

PROPOSAL SKRIPSI

**ANALISIS PERBANDINGAN *METODE RECURRENT NEURAL
NETWORK* DAN *GRADIENT BOOSTING* UNTUK PREDIKSI
BANJIR**

DISUSUN OLEH

NURHASANAH

2109020009



**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
MEDAN**

2025

**ANALISIS PERBANDINGAN *METODE RECURRENT NEURAL
NETWORK DAN GRADIENT BOOSTING* UNTUK PREDIKSI
BANJIR**

SKRIPSI

**Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana
Komputer (S.Kom) dalam Program Studi Teknologi Informasi pada
Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas
Muhammadiyah Sumatera Utara**

**NURHASANAH
2109020009**



**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA M**

LEMBAR PENGESAHAN

Judul Skripsi : ANALISIS PERBANDINGAN *METODE*
RECURRENT NEURAL NETWORK DAN
GRADIENT BOOSTING UNTUK PREDIKSI
BANJIR

Nama Mahasiswa : NURHASANAH

NPM : 2109020009

Program Studi : Teknologi Informasi

Menyetujui
Komisi Pembimbing



(Halim Maulana, S.T., M.Kom.)
NIDN. 0121119102

Ketua Program Studi



(Fatma Sari Hutagalung S.Kom., M.Kom.)
NIDN. 0117019301

Dekan



(Dr. Al-Khowarizmi, S.Kom., M.Kom.)
NIDN. 0127099201

PERNYATAAN ORISINALITAS

ANALISIS PERBANDINGAN METODE RECURRENT NEURAL NETWORK DAN GRADIENT BOOSTING UNTUK PREDIKSI BANJIR

SKRIPSI

Saya menyatakan bahwa karya tulis ini adalah hasil karya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya.

Medan, September 2025
Yang membuat pernyataan



**Nurhasanah
210902000**

**PERNYATAAN PERSETUJUAN PUBLIKASI
KARYA ILMIAH UNTUK KEPENTINGAN
AKADEMIS**

Sebagai sivitas akademika Universitas Muhammadiyah Sumatera Utara, saya bertanda tangan dibawah ini:

Nama	: Nurhasanah
NPM	: 2109020009
Program Studi	: Teknologi Informasi
Karya Ilmiah	: Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Muhammadiyah Sumatera Utara Hak Bedas Royalti Non-Eksekutif (*Non-Exclusive Royalty free Right*) atas penelitian skripsi saya yang berjudul:

**ANALISIS PERBANDINGAN METODE RECURRENT NEURAL
NETWORK DAN GRADIENT BOOSTING UNTUK PREDIKSI BANJIR**

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksekutif ini, Universitas Muhammadiyah Sumatera Utara berhak menyimpan, mengalih media, memformat, mengelola dalam bentuk database, merawat dan mempublikasikan Skripsi saya ini tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis dan sebagai pemegang dan atau sebagai pemilik hak cipta.

Demikian pernyataan ini dibuat dengan sebenarnya.

Medan, September 2025

Yang membuat pernyataan



Nurhasanah
2109020009

RIWAYAT HIDUP

DATA PRIBADI

Nama Lengkap : NURHASANAH
Tempat dan Tanggal Lahir : MEDAN, 08 MARET 2003
Alamat Rumah : JL.BERSAMA NO 84 LK III MEDAN
Telepon/Faks/HP : 081536690049
E-mail : hanafaith23@gmail.com
Instansi Tempat Kerja : -
Alamat Kantor : -

DATA PENDIDIKAN

SD	: SDN 064974	TAMAT: 2015
SMP	: SMP Swasta Budisatrya Medan	TAMAT: 2018
SMA	: SMK Swasta Budisatrya Medan	TAMAT: 2021

KATA PENGANTAR



Assalamu'alaikum warahmatullahi wabarakatuh.

Segala puji dan syukur penulis panjatkan ke hadirat Allah Subhanahu wa Ta'ala yang telah melimpahkan rahmat, taufik, serta hidayah-Nya sehingga penulis dapat menyelesaikan skripsi dengan judul “Analisis Perbandingan Metode Recurrent Neural Network dan Gradient Boosting untuk Prediksi Banjir”. Shalawat serta salam semoga selalu tercurah kepada junjungan kita Nabi Muhammad Shallallahu ‘alaihi wasallam, beserta keluarga, sahabat, dan umatnya hingga akhir zaman.

Skripsi ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana pada Program Studi Teknologi Informasi Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Muhammadiyah Sumatera Utara Penulis menyadari sepenuhnya bahwa dalam proses penyusunan skripsi ini banyak hambatan dan tantangan yang dihadapi. Namun, berkat pertolongan Allah SWT, doa, dukungan, serta bantuan dari berbagai pihak, akhirnya skripsi ini dapat terselesaikan.

Pada kesempatan ini penulis ingin menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Terimakasih kepada Bapak Prof. Dr. Agussani, M.AP., Rektor Universitas Muhammadiyah Sumatera Utara (UMSU)
2. Terimakasih kepada Bapak Dr. Al-Khowarizmi, S.Kom., M.Kom. Dekan

Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.

3. Terimakasih kepada Ibu Fatma Sari Hutagalung, S.Kom., M.Kom., selaku Ketua Program Studi Teknologi Informasi Sekaligus Dosen Penguji penulis
4. Terimakasih kepada Bapak Muhammad Basri, S.Kom., M.Kom selaku Sekretaris Program Studi Teknologi Informasi
5. Terimakasih kepada Bapak Halim Maulana, S.T., M.Kom selaku dosen pembimbing skripsi saya
6. Terimakasih kepada Seluruh dosen dan staf akademik di lingkungan Fakultas Ilmu Komputer dan Teknologi Informasi, khususnya Program Studi Teknologi Informasi, yang telah memberikan ilmu dan pengalaman berharga selama masa studi
7. Terimakasih kepada Kedua Orang Tua saya yang saya sayangi dan saya banggakan Almarhum Bapak Sulaiman dan Almarhumah Ibu Mariani
8. Terimakasih kepada abang dan kakak saya yang saya sayangi dan saya banggakan yang telah mengurus saya Abang saya Muhammad Saleh, dan Kakak saya Erlina, yang telah mendukung dan mensupport saya sehingga saya bisa sampai ketitik ini.
9. Terimakasih kepada teman-teman seperjuangan saya Rifdah, Afrida, Nikma, Miranda, Bella, Surya, dan Handhalla yang telah membantu saya selama menempuh studi di UMSU.
10. Terimakasih kepada Teman-teman Kelas TI A1 Stambuk 21 sudah menemani penulis sedari awal kegiatan perkuliahan hingga akhir

perkuliahan

11. Terimakasih kepada Semua pihak yang terlibat langsung ataupun tidak langsung yang tidak dapat penulis ucapkan satu-persatu yang telah membantu penyelesaian skripsi ini.

Penulis menyadari bahwa skripsi ini masih jauh dari kesempurnaan. Oleh karena itu, kritik dan saran yang membangun sangat penulis harapkan demi kesempurnaan karya ini. Semoga skripsi ini dapat memberikan manfaat, khususnya dalam pengembangan ilmu pengetahuan di bidang teknologi informasi serta dapat menjadi amal jariyah yang bernilai di sisi Allah SWT.

Wassalamu'alaikum warahmatullahi wabarakatuh.

ANALISIS PERBANDINGAN METODE RECURRENT NEURAL NETWORK DAN GRADIENT BOOSTING UNTUK PREDIKSI BANJIR

ABSTRAK

Banjir merupakan salah satu bencana alam yang sering melanda Indonesia dan memberikan dampak besar terhadap infrastruktur, perekonomian, lingkungan, serta kehidupan sosial masyarakat. Untuk mengurangi risiko dan dampaknya, dibutuhkan sistem prediksi yang akurat dan dapat diandalkan. Penelitian ini bertujuan untuk membandingkan kinerja metode Recurrent Neural Network (RNN) dan Gradient Boosting dalam memprediksi kejadian banjir berdasarkan data historis yang mencakup curah hujan, suhu, kelembapan, dan ketinggian air. Data diperoleh dari Badan Meteorologi, Klimatologi, dan Geofisika (BMKG) serta sumber terkait lainnya, kemudian diproses melalui tahapan pembersihan, normalisasi, pembangunan model, pelatihan, dan pengujian. Hasil penelitian menunjukkan bahwa RNN lebih unggul dalam mengolah data time series karena kemampuannya mengenali pola berurutan, meskipun membutuhkan waktu pelatihan yang lebih lama. Di sisi lain, Gradient Boosting terbukti lebih cepat, efisien, dan fleksibel dalam menangani data tabular yang mitigasi bencana.

Kata Kunci : Prediksi Banjir, Recurrent Neural Network, Gradient Boosting, Machine Learning

COMPARATIVE ANALYSIS OF RECURRENT NEURAL NETWORK AND GRADIENT BOOSTING METHODS FOR FLOOD PREDICTION

ABSTRACT

Flooding is one of the most frequent natural disasters in Indonesia, causing significant impacts on infrastructure, the economy, the environment, and social life. To reduce its risks and consequences, an accurate and reliable prediction system is required. This study aims to compare the performance of Recurrent Neural Network (RNN) and Gradient Boosting methods in predicting flood events based on historical data, including rainfall, temperature, humidity, and water level. The data were obtained from the Meteorology, Climatology, and Geophysics Agency (BMKG) and other relevant sources, then processed through several stages such as cleaning, normalization, model construction, training, and testing. The results show that RNN performs better in processing time series data due to its ability to recognize sequential patterns, although it requires longer training time. On the other hand, Gradient Boosting proves to be faster, more efficient, and more flexible in handling complex tabular data, as well as easier to interpret. In conclusion, the choice of the most suitable method depends on the data characteristics and the needs of early warning system implementation. This research is expected to contribute to the development of more effective and applicable flood prediction technology for both communities and governments in disaster mitigation.

Keywords: Flood Prediction, Recurrent Neural Network, Gradient Boosting, Machine Learning.

DAFTAR ISI

LEMBAR PENGESAHAN.....	i
PERNYATAAN ORISINALITAS.....	ii
PERNYATAAN PERSETUJUAN PUBLIKASI.....	iii
RIWAYAT HIDUP.....	iiiv
KATA PENGANTAR.....	v
ABSTRAK.....	vviii
ABSTRACT.....	ix
DAFTAR ISI.....	x
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR.....	xiv
BAB I.....	1
PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	4
BAB II.....	6
LANDASAN TEORI.....	6
2.1 Studi Terdahulu.....	6
2.2 Recurrent Neural Network (RNN).....	14
2.2.1 Pengertian Recurrent Neural Network (RNN).....	14
2.2.2 Cara kerja Recurrent Neural Network (RNN).....	15
2.2.3 Keunggulan dan Kelemahan RNN.....	17
2.3 Gradient Boosting.....	19
2.3.1 Pengertian Gradient Boosting.....	19
2.3.2 Cara kerja Gradient Boosting.....	20
2.3.3 Keunggulan dan Kelemahan Gradient Boosting.....	24
2.4 Perbandingan RNN dan Gradient Boosting dalam Prediksi Banjir.....	25
BAB III.....	31
METODOLOGI PENELITIAN.....	31
3.1 Metodologi Penelitian.....	31
3.1.1 Teknik Pengolahan Data.....	33
3.2 Tempat dan waktu penelitian.....	34

3.2.1 Tempat Penelitian.....	34
3.2.2 Waktu dan Jadwal Pelaksanaan Penelitian.....	34
3.3 Sumber dan teknik Pengumpulan Data	35
3.4 Teknik Pengolahan dan Analisis Data.....	36
3.4.1 Prepropcessing data.....	36
3.4.2 Pembangunan Model.....	49
3.4.3 Pelatihan dan Pengujian Model.....	52
3.4.4 Evaluasi Model.....	54
3.4.5 Analisis dan Kesimpulan.....	55
3.4 Evaluasi Penelitian	56
BAB IV	58
HASIL DAN PEMBAHASAN	58
4.1 Hasil Penelitian	58
4.1.1 Pengujian Hasil	58
4.1.2 Pengujian Akurasi Sistem	59
4.2 Perhitungan Hasil	60
4.3 Perhitungan RNN (RECURRENT NEURAL NETWORK).....	61
4.3.1 Hitung Data Januari 2022.....	61
4.3.2 Hasil Data April 2023.....	63
4.3.3 Hasil Data Januari 2024	64
4.4 Perhitungan Gradient Boosting	67
4.4.1 Data Januari 2022.....	67
4.4.2 Hasil Data April 2023.....	69
4.4.3 hasil data Januari 2024	68
BAB V.....	88
KESIMPULAN DAN SARAN	88
5.1 Kesimpulan	88
5.2 Saran	87
DAFTAR PUSTAKA	89
LAMPIRAN.....	91

DAFTAR TABEL

Tabel 2. 1 studi terdahulu.....	6
Tabel 2. 2 Penjelasan Flowchart atau alur	16
Tabel 2. 3 penjelasan flowchart atau alur.....	23
Tabel 2. 4 Tabel Perbandingan prediksi.....	25
Tabel 2. 5 Perbandingan akurasi	28
Tabel 3. 1 Jadwal Pelaksanaan Penelitian.....	34
Tabel 3. 2 Jenis Data RNN dan Gradient Boosting.....	36
Tabel 3. 3 Tahapan dari Preprocessing Data.....	37
Tabel 3. 4 Tabel prediksi	42
Tabel 3. 5 Prediksi banjir di hari ke-2.....	42
Tabel 3. 6 Tabel residual.....	44
Tabel 3. 7 Tabel prediksi ke F.....	44
Tabel 3. 8 Pembangunan Model	50
Tabel 3. 9 Tabel pelatihan dan pengujian model	53
Tabel 3. 10 Tabel Evaluasi Model	55
Tabel 4 1 Tabel Data.....	58
Tabel 4 2 Tabel Akurasi Sistem.....	60
Tabel 4 3 Tabel data.....	70
Tabel 4 4 Tabel Hasil Akhir Prediksi.....	83

DAFTAR GAMBAR

Gambar 2. 1 flowchart RNN	16
Gambar 2. 2 flowchart Gradient Boosting	23
Gambar 3. 1 Tempat Penelitian	34
Gambar 3. 2 Data Dummy Prediksi banjir	39
Gambar 3. 3 fitur X	40
Gambar 3. 4 fitur x setelah normalisasi	40
Gambar 3. 5 Histogram	46
Gambar 3. 6 Boxplot untuk deteksi outlier	47
Gambar 3. 7 Heatmap korelasi antar fitur dalam dataset	49
Gambar 4.1 Info Dataset	71
Gambar 4.2 Grafik	72
Gambar 4.3 Heatmap	74
Gambar 4.4 Scatter plot	74
Gambar 4.5 Label Banjir	76
Gambar 4.6 Label Banjir Berdasarkan Curah Hujan	77
Gambar 4.7 Label Banjir Berdasarkan Kelembapan	78
Gambar 4.8 Label Banjir Berdasarkan Suhu	79
Gambar 4.9 Evaluasi Model	80
Gambar 4.10 Grafik Yang Mempengaruhi Banjir	85
Gambar 4.11 Grafik Time Series	86

BAB I

PENDAHULUAN

1.1 Latar Belakang

Banjir merupakan salah satu bencana alam paling merusak yang sering terjadi di berbagai wilayah dunia, termasuk Indonesia. Banjir tidak hanya menimbulkan kerusakan fisik seperti rusaknya infrastruktur, hilangnya tempat tinggal, dan terhambatnya layanan esensial (listrik, air bersih, dan komunikasi), tetapi juga berdampak besar pada sektor ekonomi dan sosial masyarakat. Dampak banjir diperparah oleh perubahan iklim yang menyebabkan pola curah hujan menjadi tidak menentu serta meningkatnya intensitas hujan ekstrem.

