

Penerapan Model *Recurrent Neural Network* (RNN) untuk Prediksi Curah Hujan Berbasis Data Historis

Imam Farisi ^{1,*}, Jafar Shadiq ², Wiwit Priyadi ³, Dani Maulana ⁴, Acep ⁵, Sonia F. Gusril ⁶

^{1,4} Sistem Informasi; Universitas Bina Insani; Jl.Siliwangi No.6; e-mail: imam@binainsani.ac.id , danimaulana@icloud.com

^{2,6} Teknik Informatika; Universitas Bina Insani; Jl.Siliwangi No.6; e-mail: jafarshadiq@binainsani.ac.id , friskasonia18@gmail.com

^{3,5} Rekayasa Perangkat Lunak; Universitas Bina Insani; Jl.Siliwangi No.6; e-mail: wiwit@binainsani.ac.id , acepmctominay@gmail.com

* Korespondensi: e-mail: imam@binainsani.ac.id

Diterima: 04 Desember 2024 ; Review: 05 Desember 2024 ; Disetujui: 17 Desember 2024

Cara citasi: Farisi I, Shadiq J, Priyadi W, Maulana D, Acep A, Gusril SF. 2024. Penerapan Model *Recurrent Neural Network* (RNN) untuk Prediksi Curah Hujan Berbasis Data Historis. Information System for Educators and Professionals. Vol 9(2): 217-226.

Abstrak: Prediksi curah hujan yang akurat penting untuk mitigasi bencana, perencanaan pertanian, dan pengelolaan sumber daya air, khususnya di Jakarta yang rentan terhadap banjir. Penelitian ini menerapkan model *Recurrent Neural Network* (RNN) untuk memprediksi curah hujan berbasis data historis dengan memanfaatkan data curah hujan interval 1 jam sepanjang Januari hingga Desember 2022. Model RNN dipilih karena kemampuannya menangani data berurutan dan menangkap pola temporal dalam data time-series. Data diproses melalui tahap normalisasi dan pembagian dataset dengan tiga skenario yaitu 50:50, 70:30, dan 90:10. Evaluasi model menggunakan metrik *Root Mean Square Error* (RMSE) menunjukkan bahwa skenario 90:10 memberikan performa terbaik dengan nilai RMSE terkecil sebesar 0,0925, dibandingkan dengan skenario 50:50 (0,1679) dan 70:30 (0,1962). Model ini berpotensi mendukung sistem peringatan dini banjir serta pengambilan keputusan strategis dalam tata kelola lingkungan di Jakarta. Pengembangan lebih lanjut disarankan untuk mempertimbangkan variabel tambahan seperti pola angin dan anomali iklim global lainnya guna meningkatkan akurasi prediksi.

Kata kunci: Prediksi Curah Hujan, *Recurrent Neural Network*, Data Historis, *Root Mean Square Error*

Abstract: Accurate rainfall prediction is crucial for disaster mitigation, agricultural planning, and water resource management, particularly in Jakarta, which is highly prone to flooding. This study implements a *Recurrent Neural Network* (RNN) model to predict rainfall based on historical data, utilizing hourly rainfall data from January to December 2022. The RNN model was chosen for its ability to handle sequential data and capture temporal patterns in time-series datasets. The data underwent normalization and was divided into three dataset scenarios: 50:50, 70:30, and 90:10. Model evaluation using the *Root Mean Square Error* (RMSE) metric indicated that the 90:10 scenario achieved the best performance, with the lowest RMSE value of 0.0925, compared to 0.1679 for the 50:50 scenario and 0.1962 for the 70:30 scenario. This model has the potential to support early flood warning systems and inform strategic decision-making in environmental management in Jakarta. Further development is recommended to incorporate additional variables, such as wind patterns and other global climate anomalies, to enhance prediction accuracy.

Keywords: Rainfall Prediction, *Recurrent Neural Network*, Historical Data, *Root Mean Square Error*

1. Pendahuluan

Jakarta, sebagai salah satu kota terbesar di Indonesia, menghadapi tantangan besar terkait dengan perubahan pola curah hujan [1]. Sebagai wilayah dengan populasi yang padat dan infrastruktur urban yang berkembang pesat, ketidakaturan curah hujan sering kali menjadi penyebab banjir besar yang mengganggu aktivitas ekonomi dan mengancam keselamatan warga. Fenomena ini menjadi semakin kompleks dengan adanya peningkatan urbanisasi yang mengurangi daya serap tanah [2], serta sistem drainase yang tidak optimal [3]. Semua faktor ini memperburuk dampak curah hujan yang tinggi, menyebabkan kerugian material dan sosial yang signifikan.

