Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK) DOI: 10.25126/jtiik.938718
Vol. 11, No. 3, Juni 2024, hlm. 637-646 p-ISSN: 2355-7699
Akreditasi KEMENRISTEKDIKTI, No. 36/E/KPT/2019 e-ISSN: 2528-6579

IDENTIFIKASI DINI CURAH HUJAN BERPOTENSI BANJIR MENGGUNAKAN ALGORITMA *LONG SHORT-TERM MEMORY* (LSTM) DAN *ISOLATION FOREST* STUDI KASUS WILAYAH SEMARANG

Ahmad Wijayanto*[1], Aris Sugiharto[2] , Rukun Santoso[3]
[1,2,3] Universitas Diponegoro, Semarang
Email: [1]ahyat2011@gmail.com, [2]aris.sugiharto@live.undip.ac.id, [3]rukunsantoso@lecturer.undip.ac.id
[*]Penulis Korespondensi

(Naskah masuk: 09 Februari 2024, diterima untuk diterbitkan: 11 Juni 2024)

Abstrak

Curah hujan yang tinggi merupakan faktor utama yang dapat mengakibatkan banjir di suatu daerah. Pola curah hujan yang semakin tidak teratur dan peningkatan curah hujan ekstrem membuat pengendalian banjir semakin sulit. Identifikasi dini diperlukan untuk memahami peran curah hujan dalam manajemen sumber daya air dan perancangan infrastruktur air yang tangguh untuk daerah rawan banjir. Dengan keterbatasan data dan parameter input tunggal, model yang diusulkan menghadapi tantangan dalam *forecasting* pola curah hujan jangka panjang dan generalisasi data. Studi ini memproses data curah hujan BMKG untuk menghasilkan *forecasting* menggunakan *Long Short-Term Memory* (LSTM) berdasarkan pola data series dan hubungan jangka panjang. Algoritma *Isolation Forest* kemudian digunakan untuk mengidentifikasi secara otomatis curah hujan dengan potensi banjir. Probabilitas curah hujan tinggi diidentifikasi untuk menghitung ketahanan infrastruktur air dan menetapkan standar yang sesuai untuk daerah beriklim hujan dan rawan banjir. Prediksi LSTM dievaluasi menggunakan *Mean Square Error* (terbaik 19,11) dan *Root Mean Square Error* (terbaik 4,37) sebelum dilakukan *forecasting* jangka panjang. Model yang diusulkan bertujuan untuk membantu pemangku kepentingan secara cepat mengidentifikasi probabilitas curah hujan tinggi jangka panjang, khususnya di daerah Semarang.

Kata kunci: LSTM *Forecasting*, *Isolation forest*, curah hujan, deret waktu.

EARLY IDENTIFICATION OF RAINFALL WITH FLOOD POTENTIAL USING LONG SHORT-TERM MEMORY (LSTM) AND ISOLATION FOREST ALGORITHMS CASE STUDY OF SEMARANG AREA

Abstract

High rainfall is a key factor causing floods in an area. Increasingly irregular rainfall patterns and rising extreme rainfall make it more challenging to control floods. Early identification is needed to understand rainfall's role in water resource management and designing resilient water infrastructure for flood-prone areas. With limited data and single input parameters, the proposed model faces challenges in long-term rainfall pattern forecasting and data generalization. This study processes BMKG rainfall data to generate forecasts using Long Short-Term Memory (LSTM) based on data series patterns and long-term relationships. The Isolation Forest algorithm is then used to automatically identify rainfall with flood potential. The probability of high rainfall is identified to calculate water infrastructure resilience and set appropriate standards for rainy, flood-prone areas. LSTM predictions are evaluated using Mean Square Error (best 19.11) and Root Mean Square Error (best 4.37) before conducting long-term forecasting. The proposed model aims to help stakeholders quickly identify the probability of long-term high rainfall, particularly in the Semarang area.

Keywords: LSTM *Forecasting, Isolation Forest, rainfall, time series*

1. PENDAHULUAN

Banjir kritis terjadi saat curah hujan tinggi dan pasang surut air laut tinggi (Junaidi et al., 2018) berdampak kerusakan terhadap rumah, bisnis, infrastruktur, hingga menyebabkan korban jiwa (Marzukhi et al., 2018). Perubahan iklim dan pola cuaca yang tidak stabil semakin memperumit tantangan perencanaan infrastruktur bangunan air. Fluktuasi curah hujan yang lebih intens dan frekuensi curah hujan yang tidak terduga dapat meningkatkan risiko banjir dan mengganggu distribusi sumber daya air yang efisien (Miller & Hutchins, 2017).

Studi analisis mendapatkan parameter distribusi teoritis untuk curah hujan melibatkan beberapa

tantangan. Salah satunya adalah perdebatan mengenai model yang paling tepat untuk menggambarkan curah hujan. Keterbatasan dalam aplikabilitas model dan ketidakakuratan dalam mencari probabilitas curah hujan jangka panjang menjadi faktor yang sulit diatasi. Beberapa metode yang digunakan, seperti MLE, MoM, dan POME, memiliki kelebihan dan keterbatasan masing-masing. Pendekatan lain mencakup penggunaan distribusi *gamma*, *log-normal*, eksponensial, *Weibull*, dan distribusi probabilitas hibrida. Namun, belum ada konsensus tentang model distribusi teoritis yang paling sesuai untuk curah hujan. Masalah distribusi ini, perlu penelitian lebih lanjut untuk mengembangkan pendekatan yang lebih baik dalam menentukan model distribusi secara akurat dengan kriteria evaluasi yang ketat dan ukuran sampel yang flexible (Shen & Xiang, 2023). Oleh karena itu, diperlukan pendekatan yang lebih canggih dan terperinci dalam menghasilkan *forecasting* distribusi curah hujan, terutama curah hujan tinggi dengan deteksi anomali yang dapat menyebabkan situasi darurat.

Penelitian ini, menggunakan dua algoritma yaitu *Long Short Term Memory* (LSTM) dan *Isolation Forest*. Algoritma LSTM merupakan jenis *Recurrent Neural Network* (RNN) yang dapat memodelkan dan memprediksi data sekuensial, seperti data curah hujan. LSTM mampu mengenali pola dan hubungan jangka panjang dalam data serta mempertahankan informasi penting dalam memori jangka panjang (Pyo et al., 2023). Dalam penelitian ini, LSTM digunakan untuk memprediksi untuk menghasilkan *forecasting* curah hujan masa depan berdasarkan data historis curah hujan. Selanjutnya, algoritma *Isolation Forest* digunakan untuk mengidentifikasi *anomaly* (Liu et al., 2008) pada data curah hujan yang berpotensi menyebabkan banjir.

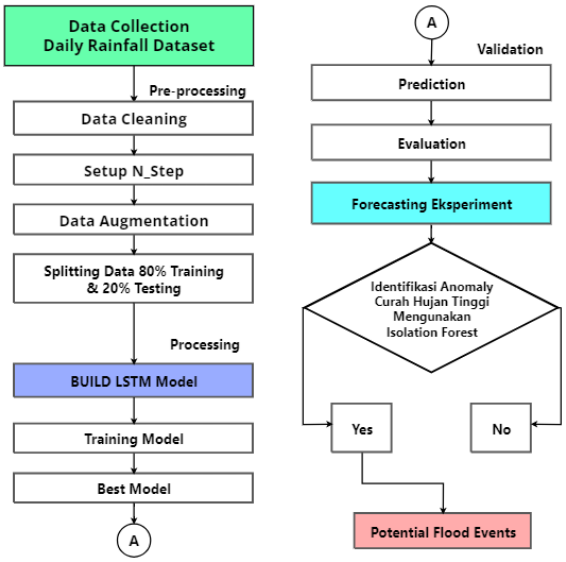
Penggunaan LSTM dan *Isolation Forest* diharapkan secara efektif dan cepat digunakan untuk manajemen sumber daya air dan perancangan infrastruktur air yang tangguh untuk daerah rawan banjir, sehingga infrastruktur bangunan air pada daerah hujan dapat dirancang dan direncanakan dengan standard yang diperlukan untuk mengatasi kemungkinan banjir dikemudian hari.