Secara umum, banjir terjadi ketika volume air yang masuk ke suatu wilayah melebihi kapasitas penampungannya, sehingga air meluap dan merendam area di sekitarnya. Arus air yang deras kerap kali merusak bangunan, memutus jalur komunikasi, dan menyulitkan proses evakuasi. Oleh karena itu, prediksi banjir yang akurat dan cepat menjadi sangat penting untuk mengurangi risiko dan dampak bencana tersebut.

Dalam beberapa tahun terakhir, kemajuan teknologi kecerdasan buatan (Artificial Intelligence/AI) telah memberikan solusi baru dalam prediksi bencana, termasuk banjir. Dua metode AI yang populer untuk prediksi data sekuensial dan kompleks adalah Recurrent Neural Network (RNN) dan Gradient Boosting. RNN dikenal mampu mengolah data time series seperti curah hujan dan ketinggian air dengan baik karena kemampuannya mengingat informasi jangka panjang. Sementara itu, Gradient Boosting efektif dalam menangani data heterogen dan menghasilkan prediksi yang presisi melalui pendekatan pohon keputusan berlapis.

Meskipun keduanya telah digunakan dalam berbagai aplikasi, belum banyak

penelitian yang membandingkan secara langsung kinerja RNN dan Gradient Boosting dalam konteks prediksi banjir. Beberapa studi masih terfokus pada satu metode saja atau terbatas pada aplikasi tertentu tanpa evaluasi komparatif yang menyeluruh. Selain itu, tantangan seperti data tidak lengkap, variasi lingkungan, serta kebutuhan komputasi turut memengaruhi efektivitas masing-masing metode.

Tinjauan literatur lima tahun terakhir menunjukkan bahwa AI telah banyak diterapkan dalam mitigasi banjir, misalnya melalui klasifikasi korban banjir menggunakan algoritma pohon keputusan atau analisis dampak banjir terhadap sektor pertanian. Namun, masih terdapat kesenjangan dalam studi komparatif metode AI untuk prediksi banjir dengan kondisi dan dataset yang setara.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang diuraikan, rumusan masalah dalam penelitian ini adalah:

1. kinerja metode Recurrent Neural Network (RNN) dibandingkan dengan Gradient Boosting dalam memprediksi kejadian banjir berdasarkan data historis curah hujan dan ketinggian air?
2. Seberapa besar tingkat akurasi prediksi masing-masing metode (RNN dan Gradient Boosting) dalam konteks prediksi banjir ?
3. Metode mana yang lebih efisien dalam hal waktu pelatihan, waktu prediksi, dan adaptabilitas terhadap variasi data lingkungan?
4. Apa saja kelebihan dan kekurangan masing-masing metode dalam pengembangan sistem peringatan dini banjir yang efektif dan efisien?
5. Bagaimana perbandingan antara metode Recurrent Neural Network dan Gradient Boosting

1.3 Batasan Masalah

Penelitian ini memiliki batasan masalah sebagai berikut:

1. Metode yang dikaji dalam penelitian ini terbatas pada dua algoritma kecerdasan buatan, yaitu Recurrent Neural Network (RNN) dan Gradient Boosting,
2. Data yang digunakan terbatas pada data historis yang berkaitan dengan kejadian banjir, seperti curah hujan, tinggi muka air sungai, dan parameter hidrometeorologi lainnya,
3. Penelitian ini tidak mencakup aspek implementasi sistem peringatan dini secara penuh (misalnya, antarmuka pengguna atau integrasi dengan sistem evakuasi), tetapi hanya fokus pada aspek pemodelan dan prediksi kejadian banjir.
4. Data yang di ambil hanya data dari BMKG JL.Meteorologi Raya

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Menganalisis Kinerja Metode RNN dan Gradient Boosting dalam Prediksi Banjir.
2. Mengukur Akurasi Prediksi Menggunakan Metrik Evaluasi MAE dan RMSE.
3. Mengevaluasi Efisiensi Proses Pelatihan dan Prediksi.
4. Menilai Kemampuan Adaptasi Model terhadap Variasi Data dan Kondisi Lingkungan Banjir.
5. Memberikan Rekomendasi Metode AI yang Optimal untuk Sistem Peringatan Dini Banjir di Indonesia.

Dengan tujuan-tujuan tersebut, penelitian ini diharapkan dapat memberikan kontribusi penting baik dari segi teori maupun praktik dalam bidang mitigasi bencana banjir.

1.5 Manfaat Penelitian

Penelitian ini memberikan berbagai manfaat baik dari sisi teoritis maupun praktis. Secara teoritis, penelitian ini memperkaya literatur mengenai penerapan kecerdasan buatan, khususnya metode Recurrent Neural Network (RNN) dan Gradient Boosting, dalam prediksi bencana banjir. Penelitian ini juga menawarkan wawasan baru mengenai perbandingan kinerja kedua metode tersebut dalam konteks yang sama, yang masih jarang dibahas dalam studi sebelumnya.

Secara praktis, hasil penelitian ini dapat dimanfaatkan oleh pemerintah dan lembaga terkait dalam mengembangkan sistem peringatan dini banjir yang lebih efektif dan efisien. Dengan metode prediksi yang terbukti akurat, kesiapsiagaan masyarakat dan instansi terkait dalam menghadapi bencana banjir akan meningkat, sehingga kerugian materi dan korban jiwa dapat diminimalisir. Selain itu, penelitian ini juga membantu dalam pengelolaan sumber daya secara lebih efisien dengan mempertimbangkan waktu pelatihan dan penggunaan komputasi.

Dari sisi sosial, peningkatan akurasi prediksi banjir berpotensi melindungi masyarakat serta infrastruktur vital, sekaligus menjaga kelangsungan aktivitas sosial dan ekonomi di wilayah terdampak. Penelitian ini juga turut mendukung peningkatan ketahanan dan kesadaran masyarakat terhadap risiko banjir.

Secara lingkungan, prediksi banjir yang lebih tepat akan memudahkan pengambilan tindakan pencegahan yang efektif untuk mengurangi dampak negatif seperti pencemaran air, kerusakan ekosistem, dan kerugian sektor pertanian. Dari sisi ekonomi, penerapan metode AI ini diharapkan mampu mengurangi kerugian yang ditimbulkan oleh banjir dan mempercepat proses pemulihan pasca-bencana.

Dengan semua manfaat tersebut, penelitian ini diharapkan dapat berkontribusi dalam upaya mitigasi banjir, baik di Indonesia maupun di negara lain yang menghadapi

tantangan serupa, serta meningkatkan kesiapsiagaan masyarakat melalui teknologi prediksi yang lebih akurat.

BAB II LANDASAN TEORI

2.1 Studi Terdahulu

Tabel 2. 1 studi terdahulu

No.	Penulis (Tahun)	Judul penelitian	Hasil Penelitian
1	Pangad Eko Putra, Muhamad Azhri Amrullah, Yahya Hasani Fauzi , Refy Fitriani Saputri , Lelly Clodia RF (2025)	Penerapan Algoritma C4.5 untuk Klasifikasi Tingkat Korban Banjir di Indonesia	1. Gradient Boosting lebih efisien dan akurat untuk prediksi banjir pada data yang beragam dan tidak sepenuhnya time-series. 2. RNN (Recurrent Neural Network) lebih unggul untuk data time-series (berurutan) seperti curah hujan dan tinggi muka air, namun butuh waktu pelatihan lebih lama. 3. Algoritma C4.5 berhasil mengklasifikasikan tingkat korban banjir di Indonesia dengan akurasi sangat tinggi (97,5%). 4. Pemilihan metode terbaik tergantung pada karakteristik data: Gradient Boosting cocok untuk implementasi luas dan data heterogen. RNN cocok untuk data historis yang kontinu. 5. Penelitian ini memberikan rekomendasi teknis untuk sistem peringatan dini banjir di Indonesia agar lebih efektif dan efisien.
			Hasil penelitian ini menunjukkan bahwa luas genangan banjir rob di Kota Pekalongan pada bulan

2	Syafrei Adi Iskandar, Muhammad Helmi, Muslim, Sugeng Widada dan Baskoro Rochaddi (2020)	Analisis Geospasial Area Genangan Banjir Rob dan Dampaknya pada Penggunaan Lahan Tahun 2020 - 2025 di Kota Pekalongan Provinsi Jawa Tengah	Februari 2020 adalah sekitar 477,57 hektar. Prediksi untuk tahun 2025 menunjukkan peningkatan luas genangan yang cukup signifikan menjadi 1.877,07 hektar. Faktor utama yang menyebabkan perluasan genangan ini adalah penurunan muka tanah yang terjadi dengan laju rata-rata sekitar 20–24 cm per tahun di berbagai wilayah Pekalongan, yang ternyata memberikan pengaruh lebih besar dibandingkan kenaikan muka air laut yang tercatat sebesar 4,3 mm per tahun. Dengan demikian, penurunan muka tanah menjadi faktor dominan dalam perubahan luas genangan banjir rob di Pekalongan, dan hal ini berdampak pada perubahan penggunaan lahan serta meningkatkan risiko terhadap wilayah pesisir kota tersebut
3	Zahirulhaq, Sidik Nurcahyo, Imam Saukan (2025)	Rancang Bangun Sistem Peringatan Dini dan Pencegahan Banjir pada Bendungan Menggunakan Metode Fuzzy	Hasil penelitian ini menunjukkan bahwa sistem peringatan dini dan pencegahan banjir pada bendungan yang dirancang menggunakan metode fuzzy dapat memantau dan mengontrol ketinggian air secara otomatis dan real-time. Sistem ini memanfaatkan sensor ultrasonik untuk mengukur ketinggian air dan sensor hujan untuk memantau curah hujan, dengan data yang diproses oleh mikrokontroler dan ditampilkan melalui LCD serta aplikasi IoT Blynk. Pengujian membuktikan bahwa sistem mampu menjaga ketinggian air tetap pada batas aman, dan hasil deteksi potensi banjir yang dihasilkan sangat mendekati hasil simulasi Matlab, dengan rata-rata error hanya 7,54%. Sistem ini dinilai efektif dalam memberikan peringatan dini dan pengendalian banjir secara otomatis pada bendungan.
			Hasil penelitian ini menunjukkan

4	Ryan Septianto, Wahyu Widji Pamungkas, M Arif Yuana (2022)	Sistem Peringatan Dini Banjir Menggunakan Logika Fuzzy	bahwa sistem peringatan dini banjir berbasis logika fuzzy yang dikembangkan mampu memberikan informasi status siaga banjir secara otomatis dan real-time. Sistem ini menggunakan sensor ultrasonik yang terhubung dengan Arduino untuk memantau ketinggian air pada sebuah miniatur kali yang mewakili kondisi sungai sebenarnya. Data curah hujan yang diperoleh dari BMKG dipadukan dengan hasil pengukuran debit air untuk menentukan tingkat risiko banjir. Dengan pendekatan logika fuzzy, sistem dapat mengklasifikasikan kondisi sungai ke dalam beberapa level siaga, mulai dari normal, waspada, siaga, hingga awas dan bencana, berdasarkan kecepatan kenaikan permukaan air dan intensitas hujan. Pengujian pada prototipe menunjukkan bahwa sistem ini dapat memprediksi waktu terjadinya banjir berdasarkan volume dan debit air yang masuk, sehingga masyarakat dan petugas dapat menerima peringatan lebih awal dan memiliki waktu yang cukup untuk melakukan persiapan atau evakuasi. Dengan demikian, sistem ini dinilai efektif dalam meningkatkan kesiapsiagaan terhadap bencana banjir di wilayah perkotaan yang rawan.
5	Leo Riska (2021)	Prediksi Lokasi Banjir dari Data Twitter Menggunakan Hidden Markov Model	Berdasarkan penelitian dalam tesis ini, penggunaan data Twitter yang dianalisis dengan metode <i>Random Forest</i> untuk klasifikasi dan <i>Hidden Markov Model</i> untuk prediksi terbukti mampu memetakan dan memprediksi lokasi banjir secara lebih spesifik dan akurat dibandingkan informasi prediksi banjir konvensional. Model yang dibangun dapat mengidentifikasi pola-pola

			kemunculan banjir berdasarkan tweet masyarakat, sehingga prediksi lokasi banjir menjadi lebih tepat sasaran. Hasil pengujian menunjukkan bahwa integrasi kedua metode ini efektif dalam mengolah data media sosial untuk mendukung sistem peringatan dini banjir, sehingga masyarakat dan pihak terkait dapat memperoleh informasi yang lebih cepat dan relevan mengenai potensi lokasi banjir di masa mendatang.
--	--	--	---