Tabel 1. Data Banjir Jakarta Tahun 2002 sampai 2021

Tanggal Kejadian	2 Februari 2002	2 Februari 2007	17 Januari 2013	11 Februari 2015	1 Januari 2020	20 Februari 2021
Curah Hujan Tertinggi (mm/hari)	168	340	100	277	377	226
RW Tergenang Luas (km²)	353	955	599	702	390	113
Area Strategis Pengungsi	168	455	240	281	156	4
Jumlah Lokasi Pengungsian	Ya	Ya	Ya	Ya	Tidak	Tidak
Korban Meninggal	154.270	276.333	90.913	45.813	36.445	3.311
Waktu Surut Genangan (hari)	N.A	N.A	1.250	409	269	44
Korban Meninggal	32	48	40	5	19	5
Waktu Surut Genangan (hari)	6	10	7	7	4	1

Menurut Badan Penanggulangan Bencana Daerah (BPBD) Jakarta tercatat data banjir Jakarta dari tahun 2002 hingga 2021 menunjukkan variasi intensitas dan dampak seperti pada Tabel 1. Curah hujan tertinggi tercatat pada tahun 2020, sebesar 377 mm/hari, mengakibatkan 390 RW tergenang dan 36.445 pengungsi. Tahun 2007 merupakan yang terdampak paling luas, dengan 955 RW dan 455 km² area tergenang. Area strategis terdampak pada sebagian besar kejadian, kecuali pada 2020 dan 2021. Jumlah korban meninggal bervariasi, tertinggi 48 orang pada 2007, sedangkan waktu surut genangan berkisar antara 1-10 hari. Lokasi pengungsian mulai tercatat sejak 2013, dengan puncaknya sebanyak 1.250 lokasi.

Selain faktor-faktor lokal, perubahan iklim global turut berperan dalam mempengaruhi pola cuaca di Jakarta, dengan menyebabkan intensitas curah hujan yang lebih sulit diprediksi dan pola musim yang semakin tidak stabil [4]. Kondisi ini semakin mendesak kota untuk mengambil langkah-langkah adaptasi yang lebih baik guna mengurangi risiko yang ditimbulkan. Oleh karena itu, diperlukan strategi mitigasi yang lebih efektif untuk menghadapi curah hujan ekstrem dan dampaknya, termasuk prediksi cuaca yang lebih akurat, pengelolaan infrastruktur yang lebih baik, dan kesadaran masyarakat akan pentingnya kesiapsiagaan terhadap bencana alam [5].

Prediksi curah hujan yang akurat di Jakarta sangat penting untuk mendukung berbagai aspek perencanaan kota, termasuk pengelolaan sumber daya air dan mitigasi risiko bencana banjir. Dengan tingkat urbanisasi yang tinggi dan rentan terhadap perubahan iklim, memiliki kemampuan untuk memprediksi pola curah hujan secara tepat dapat membantu pemerintah dan masyarakat dalam merencanakan respons yang lebih baik terhadap bencana dan memastikan ketersediaan air yang cukup. Prediksi yang akurat juga berperan dalam perencanaan pembangunan infrastruktur, seperti drainase dan pengelolaan ruang terbuka hijau, yang sangat penting untuk mengurangi dampak negatif dari hujan ekstrem. Namun, prediksi cuaca tradisional sering kali terbatas dalam memproyeksikan pola curah hujan yang dinamis, mengingat tingginya variabilitas data yang kompleks dan banyaknya faktor eksternal yang mempengaruhi curah hujan. Faktor-faktor seperti perubahan iklim, kondisi atmosfer, dan interaksi antara berbagai elemen cuaca menjadi tantangan besar dalam menghasilkan prediksi yang tepat. Oleh karena itu, dibutuhkan pendekatan yang lebih bagus, seperti penggunaan algoritma *machine learning* dan *deep learning*, untuk meningkatkan akurasi prediksi curah

hujan, sehingga dapat lebih baik mendukung perencanaan dan kebijakan mitigasi risiko di Jakarta.

Recurrent Neural Network (RNN) adalah algoritma dalam *deep learning* yang dapat mengolah data berurutan atau *time series* [6], sehingga cocok untuk prediksi berbasis data historis curah hujan. Dengan memanfaatkan data historis curah hujan di Jakarta, algoritma RNN mampu menangkap pola musiman dan pergeseran tren yang mungkin terjadi dari waktu ke waktu. Kemampuan RNN untuk mengingat data-data sebelumnya [7] menjadikannya efektif dalam memproyeksi curah hujan di masa depan dengan tingkat keakuratan yang lebih baik dibandingkan metode konvensional.

Penelitian sebelumnya yang memprediksi curah hujan menggunakan jaringan syaraf tiruan [8]; *backpropagation neural network* [9]; *artificial neural network* [10]. Kemudian penelitian yang menggunakan algoritma RNN antara lain menggunakan data iridiasi matahari [11]; data baja canai panas [12]; data beras [13]. Penelitian dengan menggunakan data beras dan algoritma RNN mendapatkan hasil yang lebih bagus dari pada algoritma LSTM [13]. Oleh karena itu pada penelitian ini akan digunakan data curah hujan dengan menggunakan algoritma RNN. Diharapkan dengan algoritma RNN meningkatkan hasil prediksi yang lebih akurat dan bisa digunakan sebagai upaya mitigasi banjir dan perencanaan infrastruktur kota, serta ketahanan terhadap perubahan iklim. Evaluasi model akan dilakukan dengan membandingkan hasil prediksi terhadap data aktual, menggunakan berbagai metrik evaluasi.

2. Metode Penelitian

Metode yang digunakan dalam penelitian ini adalah CRISP-DM (*Cross-Industry Standard Process for Data Mining*), yang merupakan metodologi yang sering digunakan dalam proses data mining untuk berbagai aplikasi [14]. CRISP-DM terdiri dari beberapa tahapan yang sistematis dan terstruktur untuk menyelesaikan masalah data mining, mulai dari pemahaman bisnis, pengumpulan data, persiapan data, hingga pembentukan model dan evaluasi [15]. Tahapan-tahapan ini sangat berguna untuk membangun model prediksi yang andal dan efektif. Dalam konteks penelitian ini, metode CRISP-DM akan digunakan untuk membangun model prediksi curah hujan yang berbasis pada algoritma *Recurrent Neural Network* (RNN), yang dikenal efektif dalam menangani data berurutan seperti data cuaca.

