida yang didasarkan pada pelatihan awal yang dipandu oleh model fisika. Metode model umumnya terdiri dari lapisan input, lapisan tersembunyi, dan lapisan penerima.

2.5.6 Jaringan Saraf Berulang

Jaringan Saraf Berulang telah membuat kemajuan signifikan di bidang-bidang seperti pemrosesan bahasa alami dan pengenalan ucapan. Selain itu, metode ini telah banyak digunakan dalam bidang pengenalan tulisan sejak tahun 1986 karena memiliki kemampuan untuk menangani input deret waktu. RNN dikembangkan untuk menangani data berurutan dengan memanfaatkan loop umpan balik untuk mempertahankan informasi masa lalu. Data yang telah

diproses awal dan diubah dari masukan dapat disimpan dengan metode ini. Hal ini karena RNN menawarkan kegunaan tertentu ketika karakteristik temporal sinyal memerlukan pertimbangan. Namun, RNN memiliki keterbatasan dalam menangani kondisi jangka panjang, hal ini disebabkan oleh masalah gradien yang menghilang, di mana informasi dari awal hilang seiring waktu.

$$h_t = F(W_{hh}h_{t-1} + W_{xh}x_t + b_h) \quad (10)$$

$$y_t = \sigma(W_{hy}h_t + b_y) \quad (11)$$

Dalam model RNN, keadaan tersembunyi pada waktu t (h_t) berfungsi sebagai memori yang menyimpan informasi dari langkah sebelumnya dan membantu menghasilkan prediksi pada langkah berikutnya. Nilai ini diperbarui melalui fungsi aktivasi non-linear F , seperti tanh atau ReLU, yang memberikan kemampuan pemodelan hubungan kompleks dalam data. Pembaruan h_t dipengaruhi oleh bobot W_{hh} yang menghubungkan h_{t-1} ke h_t , bobot W_{xh} yang menghubungkan input x_t ke h_t , serta bias b_h yang menyesuaikan nilai pada lapisan tersembunyi. Selanjutnya, output jaringan pada waktu t (y_t) dihitung melalui fungsi aktivasi output σ , misalnya softmax untuk klasifikasi atau sigmoid untuk prediksi biner. Proses ini memanfaatkan bobot W_{hy} yang menghubungkan keadaan tersembunyi ke lapisan keluaran dan bias b_y yang menyesuaikan nilai pada output, sehingga jaringan mampu menghasilkan prediksi berdasarkan kombinasi informasi masa lalu dan input saat ini.

2.5.7 Long Short-Term Memory

Long Short-Term Memory adalah versi terbaru dari RNN dalam mengatasi masalah gradien menghilang dengan menggunakan sel memori dan tiga gerbang yaitu gerbang input, gerbang lupa, dan gerbang output. Mekanisme ini memungkinkan LSTM unggul dalam pemodelan deret waktu secara efektif karena mampu mengekstraksi fitur temporal secara efisien dari riwayat perubahan bentuk, posisi, dan ukuran, menjadikannya ideal untuk analisis deret waktu. Metode ini unggul dalam memodelkan data deret waktu dan dapat secara efektif menangkap hubungan dan tren keseluruhan di antara titik data. LSTM sering digunakan dalam pemodelan prediksi deret waktu yang bertujuan untuk memprediksi nilai di masa depan menggunakan data input historis dan saat ini. Kedua model ini disukai dengan data deret waktu yang memiliki ketergantungan jangka panjang dan deteksi pola yang rumit, karena terbukti mampu menangkap non-linearitas yang tidak dapat dilakukan oleh model linier tradisional. Selain itu, ketersediaan kumpulan data besar dan sumber daya komputasi akan mendukung hasil prediksi yang dihasilkan.

3. HASIL DAN PEMBAHASAN

3.1 Hasil Pengujian Metode

Sebagai bagian dari evaluasi kinerja model dalam memprediksi data cuaca per harian, dilakukan visualisasi hasil prediksi menggunakan dua pendekatan berbasis *Jaringan Saraf Berulang*, yaitu RNN standar dan *Long Short-Term Memory* (LSTM). Visualisasi ini bertujuan untuk melihat seberapa baik model dapat mengikuti pola data asli pada empat parameter cuaca utama, yaitu curah hujan, suhu udara, arah angin, dan kecepatan angin. Dengan membandingkan hasil prediksi dengan data aktual, akurasi dan kemampuan model untuk menangkap pola waktu dinamis dari setiap parameter dapat dianalisis.

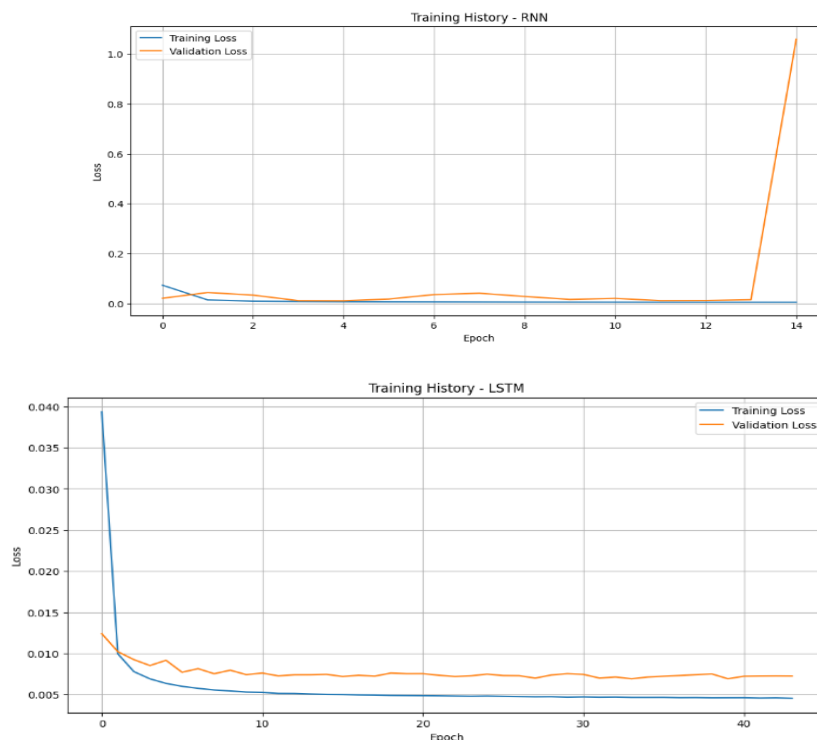
3.1.1 Pembangunan dan Pengujian Model RNN dan LSTM