2. METODE PENELITIAN

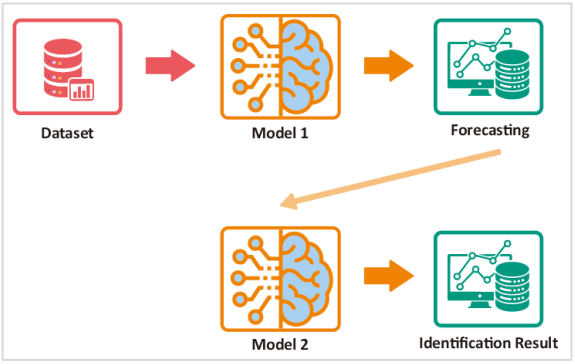
Tahapan dalam penelitian ini merupakan langkah – langkah, bagaimana data di proses hingga menghasilkan keluaran untuk mencapai yang diharapkan dari tujuan dan manfaat penelitian sesuai penjelasan pada Gambar 1 dan Gambar 2.

2.1 Dataset Curah Hujan

Penelitian dimulai dengan mengumpulkan dataset curah hujan. Dataset bersumber dari data BMKG dalam format .csv dengan 1 parameter curah hujan. Data curah hujan adalah hasil pembacaan sensor dari stasiun hujan wilayah Semarang, Jawa Tengah. Tabel 1 merupakan statistik deskriptif dari dataset yang digunakan.



Gambar 1 Tahapan Penelitian



Gambar 2 Proses Implementasi Pada Model

Tabel 1. Statistik Deskriptif

Statistik	Nilai
Jumlah data	4.030
Rata-rata	6,6 (mm)
Standar deviasi	14,83 (mm)
Nilai minimum	0 (mm)
Kuartil pertama (Q1)	0 (mm)
Median	0 (mm)
Kuartil ketiga (Q3)	6 (mm)
Nilai maksimum	171 (mm)
T-Statistik	28,24
P-Value	2.56

Penelitian ini, menggunakan dataset curah hujan selama 11 tahun dengan total 4.030 data *time series*. Data tersebut dibagi menjadi rasio 80:20, di mana 80% dari data digunakan sebagai data *training* (*train_x*, *train_y*) dan 20% digunakan sebagai data *validation* (*val_x*, *val_y*).

2.2 *Preprocessing* Data

Tahapan *Pre-processing* data curah hujan harian melibatkan beberapa langkah. Tahapan *preprocessing* sangat penting untuk mempersiapkan data sebelum digunakan dalam *training* model LSTM. Pertama, data dibaca dari file dengan format .csv dan dilakukan beberapa langkah persiapan

seperti menghapus duplikat, mengatur indeks, dan mengonversi frekuensi menjadi harian. Langkah ini membantu memastikan bahwa data yang digunakan dalam model LSTM bersih dan siap digunakan. Selanjutnya, fitur independen dan dependen dipersiapkan menggunakan fungsi *library* pada bahasa pemrograman *phyton* yaitu *Timeseries Generator* dari *Keras*. Fungsi tersebut memungkinkan data *time series* dibagi menjadi rangkaian waktu dengan panjang yang ditentukan dan membangkitkan pasangan input dan output berdasarkan format yang diterima model LSTM.

Tabel 1 Data Sebelum Augmentasi

Before Augmentation		
No.	**Date**	**Daily Rainfall (mm)**
1	01/01/2010	5,22
2	02/01/2010	2,13
3		

Tabel 2 Data Setelah Augmentasi

After Augmentation			
No.	**Date**	**Augmented_X**	**Augmented_y**
1	01/01/2010	[5,37877836, 5,19090844, 4,97448175, ..., 5,01443837]	5,2
2	02/01/2010	[[[2,12331855], [2,13660104], [2,23849684], ... ,[2,26118932]]]	2,1
3	…	…	…

Teknik augmentasi yang ditunjukan pada Tabel 2 dan Tabel 3 dilakukan dengan menambahkan gangguan acak ke setiap sampel data sehingga dari 4.030 data menjadi 20.100 data. Ini membantu menciptakan variasi dalam data dan mencegah *overfitting*. Dalam kode yang diberikan, gangguan acak ditambahkan menggunakan distribusi normal dengan *mean* 0 dan standard deviasi 0.1. Faktor augmentasi, yang ditentukan oleh *augment_factor*, menentukan seberapa banyak setiap sampel diganggu untuk menciptakan variasi yang lebih besar. Selanjutnya, data *augmented* diubah bentuknya dari [*samples*, *timesteps*] menjadi [*samples*, *timesteps*, *features*]. Model LSTM menerima input dalam format tiga dimensi yang mencakup jumlah sampel, jumlah waktu, dan jumlah fitur. Dalam kasus ini, jumlah fitur ditentukan sebagai 1 karena hanya ada satu fitur yaitu curah hujan.

Tahapan selanjutnya, data dibagi menjadi set *training* dan *validation* menggunakan fungsi *train_test_split* dari *library scikit-learn*. Pembagian data memungkinkan validasi kinerja model pada set data *validation* yang terpisah. Data *training* dan *validation* disimpan dalam variabel (*x_train_aug, x_val_aug, y_train_aug, y_val_aug*) yang sesuai untuk digunakan dalam *training* dan *validation* model LSTM. Dengan melakukan tahapan *preprocessing* ini, data telah dipersiapkan dengan baik sehingga dapat digunakan dalam *training* dan *validation* model LSTM dengan harapan mencapai hasil yang lebih baik. Kemudian dalam proses *input sequence*, dapat digunakan untuk mengatur seberapa lama melihat kebelakang dan melakukan pergeseran data *series* yang digunakan yang selanjutnya disebut *N_Step*.

2.3 Implementasi Model LSTM

Implementasi model merupakan suatu proses penentuan nilai melalui tahapan eksperimen sehingga menghasilkan tingkat akurasi terbaik dari beberapa model. Jumlah unit LSTM dipilih sebesar 50 untuk menghindari *overfitting*. Jika model memiliki terlalu banyak unit LSTM, maka model cenderung akan "menghafal" data *training* dan tidak bisa melakukan generalisasi dengan baik pada data baru (Sennhauser & Berwick, 2018). Aktivasi *'tanh'* dipilih sebagai fungsi aktivasi pada layer LSTM karena menghasilkan rentang keluaran yang lebih luas -1 ke 1 dibandingkan dengan fungsi aktivasi *sigmoid*. Dalam kasus prediksi curah hujan, rentang nilai yang luas diperlukan untuk memodelkan variasi yang signifikan dalam curah hujan yang dapat terjadi. *Dropout* layer ditambahkan setelah layer LSTM dan *Dense* untuk mengurangi *overfitting*.