Tahap pertama dalam CRISP-DM adalah memahami tujuan bisnis atau masalah yang ingin diselesaikan, dalam hal ini adalah prediksi curah hujan di wilayah Jakarta. Setelah itu, data yang diperlukan untuk pelatihan model akan dikumpulkan dan dipersiapkan dengan baik, memastikan bahwa data yang digunakan memiliki kualitas yang baik dan relevansi yang tinggi terhadap masalah yang dihadapi. Tahap selanjutnya adalah pembentukan model, di mana algoritma RNN akan diterapkan untuk mempelajari pola-pola dalam data historis yang terkait dengan curah hujan. Setelah model terbentuk, tahap evaluasi akan dilakukan untuk mengukur performa model, menggunakan metrik RMSE untuk memastikan bahwa model dapat memberikan prediksi yang akurat dan berguna dalam prakiraan cuaca di masa depan. Semakin nilai RMSE mendekati 0 maka kinerja model akan semakin baik [16]. Dengan pendekatan CRISP-DM yang terstruktur, penelitian ini bertujuan untuk menghasilkan model prediksi curah hujan yang optimal di wilayah Jakarta.

Business Understanding

Penelitian ini dimulai dengan pemahaman atas tantangan yang dihadapi Jakarta terkait curah hujan yang tidak menentu dan seringkali ekstrem. Sebagai kota besar dengan tingkat urbanisasi yang tinggi, Jakarta sering kali menghadapi permasalahan seperti banjir, kemacetan, dan kerusakan infrastruktur akibat hujan lebat. Pola curah hujan yang tidak stabil memperumit perencanaan dan pengelolaan kota, termasuk dalam pengendalian banjir, pengelolaan drainase, serta mitigasi dampak bencana. Oleh karena itu, kebutuhan akan sistem prediksi curah hujan yang akurat sangat dibutuhkan agar berbagai sektor dapat lebih siap menghadapi risiko yang ditimbulkan oleh perubahan pola cuaca.

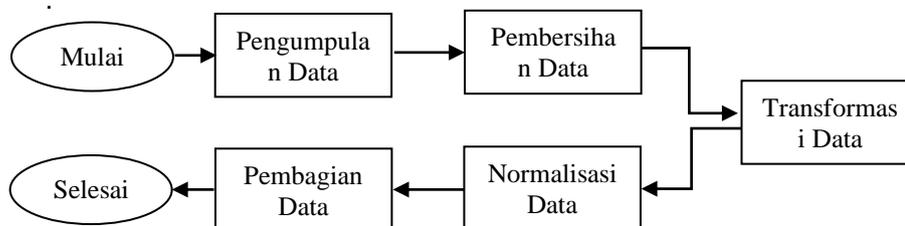
Tujuan utama dari penelitian ini adalah membangun model prediksi curah hujan berbasis data historis menggunakan algoritma *Recurrent Neural Network* (RNN). Model ini diharapkan mampu memberikan proyeksi curah hujan dengan tingkat keakuratan yang tinggi, sehingga dapat mendukung perencanaan strategis dan mitigasi risiko di Jakarta. Dengan mengidentifikasi pola musiman, fluktuasi, dan tren curah hujan secara lebih detail, hasil prediksi ini diharapkan membantu pihak berwenang, pengelola bencana, dan para pemangku

kepentingan lainnya dalam mengambil langkah-langkah yang lebih proaktif untuk meminimalisasi dampak negatif cuaca ekstrem.

Data Preparation

Tahapan *data preparation* merupakan langkah penting dalam proses data mining yang melibatkan beberapa aktivitas untuk memastikan kualitas data yang digunakan dalam membangun model prediksi. Langkah pertama adalah pengumpulan data, yang mencakup pengumpulan informasi yang relevan untuk masalah yang ingin diselesaikan, dalam hal ini adalah data curah hujan yang melibatkan berbagai parameter cuaca. Setelah data terkumpul, tahap berikutnya adalah pembersihan data, di mana data yang hilang, duplikat, atau tidak konsisten akan diperbaiki atau dihapus untuk memastikan bahwa model dilatih menggunakan data yang bersih dan valid.

Selain itu, tahap *data preparation* juga mencakup transformasi data untuk mengubah data dalam format yang lebih sesuai untuk analisis, seperti mengkonversi data kategorikal menjadi numerik atau menggabungkan beberapa variabel yang relevan. Selanjutnya, normalisasi data dilakukan untuk memastikan bahwa semua variabel berada dalam skala yang sama, menghindari dominasi fitur dengan skala yang lebih besar. Terakhir, data dibagi menjadi data latih dan data uji, di mana data latih digunakan untuk melatih model dan data uji digunakan untuk mengukur performa model. Semua tahapan ini sangat penting untuk membangun model yang akurat dan dapat diandalkan.