Data untuk tahun 2021-2024, jumlah baris data yang difilter: 1363

Tabel 1. Data lima baris pertama yang difilter

Tanggal	Curah Hujan	Suhu
2021-01-01	70,864438	27,462908
2021-01-02	80,753581	27,892496
2021-01-03	72,450880	28,228133
2021-01-04	67,234703	29,205613
2021-01-05	50,782595	26,928717

Tabel 1 selama periode lima hari pertama di bulan Januari 2021, data menunjukkan adanya fluktuasi yang menarik pada curah hujan dan suhu. Curah hujan tertinggi terjadi pada tanggal 2 Januari (80,75 mm), kemudian cenderung menurun hingga mencapai titik terendah pada tanggal 5 Januari (50,78 mm). Sementara itu, suhu menunjukkan tren kenaikan dari awal bulan hingga puncaknya di tanggal 4 Januari (29,21 derajat), sebelum akhirnya menurun drastis ke suhu terendah di tanggal 5 Januari (26,93 derajat), hari yang sama dengan curah hujan terendah.



Gambar 4. Model Training RNN dan LSTM

Gambar 4 menunjukkan riwayat pelatihan (training history) untuk dua model pembelajaran mesin, yaitu RNN dan LSTM, yang digunakan untuk memprediksi data, dengan fokus pada nilai *loss* (kerugian) selama proses pelatihan. *Loss* ini menunjukkan seberapa baik model mempelajari data, di mana nilai yang lebih rendah menandakan performa yang lebih baik.

Untuk model RNN (bagian atas), grafik menunjukkan bahwa *training loss* (biru) dan *validation loss* (oranye) awalnya stabil di kisaran rendah (sekitar 0,1) hingga *epoch* ke-12. Namun, setelah itu, *validation loss* tiba-tiba melonjak tajam mendekati 0,9, yang menandakan model mulai overfit, artinya model tidak lagi bekerja baik untuk data baru.

Sebaliknya, untuk model LSTM (bagian bawah), grafik menunjukkan penurunan yang konsisten pada *training loss* (biru) dan *validation loss* (oranye) dari awal hingga sekitar *epoch* ke-10, lalu stabil di kisaran sangat rendah (sekitar 0,01). Hal ini menunjukkan bahwa LSTM belajar dengan efektif dan tetap konsisten tanpa tanda-tanda *overfitting*, menjadikannya lebih andal dibandingkan RNN dalam konteks ini.

Tabel 2. Model Training History RNN dan LSTM

	Evaluasi Model RNN (Training)	Evaluasi Model LSTM (Training)	Evaluasi Model LSTM (Testing)
RMSE	0.0952	0.0633	0.0832
MSE	0.0091	0.0040	0.0112
MAE	0.0621	0.0367	0.0669
MAPE	60.16%	43.35%	65.64%
R ²	0.6881	0.8684	0.7071

Tabel 2 menunjukkan data evaluasi model, model LSTM menunjukkan performa yang lebih unggul dibandingkan model RNN saat dilatih (training). Hal ini terlihat dari semua metrik evaluasi: LSTM memiliki nilai RMSE (0,0633), MSE (0,0040), dan MAE (0,0367) yang lebih rendah, serta nilai R² (0,8684) yang jauh lebih tinggi daripada model RNN. Meskipun performa model LSTM saat diuji (testing) sedikit menurun dibandingkan saat dilatih, seperti yang ditunjukkan oleh peningkatan nilai RMSE (0,0832), MSE (0,0112), dan MAE (0,0669), model ini tetap menunjukkan kemampuan generalisasi yang baik dengan nilai R² sebesar 0,7071. Secara keseluruhan, model LSTM adalah pilihan yang lebih baik karena memiliki akurasi yang lebih tinggi dan kesalahan prediksi yang lebih kecil.

Tabel 3. Evaluasi Kinerja Model Komparatif

Model	Metrik	Kinerja pada Data Pelatihan	Kinerja pada Data Pengujian
RNN	MAE	0.0621	0.0763
	RMSE	0.0952	0.1035
LSTM	MAE	0.0367	0.0730
	RMSE	0.0633	0.1036

Untuk mengevaluasi kinerja dan kemampuan generalisasi model, dilakukan perbandingan hasil prediksi pada data pelatihan (*training set*) dan data pengujian (*testing set*). Data pelatihan adalah data yang dilihat oleh model selama proses belajar, sementara data pengujian adalah data baru yang digunakan untuk mengukur performa model di dunia nyata. Tabel di atas menyajikan hasil evaluasi komparatif dari kedua model menggunakan metrik *Mean Absolute Error* (MAE) dan *Root Mean Square Error* (RMSE). Dari tabel tersebut, dapat diamati bahwa kedua model menunjukkan nilai error yang sangat rendah pada data pelatihan. Namun, evaluasi yang paling penting adalah kinerja pada data pengujian, karena ini mencerminkan kemampuan model untuk memprediksi data yang belum pernah ia lihat sebelumnya. Pada data pengujian, model LSTM menunjukkan keunggulan dengan nilai MAE 0.0730 dan RMSE 0.1036, yang sedikit lebih baik dibandingkan model RNN.