Tabel 3 Tabel Hyperparameter Terbaik

Hyperparameter	**Nilai**
lstm(LSTM)	50
Activation_1	Tanh
Dropout_1 (Dropout)	0,3
Dense	50
Activation_2	Relu
Dropout_2 (Dropout)	0,3
Dense_2	1
Optimizer	Adam
Learning rate	0,001
Epoch	Early Stoping (Optional)

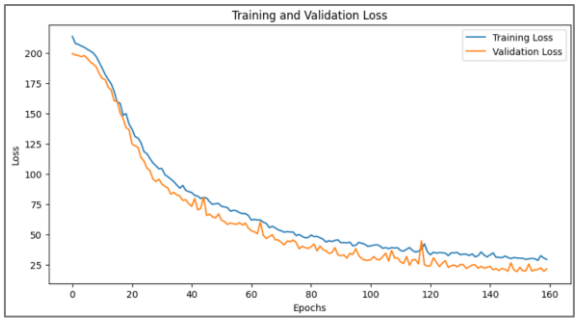
Dropout layer secara acak menonaktifkan beberapa unit selama *training*, sehingga memaksa model untuk belajar secara lebih umum dan mengurangi ketergantungan yang kuat pada unit-unit tertentu. Dalam kasus ini, *dropout rate* sebesar 0.3 dipilih untuk menonaktifkan sebesar 30% unit pada setiap iterasi *training*. Regularisasi L1L2 diterapkan pada layer *Dense* untuk mencegah *overfitting* dan meningkatkan generalisasi. Regularisasi L1L2 memberikan penalti pada bobot model, mendorong bobot untuk tetap kecil dan menghindari nilai yang terlalu ekstrem. Ini membantu mengontrol kompleksitas model dan mengurangi *overfitting*.

Aktivasi ReLU (*Rectified Linear Unit*) pada layer dense ke-2 memberikan non-linearitas pada output dari layer sebelumnya. Fungsi *'relu'* menghasilkan output yang positif jika inputnya positif, sementara outputnya menjadi nol jika inputnya negatif. Hal ini membantu model untuk mempelajari hubungan yang kompleks dalam data atau perubahan tajam dalam data (Nwankpa et al., 2018). Nilai *learning rate* 0.001 pada *optimizer* adam dipilih berdasarkan eksperimen dan dapat disesuaikan tergantung pada dataset dan pengaturan *training* yang spesifik. *learning rate* yang terlalu besar menyebabkan model melompati minimum lokal yang seharusnya menjadi titik optimal dalam

proses optimisasi, sedangkan *learning rate* yang terlalu kecil dapat menyebabkan *training* menjadi lambat dan sulit mencapai hasil yang baik. *early stopping* digunakan untuk menghentikan proses *training* jika tidak ada peningkatan dalam *loss* pada set *validation* setelah beberapa *epoch* seperti ditunjukan pada Gambar 5. Hal ini membantu mencegah proses *training* berlanjut terlalu lama dan menghindari *overfitting* (Bai et al., 2021). *Patience* (kesabaran) sebesar 10 berarti *training* akan dihentikan jika tidak ada peningkatan dalam *loss* pada set *validation* selama 10 *epoch* berturut-turut.

Forecasting curah hujan dilakukan dengan menggunakan model LSTM yang telah dilatih. Dalam tahap ini, data curah hujan hingga saat ini yang tersedia digunakan sebagai *input* untuk model LSTM. Model akan menghasilkan prediksi curah hujan di masa depan berdasarkan pola dan tren yang dipelajari selama *training*. Prediksi ini dapat digunakan untuk menginformasikan perkiraan curah hujan di suatu wilayah pada periode waktu tertentu. Dalam hal ini, distribusi curah hujan tinggi menjadi topik penting untuk keberlanjutan dalam penelitian.

Hasil *loss function* pada Gambar 5 menunjukkan bahwa model telah mencapai penyesuaian yang baik terhadap data dan tidak menunjukkan tanda-tanda *overfitting* dikarenakan penggunaan *early stoping*. Hasil penurunan awal pada *loss training* maupun *validation* dimulai relatif tinggi dan menurun dengan cepat selama *epoch* awal. Hal ini menunjukkan bahwa model sedang belajar secara efektif dan mengurangi kesalahannya saat *training*.



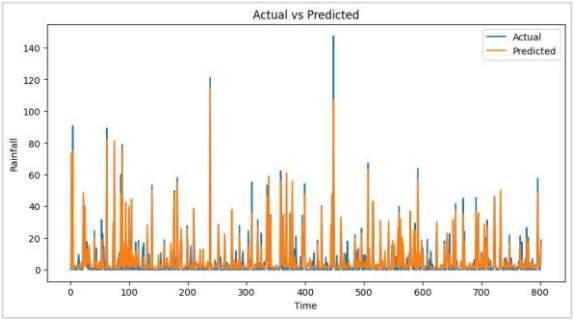
Gambar 1 Grafik *loss function training* dan *validation*

Grafik loss *training* dan *validation* konvergen ke nilai yang relatif stabil setelah sejumlah *epoch* tertentu. Hal ini menunjukkan bahwa model telah mencapai kinerja optimalnya dan *training* lebih lanjut mungkin tidak menghasilkan peningkatan yang signifikan. Kesenjangan antara kurva loss *training* dan *validation* relatif kecil, yang merupakan pertanda baik. Kesenjangan yang besar dapat mengindikasikan *overfitting* (Ying, 2019), penggunaan pemberhentian dini *early stoping* telah secara efektif mencegah *overfitting* dengan menghentikan proses *training* sebelum *loss validation* mulai meningkat. Hal ini terlihat dari kenyataan bahwa *loss validation* tampaknya telah mencapai titik jenuh sebelum mencapai nilai minimumnya.

2.4 Hasil *Validation* LSTM

Model LSTM digunakan untuk melakukan prediksi pada data *validation* setelah melalui proses *training* menggunakan data *training*. Setelah melatih model dengan data *training*, peneliti menggunakan model tersebut untuk membuat prediksi pada data *validation*. Setelah mendapatkan hasil prediksi, peneliti dapat mengukur tingkat akurasi menggunakan metrik *Mean Squared Error* (MSE) dan *Root Mean Squared Error* (RMSE).

a) *Mean Squared Error* (MSE) adalah metrik yang mengukur rata-rata kuadrat dari selisih antara nilai aktual (dalam hal ini *val_y*) dan nilai prediksi (*val_predictions*). MSE akan memberikan nilai kesalahan prediksi dalam bentuk kuadrat, dan semakin rendah nilai MSE, semakin baik kinerja model.
b) *Root Mean Squared Error* (RMSE) adalah akar kuadrat dari MSE. RMSE memberikan kesalahan prediksi dalam satuan yang sama dengan variabel target. RMSE membantu dalam memberikan gambaran yang lebih intuitif tentang tingkat kesalahan prediksi, dan semakin rendah nilai RMSE, semakin baik kinerja model.



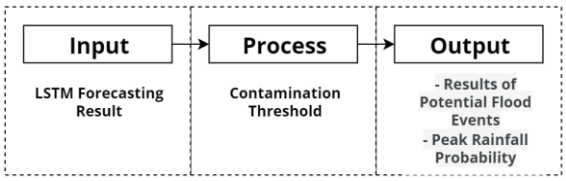
Gambar 2 Grafik Validasi Hasil Prediksi Data Training

Pada Gambar 6, dapat dilihat bahwa data curah hujan aktual dan curah hujan yang diprediksi memiliki pola yang sama. Artinya, model dapat memprediksi pola curah hujan dengan cukup baik. Hal ini menunjukan, LSTM dapat mengenali dan memahami pola dan ketergantungan sekuensial dalam data untuk menghasilkan prediksi (Qian, 2022).