Gambar 1. Tahapan *Data Preparation*

Pada Gambar 1. menjelaskan beberapa tahapan dari *data preparation*. Tahap pertama yaitu melakukan pengambilan data yang didapatkan dari website Sistem Informasi Banjir Jakarta. Bentuk data masih dalam bentuk JSON file dengan beberapa properties yang tersedia. Kemudian data di bersihkan dengan cara menghapus beberapa data yang tidak terisi ataupun properties yang tidak perlukan. Properti yang akan dipakai hanya curah hujan dan lokasi. Setelah dibersihkan kemudian data di ubah ke bentuk object python agar mudah di proses. Kemudian dilakukan normalisasi agar data menjadi seragam tidak ada yang terlalu besar atau kecil. Tahap terakhir dilakukan pembagian data dengan prosentase 50 data latih dan 50 data uji, 70 data latih dan 30 data uji dan 90 data latih dan 10 data uji.

Modeling

Pada tahap modeling, algoritma RNN diterapkan untuk membangun model prediksi berdasarkan data yang tersedia. Dalam proses ini, model dilatih untuk mempelajari pola-pola dalam data yang dapat digunakan untuk memprediksi hasil di masa depan. Meskipun konfigurasi hyperparameter seperti jumlah lapisan tersembunyi, fungsi aktivasi, dan jumlah iterasi pelatihan dapat memengaruhi performa model, pada tahap ini tidak ada perubahan pada hyperparameter yang digunakan. Fokus utama adalah pada pengujian model dengan tiga proporsi pembagian dataset yang berbeda untuk melihat bagaimana pembagian data tersebut mempengaruhi akurasi model.

Dengan membandingkan hasil dari ketiga proporsi ini, model dievaluasi untuk menentukan pengaruh pembagian data terhadap performa prediksi. Meskipun hyperparameter tetap konstan, eksperimen dengan pembagian dataset yang berbeda memberikan gambaran mengenai bagaimana ukuran data latih dan uji mempengaruhi hasil prediksi dan seberapa baik model dapat menangkap pola dalam data yang diberikan. Hasil ini akan memberikan insight tentang proporsi data yang paling optimal untuk menghasilkan prediksi yang akurat.

Evaluation

Tahap selanjutnya adalah Evaluation, di mana model yang telah dibuat akan dievaluasi menggunakan metrik kinerja yaitu Root Mean Square Error (RMSE) untuk memastikan akurasi dan reliabilitas prediksi.

3. Hasil dan Pembahasan

Pengumpulan dataset dilakukan di Sistem Informasi Banjir Jakarta dengan alamat url <https://pantabanjir.jakarta.go.id/pemantauan-banjir> menggunakan teknik *web scraping*.



Gambar 2. Dataset Curah Hujan

Pada Gambar 2. merupakan contoh *dataset* curah hujan dengan interval 1 jam dari tanggal 1 Januari 2022 sampai 31 Desember 2022. Dataset tersebut berbentuk JSON file yang kemudian dirubah ke bentuk CSV file. JSON di atas menggambarkan sebuah data pengamatan curah hujan yang mencakup informasi mengenai waktu, lokasi, dan petugas yang melakukan pengukuran. Data ini berisi identifikasi unik untuk setiap pengamatan, termasuk waktu pengamatan yang tercatat pada tanggal dan jam tertentu, serta informasi tentang lokasi pengukuran, seperti nama lokasi dan alamat lengkapnya. Selain itu, terdapat informasi mengenai petugas yang melakukan pengamatan. JSON ini menyusun data secara terstruktur, memungkinkan untuk pemrosesan dan analisis yang lebih mudah terkait dengan pengamatan curah hujan di lokasi tertentu pada waktu yang spesifik.

Tabel 2. Properties dari JSON File

Nama	Keterangan	Tipe Data
intensitas	Jumlah curah hujan yang turun dalam satuan waktu tertentu	Integer
tanggal	Tanggal pengamatan	Date
jam	Jam pengamatan	Time
lokasi	Lokasi pengamatan	String

Pada Tabel 2. menjelaskan properti-properti yang terdapat dalam file JSON yang digunakan sebagai sumber data. Properti pertama adalah intensitas yang merepresentasikan jumlah curah hujan yang tercatat dalam satuan waktu tertentu dengan tipe data berupa integer. Properti ini penting untuk menganalisis pola curah hujan pada berbagai kondisi waktu dan lokasi. Properti kedua adalah tanggal, yang menunjukkan waktu pengamatan curah hujan dalam format tanggal. Data ini menggunakan tipe data date untuk memastikan validitas dan kemudahan pengolahan waktu. Selanjutnya, properti jam mencatat waktu spesifik dari pengamatan curah hujan dalam format time, memungkinkan analisis yang lebih mendetail, seperti mengidentifikasi tren harian. Properti terakhir adalah lokasi, yang menunjukkan tempat pengamatan curah hujan dilakukan, dengan tipe data berupa string. Informasi ini berfungsi untuk membandingkan variasi curah hujan di berbagai lokasi, memberikan dimensi geografis pada analisis data. Tabel ini memberikan gambaran lengkap mengenai struktur data yang tersedia dalam file JSON, yang akan digunakan sebagai dasar analisis. Kemudian tahap berikutnya dataset ditransformasi ke dalam bentuk CSV file.