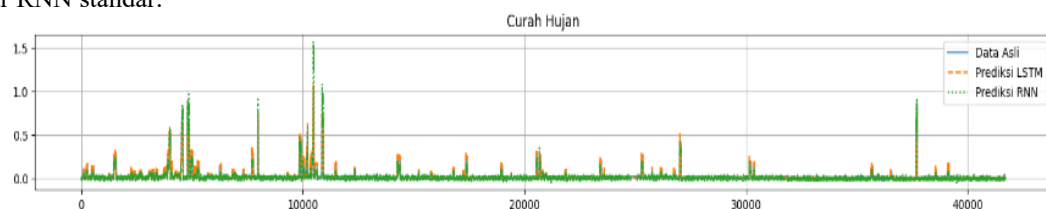
Perbandingan antara kinerja pada data pelatihan dan pengujian juga memberikan wawasan tentang *overfitting*. Terlihat bahwa nilai error untuk kedua model sedikit meningkat saat diuji pada data pengujian, yang merupakan hal wajar. Namun, konsistensi kinerja LSTM yang relatif lebih stabil antara set pelatihan dan pengujian menunjukkan bahwa arsitekturnya lebih efektif dalam menangkap pola yang dapat digeneralisasi, sejalan dengan hasil pada grafik riwayat pelatihan.

Untuk memastikan bahwa perbandingan antara kinerja model RNN dan LSTM bersifat adil (*faircomparison*), desain eksperimen dirancang untuk menjaga kondisi dan parameter utama tetap identik untuk kedua model. Dengan menyamakan parameter-parameter ini, perbedaan kinerja yang teramati dapat diatribusikan secara lebih valid kepada perbedaan arsitektural fundamental antara RNN dan LSTM, bukan karena faktor eksternal.

Parameter-parameter kunci yang diseragamkan dalam eksperimen ini adalah sebagai berikut :

- Arsitektur Jaringan:** Kedua model dibangun dengan arsitektur yang sebanding, yaitu terdiri dari satu lapisan rekuren (baik itu lapisan RNN atau LSTM) yang diikuti oleh lapisan *output* (Dense). Jumlah unit neuron pada lapisan rekuren tersebut juga dibuat sama untuk kedua model untuk memastikan keduanya memiliki kapasitas pemodelan yang setara.
- Parameter Pelatihan:** Proses pelatihan untuk kedua model menggunakan konfigurasi yang identik. Secara spesifik, keduanya dilatih dengan batch size 32 dan jumlah maksimum 100 epoch.
- Fungsi Optimasi dan Kerugian (*Optimizer & Loss Function*):** Kedua model menggunakan *optimizer* 'Adam' yang merupakan standar industri untuk sebagian besar kasus *deep learning*. Selain itu, fungsi kerugian (*loss function*) yang sama, yaitu *Mean Squared Error* (MSE), digunakan untuk menghitung error dan mengoptimalkan bobot pada kedua model selama pelatihan.
- Dataset yang Sama:** Proses pembagian data pelatihan dan pengujian (rasio 80:20) serta teknik pra-pemrosesan (imputasi dan normalisasi Min-Max) diterapkan secara identik pada data yang dimasukkan ke kedua model.

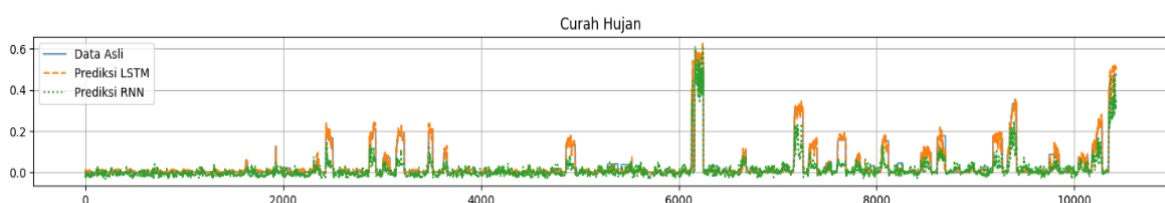
Dengan standarisasi ini, hasil evaluasi yang menunjukkan keunggulan LSTM dapat diinterpretasikan sebagai bukti dari efektivitas mekanisme internalnya (*gating mechanism*) dalam menangani data deret waktu dibandingkan arsitektur RNN standar.



Gambar 5 Prediksi pada Data Training

Gambar 5 menunjukkan data curah hujan beserta prediksi menggunakan dua model, yaitu LSTM dan RNN, selama periode waktu yang panjang. Kurva biru mewakili data asli curah hujan, yang menunjukkan variasi signifikan dengan puncak tertinggi mencapai sekitar 1,5. Kurva oranye (prediksi LSTM) dan kurva hijau (prediksi RNN) mencoba mengikuti pola ini, tetapi dengan pendekatan yang berbeda.

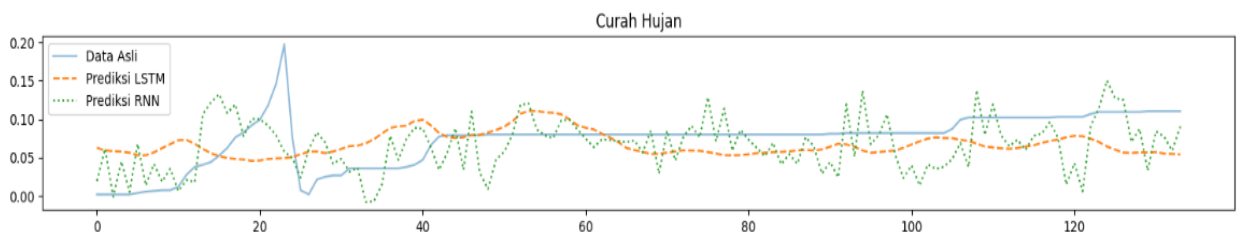
Pada awal periode, data asli menunjukkan beberapa lonjakan besar, terutama di rentang 0 hingga 10.000, yang kemudian mereda menjadi fluktuasi kecil di kisaran 0 hingga 0,5 seiring waktu berlalu hingga 40.000. Prediksi LSTM (oranye) cenderung lebih stabil dan mendekati data asli di beberapa titik, sementara prediksi RNN (hijau) menunjukkan variasi yang lebih tajam, terutama pada puncak-puncak tertentu. Secara umum, kedua model memberikan gambaran yang cukup baik tentang tren curah hujan, meskipun ada perbedaan dalam menangkap detail fluktuasi data.