2.5 Proses Identifikasi Menggunakan I-Forest

Isolation Forest digunakan untuk mendeteksi *outlier* dalam data. Dalam konteks ini, *outlier* mengacu pada nilai-nilai yang dianggap tidak biasa atau tidak sesuai dengan pola umum (Karczmarek et al., 2020) dalam data curah hujan. Konsep "kontaminasi" dalam *Isolation Forest* mengacu pada proporsi *outlier* yang membangun pohon biner untuk setiap pecahan data (Fang et al., 2022). Proses ini berulang secara rekursif hingga semua data terisolasi dalam *leaf node*. Nilai *anomaly score* kemudian dihitung berdasarkan rata-rata panjang jalur yang diperlukan untuk mencapai setiap data dalam pohon. Semakin pendek jalur yang diperlukan, semakin

tinggi skor *anomaly* dan semakin mungkin data tersebut dianggap sebagai *outlier*.



Gambar 3 Proses Identifikasi *Isolation Forest*

Nilai kontaminasi dalam *Isolation Forest* mengendalikan proporsi *outlier* yang diharapkan dalam data. Penelitian ini menggunakan tingkat kontaminasi sebesar 0,05 (5%), maka 5% dari data diharapkan sebagai *outlier* hal ini mengacu pada penelitian terdahulu oleh Aldrich & Liu, (2024).

3. LANDASAN KEPUSTAKAAN

3.1 Penelitian Terdahulu

Penelitian sebelumnya yang dilakukan Poornima & Pushpalatha, Tahun (2019) menghasilkan pengembangan dan perbandingan berbagai model prediksi curah hujan. Hasil penelitian menunjukkan bahwa model *Intensified* LSTM mampu memberikan hasil prediksi terbaik dengan akurasi dan RMSE yang lebih baik disbanding metode lain. Dalam kasus ini, *softmax* digunakan untuk menghasilkan distribusi probabilitas dari keluaran model. Model ini memiliki kemampuan untuk mengelola *dataset* yang besar dan mengatasi masalah gradien yang menghilang. Selain itu, penelitian ini juga menjelaskan pentingnya peran fungsi aktivasi dan penggunaan LSTM dalam jaringan saraf berulang untuk prediksi curah hujan. Tantangan matematis dalam pembelajaran dependensi jangka panjang dalam jaringan saraf berulang juga dikaji dan diusulkanlah metode *Intensified* LSTM sebagai solusi.

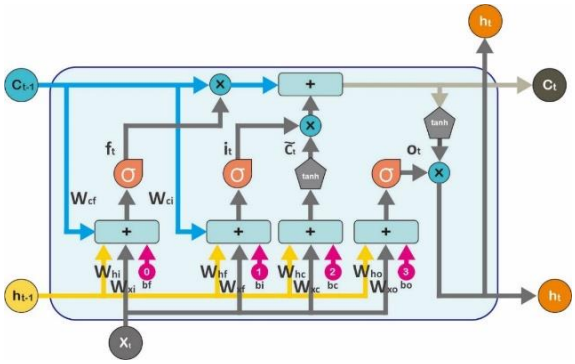
Penelitian sebelumnya dari Septian & Kartini, (2023) LSTM digunakan untuk memprediksi jangka pendek menghasilkan *forecasting* dalam waktu 1 hari dengan hasil *validation* cukup baik dibanding metode lain seperti *Convolutional Neural Network* (CNN), meskipun hasil *Deep Neural Investigation Network* (DNIN) lebih baik, namun masih bisa mengimbangi untuk hasilnya. Kemudian pada penelitian berikutnya oleh Carnegie & Chairani, (2023) dengan melakukan perbandingan antara LSTM dan GRU dalam prediksi curah hujan, LSTM berhasil mencapai akurasi yang lebih baik dibandingkan GRU dalam memprediksi curah hujan.

Penelitian yang dilakukan (Wang & Qi, 2023) menyebutkan *gaussian noise* adalah jenis *noise* yang mengikuti distribusi normal. Teknik ini digunakan untuk membuat variasi acak pada data untuk keperluan eksperimen. Pada penelitian ini, penambahan *gaussian noise* pada tahapan augmentasi dan deteksi *anomaly* akan memberikan referensi lain terkait pengembangan LSTM sehingga LSTM mampu menghasilkan distribusi *forecasting* dalam jangka panjang dan curah hujan tinggi dapat teridentifikasi secara cepat menggunakan *isolation forest*. Hasil penelitian sebelumnya yang berfokus pada akurasi menjadi dasar penelitian ini dilakukan, karena LSTM telah diuji dalam melakukan prediksi. Setelah dilakukan prediksi, nilai *forecasting* menjadi tantangan selanjutnya untuk menghasilkan generalisasi data baru jangka panjang untuk menjawab tantangan distribusi model seperti yang di sebutkan oleh Shen & Xiang, tahun 2023.

3.2 *Long Short-Term Memory* (LSTM)

Algoritma *Long Short Term Memory* (LSTM) diperkenalkan sebagai tipe khusus dari RNNs yang dapat mengingat informasi dalam jangka waktu yang lebih lama (Goodfellow et al., 2016). Pendekatan *lookback* dalam penelitian ini mencakup penggunaan informasi dari langkah waktu sebelumnya dalam perhitungan saat ini. Dengan menggunakan mekanisme *lookback*, jaringan dapat mengatasi tantangan yang melibatkan data berurutan dengan mempertahankan dan memanfaatkan informasi kontekstual dari langkah waktu sebelumnya (Hochreiter & Schmidhuber, 1997).

Persamaan 1-5 secara umum memberikan gambaran langkah-langkah penting dalam model LSTM. Pada setiap langkah waktu, *gate input* (i_t) mengontrol seberapa banyak informasi baru yang akan masuk ke dalam *memory cell*, sedangkan *gate forget* (f_t) mengontrol seberapa banyak informasi yang akan dihapus dari *memory cell* sebelumnya. Kemudian, *cell state* (c_t) menggabungkan informasi dari langkah waktu sebelumnya dengan informasi baru yang masuk, sehingga *memory cell* dapat menyimpan informasi yang relevan dari masa lalu. Kemudian, *gate output* (o_t) mengontrol seberapa banyak informasi yang akan dikirimkan dari *memory cell* ke output pada langkah waktu tersebut. Selengkapnya ditunjukan pada Gambar 8.