dari penelitian sebelumnya oleh Pangad Eko Putra, Muhamad Azhri Amrullah, Yahya Hasani Fauzi , Refy Fitriani Saputri , Lelly Clodia RF menyatakan bahwa penggunaan teknik data mining dengan algoritma C4.5 terbukti sangat efektif dalam mengklasifikasikan tingkat dampak korban banjir berdasarkan data yang diperoleh dari Badan Pusat Statistik (BPS) tahun 2023. Penelitian ini mengolah data dari 38 provinsi di Indonesia, yang mencakup jumlah korban banjir meninggal dan hilang, korban luka-luka, serta korban yang terdampak dan mengungsi, untuk menghasilkan sebuah model pohon keputusan yang mampu mengelompokkan wilayah ke dalam tiga kategori tingkat dampak, yaitu Rendah, Sedang, dan Tinggi. Melalui proses seleksi dan pengolahan data menggunakan perangkat lunak RapidMiner dan Altair AI Studio, pohon keputusan yang terbentuk menunjukkan bahwa atribut "jumlah korban terdampak dan mengungsi" menjadi faktor utama dalam penentuan tingkat dampak korban banjir. Model klasifikasi ini menghasilkan tingkat akurasi yang sangat tinggi, yakni sebesar 97,50%, dengan margin kesalahan $\pm 7,91\%$ dan tingkat presisi serta recall yang sangat memuaskan, terutama pada kategori Sedang dan Rendah. Hal ini menunjukkan bahwa algoritma C4.5 tidak hanya mampu mengolah data dengan baik, tetapi juga dapat memberikan hasil klasifikasi yang akurat dan dapat diandalkan untuk keperluan mitigasi bencana. (Riska, 2022) Model ini memungkinkan para pengambil

kebijakan untuk mengidentifikasi wilayah-wilayah yang memiliki risiko tinggi dan membutuhkan perhatian lebih besar dalam penanggulangan bencana, serta membantu dalam perencanaan distribusi bantuan, sumber daya, dan strategi penanggulangan secara lebih tepat sasaran. Meskipun performa model secara umum sangat baik, hasil penelitian ini juga menunjukkan bahwa masih ada ruang untuk peningkatan, khususnya dalam menyempurnakan akurasi klasifikasi pada kategori Tinggi, agar kesalahan prediksi dapat diminimalkan lebih lanjut. Oleh karena itu, pengembangan model lanjutan dengan mempertimbangkan lebih banyak variabel atau integrasi dengan pendekatan lain dapat dilakukan di masa mendatang untuk meningkatkan ketepatan klasifikasi. Secara keseluruhan, penelitian ini memberikan kontribusi penting dalam pemanfaatan teknologi dan analisis data untuk mendukung kesiapsiagaan bencana, dengan harapan hasilnya dapat dijadikan landasan dalam pengambilan keputusan yang lebih tepat dalam merancang kebijakan mitigasi banjir di Indonesia. (Riska, 2022)

Selanjutnya, penelitian oleh Syafrei Adi Iskandar, Muhammad Helmi, Muslim, Sugeng Widada dan Baskoro Rochaddi Berdasarkan hasil penelitian yang berjudul "Analisis Geospasial Area Genangan Banjir Rob dan Dampaknya pada Penggunaan Lahan Tahun 2020 - 2025 di Kota Pekalongan Provinsi Jawa Tengah", dapat disimpulkan bahwa fenomena banjir rob di Kota Pekalongan merupakan ancaman serius yang dipicu oleh kombinasi dari kenaikan muka air laut akibat perubahan iklim dan penurunan muka tanah yang signifikan akibat aktivitas manusia serta kondisi geologis daerah tersebut. Berdasarkan analisis pasang surut, Kota Pekalongan memiliki tipe pasang campuran condong ke harian ganda dengan perbedaan elevasi antara muka air laut tertinggi (HHWL) dan muka rata-rata laut (MSL) sebesar 45,99 cm. Laju kenaikan muka air laut tercatat sebesar 4,3 mm per tahun, sedangkan laju penurunan muka tanah sangat tinggi, berkisar antara 16,74 cm hingga 27,51 cm per tahun, dengan

nilai tertinggi di Kecamatan Pekalongan Utara. Pemodelan geospasial menunjukkan bahwa luas area genangan banjir rob pada tahun 2020 mencapai 477,57 hektar, sedangkan prediksi untuk tahun 2025 menunjukkan lonjakan yang drastis hingga mencapai 1877,07 hektar.(Riska, 2022) Hal ini berarti dalam kurun waktu lima tahun akan terjadi peningkatan lebih dari tiga kali lipat luas wilayah yang terdampak banjir rob, di mana penurunan muka tanah berperan jauh lebih dominan dibandingkan dengan kenaikan muka air laut. Kecamatan Pekalongan Utara menjadi wilayah yang paling parah terdampak dengan luas genangan mencapai lebih dari 1.000 hektar pada tahun 2025. Dampak banjir rob ini tidak hanya terbatas pada genangan air semata, tetapi juga membawa konsekuensi serius terhadap perubahan penggunaan lahan, khususnya pada lahan pemukiman, industri, pertanian, dan tambak. Genangan yang terus meluas mengancam kelangsungan ekonomi masyarakat dan stabilitas sosial, terutama di wilayah pesisir.(Riska, 2022) Temuan ini menunjukkan urgensi perencanaan tata ruang yang adaptif, pembangunan infrastruktur pengendali rob, serta perlunya mitigasi terhadap faktor penurunan muka tanah sebagai penyebab utama. Dengan demikian, hasil penelitian ini dapat dijadikan dasar dalam pengambilan keputusan dan kebijakan untuk penanggulangan banjir rob secara berkelanjutan di wilayah pesisir Kota Pekalongan.(Iskandar et al., 2020)

Penelitian lainnya oleh Zahirulhaq, Sidik Nurcahyo, Imam Saukan bahwa sistem yang dikembangkan berhasil berfungsi dengan baik dalam memantau dan mengendalikan potensi banjir secara otomatis melalui pendekatan logika fuzzy.(Septianto et al., 2022) Sistem ini menggabungkan sensor hujan dan sensor ultrasonik untuk mendeteksi tren curah hujan dan kenaikan ketinggian air, yang kemudian dianalisis oleh mikrokontroler ESP32 untuk mengatur kran ball valve secara otomatis melalui motor servo MG996R. Hasil pengujian menunjukkan bahwa sistem

mampu memberikan respons yang adaptif terhadap kondisi yang terdeteksi, dengan potensi kesalahan rata-rata yang rendah (7,54%) jika dibandingkan dengan hasil simulasi Matlab, bahkan dalam beberapa pengujian menunjukkan error hanya sebesar 0,8%. Sistem juga berhasil memberikan peringatan dini yang ditampilkan secara real-time melalui layar LCD dan aplikasi Blynk berbasis IoT, sehingga dapat diakses dengan mudah. Dengan kemampuan monitoring dan kontrol yang terintegrasi dan responsif, sistem ini berpotensi menjadi solusi efektif dalam mendukung upaya mitigasi banjir di area bendungan dan dapat dikembangkan lebih lanjut dengan peningkatan akurasi sensor, sistem cadangan daya, serta integrasi dengan platform pemantauan yang lebih luas.(Septianto et al., 2022)

Selanjutnya penelitian dari Ryan Septianto,Wahyu Widji Pamungkas,M Arif Yuana adalah bahwa sistem yang dikembangkan mampu memprediksi rentang waktu kenaikan air dengan cukup baik serta membantu mengurangi risiko kerugian yang lebih besar akibat banjir. Sistem ini menggunakan sensor ultrasonik untuk mendeteksi ketinggian air dan Arduino sebagai pemroses utama, sementara logika fuzzy digunakan untuk mengambil keputusan berdasarkan data ketinggian tersebut. Hasilnya ditampilkan melalui LCD dan dikirim melalui SMS menggunakan modul GSM, sehingga masyarakat dapat menerima peringatan secara jarak jauh. Meskipun sistem ini memiliki kekurangan seperti belum otomatisnya integrasi antara data curah hujan dan sistem fuzzy, serta input data yang masih manual, kelebihanannya terletak pada kemampuannya dalam memberikan prediksi waktu luapan air dan memberikan peringatan dini yang praktis dan efektif. Dengan pengembangan lebih lanjut, sistem ini berpotensi menjadi solusi teknologi yang berguna dalam mitigasi bencana banjir, terutama di wilayah padat penduduk.(Septianto et al., 2022)

Penelitian lainya oleh Leo Rizka adalah bahwa penggunaan data media sosial,

khususnya Twitter, terbukti dapat dimanfaatkan secara efektif untuk memprediksi lokasi bencana banjir dengan pendekatan teknologi terkini. Penelitian ini berhasil membangun sebuah model prediktif berbasis *Hidden Markov Model (HMM)* yang sebelumnya didukung oleh proses klasifikasi tweet menggunakan *Random Forest*, guna menyaring informasi yang relevan terkait banjir. (Riska, 2022) Dengan memanfaatkan tweet berisi laporan masyarakat yang dilengkapi dengan informasi lokasi dan waktu kejadian, sistem dapat mengidentifikasi pola dan tren yang berkaitan dengan potensi terjadinya banjir. Model prediksi ini memberikan hasil yang menjanjikan karena mampu mengatasi kekurangan dari data prediksi konvensional, seperti informasi lokasi banjir yang terlalu umum atau kurang spesifik. HMM memberikan keunggulan dalam menangani data deret waktu (*time series*) sehingga mampu memproyeksikan lokasi banjir secara lebih akurat berdasarkan urutan observasi. Selain itu, pendekatan ini juga memungkinkan integrasi data dari berbagai sumber—baik data historis kejadian banjir dari pemerintah maupun informasi real-time dari masyarakat—sehingga meningkatkan keandalan sistem prediksi.

Meski demikian, penelitian ini juga menyadari adanya tantangan, seperti ketergantungan pada kualitas dan kelengkapan data dari Twitter yang bisa saja tidak representatif atau bias. Namun, dengan adanya proses validasi dan perbandingan terhadap data aktual dari instansi resmi, model ini tetap menunjukkan potensi besar dalam mendukung pengambilan keputusan dalam mitigasi bencana. Secara keseluruhan, penelitian ini memberikan kontribusi yang signifikan dalam pengembangan sistem peringatan dini berbasis data terbuka dan media sosial, serta menjadi landasan yang kuat untuk pengembangan model serupa dalam penanganan bencana alam lainnya di masa depan. (Riska, 2022)

Secara keseluruhan, penelitian saya lebih berfokus pada pendekatan terkini

dalam memprediksi banjir dengan membandingkan dua metode kecerdasan buatan yang populer, yaitu Recurrent Neural Network dan Gradient Boosting. Berbeda dengan penelitian-penelitian lain yang lebih fokus pada klasifikasi dampak, pemetaan geospasial, atau sistem peringatan lokal berbasis sensor, penelitian Anda menawarkan analisis prediktif berbasis data historis dan metrik evaluatif yang kuat seperti MAE dan RMSE. Sementara beberapa studi seperti penggunaan Hidden Markov Model dari data Twitter atau logika fuzzy pada bendungan hanya menggunakan satu pendekatan dan terbatas secara cakupan, pendekatan Anda bersifat lebih adaptif, luas, dan aplikatif dalam skala nasional. Penelitian ini tidak hanya memperkaya literatur teknis, tetapi juga menyoroti kontribusi praktis dalam pengambilan keputusan kebijakan mitigasi bencana di Indonesia melalui sistem peringatan dini yang berbasis data dan efisien. (Riska, 2022)

2.2 Recurrent Neural Network (RNN)

2.2.1 Pengertian Recurrent Neural Network (RNN)

Neural Network atau Jaringan Saraf Tiruan (JST) adalah suatu metode komputasi yang terinspirasi oleh cara kerja otak manusia, di mana sejumlah unit pemrosesan yang disebut neuron saling terhubung dan bekerja secara paralel untuk memproses informasi. Dalam konteks prediksi banjir, JST digunakan untuk menganalisis dan memodelkan data hidrologi yang kompleks dan bersifat non-linear, seperti data curah hujan, debit sungai, tinggi muka air, serta faktor lingkungan lainnya yang berpengaruh terhadap terjadinya banjir. Keunggulan JST terletak pada kemampuannya untuk belajar dari data historis melalui proses pelatihan, sehingga dapat mengenali pola-pola tersembunyi dalam data yang sulit diidentifikasi dengan metode statistik konvensional. Dengan demikian, JST mampu memberikan prediksi yang lebih akurat dan responsif terhadap perubahan kondisi lingkungan yang dinamis. Penggunaan JST dalam sistem peringatan

dini banjir memungkinkan deteksi dini terhadap potensi banjir, sehingga dapat membantu pengambilan keputusan mitigasi dan evakuasi yang lebih efektif, mengurangi risiko kerugian materiil dan korban jiwa. Berbagai penelitian telah membuktikan efektivitas JST dalam meningkatkan akurasi prediksi banjir dibandingkan dengan metode tradisional, terutama ketika data yang tersedia bersifat tidak lengkap atau mengandung noise. (Mt et al., 2025)

2.2.2 Cara kerja Recurrent Neural Network (RNN)

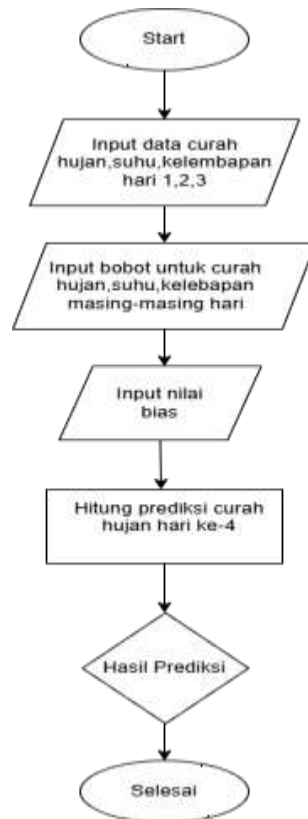
Recurrent Neural Network (RNN) adalah algoritma kecerdasan buatan yang dirancang untuk mengolah data berurutan, seperti data curah hujan atau tinggi muka air sungai, yang sangat relevan dalam prediksi banjir. RNN memiliki kemampuan untuk mengenali pola temporal dalam data time series, memungkinkan model untuk memprediksi kejadian banjir berdasarkan data historis. (Farisi et al., 2024)