Gambar 6. Prediksi pada Data Testing

Gambar 6 menampilkan data curah hujan beserta prediksi menggunakan dua model, yaitu LSTM dan RNN, selama periode waktu tertentu. Kurva biru mewakili data asli curah hujan, yang menunjukkan fluktuasi kecil dengan puncak tertinggi mencapai sekitar 0,6. Kurva oranye (prediksi LSTM) dan kurva hijau (prediksi RNN) mencoba mengikuti pola ini dengan pendekatan masing-masing.

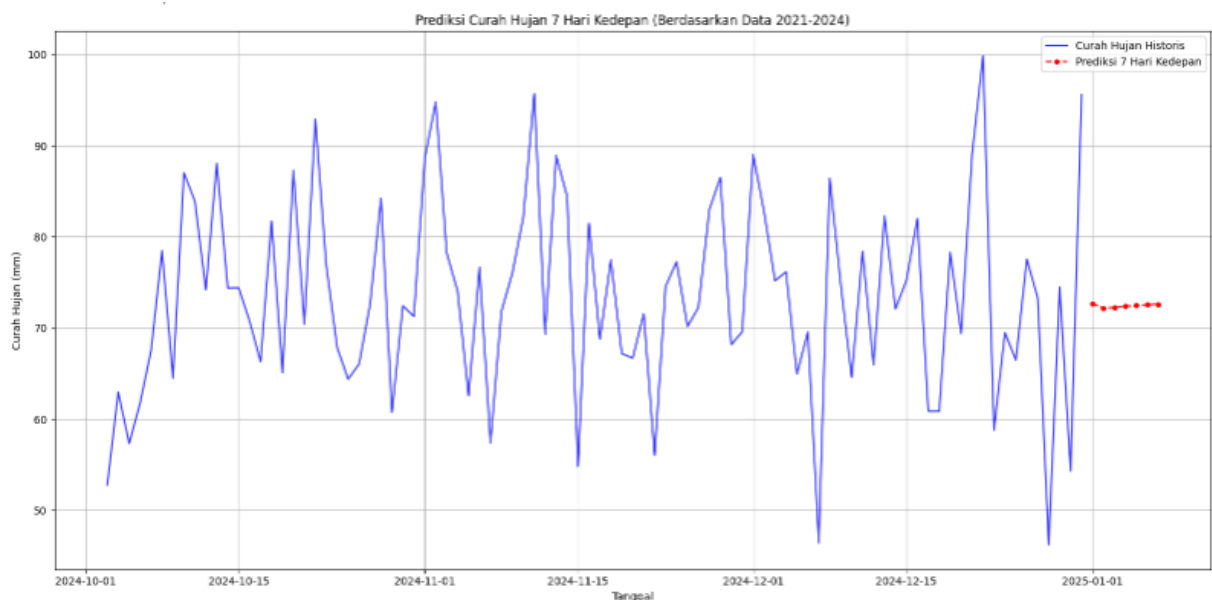
Data asli menunjukkan variasi kecil di kisaran 0 hingga 0,2 sepanjang periode, dengan beberapa lonjakan tajam yang mencapai 0,4 hingga 0,6, terutama di sekitar titik 2000, 6000, dan 10000. Prediksi LSTM (oranye) cenderung lebih dekat dengan data asli pada puncak-puncak tertentu, sementara prediksi RNN (hijau) menunjukkan variasi yang sedikit lebih konsisten namun tetap mengikuti tren umum. Secara keseluruhan, kedua model memberikan gambaran yang cukup baik tentang pola curah hujan, dengan perbedaan kecil dalam menangkap lonjakan data.



Gambar 7. Prediksi Cuaca menggunakan RNN dan LSTM

Gambar 7. menampilkan data curah hujan yang dibandingkan dengan dua jenis prediksi, yaitu LSTM dan RNN, untuk memahami pola hujan dari waktu ke waktu. Kurva biru mewakili data asli curah hujan, yang menunjukkan fluktuasi dengan puncak tertinggi mencapai sekitar 0,20 pada awal periode. Kurva oranye (prediksi LSTM) dan kurva hijau (prediksi RNN) mengikuti pola data asli, tetapi dengan sedikit perbedaan.

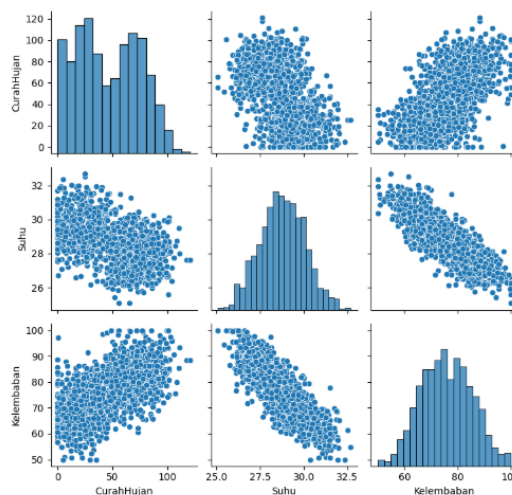
Pada awalnya, data asli menunjukkan lonjakan besar, yang kemudian diikuti oleh fluktuasi yang lebih stabil di kisaran 0,00 hingga 0,10. Prediksi LSTM (oranye) cenderung lebih dekat dengan data asli secara keseluruhan, dengan garis yang lebih halus dan konsisten. Sementara itu, prediksi RNN (hijau) menunjukkan variasi yang lebih tajam, terutama di bagian akhir, tetapi tetap mengikuti tren umum. Secara keseluruhan, kedua model prediksi memberikan gambaran yang cukup baik tentang pola curah hujan, meskipun masing-masing memiliki pendekatan yang sedikit berbeda dalam menangkap fluktuasi data.