name	intensity	date	date_time
Pintu Air Manggarai	0	01/01/2022	01/01/2022 07.00
Pintu Air Manggarai	0	01/01/2022	01/01/2022 07.00
Pintu Air Manggarai	0	01/01/2022	01/01/2022 07.00
Pintu Air Manggarai	0	01/01/2022	01/01/2022 07.00
Pintu Air Manggarai	0	01/01/2022	01/01/2022 07.00
Pintu Air Manggarai	0	01/01/2022	01/01/2022 07.00
Pintu Air Manggarai	0	01/01/2022	01/01/2022 07.00

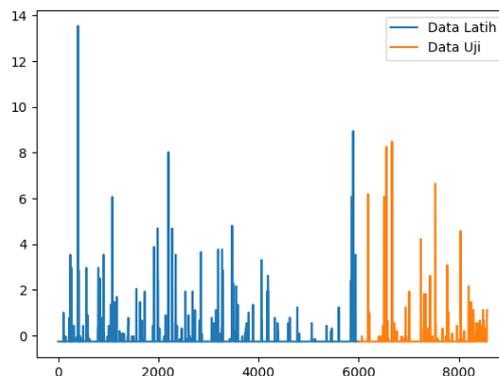
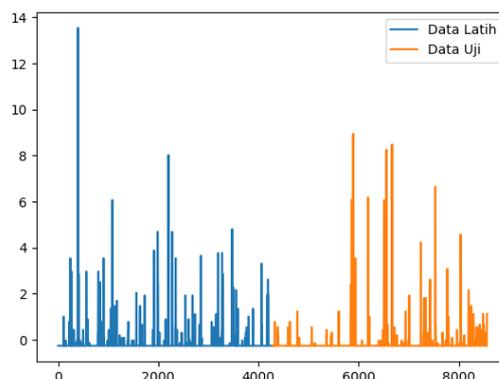
Gambar 3. Struktur CSV File

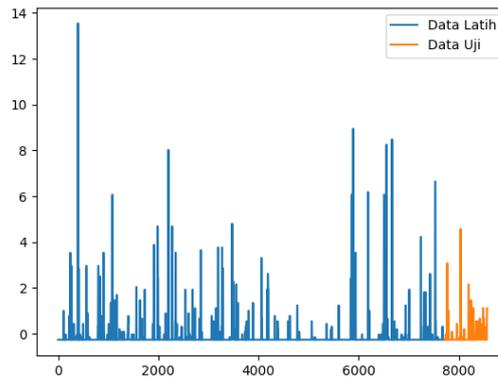
Gambar 3. menunjukkan gambar struktur dari CSV file yang sudah di bersihkan dan hanya empat properties yang digunakan. Kemudian tahap selanjutnya adalah proses normalisasi menggunakan Standar *Scaler*. Normalisasi ini mengubah nilai agar memiliki rata-rata 0 dan standar deviasi 1.

Tabel 3. Pembagian Dataset

Pembagian	Data Latih	Data Uji
1	50%	50%
2	70%	30%
3	90%	10%

Tabel 3. menjelaskan tiga skenario pembagian dataset yang digunakan untuk pelatihan dan pengujian model, yaitu 50:50, 70:30, dan 90:10 seperti pada Gambar 4, 5, dan 6. Pada pembagian pertama, data latih dan data uji masing-masing dibagi secara merata sebesar 50%. Pendekatan ini memastikan bahwa jumlah data yang tersedia untuk melatih model sama dengan jumlah data yang digunakan untuk mengevaluasi performa model, memberikan keseimbangan antara pelatihan dan pengujian. Pembagian kedua, yaitu 70% untuk data latih dan 30% untuk data uji, dirancang untuk memberikan lebih banyak data bagi model untuk mempelajari pola yang ada, sambil tetap menyediakan jumlah data uji yang memadai untuk mengevaluasi hasil prediksi. Pada pembagian ketiga, data latih mengambil porsi terbesar sebesar 90%, sementara hanya 10% data yang digunakan untuk pengujian. Pendekatan ini bertujuan memaksimalkan kemampuan model dalam belajar dari data yang lebih banyak, sehingga dapat menghasilkan prediksi yang lebih akurat, meskipun dengan data uji yang terbatas. Variasi pembagian ini memberikan gambaran mengenai pengaruh proporsi data terhadap kinerja model secara keseluruhan.

Gambar 4. Pembagian *Dataset* 70% Data Latih dan 30% Data UjiGambar 5. Pembagian *Dataset* 50% Data Latih dan 50% Data Uji



Gambar 6. Pembagian Dataset 90% Data Latih dan 10% Data Uji

Berikut potongan source code dalam bahasa Python untuk memodelkan menggunakan algoritma RNN.

```
def modeling(X_latih, y_latih, X_uji, y_uji):
    # Membangun model RNN
    model = Sequential()
    model.add(SimpleRNN(units=50, activation='relu', input_shape=(X_latih.shape[1], 1)))
    model.add(Dense(units=1))

    # Kompilasi dan training model
    model.compile(optimizer=Adam(), loss='mean_squared_error')
    model.fit(X_latih, y_latih, epochs=20, batch_size=32, validation_data=(X_uji, y_uji))

    # Prediksi menggunakan model hasil training
    predictions = model.predict(X_uji)

    # Membalikkan skala prediksi ke nilai asli
    data_asli = scaler_rf.inverse_transform(y_uji)
    data_prediksi = scaler_rf.inverse_transform(predictions)

    # Pengukuran nilai RMSE
    RMSE = model.evaluate(X_uji, y_uji, batch_size=1000)
```

Pada potongan *source code* di atas terdapat inputan berupa X_latih, y_latih, X_uji, y_uji. Empat inputan tersebut mewakili data latih dan data uji dengan pembagian seperti pada Tabel 2. Fungsi modeling bertujuan untuk membangun, melatih, dan mengevaluasi model RNN (Recurrent Neural Network) dengan input data latih dan uji. Model ini terdiri dari satu lapisan RNN dengan 50 unit dan fungsi aktivasi ReLU, diikuti oleh lapisan Dense dengan satu unit output. Model dikompilasi menggunakan optimizer Adam dan fungsi loss mean squared error, serta dilatih selama 20 epoch dengan ukuran batch 32. Selain itu, fungsi ini juga menggunakan data uji untuk validasi selama proses pelatihan.