Dalam konteks prediksi banjir, RNN bekerja dengan memproses data historis curah hujan dan tinggi muka air sungai untuk memprediksi kemungkinan terjadinya banjir di masa depan. Model ini dapat dilatih menggunakan data historis yang tersedia, seperti data curah hujan interval 1 jam sepanjang Januari hingga Desember 2022, untuk menghasilkan prediksi yang akurat. (Farisi et al., 2024)

Selain RNN, varian seperti Long Short-Term Memory (LSTM) juga telah digunakan dalam prediksi banjir. LSTM memiliki kemampuan untuk memproses data sekuensial dalam jangka waktu yang panjang, memungkinkan model untuk menangkap dependensi jangka panjang dalam data. Misalnya, dalam studi kasus di Jakarta, model RNN digunakan untuk memprediksi tinggi muka air di pintu air berdasarkan data historis cuaca dan tinggi muka air. (Nurashila et al., 2023)

Secara keseluruhan, penggunaan RNN dan LSTM dalam prediksi banjir di Indonesia telah terbukti efektif dalam mengolah data historis untuk menghasilkan

prediksi yang akurat, membantu dalam sistem peringatan dini dan mitigasi bencana banjir. (Nurashila et al., 2023)

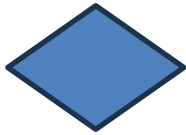


Gambar 2. 1flowchart RNN

Di bawah ini terdapat penjelasan dari flowchart atau bagaimana alur dari proses *kerja Recurrent Neural Network*

Tabel 2. 2 Penjelasan Flowchart atau alur

Proses	Simbol	Keterangan
Start		Titik awal proses prediksi curah hujan, menandai dimulainya alur eksekusi model
Input data curah hujan, suhu, kelembapan		Data input berupa nilai-nilai historis dari 3 hari sebelumnya untuk curah hujan, suhu, dan kelembapan. Digunakan sebagai fitur prediksi
Input bobot untuk		Bobot adalah nilai parameter yang

curah hujan, suhu dan kelembapan		menunjukkan seberapa besar pengaruh setiap fitur dalam perhitungan prediksi
Input nilai bias		Bias adalah nilai tambah dalam perhitungan prediksi untuk menyesuaikan output, agar tidak selalu bernilai nol saat input nol
Hitung prediksi curah hujan, suhu dan kelembapan		Perhitungan dilakukan menggunakan rumus prediksi linier
Hasil prediksi		Output dari model, yaitu nilai prediksi curah hujan untuk hari ke-4 berdasarkan input dan bobot sebelumnya
Selesai		Proses selesai setelah menghasilkan prediksi. Tidak ada langkah lanjut dalam alur ini

2.2.3 Keunggulan dan Kelemahan RNN

Recurrent Neural Network (RNN) memiliki keunggulan utama dalam kemampuannya memproses dan menganalisis data berurutan, yang sangat relevan untuk prediksi banjir yang bergantung pada data historis seperti curah hujan, tinggi muka air, dan debit sungai. Model ini dirancang untuk mempertahankan informasi dari waktu ke waktu melalui struktur hidden state, sehingga mampu mengenali pola temporal dalam data. Dalam konteks sistem peringatan dini banjir, hal ini memungkinkan RNN untuk memberikan prediksi yang dinamis dan terus diperbarui berdasarkan data terbaru. RNN juga fleksibel dalam menangani berbagai jenis format input dan output, baik itu satu-ke-satu, satu-ke-banyak, maupun banyak-ke-banyak, menjadikannya berguna untuk aplikasi

prediksi jangka pendek maupun jangka panjang.(Putra, Amrullah, Fauzi, & Saputri, 2025)

Namun, meskipun memiliki potensi besar, RNN tidak lepas dari kelemahan. Salah satu keterbatasan paling menonjol adalah ketidakmampuannya dalam mengingat pola jangka panjang secara efektif karena masalah vanishing gradient. Masalah ini menyebabkan sinyal pembelajaran melemah seiring bertambahnya panjang urutan data, sehingga pola yang terjadi jauh sebelumnya cenderung diabaikan oleh model. Selain itu, proses pelatihan RNN relatif lambat dan memerlukan banyak iterasi serta sumber daya komputasi yang besar, terutama ketika arsitektur diperluas atau data yang digunakan berskala besar. RNN juga cenderung sensitif terhadap parameter pelatihan dan memerlukan tuning yang cermat agar dapat bekerja optimal. Dalam kondisi di mana data yang digunakan mengandung noise atau tidak lengkap, kinerja RNN bisa menurun drastis jika tidak dilakukan preprocessing data secara tepat.(Putra, Amrullah, Fauzi, & Saputri, 2025)

Keunggulan dan kelemahan ini telah dibahas dalam sejumlah literatur ilmiah seperti oleh Goodfellow et al. (2016) dalam Deep Learning (MIT Press), serta dalam ulasan oleh Lipton et al. (2015, diperbarui 2021) dalam makalah mereka A Critical Review of Recurrent Neural Networks for Sequence Learning (arXiv:1506.00019). Di Indonesia, penerapan RNN dalam konteks prediksi banjir juga telah diteliti oleh Rahman et al. (2022) melalui studi kasus di Jakarta, di mana mereka menunjukkan bahwa RNN efektif dalam menangkap pola curah hujan dan tinggi muka air, namun memerlukan waktu pelatihan dan tuning yang signifikan untuk mencapai akurasi yang tinggi. (Goodfellow, I., Bengio, Y., & Courville, A. 2016).

2.3 Gradient Boosting

2.3.1 Pengertian Gradient Boosting

Gradient Boosting adalah suatu metode dalam machine learning yang termasuk dalam kategori teknik ensemble learning, yang menggabungkan banyak model prediksi lemah (biasanya berupa decision tree sederhana) untuk membentuk sebuah model yang kuat dan akurat. Pendekatan ini bekerja secara iteratif dengan tujuan utama mengurangi kesalahan prediksi dari model sebelumnya melalui proses pembelajaran berbasis gradien. Pada setiap iterasi, model baru dibangun dengan mempelajari *residual* atau selisih antara nilai aktual dan nilai prediksi sebelumnya, lalu model tersebut digunakan untuk memperbaiki kesalahan tersebut. Proses ini berlanjut hingga jumlah iterasi yang ditentukan tercapai atau hingga model mencapai performa optimal. (Putra, Amrullah, Fauzi, & Saputri, 2025)

Salah satu keunggulan utama Gradient Boosting adalah fleksibilitasnya dalam menangani berbagai jenis data dan masalah, baik untuk regresi maupun klasifikasi. Selain itu, metode ini mampu menangani data yang kompleks dan tidak terstruktur, serta memberikan akurasi tinggi dalam prediksi jika disetel dengan baik. Gradient Boosting juga mendukung penggunaan berbagai fungsi loss, seperti mean squared error untuk regresi atau log-loss untuk klasifikasi, sehingga dapat diadaptasi sesuai dengan kebutuhan spesifik dari masalah yang dihadapi. Model ini juga dikenal karena kemampuannya menangkap hubungan non-linear dan interaksi antar fitur, menjadikannya populer dalam berbagai kompetisi dan aplikasi nyata. (Putra, Amrullah, Fauzi, & Saputri, 2025)

Meskipun memiliki banyak keunggulan, Gradient Boosting juga memiliki beberapa tantangan. Metode ini cenderung lebih lambat dalam proses pelatihan karena sifat iteratifnya, serta memerlukan pemilihan dan penyetelan hyperparameter yang

cermat, seperti learning rate, jumlah pohon, dan kedalaman pohon. Jika tidak ditangani dengan baik, Gradient Boosting juga berisiko mengalami overfitting, terutama ketika jumlah pohon terlalu banyak atau model terlalu kompleks.

Konsep Gradient Boosting pertama kali diperkenalkan secara formal oleh Jerome H. Friedman dalam artikelnya yang berjudul “Greedy Function Approximation: A Gradient Boosting Machine” yang diterbitkan dalam *Annals of Statistics* pada tahun 2001. Artikel ini menjadi landasan penting dalam pengembangan metode boosting modern. Selain itu, penelitian lanjutan oleh Chen dan Guestrin pada tahun 2016 menghasilkan implementasi XGBoost, yang secara signifikan meningkatkan efisiensi dan skalabilitas Gradient Boosting, menjadikannya salah satu algoritma paling populer di kalangan praktisi data saat ini. (Chen, T., & Guestrin)

2.3.2 Cara kerja Gradient Boosting

Cara kerja Gradient Boosting merupakan suatu pendekatan pembelajaran mesin yang sangat efisien dalam meningkatkan akurasi prediksi melalui penggabungan model-model lemah secara bertahap. Teknik ini bekerja berdasarkan prinsip iteratif, di mana setiap model baru dibangun untuk memperbaiki kesalahan dari model sebelumnya. Proses ini menggunakan pendekatan optimasi berbasis gradien, yaitu dengan meminimalkan fungsi kerugian (loss function) secara berulang melalui penambahan model-model yang secara khusus dirancang untuk memprediksi sisa kesalahan (residual error) dari model sebelumnya.

Pada tahap awal, Gradient Boosting dimulai dengan membentuk sebuah model sederhana (biasanya decision tree berukuran kecil) yang digunakan untuk memprediksi target atau label. Hasil prediksi ini kemudian dibandingkan dengan nilai aktual untuk menghitung error atau residual. Error ini menunjukkan seberapa besar kesalahan model dalam memprediksi target. Setelah itu, sebuah model baru dibangun dengan tujuan

mempelajari pola dari residual tersebut, sehingga dapat memperbaiki kesalahan yang telah dibuat oleh model sebelumnya. Proses ini terus diulang: setiap model selanjutnya akan dilatih untuk memprediksi kesalahan dari model gabungan sebelumnya. (Nurchahyo & Saukani, 2025)

Proses penggabungan model dilakukan secara aditif. Artinya, setiap model yang baru ditambahkan akan memperbarui hasil prediksi akhir berdasarkan prediksi model sebelumnya ditambah hasil koreksi dari model baru. Formula umum yang digunakan dalam Gradient Boosting adalah:

$$F_m(x) = F_{m-1}(x) + y \cdot h_m(x)$$

Dalam rumus ini, $F_m(x)$ merepresentasikan model gabungan pada iterasi ke- m , sedangkan $F_{m-1}(x)$ adalah model gabungan pada iterasi sebelumnya. Fungsi $h_m(x)$ merupakan pohon keputusan baru yang memprediksi kesalahan dari $F_{m-1}(x)$, dan y adalah learning rate atau faktor pengurang yang mengontrol kontribusi setiap model baru agar tidak terlalu dominan dan menghindari overfitting terhadap data pelatihan. Semakin kecil nilai y , maka proses pelatihan akan berlangsung lebih lambat, namun cenderung lebih stabil dan akurat jika dilakukan dengan jumlah iterasi yang cukup banyak.

Sebagai ilustrasi, misalnya sebuah model pertama memperkirakan curah hujan besok sebesar 60 mm, padahal nilai aktualnya adalah 70 mm. Selisih 10 mm ini merupakan residual error yang harus diperbaiki. Maka model kedua akan dibuat untuk memperkirakan nilai kesalahan tersebut. Jika model kedua berhasil memprediksi kesalahan sebesar 8 mm, maka hasil gabungan model pertama dan kedua akan memberikan prediksi sebesar 68 mm. Model ketiga akan memperkirakan sisa error yang masih ada, yakni 2 mm, dan seterusnya. Proses ini terus berlanjut hingga error menjadi sangat kecil atau sampai batas iterasi yang telah ditentukan tercapai.

Dengan mekanisme seperti ini, Gradient Boosting secara bertahap memperbaiki kelemahan-kelemahan prediksi awal dan membentuk sebuah model akhir yang sangat kuat. Keunggulan utama dari metode ini adalah kemampuannya dalam menangani data yang kompleks, non-linear, dan memiliki banyak variabel input. Selain itu, metode ini fleksibel dalam menggunakan berbagai jenis fungsi kerugian, termasuk fungsi untuk regresi, klasifikasi, dan peringkat. Namun, tantangannya adalah pengaturan parameter yang sensitif seperti learning rate, jumlah pohon, dan kedalaman pohon, yang jika tidak dioptimalkan dapat menyebabkan overfitting atau underfitting. (Nurcahyo & Saukani, 2025)

Gradient Boosting kini menjadi salah satu teknik paling populer dalam berbagai kompetisi data science dan penerapan praktis, seperti prediksi bencana alam, sistem rekomendasi, analisis risiko, dan diagnosis medis, karena kemampuannya yang luar biasa dalam membangun model prediktif yang sangat akurat.