Gambar 8 Prediksi Curah Hujan 7 Hari Kedepan

Gambar 8 menunjukkan prediksi curah hujan selama 7 hari ke depan berdasarkan data dari tahun 2021 hingga 2024. Kurva biru mewakili curah hujan historis, yang menunjukkan fluktuasi harian dari sekitar 50 mm hingga puncak tertinggi mendekati 100 mm. Kurva merah, yang merupakan prediksi untuk 7 hari ke depan, menunjukkan tren yang sedikit berbeda, dengan nilai yang lebih rendah dibandingkan puncak historis, berkisar antara 60-70 mm, dan tampak stabil menuju akhir periode.

Secara keseluruhan, grafik ini menggambarkan bahwa curah hujan cenderung bervariasi setiap hari, dengan beberapa lonjakan signifikan di masa lalu. Prediksi untuk minggu depan menunjukkan kondisi yang lebih stabil dengan curah hujan yang sedikit lebih rendah dibandingkan puncak historis, memberikan gambaran yang lebih tenang untuk perencanaan ke depan. Periode waktu yang ditampilkan dimulai dari Oktober 2024 hingga awal Januari 2025.



Gambar 9 Scatter Plot antar parameter

Gambar 9 ini memberikan wawasan penting tentang karakteristik data cuaca yang dianalisis, secara spesifik kelembaban yang tinggi berkorelasi dengan potensi hujan yang tinggi, sementara suhu yang tinggi cenderung berhubungan dengan tingkat kelembaban yang lebih rendah.

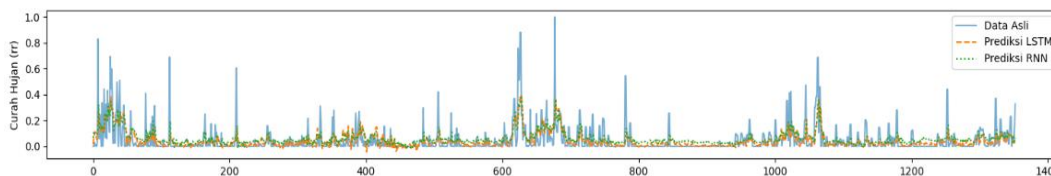
Informasi dan pola hubungan ini sangat berharga untuk pengembangan model kecerdasan buatan (AI) yang bertujuan meramal cuaca. Data ini mengonfirmasi bahwa variabel suhu dan kelembaban merupakan fitur input yang sangat relevan dan berguna. Dengan mempelajari pola-pola ini, sebuah model AI dapat dilatih untuk membuat prediksi curah hujan yang lebih akurat di masa depan.

3.2 Implikasi Analisis Korelasi pada Desain Model (Versi dengan Referensi)

Analisis korelasi yang divisualisasikan pada Gambar 9 bukan hanya sekadar tahap eksplorasi data, melainkan menjadi dasar strategis dalam perancangan fitur input untuk model prediksi. Wawasan yang diperoleh bahwa parameter seperti suhu dan kelembaban memiliki hubungan yang jelas dengan curah hujan secara langsung dimanfaatkan untuk membangun model prediksi berbasis multivariate time series, sebuah pendekatan yang telah terbukti lebih unggul dibandingkan pendekatan univariat dalam berbagai studi peramalan cuaca.

Artinya, model tidak hanya dilatih menggunakan data historis curah hujan saja (pendekatan univariat), tetapi juga diberikan fitur-fitur input tambahan yang relevan seperti suhu, kelembaban, serta arah dan kecepatan angin. Keputusan ini didasarkan pada konfirmasi dari Gambar 9 bahwa variabel-variabel tersebut merupakan fitur input yang sangat berguna, yang mana pemilihan fitur yang tepat merupakan langkah krusial dalam meningkatkan akurasi model prediktif.

Dengan memberikan konteks yang lebih kaya ini, model RNN dan LSTM didorong untuk mempelajari hubungan non-linear yang kompleks antara berbagai parameter cuaca. Sebagai contoh, model dapat belajar bahwa kombinasi spesifik antara penurunan suhu dan peningkatan kelembaban secara drastis seringkali mendahului peristiwa hujan lebat. Pemanfaatan fitur-fitur berkorelasi tinggi inilah yang secara konkret berkontribusi pada kemampuan model khususnya LSTM untuk mencapai akurasi prediksi yang lebih baik, karena prediksi tidak lagi hanya bergantung pada tren masa lalu dari satu variabel, melainkan pada interaksi dinamis dari beberapa faktor meteorologi sekaligus.



Gambar 10 Visualisasi hasil prediksi Hujan Harian menggunakan metode LSTM dan RNN

Gambar 10 terdiri dari empat subplot yang menunjukkan perbandingan antara data asli (digambarkan oleh garis biru), prediksi LSTM (garis oranye putus-putus), dan prediksi RNN (garis hijau putus-putus). Untuk parameter curah hujan, kedua model mampu mengenali lonjakan tajam, meskipun ada beberapa titik ekstrem di mana prediksi sedikit meleset. Sementara itu, untuk suhu udara, model LSTM lebih halus dan lebih akurat dalam mengikuti pola fluktuasi suhu dibandingkan dengan RNN, yang cenderung menghasilkan prediksi dengan fluktuasi dan ketidakstabilan yang lebih tinggi.