Gambar 8 Ilustrasi Algoritma LSTM
Sumber : Jurnal (Shiri M Farhad et al., 2023)

$$f_t = \sigma(x_t W_{xf} + h_{t-1} W_{hf} + c_{t-1} W_{cf} b_f) \quad (1)$$
$$i_t = \sigma(x_t W_{xi} + h_{t-1} W_{hi} + c_{t-1} W_{ci} b_i) \quad (2)$$
$$o_t = \sigma(x_t W_{xo} + h_{t-1} W_{ho} + c_t W_{co} + b_o) \quad (3)$$
$$c_t = f_t c_{t-1} + i_t \, tanh(x_t W_{xc} + h_{t-1} W_{hc} + b_c) \quad (4)$$
$$h_t = o_t \, tanh(c_t) \quad (5)$$

Dengan menggunakan persamaan 1-5, model LSTM dapat mengatur aliran informasi dengan cermat dan mengingat informasi yang penting dalam rangkaian waktu yang panjang dan melakukan prediksi waktu yang akan datang.

$$f(x) = \frac{1}{\sqrt{2\pi\sigma}}\, e\, \frac{-(x-\mu)^2}{2\sigma^2}\mu_n \quad (6)$$

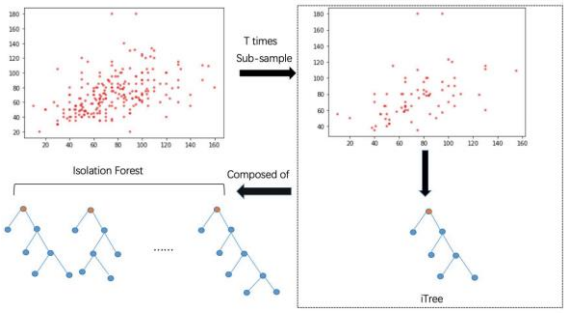
$$noise = \mu_n + \sigma_n * random\ number \quad (7)$$

$$noisy\ data = original\ data + noise \quad (8)$$

Dengan persamaan 6-7, *Gaussian noise* ditambahkan pada proses augmentasi data. Data asli diubah dengan menggunakan generator angka acak yang mengikuti distribusi normal standar dengan rata-rata 0 dan variasi yang berbeda. Melalui transformasi linear, *mean* dan standar deviasi dari angka-angka tersebut disesuaikan. Hasil transformasi tersebut kemudian ditambahkan ke data asli untuk menghasilkan data yang diberi *Gaussian Noise* (Wang & Qi, 2023).

3.3 Isolation Forest (I-Forest)

Isolation Forest merupakan algoritma untuk pengelompokan data dengan mengisolasi *outlier* yang jarang muncul ke dalam kluster yang berbeda dari data normal. Dalam hal ini, *outlier* dianggap sebagai data yang jarang dan berbeda dari mayoritas data normal (Stanton et al., 2012). Dengan melihat distribusi berdasarkan nilai atau *scoring*, dapat memberikan dampak positif secara keseluruhan, Hal ini mengkonfirmasi bahwa memiliki sedikit sensitivitas terhadap *outlier* di antara estimator dapat membantu dalam mendeteksi *anomaly*, sementara terlalu banyak sensitivitas tidak memberikan manfaat yang signifikan (Dhouib et al., 2023) untuk selengkapnya ditunjukan pada Gambar 9.



Gambar 9 Pemisahan kumpulan data untuk mengisolasi *anomaly* dari data normal
Sumber : Jurnal (Chen et al., 2023)

$$\boldsymbol{c(n) = 2H(n-1) - \frac{2(n-1)}{n}} \quad (9)$$

$$\boldsymbol{s\,(x,n) = 2^{-\frac{E\,(h(x))}{c(n)}}} \quad (10)$$

Persamaan 9 dalam algoritma *Isolation Forest* digunakan untuk menghitung faktor skala yang digunakan dalam normalisasi tinggi pohon isolasi. Fungsi *H(n-1)* adalah fungsi harmonik dengan argumen *(n-1)*, yang digunakan untuk menghitung konstanta normalisasi. Faktor koreksi *2(n-1)/n* digunakan untuk memperhitungkan kombinasi tanpa urutan dalam perhitungan.

Persamaan 10 digunakan untuk menghitung skor *anomaly* (*anomaly score*) dari titik data x dalam *dataset* dengan jumlah data n. Skor *anomaly s(x, n)* dihitung berdasarkan tinggi pohon *h(x)* yang diperlukan untuk mengisolasi titik data x dalam pohon isolasi. Faktor skala *c(n)* digunakan untuk normalisasi tinggi pohon *h(x)* berdasarkan jumlah data n. Panjang jalur yang diharapkan *E(h(x))* merupakan rata-rata tinggi pohon *h(x)* dari semua pohon isolasi dalam hutan. Dengan menggunakan persamaan 9-10, *Isolation Forest* dapat menghitung skor *anomaly* untuk setiap titik data dalam *dataset*. Skor *anomaly* tinggi menunjukkan kemungkinan adanya *anomaly*, sedangkan skor yang rendah mengindikasikan data yang lebih normal.

Identifikasi ini penting untuk dilakukan karena efektif dalam mendeteksi secara dini kejadian yang tidak biasa atau tidak diharapkan seperti distribusi curah hujan ekstrem dalam waktu tertentu. Hal ini memungkinan untuk melakukan perencanaan lebih rinci mengenai perencanaan dan mengatasi masalah pada infrastruktur dan lahan dengan mengambil tindakan yang tepat untuk mengatasi situasi sebelum dampaknya menjadi lebih serius.

4. HASIL DAN PEMBAHASAN

Hasil percobaan yang telah dilakukan pada penelitian ini, ditunjukan pada Tabel 5. Penelitian sebelumnya yang dilakukan oleh Carnegie & Chairani, (2023) serta Hendra dkk., (2023) dengan menggunakan LSTM biasa menghasilkan MSE tidak kurang dari 300 untuk data curah hujan dengan sumber BMKG. Meskipun studi kasus berbeda, data curah hujan cenderung memiliki pola yang sama untuk setiap daerah. Pada penelitian ini, penambahan data yang dilakukan dari yang digunakan dalam penelitian sebelumnya dari 5 Tahun ke 10 Tahun hasil yang didapatkan tidak jauh berbeda dalam arti masih dikategorikan terlalu rendah untuk akurasinya yaitu disekitar 300 untuk nilai MSE dan 20 untuk RMSE. Hal ini mengakibatkan pada hasil *forecasting* dengan hasil ditunjukan pada Gambar 8. Hasil Gambar 8 tersebut menunjukkan tingkat akurasi MSE diatas 200. Penting untuk diketahui, diperlukan nilai akurasi yang tinggi untuk menghasilkan fluktuasi data dan distribusi dalam jangka panjang. Model yang diusulkan, LSTM dilakukan dengan penambahan *Gaussian Noise* dengan hasil pada Tabel 5. Percobaan juga dilakukan dengan perubahan Step yang menghasilkan perubahan pada tingkat akurasi model.

Tabel 5 Hasil Validasi Percobaan Dengan Perubahan *Step*

Step	Eksperiment 1		Eksperiment 2		Eksperiment 3	
	MSE	RMSE	MSE	RMSE	MSE	RMSE
1	216,3	14,70	216,2	14,70	216,2	14,70
3	139,3	11,80	136,2	11,67	145,73	12,07
5	97,06	9,85	96,62	9,82	95,97	9,79
7	59,88	7,73	62,55	7,90	56,98	7,54
10	34,94	5,91	27,73	5,26	30,69	5,54

Step	Eksperiment 1		Eksperiment 2		Eksperiment 3	
	MSE	RMSE	MSE	RMSE	MSE	RMSE
15	19,11	4,37	20,33	4,50	22,99	4,79
20	33,56	5,79	30,04	5,48	35,37	5,94
30	26,87	5,18	27,73	5,26	25,55	5,05
60	45,76	6,76	31,46	5,60	39,14	6,25
365	60,33	7,35	66,88	8,17	67,28	8,20
730	76,32	8,73	82,18	9,06	72,49	8,51