Pola perhitungannya:

Gradient boosting menggunakan banyak pohon keputusan yang bekerja secara bertahap untuk memperbaiki kesalahan prediksi

Rumus utama:

$$F_m(x) = f_{m-1}(x) + y \cdot h_m(x)$$

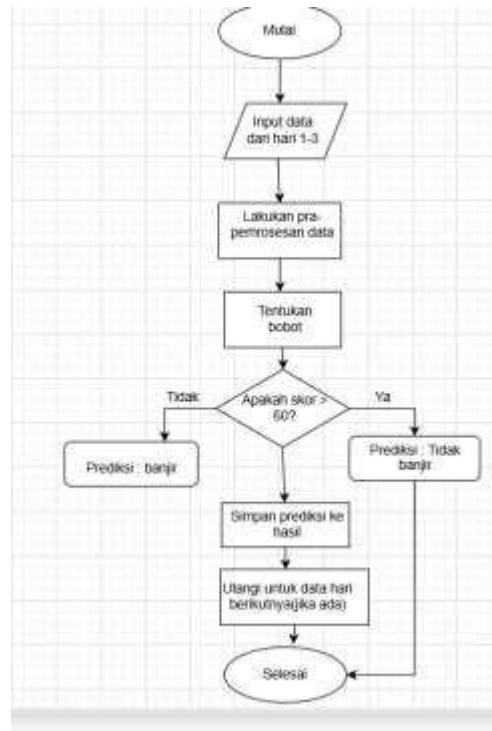
Penjelasan:

$F_m(x)$ => Model di tahap ke-m (semakin banyak tahap, semakin akurat)

$F_{m-1}(x)$ => Model sebelumnya

$h_m(x)$ => Pohon keputusan baru yang memprediksi kesalahan dari model sebelumnya

y => Faktor pengurangan supaya model tidak terlalu menyesuaikan dengan data (agar tidak overfitting)

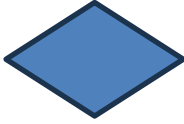


Gambar 2. 2 flowchart *Gradient Boosting*

Dibawah ini terdapat penjelasan dari flowchart atau bagaimana alur dari proses kerja *Gradient Boosting*

Tabel 2. 3 penjelasan flowchart atau alur

Proses	Simbol	Keterangan
Mulai		Titik awal dimulai proses prediksi banjir berdasarkan data curah hujan 3 hari
Input data dari hari 1-3		Menginput data histori sebelumnya dan hari ke-3 untuk proses prediksi
Lakukan prapemrosesan data		Data diberikan disiapkan (normalisasi, transformasi, penghapusan, data kosong, dll) agar siap digunakan dalam model

		prediksi
Tentukan bobot		Menentukan bobot yang sesuai untuk masing-masing fitur
Apakah skor ≥ 60 ?		Mengevaluasi skor hasil kombinasi bobot dan fitur. Jika lebih atau pas 60 banjir jika kurang dari 60 tidak banjir
Prediksi : banjir		Jika skor lebih dari 60, maka diprediksi banjir
Prediksi: Tidak banjir		Jika skor tidak lebih dari 60 atau tidak sampai 60 maka tidak banjir
Simpan prediksi ke hasil		Hasil prediksi (banjir atau tidak) di simpan ke dalam daftar hasil
Ulangi untuk data hari berikutnya		Jika masih ada data hari lainnya, proses diulang untuk melakukan prediksi kembali
Selesai		Menandakan proses prediksi selesai setelah data hari yang tersedia

2.3.3 Keunggulan dan Kelemahan Gradient Boosting

Dalam konteks prediksi banjir, keunggulan Gradient Boosting terletak pada kemampuannya menangani data lingkungan yang kompleks dan dinamis, seperti curah hujan, ketinggian muka air, suhu, kelembaban, serta data spasial dan temporal lainnya. Metode ini sangat efektif untuk mengidentifikasi pola tidak linear yang sering kali

muncul dalam data iklim dan hidrologi, sehingga prediksi kejadian banjir dapat dilakukan dengan tingkat akurasi yang tinggi. Gradient Boosting juga mampu memanfaatkan banyak fitur secara bersamaan dan tetap memberikan hasil yang stabil, selama parameter model diatur dengan benar. Hal ini penting karena prediksi banjir memerlukan integrasi dari berbagai sumber data yang berbeda-beda, baik dalam skala waktu maupun ruang.(Nurcahyo & Saukani, 2025)

Namun di sisi lain, penerapan Gradient Boosting untuk sistem peringatan dini banjir juga menghadapi tantangan. Salah satunya adalah kebutuhan komputasi yang tinggi, yang bisa menjadi masalah jika sistem harus memberikan prediksi secara real-time atau diterapkan di daerah yang memiliki keterbatasan infrastruktur teknologi. Selain itu, interpretasi hasil dari model Gradient Boosting tidak selalu intuitif bagi para pengambil kebijakan atau masyarakat awam, karena model ini terdiri dari banyak pohon keputusan yang bekerja secara bersama-sama. Jika tidak dilengkapi dengan penjelasan tambahan atau sistem visualisasi, penggunaan model ini bisa menyulitkan dalam proses pengambilan keputusan cepat saat menghadapi potensi banjir.(Nurcahyo & Saukani, 2025)

2.4 Perbandingan RNN dan Gradient Boosting dalam Prediksi Banjir

Tabel 2. 4 Tabel Perbandingan prediksi

Aspek	RNN(Recurrent Neural Network)	Gradient Boosting
Jenis Data	Time series / data berurutan	Data tabular, numerik, kategorikal
Kemampuan Temporan	Sangat baik, mampu mengingat urutan data jangka pendek /	Tidak mempertimbangkan

	panjang	urutan waktu
Akurasi	Tinggi pada data sekuensial	Sangat akurat untuk data non-sekuensial
Kecepatan Pelatihan	Lebih lambat, membutuhkan waktu dan sumber daya komputasi lebih besar	Lebih cepat dan efisien pada umumnya
Interpretabilitas	Rendah, sulit dijelaskan tanpa metode tambahan	Lebih mudah di jelaskan
Kebutuhan komputasi	Tinggi, terutama pada model LSTM/GRU	Lebih rendah, bisa di optimalkan dengan efisien
Ketahanan terhadap missing value	Kurang toleran, perlu praproses yang baik	Lebih toleran terhadap data yang tidak lengkap
Aplikasi ideal	Prediksi banjir berbasis pola histori curah hujan/debit sungai	Prediksi dampak banjir berdasarkan banyak fitur statis dan dinamis

Perbandingan antara Recurrent Neural Network (RNN) dan Gradient Boosting dalam konteks prediksi banjir mencerminkan perbedaan mendasar dalam pendekatan pemrosesan data dan karakteristik model yang digunakan, meskipun keduanya termasuk dalam kategori metode kecerdasan buatan.(Nurcahyo & Saukani, 2025)

RNN secara khusus dirancang untuk menangani data berurutan atau time series, seperti data curah hujan harian, ketinggian permukaan air sungai dari waktu ke waktu, atau debit air. Keunggulan utama RNN terletak pada kemampuannya dalam “mengingat” informasi sebelumnya melalui struktur loop dalam jaringan sarafnya. Ini

memungkinkan RNN untuk mempertahankan konteks temporal, yang sangat penting dalam prediksi banjir, karena pola bencana ini sering kali dipengaruhi oleh kondisi yang terjadi beberapa waktu sebelumnya. Selain itu, RNN sangat baik digunakan ketika urutan dan dinamika waktu dari data memiliki pengaruh kuat terhadap hasil prediksi.(Nurashila et al., 2023)

Sementara itu, Gradient Boosting lebih fleksibel dalam menangani berbagai jenis data, termasuk data kategorikal, numerik, dan bahkan data yang memiliki missing value, karena pendekatannya yang berbasis pada pohon keputusan. Model ini bekerja dengan membangun serangkaian pohon secara bertahap, di mana setiap pohon baru dirancang untuk memperbaiki kesalahan dari pohon sebelumnya. Gradient Boosting unggul dalam hal akurasi prediksi pada banyak kasus, dan biasanya lebih cepat untuk dilatih dibandingkan RNN, terutama jika data tidak terlalu panjang secara sekuensial. Namun, model ini tidak mempertahankan urutan data seperti RNN, sehingga kurang ideal jika pola temporal sangat berpengaruh.(Nurashila et al., 2023)

Dalam hal interpretabilitas, Gradient Boosting cenderung lebih mudah dijelaskan menggunakan teknik seperti feature importance atau SHAP values. Ini berbeda dengan RNN yang sering dianggap sebagai "black box" karena sulitnya menjelaskan bagaimana keputusan akhir dibuat, kecuali jika digunakan pendekatan interpretabilitas khusus seperti attention mechanism atau LSTM interpretability tools.(Nurashila et al., 2023)

Dari segi kebutuhan komputasi, RNN, terutama tipe-tipe yang kompleks seperti LSTM atau GRU, memerlukan sumber daya lebih besar untuk pelatihan dan bisa memakan waktu lebih lama. Sebaliknya, Gradient Boosting seperti XGBoost atau LightGBM dapat berjalan lebih cepat dan efisien pada banyak kasus, serta lebih mudah diimplementasikan pada sistem dengan keterbatasan daya komputasi.(Nurashila et al.,

2023)

Dengan kata lain, pemilihan antara RNN dan Gradient Boosting sangat bergantung pada jenis data yang digunakan, tujuan spesifik dari sistem peringatan dini banjir, dan kondisi lingkungan atau infrastruktur yang tersedia. RNN ideal untuk data yang kuat unsur waktunya dan memerlukan pemahaman urutan jangka panjang, sementara Gradient Boosting lebih cocok untuk data tabular dengan banyak fitur berbeda yang tidak selalu berurutan.

RNN berfokus pada pola dan urutan waktu dalam data (time series), cocok untuk memprediksi berdasarkan hubungan antar waktu. Gradient Boosting berfokus pada hubungan antar fitur dan pengurangan error secara bertahap, cocok untuk data tabular tanpa urutan waktu.(Nurashila et al., 2023)

Tabel 2. 5 Perbandingan akurasi

Aspek	RNN(Recurrent Neural Network)	Gradient Boosting
Fokus Utama	Menangkap pola berurutan dari data time series	Menangani data kompleks dan tabular tanpa urutan waktu
Akurasi	Sangat baik untuk data yang bergantung pada urutan waktu	Tinggi untuk data tabular dan non-sekuensial
Efisiensi Komputasi	Lebih lambat, butuh lebih banyak memori dan waktu pelatihan	Lebih cepat dan efisien terutama pada data besar
Kebutuhan data	Membutuhkan data	Tidak terlalu bergantung

	berurutan(time series) dan preprocessing lebih detail	pada urutan data
Kelebihan utama	Kemampuan mempelajari pola jangka panjang dari data sekuensial	Akurat, fleksibel dan mudah diinterpretasikan
Kelemahan utama	Rentan overfitting dan butuh parameter tuning yang cepat	Bisa sensitif terhadap noise dan overfitting jika tidak teratur
Cocok untuk	Prediksi curah hujan, ketinggian air sungai, pola waktu musiman	Klasifikasi resiko, prediksi berdasarkan banyak variabel

Penjelasan dari tabel di atas:

Tabel ini membandingkan dua metode dalam dunia machine learning, yaitu Recurrent Neural Network (RNN) dan Gradient Boosting. Perbedaan utama di antara keduanya terletak pada jenis data yang mereka tangani dan bagaimana mereka mempelajarinya.

RNN dirancang khusus untuk memahami pola dari data yang berurutan atau time series. Misalnya, data seperti curah hujan harian, ketinggian air sungai dari waktu ke waktu, atau pola musim yang terjadi secara berkala. Karena RNN bisa mengingat informasi dari masa lalu, ia sangat cocok untuk memahami hubungan waktu dalam data. Namun, karena kemampuannya itu, RNN biasanya lebih lambat dan membutuhkan memori serta proses pelatihan yang lebih berat. Ia juga cukup sensitif terhadap overfitting, terutama bila data dan parameter tidak ditangani dengan tepat.(Farisi et al., 2024)

Berbeda dengan RNN, Gradient Boosting lebih unggul dalam menangani data yang kompleks tapi tidak bergantung pada urutan, seperti data tabel biasa yang berisi banyak variabel. Metode ini mampu memberikan akurasi tinggi dan lebih cepat dalam proses pelatihan, terutama jika digunakan pada dataset yang besar. Ia juga dikenal fleksibel dan mudah untuk ditafsirkan hasilnya. Meskipun begitu, Gradient Boosting juga bisa overfitting, terutama jika data yang diberikan terlalu kacau atau tidak teratur.

Dalam konteks penggunaan, RNN lebih cocok digunakan untuk prediksi yang berhubungan dengan waktu, seperti pola musim atau peristiwa yang berulang. Sedangkan Gradient Boosting lebih pas dipakai untuk klasifikasi risiko atau prediksi berdasarkan banyak variabel tanpa memperhatikan urutan waktu.(Farisi et al., 2024)

Jadi, pilihan antara RNN dan Gradient Boosting tergantung pada jenis data dan kebutuhan analisisnya. Kalau Hana sedang menangani data yang berjalan dari waktu ke waktu, RNN mungkin pilihan yang lebih tepat. Tapi kalau datanya berupa tabel biasa dan ingin hasil yang cepat serta akurat, Gradient Boosting bisa jadi pilihan yang kuat.(Farisi et al., 2024)

BAB III METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Penelitian ini dilakukan dengan pendekatan kuantitatif yang bersifat eksperimen, di mana dua metode kecerdasan buatan, yaitu Recurrent Neural Network (RNN) dan Gradient Boosting, dibandingkan untuk menilai efektivitasnya dalam memprediksi kejadian banjir. Pemilihan metode ini didasarkan pada kemampuan masing-masing algoritma dalam menangani data lingkungan yang kompleks dan dinamis, serta relevansinya terhadap data historis yang mencerminkan kondisi aktual curah hujan dan ketinggian muka air sungai. Tujuan dari penelitian ini adalah untuk memperoleh gambaran yang menyeluruh mengenai kelebihan dan kelemahan kedua metode, baik dari segi akurasi prediksi maupun efisiensi dalam proses pelatihan model.(Farisi et al., 2024)

Penelitian dilaksanakan di wilayah Kabupaten Deli Serdang, Sumatera Utara, tepatnya di Jalan Meteorologi Raya No. 17, Tembung, Kecamatan Percut Sei Tuan. Wilayah ini dipilih karena memiliki karakteristik geografis yang cukup rentan terhadap bencana banjir, sehingga sangat relevan sebagai studi kasus. Adapun pelaksanaan penelitian direncanakan berlangsung selama enam bulan, dimulai pada bulan Juni hingga November 2025. Dalam kurun waktu tersebut, kegiatan penelitian mencakup serangkaian tahapan seperti studi literatur, pengumpulan data, preprocessing data, pembangunan model, pelatihan dan pengujian, evaluasi, hingga penyusunan laporan akhir.

Data yang digunakan dalam penelitian ini merupakan data sekunder yang diperoleh dari instansi resmi seperti Badan Meteorologi, Klimatologi, dan Geofisika (BMKG), serta lembaga-lembaga hidrologi dan lingkungan hidup yang relevan. Data

tersebut terdiri dari catatan historis curah hujan, tinggi muka air, dan parameter iklim lainnya yang dapat memengaruhi kejadian banjir. Sebelum digunakan dalam pemodelan, data terlebih dahulu melalui tahap preprocessing untuk memastikan konsistensi, keakuratan, dan kesiapan dalam pengolahan. Proses ini mencakup pembersihan dari nilai-nilai yang hilang atau ekstrem, normalisasi data agar berada dalam skala yang seragam, serta pembagian dataset menjadi data pelatihan dan data pengujian dengan proporsi tertentu.(Farisi et al., 2024).