Untuk arah dan kecepatan angin, visualisasi menunjukkan bahwa baik LSTM maupun RNN mampu mengikuti tren utama data asli, meskipun RNN tampaknya menghasilkan lebih banyak noise pada data kecepatan angin. LSTM menunjukkan konsistensi yang lebih baik dalam menangani data dengan perubahan yang relatif cepat dan acak. Hasil

ini menunjukkan bahwa arsitektur LSTM memiliki keunggulan dalam menangani urutan data yang kompleks dan berjangka waktu lama, serta lebih mampu mempertahankan informasi konteks dari waktu ke waktu dibandingkan dengan model RNN standar. Dengan demikian, LSTM dapat dianggap sebagai pilihan yang lebih tepat untuk pemodelan data cuaca per hari.

Keunggulan model LSTM dalam prediksi ini tidak hanya terlihat dari visualisasi yang lebih halus dan lebih dekat dengan data asli, tetapi juga didukung oleh hasil evaluasi kuantitatif menggunakan metrik MAE dan RMSE. LSTM mencatat nilai MAE sebesar 0,0730 dan RMSE sebesar 0,1036, lebih rendah dari RNN yang memiliki MAE sebesar 0,0763 dan RMSE sebesar 0,1035. Nilai-nilai ini menunjukkan bahwa LSTM secara konsisten menghasilkan prediksi yang lebih dekat dengan data aktual dan lebih efektif dalam mengurangi kesalahan ekstrem. Ini menunjukkan bahwa pemilihan arsitektur model memiliki pengaruh besar pada kualitas prediksi dalam konteks data deret waktu cuaca.

Selanjutnya, hasil ini memberikan implikasi penting dalam penerapan model prediksi cuaca waktu nyata, khususnya di sektor-sektor yang sangat bergantung pada kondisi atmosfer seperti pertanian, transportasi, dan mitigasi bencana. Peningkatan akurasi prediksi per jam dapat membantu pengambilan keputusan yang lebih cepat dan lebih tepat. Mengingat kompleksitas dan sifat dinamis data cuaca, penggunaan LSTM merupakan alternatif yang menjanjikan untuk dikembangkan lebih lanjut, baik dengan menambahkan fitur eksogen maupun menggabungkannya dengan model hibrida untuk meningkatkan kinerja prediksi jangka panjang.

Tabel 3. Evaluasi performa metode

	RNN	LSTM
RMSE	0.1035	0.1036
MAE	0.0763	0.0730

Tabel 3 menunjukkan hasil evaluasi kinerja model RNN dan LSTM menggunakan dua metrik utama, yaitu *Root Mean Square Error* (RMSE) dan *Mean Absolute Error* (MAE). Metrik MAE menunjukkan rata-rata kesalahan absolut antara nilai yang diprediksi dan data asli, sementara RMSE lebih menekankan pada kesalahan besar karena bersifat kuadratik. Dari hasil yang ditunjukkan, model LSTM berkinerja lebih baik daripada RNN pada kedua metrik evaluasi.

3.3 Hubungan Antara Curah Hujan Dengan Beberapa Variabel Waktu Dan Trigonometri

Tabel 4. Data Frame dengan Fitur Musiman (5 baris pertama) bulan 1 – bulan 4

Tanggal	Curah Hujan	Suhu	bulan	bulan_1	bulan_2	bulan_3	bulan_4
2021-01-01	73.152421	26.942379	1	True	False	False	False
2021-01-02	85.220958	28.671291	1	True	False	False	False
2021-01-03	66.101197	28.452482	1	True	False	False	False
2021-01-04	65.363054	28.585805	1	True	False	False	False
2021-01-05	68.014241	27.653857	1	True	False	False	False

Tabel 4 menunjukkan contoh data yang telah diolah dengan menambahkan fitur musiman (seasonal features). Kolom bulan secara langsung menampilkan angka bulan saat ini (dalam kasus ini, 1 untuk Januari), sedangkan kolom bulan_1 hingga bulan_4 adalah representasi one-hot encoding dari bulan. Terlihat bahwa karena semua data berasal dari awal Januari, maka kolom bulan_1 selalu bernilai True sementara kolom bulan lainnya (2, 3, dan 4) selalu bernilai False, menunjukkan bahwa data ini secara spesifik disiapkan agar model machine learning bisa mengenali pola data berdasarkan bulannya.

Tabel 5. Data Frame dengan Fitur Musiman (5 baris pertama) bulan 5 – bulan 11

Tanggal	bulan_5	bulan_6	bulan_7	bulan_8	bulan_9	bulan_10	bulan_11
2021-01-01	False	False	False	False	False	False	False
2021-01-02	False	False	False	False	False	False	False
2021-01-03	False	False	False	False	False	False	False
2021-01-04	False	False	False	False	False	False	False
2021-01-05	False	False	False	False	False	False	False

Tabel 5 adalah kelanjutan dari fitur musiman yang sudah dijelaskan sebelumnya. Sesuai dengan data tanggal yang menunjukkan periode 1 hingga 5 Januari 2021, wajar jika semua kolom yang merepresentasikan bulan-bulan berikutnya, yaitu dari bulan 5 hingga bulan 11, memiliki nilai False. Hal ini memperjelas cara kerja metode one-hot encoding, di mana suatu fitur bulan hanya akan bernilai True jika data tersebut benar-benar termasuk dalam bulan yang diwakilinya, dan sebaliknya akan bernilai False untuk bulan-bulan lainnya.

Tabel 6. Data Frame dengan Fitur Musiman (5 baris pertama) bulan_12 day_of_year day_of_year_sin day_of_year_cos

Tanggal	bulan_12	day_of_year	day_of_year_sin	day_of_year_cos
2021-01-01	False	1	0.017202	0.999852
2021-01-02	False	2	0.034398	0.999408
2021-01-03	False	3	0.051584	0.998669
2021-01-04	False	4	0.068755	0.997634
2021-01-05	False	5	0.085906	0.996303

Tabel 6 menampilkan tambahan fitur musiman yang lebih spesifik. Selain kolom bulan_12 yang bernilai False karena data berasal dari Januari, ada juga kolom day_of_year yang menunjukkan urutan hari dalam setahun. Yang paling penting, kolom day_of_year_sin dan day_of_year_cos dibuat untuk merepresentasikan hari dalam setahun secara siklus. Dengan metode ini, model dapat memahami bahwa hari pertama dalam setahun (1) secara periodik sangat dekat dengan hari terakhir (365), yang tidak dapat ditangkap jika hanya menggunakan angka hari biasa. Ini adalah teknik yang efektif agar model mampu mengenali pola data yang berulang setiap tahun.