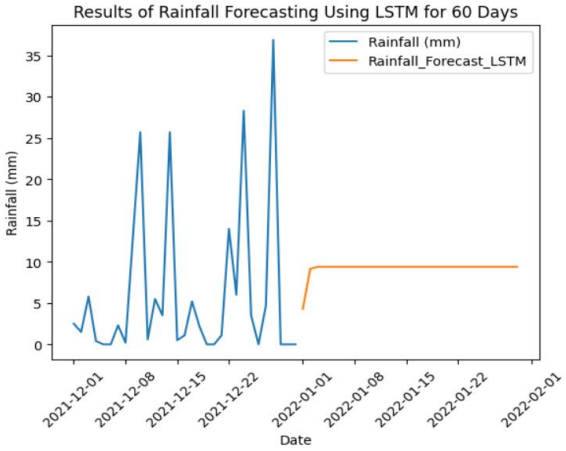
Hasil tingkat akurasi yang dihasilkan pada Tabel 5 menunjukan perlu langkah atau *step* yang tepat yang bisa digunakan dalam melakukan pengaturan parameter input untuk menghasilkan akurasi tinggi yang mampu menjawab kebutuhan *forecasting* dalam jangka panjang. Penggunaan nilai *step* 15, dapat menghasilkan *forecasting* pada Gambar 9. Untuk wilayah Kota Semarang, Hujan lebat dengan intensitas harian 50-100 mm dan sangat lebat >100 mm dari BMKG pada Tabel 6 digunakan sebagai acuan untuk menentukan potensi kejadian banjir berdasarkan nilai intensitas curah hujan.

Tabel 6 Curah Hujan Berpotensi Banjir BMKG

Keadaan curah hujan	Intensitas curah hujan (mm)		Potensi Banjir
	1 Jam	24 Jam	
Hujan sangat ringan	<1	<5	Tidak
Hujan ringan	1-5	5-20	Tidak
Hujan normal	5-20	20-50	Tidak
Hujan lebat	10-20	50-100	Ya
Hujan sangat lebat	>20	>100	Ya

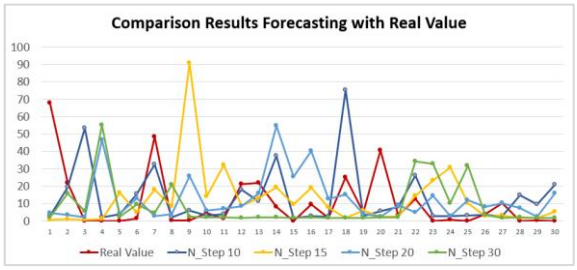
Sumber : BMKG

Penting untuk diketahui bahwa dengan hasil akurasi yang rendah akan berakibat pada hasil *forecasting* sehingga data yang dihasilkan tidak terjadi fluktuasi sehingga bisa dikatakan hanya menghasilkan *forecasting* jangka pendek 1-3 hari. Gambar 8 menunjukan kegagalan model N-Step 1 dalam melakukan fluktuasi *forecasting* selama 60hari.



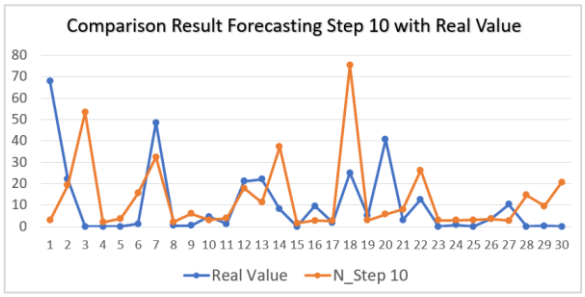
Gambar 4 Hasil *Forecasting* N-Step 1

Untuk memperbaiki hasil akurasi pada model, penelitian ini menggunakan teknik Augmentasi *Gaussian Noise* merujuk pada penelitian Wang & Qi, Tahun (2023) yang terbukti memberikan dampak pada hasil prediksi model. Penelitian tersebut menyebutkan RMSE semakin kecil dengan peningkatan tingkat *noise* (dengan meningkatnya *varians Gaussian*), membuktikan model LSTM mampu mempertahankan performa yang baik dari hasil *loss function* yang dapat mengikuti data pelatihan yang digunakan. Hasil penelitian tersebut menyebutkan nilai akurasi yang akan semakin tinggi dalam situasi penambahan variasi acak pada data. Perlu diketahui bahwa dalam penelitian tersebut menyebutkan variasi acak masih bisa terkontrol dan tidak jauh dengan data aslinya. Kemudian penelitian yang dilakukan oleh Aulia dkk., Tahun (2023) menunjukkan teknik augmentasi yang digunakan juga dapat meningkatkan akurasi model. Pada penelitian ini, kami menggunakan teknik tersebut dengan hasil sebagai berikut.



Gambar 5 Perbandingan *Real Value* dan *Forecasting*

Eksperimen dilakukan dengan mengubah Nilai *Step*, ditampilkan pada Gambar 9, yang menunjukan perbedaan hasil *forecasting* setiap Nilai *Step* yang berbeda, dan yang paling mendekati data asli dengan Nilai *Step* 10 Seperti ditunjukan pada Gambar 10. Hasil eksperimen ini menunjukan nilai dengan akurasi terbaik belum tentu bisa menghasilkan *forecasting* sesuai dengan data asli, namun hasil telah mendekati dengan Nilai *Step* dengan akurasi terbaiknya yaitu *N_Step* 15.

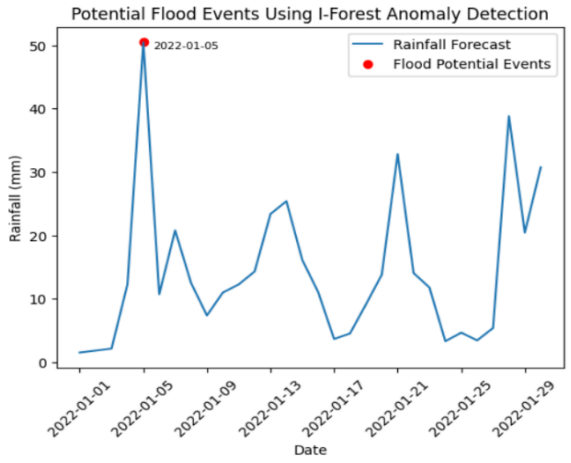


Gambar 6 Perbandingan *Real Value* dan *Forecasting N_Step* 10

Hasil dari *forecasting* selanjutnya dilakukan deteksi *anomaly* menggunakan algoritma *I-forest*, sehingga menghasilkan identifikasi curah hujan tinggi berpotensi banjir. Berdasarkan model tersebut, hasilnya ditampilkan dalam grafik "*Potential Flood Event*" pada *data frame forecast_table*. Data curah hujan pada tanggal yang telah ditandai dianggap sebagai potensi kejadian banjir berdasarkan *Isolation Forest*. Dengan informasi tersebut, pengguna dapat dengan cepat melihat dan mengidentifikasi tanggal-tanggal di mana ada potensi kejadian banjir

berdasarkan data curah hujan yang diprediksi. Ini membantu dalam pemantauan dan pengambilan keputusan terkait mitigasi risiko banjir. Hasil dari pengambilan keputusan selanjutnya dicocokan dengan data bencana kejadian banjir apakah sistem sudah dapat melakukan pengambilan keputusan dengan baik atau belum di tunjukan pada Tabel 8 yang merupakan data aktual bencana yang terjadi pada rentang tanggal 1-6 Januari 2022. Hasil Validasi menunjukan sesuai dari hasil identifikasi serta kejadian bencana.

Hasil menggunakan algoritma *Isolation Forest* seperti ditunjukan pada Gambar 11 dalam deteksi *anomaly* menjadi kebaruan dalam penelitian sehingga data curah hujan lebih mudah disimpulkan untuk selanjutnya dapat digunakan sebagai acuan curah hujan tinggi untuk perancangan bangunan air. Hal ini juga menunjukan pengambilan keputusan secara otomatis menggunakan algoritma yang dapat membantu sistem membuat keputusan dengan cepat.