Setelah proses pembersihan dan persiapan data selesai, model Recurrent Neural Network (RNN) dibangun untuk menangkap pola temporal dalam data time series. Model ini dirancang untuk memahami hubungan antara peristiwa yang terjadi pada waktu-waktu sebelumnya dengan kejadian banjir di masa mendatang. Sementara itu, model Gradient Boosting dikembangkan untuk memanfaatkan kekuatan decision tree dalam menganalisis hubungan antar variabel yang kompleks, tanpa mempertimbangkan urutan waktu. Kedua model kemudian dilatih menggunakan data pelatihan yang telah disiapkan, dan diuji pada data pengujian untuk mengukur performa prediksi. Proses pelatihan dilakukan dengan parameter awal yang disesuaikan melalui teknik tuning hyperparameter, guna mengoptimalkan hasil prediksi dari masing-masing model.

Selain mengevaluasi akurasi, penelitian ini juga menganalisis efisiensi waktu pelatihan dan pengujian masing-masing metode. Hal ini penting karena dalam konteks sistem peringatan dini bencana, waktu menjadi faktor krusial yang memengaruhi kecepatan respon terhadap potensi ancaman. Model yang cepat dan efisien dalam menghasilkan prediksi akan lebih bermanfaat bagi instansi penanggulangan bencana dibandingkan model yang meskipun akurat, namun lambat dan berat secara komputasi.(Farisi et al., 2024)

Penelitian ini pada akhirnya bertujuan untuk menarik kesimpulan yang dapat dijadikan rekomendasi praktis bagi pengembangan sistem peringatan dini banjir yang berbasis kecerdasan buatan di Indonesia. Dengan menganalisis kelebihan dan kekurangan kedua metode secara mendalam, baik dari segi teknis maupun aplikatif, diharapkan hasil penelitian ini dapat memberikan kontribusi nyata dalam pengambilan keputusan berbasis data, khususnya dalam mitigasi risiko banjir yang terus meningkat akibat perubahan iklim dan urbanisasi yang pesat.

3.1.1 Teknik Pengolahan Data

Data yang sudah di kumpulkan kemudian disusun terlebih dahulu ke dalam Microsoft Excel agar lebih terstruktur dan mudah dipahami, sehingga setiap variabel yang dibutuhkan dapat terlihat jelas. Setelah itu, data tersebut diolah lebih lanjut menggunakan Google colab, data dapat diproses secara sistematis, dilakukan pembersihan (data cleaning), visualisasi, hingga penerapan berbagai algoritma untuk mendapatkan hasil yang lebih akurat. Selain itu, pada tahap akhir pemrosesan, Google colab akan menampilkan hasil analisis seperti label banjir, tingkat akurasi, dan metrik lainnya yang mendukung evaluasi kinerja model. Proses ini tidak hanya membantu dalam mengorganisir informasi, tetapi juga memberikan fleksibilitas untuk melakukan eksplorasi data secara lebih mendalam dan efisien, sehingga hasil yang diperoleh dapat digunakan sebagai dasar pengambilan keputusan atau pembuatan laporan penelitian yang lebih berkualitas .



Gambar 3. 1 Google Colab



Gambar 3. 2 Microsoft Excel

3.2 Tempat dan waktu penelitian

3.2.1 Tempat Penelitian

Penelitian “Analisis Perbandingan Metode Recurrent Neural Network dan Gradient Bossting Untuk Prediksi Banjir”, Tempat pelaksanaan penelitian di JL.Meteorologi Raya No.17,Tembung, kec. Percut Sei Tuan, Kabupaten Deli Serdang, Sumatera Utara.



Gambar 3. 3 Tempat Penelitian

3.2.2 Waktu dan Jadwal Pelaksanaan Penelitian

Pelaksanaan waktu penelitian di laksanakan pada juni 2025, adapun jadwal penelitian sebagai berikut:

Tabel 3. 1 Jadwal Pelaksanaan Penelitian

No	Keterangan	Bulan I	Bulan II	Bulan III	Bulan IV	Bulan V	Bulan VI
1	Pengumpulan Data						
2	Preprocessing Data						
3	Pembangunan Model						
4	Pelatihan dan Pengujian Model						
5	Evaluasi Model						

6	Analisis dan Kesimpulan						
---	----------------------------	--	--	--	--	--	--

3.3 Sumber dan teknik Pengumpulan Data

Data yang digunakan dalam penelitian ini bersifat sekunder dan diperoleh dari lembaga-lembaga resmi pemerintah serta institusi yang memiliki otoritas dan kredibilitas dalam pengelolaan data iklim dan hidrologi. Di antaranya adalah Badan Meteorologi, Klimatologi, dan Geofisika (BMKG), Dinas Lingkungan Hidup, serta lembaga pengelola sumber daya air yang memiliki catatan historis terkait curah hujan, ketinggian muka air, dan parameter-parameter hidrometeorologi lainnya. Data tersebut mencakup periode waktu tertentu yang telah disesuaikan dengan kebutuhan penelitian, dengan harapan dapat merepresentasikan kondisi lapangan secara aktual dan objektif.

Pengumpulan data dilakukan melalui permintaan langsung kepada instansi terkait, baik dalam bentuk surat permohonan maupun akses terhadap repositori data online jika tersedia. Data yang dikumpulkan kemudian diorganisir ke dalam format yang seragam, sehingga dapat diproses lebih lanjut dengan metode komputasi. Dalam hal ini, validitas dan reliabilitas data sangat diperhatikan, sebab kualitas data yang digunakan akan sangat menentukan akurasi dari hasil prediksi model. Oleh karena itu, sebelum data digunakan dalam tahap pemodelan, peneliti terlebih dahulu melakukan verifikasi dan pembersihan data untuk memastikan bahwa data yang digunakan benar-benar layak untuk dianalisis secara ilmiah.

Data yang saya gunakan dalam penelitian ini adalah data sekunder dengan menggunakan metode RNN dan Gradient Boosting Adapun jenis jenis data yang saya gunakan dalam Metode RNN dan Gradient Boosting yaitu:

Tabel 3. 2 Jenis Data RNN dan Gradient Boosting

NO	RNN	Gradient Boosting
1	Curah Hujan(mm)	Curah Hujan(mm)
2	Suhu ($^{\circ}\text{C}$)	Kelembapan(%)
3	Kelembapan(%)	Ketinggian (mdpl)
4	-	Jarak ke sungai (Km)
5	-	Jenis Tanah

3.4 Teknik Pengolahan dan Analisis Data

3.4.1 Preprocessing data

1. Tahapan awal dalam proses analisis data

adalah *preprocessing*, yang merupakan fase krusial untuk memastikan bahwa data yang digunakan benar-benar bersih, relevan, dan siap untuk diolah oleh algoritma kecerdasan buatan. Dalam penelitian ini, data yang diperoleh dari instansi resmi tidak selalu dalam kondisi ideal untuk langsung digunakan. Oleh karena itu, dilakukan serangkaian proses pembersihan yang mencakup identifikasi dan penanganan nilai yang hilang (*missing values*), duplikasi data, serta data ekstrim (*outlier*) yang berpotensi mengganggu stabilitas model.(Rodhi & Saputra, 2024)

Proses pembersihan ini tidak dilakukan secara sembarangan, melainkan dengan mempertimbangkan nilai distribusi dan relevansi statistik dari tiap-tiap variabel. Setelah data dianggap bersih, langkah selanjutnya adalah normalisasi, yaitu proses menyetarakan skala nilai pada tiap fitur agar berada dalam rentang yang seragam, biasanya antara 0 dan 1. Normalisasi penting dilakukan terutama untuk model *Recurrent Neural Network* yang sangat sensitif terhadap perbedaan skala antar fitur. Setelah itu, data dibagi menjadi dua kelompok utama, yakni data latih yang digunakan

untuk membangun model dan data uji yang digunakan untuk mengukur kemampuan generalisasi model terhadap data baru. (Rodhi & Saputra, 2024)

Pembagian ini dilakukan dengan proporsi 80:20 atau 70:30 tergantung pada kondisi dataset, dan menggunakan teknik *random shuffle* untuk memastikan distribusi yang merata. Dalam beberapa kasus, dilakukan juga pelatihan dengan validasi silang agar model tidak mengalami bias akibat pembagian data yang kurang representatif. (Farisi et al., 2024).

Ada beberapa tahapan Preprocessing data yaitu:

Tabel 3. 3 Tahapan dari Preprocessing Data

No.	Tahapan	Keterangan
1	Pembersihan Data (<i>Data Cleaning</i>)	2. Mengidentifikasi dan penanganan missing values 3. Menghapus duplikasi data 4. Deteksi dan penanganan outlier
2	distribusi dan relevansi statistik	1. Menilai apakah Variabel memiliki distribusi yang sesuai untuk analisis 2. Menghapus atau merevisi
3	Normalisasi Data (<i>Feature Scaling</i>)	1. Menyamaratakan skala nilai antar fitur (fitur=kolom/variabel) 2. Umumnya menggunakan Min-Max Normalization untuk skala 0-1 3. Hal ini penting agar algoritma seperti RNN dapat bekerja secara optimal tanpa terganggu perbedaan skala antar fitur

4	Pembagian dataset	1. Data latih (<i>Training set</i>) untuk melatih model 2. Data Uji (<i>Testing Set</i>) untuk mengevaluasi kinerja model terhadap data yang belum pernah dilihat 3. Rasio umum : 80:20 atau 70:30, tergantung ukuran dataset 4. Pembagian dilakukan secara acak (<i>Random shuffle</i>) untuk menghindari bias distribusi
5	Validasi silang (Cross-Validation)	1. Diterapkan bila diperlukan untuk menghindari overfitting dan meningkatkan generalisasi model 2. Contohnya : K-Fold Cross Validation

2. Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) atau analisis data eksploratif adalah proses awal yang dilakukan untuk memahami struktur, pola, dan hubungan antar variabel dalam suatu dataset. Pada tahap ini, digunakan berbagai teknik statistik deskriptif dan visualisasi data untuk menggali informasi penting, seperti distribusi nilai setiap fitur, potensi korelasi antar fitur, serta mendeteksi nilai ekstrem (outlier).

Data yang ditampilkan terdiri dari dua bagian utama. Bagian pertama mencakup informasi harian seperti Tanggal, Curah Hujan (mm), Suhu (°C), Kelembapan (%), dan Ketinggian (mdpl). Data ini memberikan gambaran kondisi cuaca dan lingkungan pada tiap hari. Bagian kedua menambahkan informasi spasial dan jenis tanah, seperti Jarak ke Sungai (km), Jenis Tanah, dan Label Banjir, yang menunjukkan apakah terjadi banjir (1) atau tidak (0) pada kondisi tersebut.

Dengan melakukan EDA terhadap data ini, peneliti dapat mengidentifikasi tren seperti bagaimana curah hujan tinggi atau jenis tanah tertentu dapat berkaitan dengan terjadinya banjir, serta bagaimana faktor lingkungan lainnya seperti kelembapan dan ketinggian memengaruhi potensi bencana. Wawasan yang diperoleh dari tahap ini sangat penting untuk menentukan fitur yang relevan dan membangun model prediksi banjir yang akurat.

	Tanggal	Curah Hujan (mm)	Suhu (°C)	Kelembapan (%)	Ketinggian (mdpl)
0	2022-01-01	37.45	26.20	86.83	20
1	2022-01-02	95.07	25.86	82.42	25
2	2022-01-03	73.20	31.80	64.44	25
3	2022-01-04	59.87	27.50	77.88	15
4	2022-01-05	15.60	24.58	78.41	25

	Jarak ke Sungai (km)	Jenis Tanah	Label Banjir
0	4.67	Aluvial	0
1	0.68	Latosol	1
2	4.68	Aluvial	0
3	3.45	Podsolik	0
4	2.70	Latosol	0

Gambar 3. 4 Data Dummy Prediksi banjir

Fitur (x):

	Tanggal	Curah Hujan (mm)	Suhu (°C)	Kelembapan (%)
0	2022-01-01	37.45	26.20	86.83
1	2022-01-02	95.07	25.86	82.42
2	2022-01-03	73.20	31.80	64.44
3	2022-01-04	59.87	27.50	77.88
4	2022-01-05	15.60	24.58	78.41

	Jarak ke Sungai (km)	Jenis Tanah	Label Banjir
0	4.67	Aluvial	0
1	0.68	Latosol	1
2	4.68	Aluvial	0
3	3.45	Podsolik	0
4	2.70	Latosol	0

Target (y):

0	20
1	25
2	25
3	15
4	25

Gambar 3. 5 fitur X

Fitur (x) setelah normalisasi:

	Curah Hujan (mm)	Suhu (°C)	Kelembapan (%)	Jarak ke Sungai (km)
0	-0.408809	-0.985015	0.611720	1.538870
1	1.552938	-1.102391	0.229332	-1.286258
2	0.808346	0.948225	-1.329698	1.545951
3	0.354509	-0.536227	-0.164327	0.675046
4	-1.152720	-1.544274	-0.118371	0.144007

	Label Banjir
0	-0.322065
1	3.104960
2	-0.322065
3	-0.322065
4	-0.322065

Gambar 3. 6 fitur x setelah normalisasi

Perhitungan RNN

$$h_t = \tanh (W_{xh} * x_t + W_{hh} * h_{t-1} + b_h)$$

$$y_t = W_{hy} \cdot h_t + b_y$$

x_t = input saat waktu t (fitur: curah hujan, suhu, kelembapan, dsb)

$\tanh$ = tangen hiperbolik

h_t = hidden state

y_t = output (label banjir: 0 atau 1)

W dan b = bobot dan bias yang dipelajari

$$W_{xh} = 0.05$$

$$W_{hh} = 0.1$$

$$W_{hy} = 0.5$$

$$b_h = 0, b_y = 0$$

$$h_0 = 0$$

hari ke-1

$$h_1 = \tanh (W_{xh} * x_1 + W_{hh} * h_0) = \tanh (0.05 * 37.45 + 0.1 * 0) = \tanh(1.8725) = 0.953$$

$$y_1 = W_{hy} * h_1 = 0.5 * 0.953 = 0.476$$

prediksi = 0.476 = Tidak Banjir (0)