Tabel 7. Data Frame dengan Fitur Musiman (5 baris pertama) day_of_week quarter

Tanggal	day_of_week	quarter
2021-01-01	4	1
2021-01-02	5	2
2021-01-03	6	3
2021-01-04	0	4
2021-01-05	1	5

Tabel 7 menampilkan dua fitur musiman lagi. Kolom day_of_week menunjukkan hari dalam seminggu, di mana nilai 4 berarti hari Jumat, 5 hari Sabtu, 6 hari Minggu, dan seterusnya, yang sangat berguna untuk mendeteksi pola mingguan seperti perbedaan antara hari kerja dan akhir pekan. Namun, ada keanehan pada kolom quarter; seharusnya, semua tanggal di bulan Januari berada di kuartal pertama dan memiliki nilai yang sama (1), tetapi data di tabel ini malah menunjukkan urutan angka 1 hingga 5, yang tampaknya tidak sesuai dengan definisi kuartal dalam setahun dan bisa jadi merupakan kesalahan dalam data atau pelabelan.

3.2 Pembahasan

Berdasarkan hasil prediksi dan evaluasi, dapat dilihat bahwa model LSTM menunjukkan kinerja yang unggul dibandingkan dengan RNN dalam memprediksi data cuaca per jam. Hal ini dapat dilihat dari hasil visualisasi yang lebih halus dan lebih mendekati pola data asli, terutama pada variabel suhu dan arah angin. Selain itu, nilai evaluasi metrik MAE dan RMSE dari LSTM juga lebih rendah, yang menunjukkan tingkat kesalahan prediksi yang lebih kecil. Keunggulan-keunggulan ini sejalan dengan karakteristik arsitektur LSTM, yang mampu menyimpan informasi jangka panjang dan mengatasi masalah gradien menghilang yang umum terjadi pada RNN standar. Dengan demikian, LSTM lebih efektif dalam menangkap pola musiman serta fluktuasi harian dalam data cuaca.

Namun, meskipun LSTM berkinerja lebih baik secara umum, masih ada tantangan dalam memprediksi variabel yang sangat fluktuatif seperti curah hujan dan kecepatan angin. Kedua model cenderung kesulitan dalam memprediksi lonjakan ekstrem secara akurat, yang tercermin dalam penyimpangan nilai puncak. Hal ini menunjukkan bahwa untuk mencapai hasil prediksi yang lebih akurat, diperlukan pendekatan canggih seperti penyetelan hiperparameter yang lebih optimal, penggunaan fitur tambahan (misalnya tekanan udara atau kelembapan), atau penerapan model hibrida yang menggabungkan kekuatan model deret waktu dengan teknik pembelajaran mesin lainnya. Diskusi ini menekankan pentingnya pemilihan model yang tepat dan strategi pelatihan yang baik dalam pengembangan sistem prediksi cuaca yang andal.

4. KESIMPULAN

Penelitian ini menyimpulkan bahwa model Long Short-Term Memory (LSTM) menunjukkan kinerja superior secara signifikan dibandingkan model Recurrent Neural Network (RNN) untuk prediksi curah hujan harian di Kabupaten Batang, yang dibuktikan dengan tingkat kesalahan prediksi yang lebih rendah dan stabilitas pelatihan yang lebih baik. Keunggulan ini menegaskan relevansi ilmiah arsitektur LSTM dalam menangkap dependensi data cuaca jangka panjang yang kompleks, sekaligus menyoroti kontribusinya sebagai fondasi yang lebih andal untuk pengembangan sistem peringatan dini bencana hidrometeorologi di tingkat lokal. Meskipun menjanjikan, penelitian di masa depan direkomendasikan untuk fokus pada peningkatan akurasi prediksi peristiwa ekstrem melalui eksplorasi model hibrida, penggabungan variabel meteorologi tambahan seperti tekanan udara, dan optimisasi hiperparameter lebih lanjut untuk meningkatkan ketahanan terhadap perubahan iklim.