Setelah pelatihan selesai, model digunakan untuk membuat prediksi terhadap data uji. Hasil prediksi dan data asli dikembalikan ke skala semula menggunakan transformasi balik dari objek scaler, memungkinkan interpretasi dalam konteks nilai aslinya. Untuk mengevaluasi performa, fungsi menghitung nilai RMSE (Root Mean Squared Error) menggunakan data uji, memberikan ukuran kuantitatif terhadap akurasi prediksi model. Proses ini membantu mengidentifikasi sejauh mana model dapat memprediksi dengan baik berdasarkan pola yang telah dipelajari.

Tabel 4. Hasil Pengujian RMSE

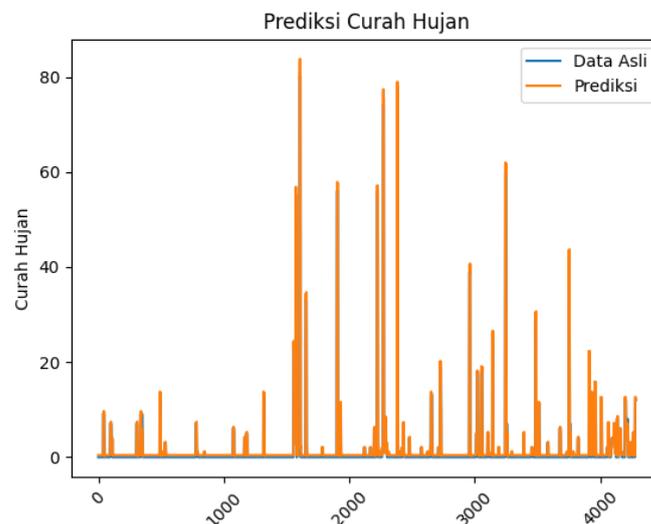
Pembagian	Data Latih	Data Uji	RMSE
1	50%	50%	0.1679
2	70%	30%	0.1962
3	90%	10%	0.0925

Tabel 4. menunjukkan hasil pengujian RMSE (Root Mean Squared Error) dari tiga skenario pembagian dataset yang berbeda, yakni 50:50, 70:30, dan 90:10. Pembagian pertama, yaitu 50% untuk data latih dan 50% untuk data uji, menghasilkan nilai RMSE sebesar 0.1679.

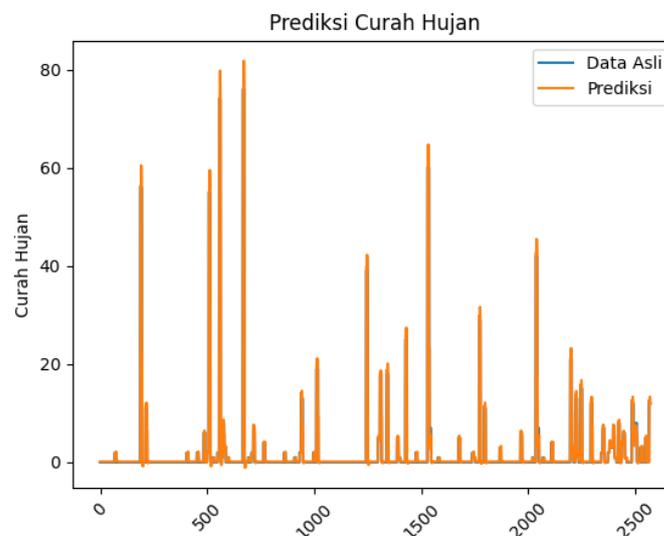
Nilai ini menunjukkan performa model yang cukup baik, meskipun dengan proporsi data latih yang terbagi secara merata dengan data uji. Hasil ini memberikan gambaran bahwa meskipun data latih dan uji memiliki porsi yang sama, model masih dapat membuat prediksi dengan error yang relatif kecil.

Pada pembagian kedua, dengan 70% data latih dan 30% data uji, nilai RMSE meningkat menjadi 0.1962. Meskipun data latih lebih banyak dibandingkan dengan data uji, nilai RMSE yang lebih tinggi ini mengindikasikan bahwa peningkatan jumlah data latih tidak selalu langsung berbanding lurus dengan penurunan error model, terutama ketika proporsi data uji tetap signifikan. Hal ini dapat disebabkan oleh ketidakseimbangan antara kemampuan model dalam mempelajari data latih yang lebih besar dan pengaruh data uji yang lebih terbatas.