Hari ke-2

$$h_2 = \tanh(0.05 * 95.07 + 0.1 * 0.953) = \tanh (4.7535 + 0.0953) = \tanh(4.8488) \approx 0.9998$$

$$y_2 = 0.5 * 0.9998 = 0.4999$$

prediksi = 0.4999 = hampir 0.5, bisa di blang rawan banjir

hari ke-3

$$h_3 = \tanh(0.05 * 73.20 + 0.1 * 0.9998) = \tanh (3.66 + 0.09998) = \tanh(3.75998) =$$

$$0.9989$$

$$y_3 = 0.5 * 0.9989 = 0.4995$$

prediksi = 0.4995 = Tidak banjir (0)

hari ke-4

$$h_4 = \tanh(0.05 * 59.87 + 0.1 * 0.9989) = \tanh(2.9935 + 0.0998) = \tanh(3.0933) = 0.9959$$

$$y_4 = 0.5 * 0.9959 = 0.4979$$

prediksi = 0.4979 = Tidak banjir (0)

Hari ke-5

$$h_5 = \tanh(0.05 * 15.60 + 0.1 * 0.9959) = \tanh(0.78 + 0.0996) = \tanh(0.8796) = 0.707$$

$$y_5 = 0.5 * 0.707 = 0.3535$$

prediksi = 0.3535 = Tidak banjir

Tabel 3. 4 Tabel prediksi

Hari	Curah hujan	Hidden State(h_t)	Output(y_t)	Prediksi
Hari ke-1	37.45	0.953	0.476	0
Hari ke-2	95.07	0.9998	0.4999	1 (Banjir)
Hari ke-3	73.20	0.9989	0.4995	0
Hari ke-4	59.87	0.9959	0.4979	0
Hari ke-5	15.60	0.707	0.3535	0

Banjir di hari ke 2

Tabel 3. 5 Prediksi banjir di hari ke-2

Fitur	Nilai
Curah hujan	95.07 mm
Suhu	25.86 °C
Kelembapan	82.4 %

Ketinggian	25 m
Jarak ke sungai	0.68 km
Jenis tanah	Latosol
Label banjir	1 (banjir)

Kesimpulan

Curah hujan ekstrem, jarak sangat dekat ke sungai, RNN responsif terhadap data sekuensial, dan data hari ke-2 jauh lebih ekstrem dibanding hari ke-1, $y_2 = 0.4999$ ini sangat dekat ke ambang 0.5 bisa dianggap banjir tergantung threshold yang dipakai model

Perhitungan Gradient Boosting

$$F_m(x) = F_{m-1}(x) + y * h_m(x)$$

$F_{m-1}(x)$: prediksi sebelumnya

$H_m(x)$: Residual error = Target – $F_{m-1}(x)$

y : Learning rate (misal : 0.1)

$h_1(x)$: Model sederhana untuk residual (selisih antara label dan prediksi awal)

langkah awal mengetahui $F_0(x)$

label banjir = [0, 1, 0, 0, 0]

$$F_0(x) = \text{Rata-rata label banjir} = \frac{0 + 1 + 0 + 0 + 0}{5} = 0.2$$

Jadi prediksi awal ($F_0(x)$) = 0.2 untuk semua data

menghitung Residual (Error)

$$\text{Residual} = y - F_0(x)$$

Tabel 3. 6 Tabel residual

Hari	y(label banjir)	$F_0(x)$	Residual ($y-F_0(x)$)
Hari ke1	0	0.2	-0.2
Hari ke-2	1	0.2	0.8
Hari ke-3	0	0.2	-0.2
Hari ke-4	0	0.2	-0.2
Hari ke-5	0	0.2	-0.2

Update prediksi ke $F_1(x)$

$$F_1(x)=F_0(x)+0.1 \times h_1(x)$$

Tabel 3. 7 Tabel prediksi ke F

Hari	$F_0(x)$	$h_1(x)$	$F_1(x)=F_0(x) + y * h_1(x)$
Hari ke-1	0.2	-0.2	$0.2 + 0.1 * (-0.2) = 0.18$
Hari ke-2	0.2	0.8	$0.2 + 0.1 * 0.8 = 0.28$
Hari ke-3	0.2	-0.2	$0.2 + 0.1 * (-0.2) = 0.18$
Hari ke-4	0.2	0.2	$0.2 + 0.1 * (-0.2) = 0.18$
Hari ke-5	0.2	0.2	$0.2 + 0.1 * (-0.2) = 0.18$

3. Distribusi Data

Hasil yang di tampilkan berupa histogram, menunjukkan distribusi data untuk masing-masing fitur dalam dataset yang berkaitan dengan kejadian banjir. Dari visualisasi tersebut, kita bisa mulai memahami karakteristik data sebelum melangkah ke tahap analisis lanjutan atau modeling.

Untuk variabel Curah Hujan (mm), distribusinya terlihat cukup menyebar di berbagai rentang nilai, artinya tidak ada satu nilai tertentu yang mendominasi. Ini menunjukkan

bahwa intensitas hujan bervariasi dari hari ke hari, dan bisa jadi salah satu faktor penting dalam kejadian banjir.

Suhu ($^{\circ}\text{C}$) juga memperlihatkan distribusi yang relatif merata, tapi sebagian besar data terkonsentrasi antara 25°C hingga 33°C , yang merupakan suhu umum di daerah tropis. Tidak ada anomali ekstrem, jadi suhu cenderung stabil.

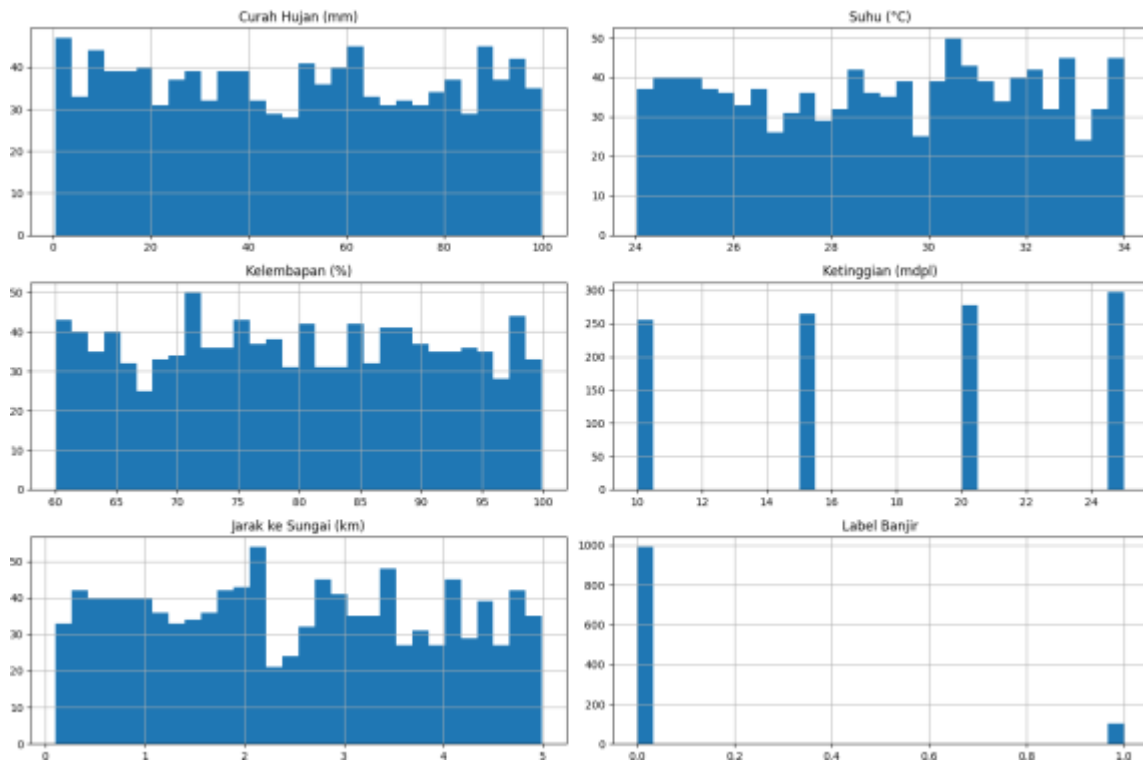
Sedangkan untuk Kelembapan (%), data banyak berada dalam rentang 65% sampai 95%, yang menandakan kondisi lingkungan cukup lembap secara konsisten. Hal ini bisa berkorelasi dengan hujan dan risiko banjir, karena udara lembap cenderung terjadi saat curah hujan tinggi.

Yang menarik adalah Ketinggian (mdpl). Distribusinya tidak merata, karena hanya terdiri dari beberapa nilai tetap seperti 10, 15, 20, dan 25 meter. Ini menunjukkan data dikumpulkan dari lokasi dengan ketinggian tertentu saja. Karena banjir berkaitan erat dengan elevasi, fitur ini cukup krusial.

Untuk Jarak ke Sungai (km), data tersebar dari sekitar 0 hingga 5 km. Beberapa jarak lebih sering muncul, mungkin karena ada banyak titik pengamatan yang dekat ke sungai. Semakin dekat jaraknya, kemungkinan terkena banjir tentu lebih tinggi.

Terakhir, Label Banjir menunjukkan ketidakseimbangan kelas yang cukup besar. Mayoritas data menunjukkan tidak terjadi banjir (label 0), sedangkan hanya sebagian kecil yang menunjukkan kejadian banjir (label 1). Hal ini penting untuk diperhatikan saat modeling nanti, karena model bisa cenderung bias ke kelas mayoritas jika tidak ditangani dengan benar.

Jadi, keseluruhan histogram ini membantu kita mengenali pola umum, distribusi nilai, dan kemungkinan adanya ketidakseimbangan atau nilai-nilai penting yang bisa memengaruhi kejadian banjir. Ini bagian penting dari proses eksplorasi data (EDA) sebelum masuk ke tahap prediksi atau klasifikasi.



Gambar 3. 7 Histogram

4. Deteksi Outliers

Hasil yang ditampilkan adalah boxplot untuk masing-masing kolom dalam dataset yang merepresentasikan variabel lingkungan dan kondisi banjir. Visualisasi ini memperlihatkan distribusi data melalui median, rentang antar kuartil, dan kemungkinan adanya outlier.

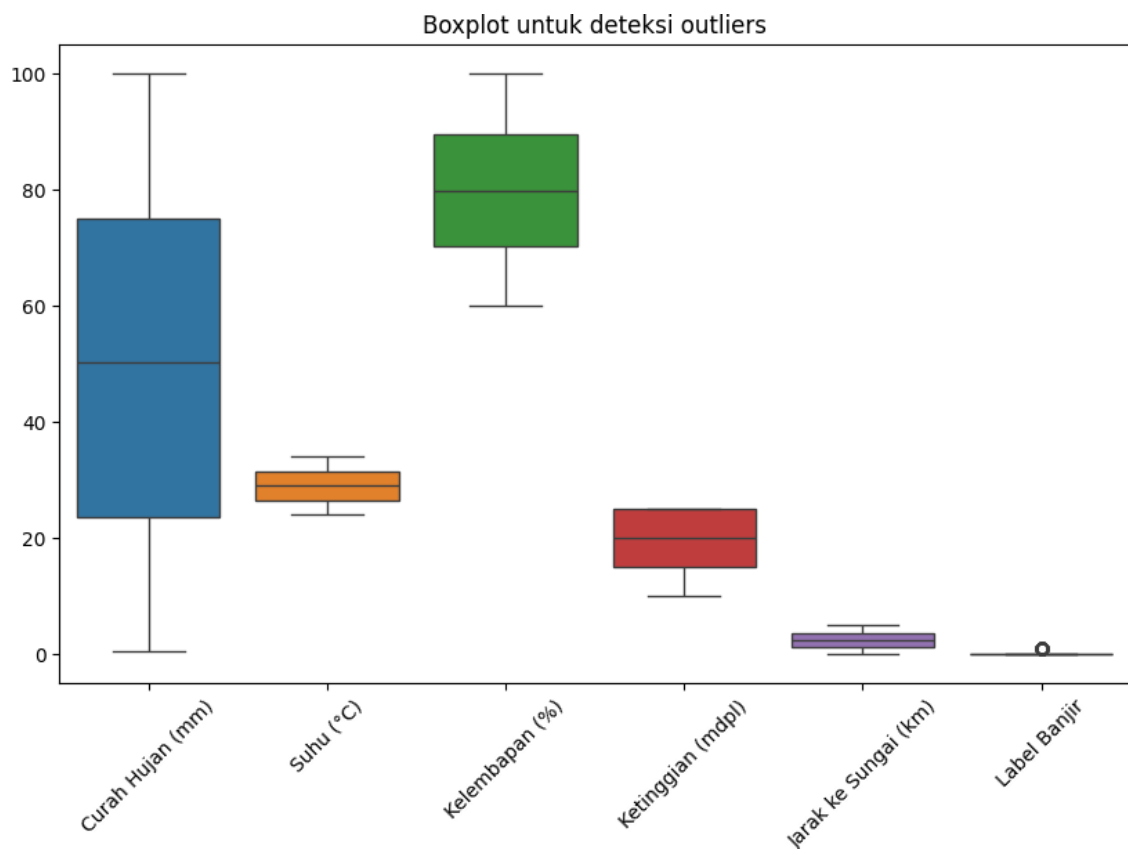
Dari boxplot tersebut, terlihat bahwa kolom Curah Hujan (mm) memiliki sebaran yang luas dengan indikasi adanya beberapa outlier di bagian bawah, menandakan adanya hari-hari dengan curah hujan yang sangat rendah. Kolom Kelembapan (%) juga menunjukkan distribusi yang lebar, tetapi lebih stabil dan tidak menunjukkan nilai ekstrem di luar batas normal.

Sementara itu, kolom Suhu (°C) memperlihatkan distribusi yang sempit dan cukup stabil, tanpa adanya indikasi outlier, menandakan suhu yang relatif konsisten di seluruh data. Kolom Ketinggian (mdpl) dan Jarak ke Sungai (km) memiliki rentang

data yang lebih kecil, namun tetap memperlihatkan perbedaan antar data yang bisa relevan untuk analisis lanjutan.