REFERENCES

- [1] J. K. Mutinda and A. K. Langat, “Stock price prediction using combined GARCH-AI models,” *Sci Afr*, vol. 26, no. July, p. e02374, 2024, doi: 10.1016/j.sciaf.2024.e02374.
- [2] G. Jerse and A. Marcucci, “Deep Learning LSTM-based approaches for 10.7 cm solar radio flux forecasting up to 45-days,” *Astronomy and Computing*, vol. 46, no. January, p. 100786, 2024, doi: 10.1016/j.ascom.2024.100786.
- [3] F. F. Mojtahedi, N. Yousefpour, S. H. Chow, and M. Cassidy, “Deep Learning for Time Series Forecasting: Review and Applications in Geotechnics and Geosciences,” *Archives of Computational Methods in Engineering*, vol. 32, no. 6, pp. 3415–3445, 2025, doi: 10.1007/s11831-025-10244-5.
- [4] Y. Man, Q. Yang, J. Shao, G. Wang, L. Bai, and Y. Xue, “Enhanced LSTM Model for Daily Runoff Prediction in the Upper Huai River Basin, China,” *Engineering*, vol. 24, pp. 229–238, 2023, doi: 10.1016/j.eng.2021.12.022.
- [5] P. H. Frederik, K. Daniel, K. Christina, and H. Markus, “MC-LSTM : Mass-Conserving LSTM,” 2021.
- [6] T. L. Rohana, “Penerapan Metode Peramalan Menggunakan Fuzzy Arma,” *MATHunesa: Jurnal Ilmiah Matematika*, vol. 13, no. 1, pp. 13–24, 2025, doi: 10.26740/mathunesa.v13n1.p13-24.
- [7] S. Sahoo and A. Govind, “Understanding Changes in the Hydrometeorological Conditions towards Climate-Resilient Agricultural Interventions in Ethiopia,” *Agronomy*, vol. 13, no. 2, 2023, doi: 10.3390/agronomy13020387.
- [8] D. A. Alif Hidayat, M. H. M. Aditya Pradana, and A. Saikhu, “Hybrid Decomposition ICEEMDAN-EWT Deep Learning Framework for Wind Speed Forecasting,” *Journal of Applied Informatics and Computing*, vol. 9, no. 4, pp. 1332–1345, 2025, doi: 10.30871/jaic.v9i4.10241.
- [9] A. López-García, O. Blasco-Blasco, M. Liern-García, and S. E. Parada-Rico, “Early detection of students’ failure using Machine Learning techniques,” *Operations Research Perspectives*, vol. 11, no. November, p. 100292, 2023, doi: 10.1016/j.orp.2023.100292.
- [10] I. D. Mienye, T. G. Swart, and G. Obaido, “Recurrent Neural Networks: A Comprehensive Review of Architectures, Variants, and Applications,” *Information (Switzerland)*, vol. 15, no. 9, pp. 1–34, 2024, doi: 10.3390/info15090517.
- [11] A. Raup, W. Ridwan, Y. Khoeriyah, S. Supiana, and Q. Y. Zaqiah, “Deep Learning dan Penerapannya dalam Pembelajaran,” *JIIP - Jurnal Ilmiah Ilmu Pendidikan*, vol. 5, no. 9, pp. 3258–3267, 2022, doi: 10.54371/jiip.v5i9.805.
- [12] D. Tarkus, S. R. U. A. Sompie, and A. Jacobus, “Implementasi Metode Recurrent Neural Network pada Pengklasifikasian Kualitas Telur Puyuh,” *Jurnal Teknik Informatika*, vol. 15, no. 2, pp. 137–144, 2020.
- [13] N. S. Islamaynita, D. Prawira, and N. Mutiah, “Prediksi Kata Selanjutnya Pada Rekam Medis Elektronik Klinik XYZ Menggunakan Bidirectional LSTM,” *Coding : Jurnal Komputer dan Aplikasi*, vol. 13, no. 2, pp. 136–148, 2025.
- [14] A. Kumar, N. Gaur, and A. Nanthaamorphong, “Machine learning RNNs, SVM and NN Algorithm for Massive-MIMO-OTFS 6G Waveform with Rician and Rayleigh channel,” *Egyptian Informatics Journal*, vol. 27, no. July, p. 100531, 2024, doi: 10.1016/j.eij.2024.100531.
- [15] L. Kristiana and D. Miyanto, “Penambahan Parameter PM2.5 dalam Prediksi Kualitas Udara : Long Short Term Memory,” *Multimedia Artificial Intelligent Networking Database (MIND)*, vol. 8, no. 2, pp. 188–202, 2023.
- [16] B. A. H. Kholifatullah and A. Prihanto, “Penerapan Metode Long Short Term Memory Untuk Klasifikasi Pada Hate Speech,” *Journal of Informatics and Computer Science (JINACS)*, vol. 04, pp. 292–297, 2023, doi: 10.26740/jinacs.v4n03.p292-297.
- [17] S. B. Timothy, C. Imam, and Y. Novanto, “Penerapan Algoritma Long Short-Term Memory (LSTM) berbasis Multi Fungsi Aktivasi Terbobot dalam Prediksi Harga Ethereum,” *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 7, no. 3, pp. 1101–1107, 2023, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [18] N. D. Alotaibi, H. Jahanshahi, Q. Yao, J. Mou, and S. Bekiros, “An Ensemble of Long Short-Term Memory Networks with an Attention Mechanism for Upper Limb Electromyography Signal Classification,” *Mathematics*, vol. 11, no. 18, 2023, doi: 10.3390/math11184004.
- [19] A. Zafar *et al.*, “Enhanced solar power forecasting in smart grids using a hybrid autoencoder and long short-term memory model,” *Energy Exploration and Exploitation*, 2025, doi: 10.1177/01445987251360490.
- [20] A. Rosyd, A. Irma Purnamasari, and I. Ali, “Penerapan Metode Long Short Term Memory (Lstm) Dalam Memprediksi Harga Saham Pt Bank Central Asia,” *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 1, pp. 501–506, 2024, doi: 10.36040/jati.v8i1.8440.
- [21] A. Herdhyanti, L. Muflikhah, and I. Cholissodin, “Prediksi Curah Hujan dengan Empat Parameter menggunakan Backpropagation (Studi Kasus: Stasiun Meteorologi Ahmad Yani),” *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 6, no. 12, pp. 5862–5870, 2022, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [22] R. Yulvina *et al.*, “Hybrid Vision Transformer and Convolutional Neural Network for Multi-Class and Multi-Label Classification of Tuberculosis Anomalies on Chest X-Ray,” *Computers*, vol. 13, no. 12, pp. 1–29, 2024, doi: 10.3390/computers13120343.
- [23] M. M. Ahsan, M. A. P. Mahmud, P. K. Saha, K. D. Gupta, and Z. Siddique, “Effect of Data Scaling Methods on Machine Learning Algorithms and Model Performance,” *Technologies (Basel)*, vol. 9, no. 3, pp. 5–9, 2021, doi: 10.3390/technologies9030052.
- [24] W. A. Firmansyach, U. Hayati, and Y. Arie Wijaya, “Analisa Terjadinya Overfitting Dan Underfitting Pada Algoritma Naive Bayes Dan Decision Tree Dengan Teknik Cross Validation,” *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 1, pp. 262–269, 2023, doi: 10.36040/jati.v7i1.6329.