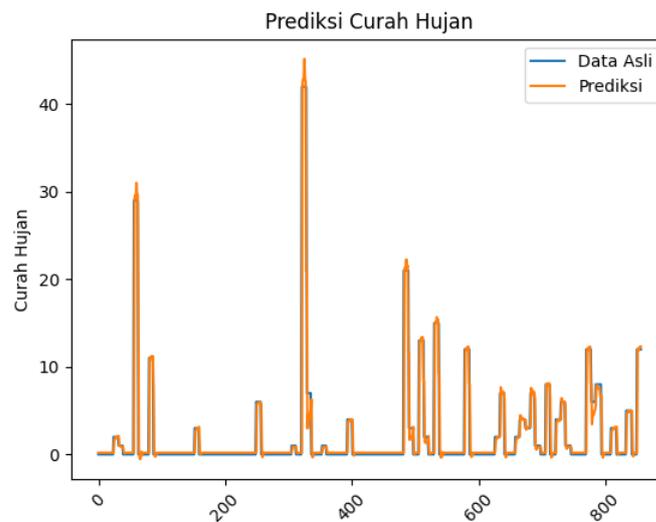
Pada pembagian ketiga, yang menggunakan 90% data latih dan hanya 10% data uji, nilai RMSE mencapai angka yang paling rendah, yaitu 0.0925. Hasil ini menunjukkan bahwa dengan menggunakan sebagian besar data untuk pelatihan, model berhasil menghasilkan prediksi yang lebih akurat, meskipun jumlah data uji yang terbatas. Pembagian 90:10 memberikan model kesempatan untuk belajar lebih banyak dari data latih, yang terbukti meningkatkan kemampuan prediksi secara signifikan. Secara keseluruhan, hasil-hasil ini menunjukkan bahwa pembagian data yang lebih banyak untuk pelatihan dapat meningkatkan performa model, tetapi perlu mempertimbangkan keseimbangan antara data latih dan data uji agar evaluasi performa tetap akurat dan representatif.



Gambar 7. Hasil Perbandingan Data Asli dan Data Prediksi Dataset 50% : 50%



Gambar 8. Hasil Perbandingan Data Asli dan Data Prediksi Dataset 70% : 30%



Gambar 9. Hasil Perbandingan Data Asli dan Data Prediksi Dataset 90% : 10%

Gambar 7, 8 da 9. yang menggambarkan hasil prediksi terhadap data nyata dari tiga pembagian dataset (50:50, 70:30, dan 90:10) menunjukkan bahwa meskipun nilai RMSE untuk ketiga pembagian tidak ada yang nilainya 0, perbedaan antara hasil prediksi dan data real secara visual tidak terlalu mencolok. Hal ini menunjukkan bahwa model, meskipun tidak sempurna, mampu menghasilkan prediksi yang cukup akurat pada ketiga skenario pembagian data. Dengan kata lain, meskipun RMSE untuk pembagian 70:30 dan 50:50 lebih tinggi dibandingkan dengan 90:10, visualisasi data menunjukkan bahwa model masih bisa mengikuti pola umum dari data nyata dengan baik.

Pada pembagian data 90:10, meskipun RMSE terendah tercatat, perbedaan antara hasil prediksi dan data asli tetap terlihat tipis secara visual, menunjukkan bahwa meskipun model dilatih dengan banyak data, prediksi yang dihasilkan tidak sepenuhnya bebas dari kesalahan. Demikian juga, pada pembagian 50:50 dan 70:30, meskipun RMSE lebih tinggi, garis prediksi masih menunjukkan kecocokan yang baik dengan data aktual. Hal ini mengindikasikan bahwa model sudah cukup baik dalam memprediksi nilai, meskipun ada sedikit variasi atau kesalahan, dan perbedaan antara prediksi dan nilai asli tidak terlalu besar, membuat grafik terlihat hampir serupa.

4. Kesimpulan

Dari hasil pengujian yang tercantum dalam Tabel 3, dapat disimpulkan bahwa proporsi pembagian dataset memiliki pengaruh signifikan terhadap performa model dalam menghasilkan prediksi. Pada pembagian 50:50, model mampu menghasilkan prediksi dengan RMSE sebesar 0.1679, yang menunjukkan performa yang cukup baik meskipun data latih dan uji terbagi rata. Pembagian ini memberikan gambaran bahwa model masih dapat belajar dengan efektif meskipun dengan jumlah data uji yang seimbang, namun tidak seakurat pembagian lainnya dengan proporsi data latih lebih besar.

Namun, pada pembagian 70:30, meskipun jumlah data latih lebih banyak, nilai RMSE meningkat menjadi 0.1962. Hal ini menunjukkan bahwa peningkatan jumlah data latih tidak selalu mengarah pada penurunan error yang signifikan, dan dapat dipengaruhi oleh ukuran data uji yang tetap cukup besar. Sebaliknya, pembagian 90:10 menunjukkan performa terbaik dengan RMSE terendah yaitu 0.0925, yang mengindikasikan bahwa dengan jumlah data latih yang sangat besar, model dapat menghasilkan prediksi yang lebih akurat meskipun data uji terbatas. Kesimpulannya, semakin banyak data latih yang digunakan, semakin baik performa model, namun pembagian data yang terlalu terbatas untuk pengujian juga harus dipertimbangkan agar hasil evaluasi tetap valid.