Gambar 7 Hasil Identifikasi Menggunakan Isolation Forest dalam Waktu 30 Hari

Hasil yang ditunjukan pada Tabel 9 model mampu melakukan *generate* data selama 5 Tahun. Dengan demikian curah hujan puncak dapat diketahui sehingga dapat digunakan sebagai perhitungan dalam desain infrastruktur air. Teknik augmentasi sangat berpengaruh pada hasil *forecasting* karena membantu model dalam generalisasi data untuk melakukan *forecasting* kejadian yang belum terjadi untuk antisipasi curah hujan di masa depan. Hasil dari *N-Step* yang memiliki akurasi yang baik sangat berpengaruh dalam hal ini karena membuat *generate* data lebih bervariasi untuk tinggi puncak maksimalnya. Kemampuan ini penting dalam perhitungan desain infrastruktur air, seperti bendungan, saluran drainase, dan sistem irigasi.

Penelitian ini memberikan alternatif metode yang digunakan untuk identifikasi potensi curah hujan tinggi. Banyak kelemahan yang masih perlu dikembangkan seperti penelitian yang hanya *generate* data sampai 5 Tahun, dengan keterbatasan waktu sehingga masih diperlukan penelitian lebih lanjut mengenai distribusi jangka panjang penggunaan metode ini. Dengan hasil penelitian, potensi curah hujan tinggi dapat diketahui pada bulan-bulan tertentu seperti di tunjukan pada Tabel 9. Hasil penelitian menunjukan potensi banjir di wilayah studi masih tinggi, meskipun menunjukkan tren penurunan. Perlu kewaspadaan dan kesiapsiagaan terhadap potensi banjir di masa depan, terutama untuk daerah hujan sehingga perlu untuk dilakukan standardisasi penyesuaian bangunan terhadap nilai curah hujan tinggi yang telah di identifikasi. Sehingga infrastruktur air yang kuat dapat menjadi pencegahan efektif risiko banjir dikemudian hari.

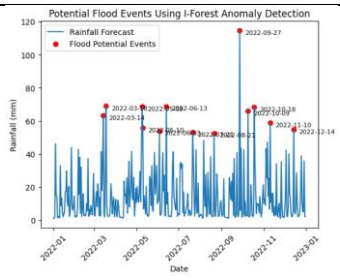
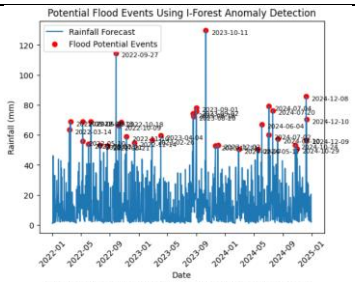
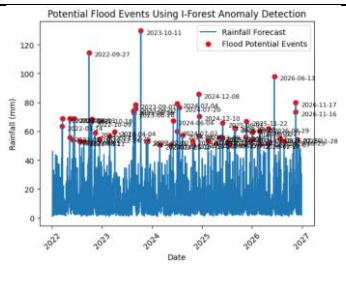
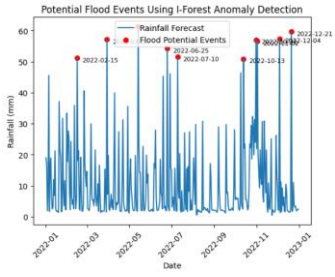
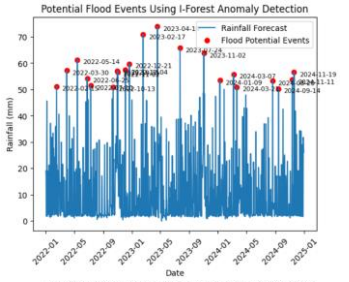
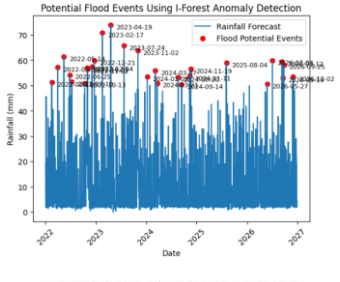
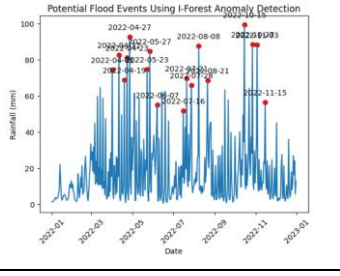
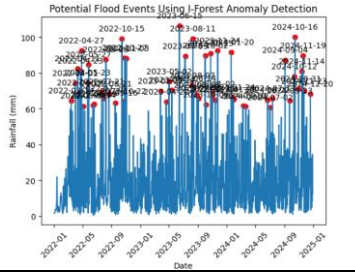
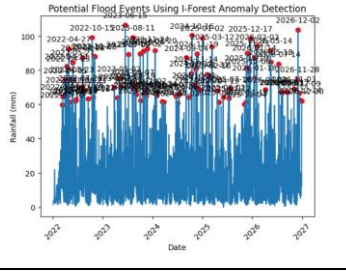
Tabel 8 Data Aktual Bencana

No.	Tnggal	Lokasi	Korban	Keterangan
1.	01/01/2022	RT.01 RW.03 Peadangsari Banyumanik 1	4	Hujan deras menyebabkan di JL. Cemara Timur 1.
2.	01/01/2022	RT.01 RW.05 Jabangan Banyumanik 1	Nihil	Sungai Sadang meluap dan talud sungai roboh akibat hujan deras.
3.	02/01/2022	RW. 04 -RW.09 Srondel Kulon Banyumanik 1	10	Hujan dengan intesitas tinggi menyebabkan tanah longsor.
4.	01/01/2022	RW.01 – RW.07 Tambakrejo Gayamsari 1	11	banjir terjadi akibat pompa yang mengarah ke Banjir Kamal Timur kurang berfungsi.
5.	01/01/2022	RW.01 – RW.03 Siwalan Gayamsari 1	Nihil	Banjir dengan ketinggian genangan air antara 10-60 cm.
6.	01/01/2022	RT.01 RW.03 Jangli Tembalang 1	Nihil	Hujan deras menyebabkan talud jalan longsor.
7.	06/01/2022	RT.07 RW.11 Kembangarum Semarang Barat 1	4	Hujan deras dan saluran air yang mampet menyebabkan air langsung menghantam tembok.

Sumber : BPBD Kota Semarang

Hasil yang telah didapatkan dalam penelitian ini, telah menjawab permasalahan penelitian yaitu menghasilkan *forecasting* probabilitas jangka panjang pada data curah hujan yang kemudian dapat diidentifikasi secara cepat dan efisien. Hasil tersebut merupakan identifikasi dini yang dapat dilakukan untuk pencegahan banjir. Masih terdapat banyak kelemahan pada model ini diantaranya akurasi *forecasting*, hal ini dipengaruhi oleh parameter input yang terbatas yang hanya 1 parameter input curah hujan. Untuk penelitian selanjutnya diharapkan dapat mengembangkan dengan banyak parameter input sehingga hasil *forecasting* akan lebih baik.

Tabel 9 Hasil Identifikasi Curah Hujan Tinggi 1 Hingga 5 Tahun

Step	1 Tahun	3 Tahun	5 Tahun
10			
15			
365			

5. KESIMPULAN

Model LSTM yang dikembangkan dengan penambahan Teknik Augmentasi *Gaussian Noise* dapat memperbaiki tingkat akurasi LSTM. Dengan perubahan nilai Step yang dilakukan Nilai Step 15 hari sebelumnya atau data *series* sebelumnya sebagai sekuens masukan mendapatkan nilai akurasi yang dapat menghasilkan fluktuasi data *forecasting*. Sehingga model dapat melakukan *forecasting* dalam jangka panjang dan menghasilkan identifikasi dini dengan cepat menggunakan *i-forest* pada pola jangka panjang yang dihasilkan dan menjawab permasalahan penelitian. Hasil eksperimen menunjukkan bahwa dalam melakukan *forecasting*, *N_Step* 10 hari memiliki tingkat keakuratan lebih baik dibandingkan dengan prediksi menggunakan *N_Step* 15. Hasil uji RMSE *N_Step* 15 lebih baik, namun jarak waktu antara keduanya tidak terlalu jauh. Hasil penelitian ini penting dalam pengambilan keputusan terkait mitigasi risiko banjir.

DAFTAR PUSTAKA

ALDRICH, C., & LIU, X. 2024. Monitoring of Mineral Processing Operations with Isolation Forests. *Minerals*, *14*(1), 76. https://doi.org/10.3390/min14010076

AULIA, F., FARISI, A., SETYA PERDANA, R., ADIKARA, P. P., & KORESPONDENSI, P. 2023. Klasifikasi Intensi Dengan Metode Long Short-Term Memory Pada Chatbot Bahasa Indonesia. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, *10*(7). https://doi.org/10.25126/jtiik.2023108000

BAI, Y., YANG, E., HAN, B., YANG, Y., LI, J., MAO, Y., NIU, G., & LIU, T. 2021. Understanding and Improving Early Stopping for Learning with Noisy Labels. *35th Conference on Neural Information Processing Systems*. https://github.com/tmllab/PES.

CARNEGIE, D. A., & CHAIRANI. 2023. *Perbandingan Long Short Term Memory (LSTM) dan Gated Recurrent Unit (GRU) Untuk Memprediksi Curah Hujan*. *7*(3), 1022–1032. https://doi.org/10.30865/mib.v7i3.6213

CHEN, J., ZHANG, J., QIAN, R., YUAN, J., & REN, Y. 2023. An Anomaly Detection Method for Wireless Sensor Networks Based on the Improved Isolation Forest. *Applied Sciences (Switzerland)*, *13*(2). https://doi.org/10.3390/app13020702

DHOUIB, H., WILMS, A., & BOES, P. 2023. *Distribution and volume based scoring for Isolation Forests*. http://arxiv.org/abs/2309.11450

FANG, N., FANG, X., & LU, K. 2022. Anomalous Behavior Detection Based on the Isolation Forest Model with Multiple Perspective Business Processes. *Electronics (Switzerland)*, *11*(21). https://doi.org/10.3390/electronics11213640

GOODFELLOW, I., BENGIO, Y., & COURVILLE, A. 2016. *Deep Learning*. The MIT Press.

HENDRA, Y., MUKHTAR, H., & HAFSARI, R. 2023. *Prediksi Curah Hujan Di Kota Pekanbaru menggunakan LSTM (Long Short Term Memory)* (Vol. 3, Número 2).

HOCHREITER, S., & SCHMIDHUBER, J. 1997. Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

JUNAIDI, A., NURHAMIDAH, N., & DAOED, D. 2018. Future flood management strategies in Indonesia. *MATEC Web of Conferences*, *229*. https://doi.org/10.1051/matecconf/201822901014

KARCZMAREK, P., KIERSZTYN, A., PEDRYCZ, W., & AL, E. 2020. K-Means-based isolation forest ☆. *Knowledge Based Systems*, *195*, 105659. https://doi.org/10.1016/j.knosys

LIU, F. T., TING, K. M., & ZHOU, Z. H. 2008. Isolation forest. *Proceedings - IEEE International Conference on Data Mining, ICDM*, 413–422. https://doi.org/10.1109/ICDM.2008.17

MARZUKHI, S., SIDIK, M. A. S. M., NASIR, H. M., ZAINOL, Z., & ISMAIL, M. N. 2018. Flood Detection and Warning System (FLoWS). *ACM International Conference Proceeding Series*. https://doi.org/10.1145/3164541.3164623

MILLER, J. D., & HUTCHINS, M. 2017. The impacts of urbanisation and climate change on urban flooding and urban water quality: A review of the evidence concerning the United Kingdom. Em *Journal of Hydrology: Regional Studies* (Vol. 12, p. 345–362). Elsevier B.V. https://doi.org/10.1016/j.ejrh.2017.06.006

NWANKPA, C., IJOMAH, W., GACHAGAN, A., & MARSHALL, S. 2018. *Activation Functions: Comparison of trends in Practice and Research for Deep Learning*. http://arxiv.org/abs/1811.03378

POORNIMA, S., & PUSHPALATHA, M. 2019. Prediction of rainfall using intensified LSTM based recurrent Neural Network with Weighted Linear Units. *Atmosphere*, *10*(11). https://doi.org/10.3390/atmos10110668

PYO, J. C., PACHEPSKY, Y., KIM, S., ABBAS, A., KIM, M., KWON, Y. S., LIGARAY, M., & CHO, K. H. 2023. Long short-term memory models of water quality in inland water environments. Em *Water Research X* (Vol. 21). Elsevier Ltd. https://doi.org/10.1016/j.wroa.2023.100207

QIAN, H. 2022. Stock Predicting based on LSTM and ARIMA. *Proceedings of the 2022 2nd International Conference on Economic Development and Business Culture (ICEDBC 2022)*, 485–490. https://doi.org/10.2991/978-94-6463-036-7_72

SENNHAUSER, L., & BERWICK, R. C. 2018. Evaluating the Ability of LSTMs to Learn Context-Free Grammars. *Proceedings of the EMNLP Workshop BlackboxNLP*, 115–124. https://doi.org/https://doi.org/10.48550/arXiv.1811.02611

SEPTIAN, B. A., & KARTINI, U. T. 2023. Pemodelan Peramalan Beban Jangka Pendek untuk Subsistem Krian-Gresik Menggunakan Deep Learning LSTM-NN. *Jurnal Teknik Elektro UNESA*, *12*(2), 1–5.

SHEN, T., & XIANG, Y. 2023. Optimization of Probability Density Functions Applicable for Hourly Rainfall. *Atmosphere*, *14*(7). https://doi.org/10.3390/atmos14071100

SHIRI M FARHAD, PERUMAL THINAGARAN, MUSTAPHA NORMAWATI, & MOHAMED RAIHANI. 2023. *A Comprehensive Overview and Comparative Analysis on Deep Learning Models: CNN, RNN, LSTM, GRU*. https://doi.org/10.48550/arXiv.2305.17473

STANTON, C., KATZ, GI., & SONG, D. 2012. Isolation Forest for Anomaly Detection. Em *ACM Transactions on Knowledge Discovery from Data* (Vol. 6, Número 1).

WANG, Z., & QI, Z. 2023. Future Stock Price Prediction Based on Bayesian LSTM in CRSP. *Proceedings of the 3rd International Conference on Internet Finance and Digital Economy (ICIFDE 2023)*, 219–230. https://doi.org/10.2991/978-94-6463-270-5_1

YING, X. 2019. An Overview of Overfitting and its Solutions. *Journal of Physics: Conference Series*, *1168*(2). https://doi.org/10.1088/1742-6596/1168/2/022022