Berdasarkan kesimpulan di atas penelitian selanjutnya dapat mengeksplorasi penggunaan model atau algoritma yang lebih kompleks dan adaptif terhadap perubahan proporsi data latih dan uji. Kemudian juga bisa melakukan optimasi parameter pada algoritma yang digunakan untuk lebih meningkatkan akurasi tingkat prediksinya.

Referensi

- [1] D. Bharath, M. Dolok, and Hugeng, "PERANCANGAN DASHBOARD SEBAGAI PERBANDINGAN VISUALISASI DATA IKLIM BMKG DI JAKARTA," *J. Inf. Technol. Comput. Sci.*, vol. 7, no. 6, 2024.
- [2] D. F. Bagaskara, Mohammad Bisri, and Ussy Andawayanti, "Studi Evaluasi Sumur Resapan di Kelurahan Cipinang Besar, Kecamatan Jatinegara, Kota Jakarta Timur," *J. Teknol. dan Rekayasa Sumber Daya Air*, vol. 4, no. 1, 2024.
- [3] W. Anggraini, A. Jayady, R. K. Natadipura, and H. Tunafiah, "Desain Drainase Efektif dalam Mengatasi Banjir di Ruas Jalan Kebon Pala I Jakarta," *Ikraith Teknol.*, vol. 8, no. 3, 2024.
- [4] F. Azzahrotunnisa, M. I. Mahfud, N. S. Ishak, and R. Kurniawan, "Perbandingan Metode Supervised Machine Learning untuk Prediksi Kelembapan Tanah di Jakarta," *Semin. Nas. Sains Data*, 2024.
- [5] S. A. Saputra and H. Soetanto, "IMPLEMENTASI NAÏVE BAYES UNTUK KLASIFIKASI PREDIKSI SERTA ANALISIS DATA BANJIR DI WILAYAH JAKARTA PUSAT," *Semin. Nas. Mhs. Fak. Teknol. Inf.*, vol. 3, no. 2, 2024.
- [6] R. Al Kiramy, I. Permana, A. Marsal, M. R. Munzir, and Megawati, "Perbandingan Performa Algoritma RNN dan LSTM dalam Prediksi Jumlah Jamaah Umrah pada PT. Hajar Aswad," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 4, no. 4, 2024.
- [7] A. Sujjada, F. Sembiring, and Febriansyah, "Prediksi Harga Forex Menggunakan Algoritma Long Short-Term Memory," *Jnanaloka*, vol. 9, no. 1, 2024.
- [8] S. F. Prasetyo, T. F. Efendi, and Muqorobin, "Implementasi Sistem Prediksi Curah Hujan Dengan Penerapan Jaringan Syaraf Tiruan Berbasis Website," *Jurtikom*, vol. 1, no. 2, pp. 80–96, 2024.
- [9] R. Utami and A. Hidayatullah, "Prediksi Curah Hujan Pada Stasiun BMKG (CITEKO) Menggunakan Metode Backpropogation Neural Network," *J. Ilm. Elektron. DAN Komput.*, vol. 17, no. 1, pp. 251–255, 2024.
- [10] R. Aprianto, P. A. D. Puspitasari, S. Fitriyanto, and ..., "Analisis Potensi Bencana Banjir Berdasarkan Hasil Prediksi Curah Hujan di Kabupaten Sumbawa," *Titian Ilmu J. ...*, vol. 16, no. 2, pp. 124–133, 2024, [Online]. Available: <http://journal.unuha.ac.id/index.php/JTI/article/view/3436%0Ahttp://journal.unuha.ac.id/index.php/JTI/article/download/3436/967>
- [11] R. A. Rahman, P. Risma, Y. Oktarina, and H. M. Yudha, "Prediksi Temperatur Lingkungan dengan Recurrent Neural Network Menggunakan Data Historis Iradiasi Matahari," *J. Appl. SMART Electr. Netw. Syst. (Jasen.)*, vol. 5, no. 1, pp. 16–21, 2024.
- [12] T. Yuhono, C. R. Hassolthine, and R. Sahara, "Prediksi Harga Steel Hot-Rolled (Hrc) Dengan Model Recurement Neural Network (Rnn) Kata kunci: Artificial Neural Network , Recurrent Neural Network, Root Mean Square Propagation, Steel Hot-rolled Coil," *JEKIN (Jurnal Tek. Inform.)*, vol. 4, no. 1, pp. 1–9, 2024.
- [13] A. Santoso, A. Irma Purnamasari, and Irfan Ali, "Prediksi Harga Beras Menggunakan Metode Recurrent Neural Network Dan Long Short-Term Memory," *PROSISKO J. Pengemb. Ris. dan Obs. Sist. Komput.*, vol. 11, no. 1, pp. 128–136, 2024, doi: 10.30656/prosisko.v11i1.7921.
- [14] R. W. Arifin, M. Y. Putra, and D. I. Putri, "Comparison of Latent Semantic Analysis (LSA) and Doc2Vec Algorithms of Thesis Similarity Detection," *PIKSEL*, vol. 12, no. 2, 2024.
- [15] D. Larose and C. Larose, *Data Mining and Predictive Analytics*. 2015. doi: 10.1016/b978-0-12-800229-2.00003-1.
- [16] S. Handayani, Taslim, and D. Toresa, "Convolutional Neural Network – Long Short Term Memory Untuk Prediksi Harga Emas Indonesia," *Indones. J. Comput. Sci.*, vol. 11, no. 1, pp. 566–576, 2022.