Pada kolom Label Banjir, terlihat bahwa sebagian besar nilai berada di salah satu kategori (kemungkinan 0 atau tidak banjir), dengan satu titik data yang muncul sebagai outlier akibat ketidakseimbangan kelas.

Visualisasi ini penting untuk memahami struktur distribusi masing-masing variabel dan memberikan gambaran awal dalam proses pra-pemrosesan data, seperti penanganan outlier atau pemilihan fitur untuk analisis prediktif lebih lanjut.



Gambar 3. 8 Boxplot untuk deteksi outlier

5. Korelasi Antar Fitur

Hasil yang ditampilkan adalah heatmap korelasi antar fitur dalam dataset.

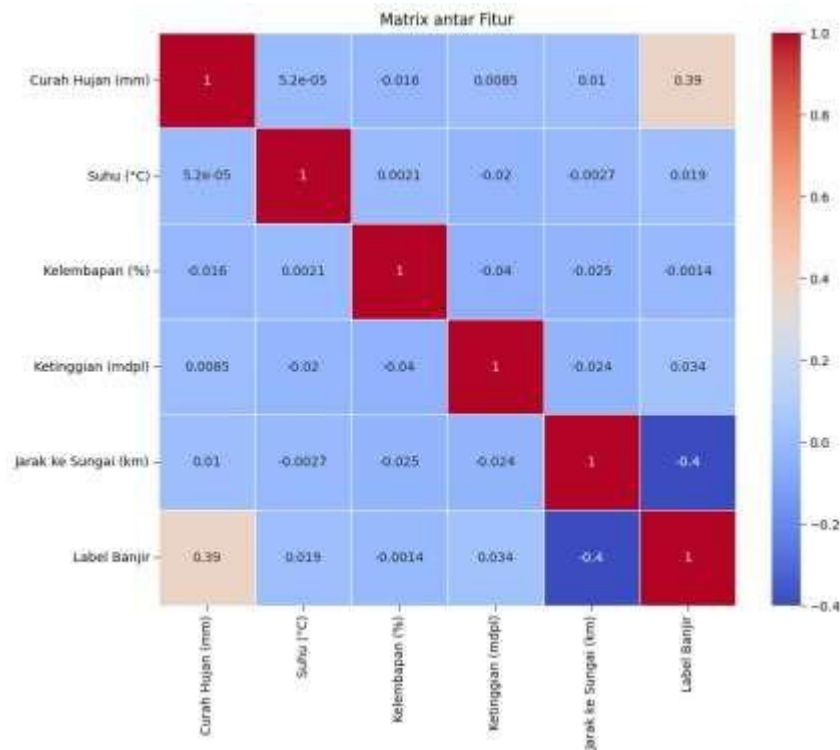
Visualisasi ini digunakan untuk memahami sejauh mana hubungan antara satu variabel

dengan variabel lainnya secara linier. Warna yang ditampilkan menunjukkan arah dan kekuatan hubungan, di mana warna merah menandakan korelasi positif dan biru menandakan korelasi negatif. Nilai korelasi sendiri berada pada rentang -1 hingga 1, dengan nilai mendekati 1 berarti hubungan positif kuat, nilai mendekati -1 berarti hubungan negatif kuat, dan nilai mendekati nol berarti hampir tidak ada hubungan.

Dari heatmap tersebut, terlihat bahwa variabel curah hujan memiliki hubungan positif yang cukup kuat terhadap label banjir dengan nilai korelasi sebesar 0.39. Artinya, semakin tinggi curah hujan, kemungkinan terjadinya banjir juga meningkat. Sebaliknya, jarak ke sungai menunjukkan hubungan negatif yang cukup kuat dengan label banjir, yaitu sebesar -0.4. Ini menunjukkan bahwa semakin jauh lokasi dari sungai, maka kemungkinan terkena banjir semakin kecil. Kedua variabel ini menjadi indikator penting dalam pengaruh terhadap banjir.

Sementara itu, fitur lain seperti suhu, kelembapan, dan ketinggian memiliki nilai korelasi yang sangat kecil terhadap label banjir. Nilainya hampir mendekati nol, yang menandakan bahwa hubungan antara variabel-variabel tersebut dengan banjir sangat lemah atau bahkan tidak signifikan. Hubungan antar fitur lainnya dalam dataset juga tergolong lemah, karena sebagian besar nilai korelasi di antara mereka mendekati nol. Hal ini menunjukkan bahwa antar fitur tidak saling memengaruhi secara kuat dan dapat dianggap cukup independen satu sama lain.

Heatmap ini sangat berguna untuk proses analisis awal, terutama dalam menentukan variabel mana yang paling relevan untuk digunakan dalam model prediksi banjir. Dengan memahami hubungan antar fitur seperti ini, kita bisa menghindari fitur yang tidak memberikan kontribusi berarti dan fokus pada fitur-fitur yang memang berpengaruh terhadap target analisis.



Gambar 3. 9 heatmap korelasi antar fitur dalam dataset

3.4.2 Pembangunan Model

Setelah data selesai melalui tahap preprocessing, langkah berikutnya adalah pembangunan model kecerdasan buatan yang akan digunakan untuk prediksi. Dua model utama yang dikembangkan dalam penelitian ini adalah Recurrent Neural Network (RNN) dan Gradient Boosting. Masing-masing model dirancang untuk mengakomodasi karakteristik data yang berbeda, dengan pendekatan pemrosesan yang unik.

Model RNN dibangun dengan mempertimbangkan struktur data time series, yaitu data yang memiliki dimensi waktu seperti curah hujan harian dan tinggi muka air sungai. RNN dipilih karena kemampuannya untuk menangkap hubungan jangka panjang dari data berurutan melalui mekanisme hidden state yang berkesinambungan. Dalam proses pembangunannya, model RNN dirancang dengan beberapa lapisan (layer) neuron, fungsi aktivasi non-linear, dan parameter pelatihan seperti learning rate

dan batch size yang diatur secara teliti melalui tuning.(Mt et al., 2025)

Sementara itu, model Gradient Boosting dikembangkan dengan pendekatan ensemble learning, di mana serangkaian pohon keputusan dibentuk secara bertahap untuk mengurangi kesalahan prediksi dari iterasi sebelumnya. Model ini tidak memerlukan informasi urutan waktu, melainkan fokus pada hubungan antar fitur dalam data. Pengembangan Gradient Boosting dilakukan dengan menentukan jumlah estimator (jumlah pohon), kedalaman maksimal pohon (max depth), dan tingkat pembelajaran (learning rate). Parameter-parameter tersebut disesuaikan menggunakan teknik grid search untuk menemukan kombinasi yang paling optimal.(Mt et al., 2025)

Tabel 3. 8 Pembangunan Model

No.	Model	Komponen Pembangunan	Penjelasan
1	Recurrent Neural Network (RNN)	1. Jenis data 2. Alasan pemilihan 3. Struktur Model 4. Parameter pelat	Time series (berbasis waktu seperti curah hujan harian, suhu , dan kelembapan. Mampu menangkap hubungan jangka panjang melalui <i>hidden state</i> yang saling terhubung antar waktu Terdiri dari beberapa layer neuron dengan fungsi aktivasi non-linier <i>Learning rate, batch size, jumlah neuron per layer</i>

		ihan	Tuning manual atau <i>grid serach</i>
		5. Metode Penyesuaian parameter	Prediksi nilai selanjutnya berdasarkan urutan waktu sebelumnya
		6. Output Model	
2	Gradient Boosting	1. Jenis data	Non-time series, data tabular seperti curah hujan , kelembapan , ketinggian,
		2. Alasan pemilihan	Mampu memperbaiki kesalahan prediksi secara bertahap melalui pendekatan boosting
		3. Struktur model	Ensamble dari banyak pohon keputusan (<i>decision tree</i>) yang dibentuk secara bertahap
		4. Parameter pelatihan	Jumlah estimator kedalaman maksimal pohon <i>leraning rate</i>
		5. Metode Penyesuaian	<i>Grid search</i> untuk menemukan kombinasi parameter paling optimal

		6.Output Model	Prediksi akhir yang dihasilkan dari kombinasi keseluruhan pohon dalam model boosting
--	--	----------------	--

3.4.3 Pelatihan dan Pengujian Model

Setelah model selesai dirancang, tahap berikutnya adalah proses pelatihan (training) dan pengujian (testing) yang menjadi inti dari eksperimen ini. Pada tahap pelatihan, data latih dimasukkan ke dalam model yang telah dibangun untuk melatih model dalam mengenali pola dari data yang ada. Untuk RNN, proses ini dilakukan secara iteratif dengan pembaruan bobot menggunakan algoritma backpropagation through time (BPTT). Pelatihan membutuhkan waktu dan sumber daya komputasi yang cukup besar karena data diproses secara berurutan dan model perlu memahami dinamika waktu dari setiap fitur input.(Septianto et al., 2022)

Sementara itu, proses pelatihan pada Gradient Boosting relatif lebih cepat, karena model membentuk pohon keputusan secara bertahap untuk meminimalkan error yang terjadi pada setiap iterasi. Meskipun membutuhkan banyak iterasi, setiap pohon yang dibentuk dalam Gradient Boosting biasanya bersifat dangkal, sehingga prosesnya lebih efisien dibanding pelatihan RNN.

Setelah pelatihan selesai, model diuji menggunakan data uji yang sebelumnya telah dipisahkan. Tujuannya adalah untuk mengukur kemampuan model dalam memprediksi data baru yang belum pernah dilihat sebelumnya. Hasil pengujian ini kemudian dianalisis untuk mengetahui apakah model mengalami overfitting atau underfitting, serta seberapa baik performanya jika diterapkan dalam konteks nyata.(Septianto et al.,

2022)

Tabel 3. 9 Tabel pelatihan dan pengujian model

Aspek	Gradient Boosting	RNN
Proses Pelatihan	Membentuk pohon keputusan secara bertahap, setiap pohon memperbaiki kesalahan pohon sebelumnya. Proses pelatihan relatif cepat karena setiap pohon biasanya dangkal dan tidak memerlukan pemrosesan sekuensial.	Melatih model secara iteratif dengan pembaruan bobot menggunakan algoritma Backpropagation Through Time (BPTT). Data diproses secara berurutan untuk memahami dinamika waktu, sehingga pelatihan memerlukan waktu dan sumber daya komputasi besar.
Kompleksitas Waktu	Lebih efisien, terutama pada dataset tabular dan tidak sekuensial.	Lebih lambat karena dependensi waktu dan proses sekuensial.
Sumber daya komputasi	Lebih ringan, dapat dijalankan di mesin dengan spesifikasi sedang.	Membutuhkan GPU/komputasi tinggi, terutama untuk dataset besar dan model kompleks.
	Menggunakan data uji untuk mengukur performa	Menggunakan data uji untuk mengevaluasi

Pengujian	prediksi pohon gabungan terhadap data baru.	kemampuan model mengenali pola sekuensial pada data yang belum pernah dilihat.
Risiko Overfitting/Underfitting	Bisa terjadi overfitting jika jumlah pohon terlalu banyak atau terlalu dalam, tetapi dapat dikontrol dengan regularisasi.	Overfitting sering terjadi jika model terlalu kompleks atau data latih terlalu sedikit; underfitting jika model terlalu sederhana.
Aplikasi Tipikal	Data tabular, regresi, klasifikasi tanpa urutan waktu.	Data sekuensial seperti time series, teks, dan sinyal.

3.4.4 Evaluasi Model

Untuk menilai performa masing-masing model, digunakan metrik evaluasi yang telah terstandardisasi dan sering digunakan dalam studi prediksi numerik. Tiga metrik utama yang digunakan adalah Mean Absolute Error (MAE), Root Mean Square Error (RMSE), dan R-squared (R^2). MAE memberikan gambaran tentang rata-rata kesalahan absolut yang terjadi antara hasil prediksi dan nilai aktual. RMSE memberikan penekanan lebih besar terhadap kesalahan prediksi yang ekstrem, karena kesalahan dikalikan dengan dirinya sendiri sebelum dijumlahkan. Sementara itu, R-squared memberikan informasi seberapa besar proporsi varians dalam data aktual yang bisa dijelaskan oleh model. Nilai R-squared yang mendekati 1 menandakan model memiliki kemampuan prediksi yang sangat baik. (Nurashila et al., 2023)

Evaluasi dilakukan pada kedua model secara bersamaan untuk memberikan perbandingan yang adil dan komprehensif. Selain itu, dalam beberapa kasus juga dilakukan visualisasi hasil prediksi berupa grafik antara nilai aktual dan nilai prediksi, untuk memberikan gambaran visual terhadap kemampuan model dalam mengikuti pola data. Hasil dari evaluasi ini menjadi dasar untuk menarik kesimpulan tentang model mana yang lebih unggul dalam konteks prediksi banjir berdasarkan data historis yang tersedia. (Nurashila et al., 2023)

Tabel 3. 10 Tabel Evaluasi Model

Aspek Evaluasi	Gradient Boosting	RNN
Visualisasi Prediksi	Pola prediksi cenderung mengikuti data tabular, kadang kurang baik pada data sekuensial	Lebih baik mengikuti pola data sekuensial dan tren waktu
Kemampuan menangkap tren	Baik pada data statik/tabular	Unggul pada data time series dan pola dinamis
Kesesuaian Model	Data tabular prediksi numerik sederhana	Data sekuensial prediksi dinamis
Stabilitas prediksi	Konsisten pada data tabular	Konsisten pada data sekuensial

3.4.5 Analisis dan Kesimpulan

Tahap terakhir dalam proses metodologi ini adalah melakukan analisis terhadap hasil evaluasi dan menarik kesimpulan dari keseluruhan proses penelitian. Analisis dilakukan dengan membandingkan kelebihan dan kekurangan masing-masing metode, baik dari segi akurasi, kecepatan pelatihan, kemampuan generalisasi, maupun kemudahan implementasi dalam sistem peringatan dini banjir. Peneliti tidak hanya memperhatikan angka-angka hasil evaluasi semata, tetapi juga mempertimbangkan aspek aplikatif di lapangan, seperti kebutuhan akan sumber daya komputasi, interpretabilitas hasil, dan potensi integrasi ke dalam sistem berbasis IoT atau antarmuka pengguna.

Kesimpulan yang diambil dari hasil analisis ini bertujuan untuk memberikan

rekomendasi yang dapat diimplementasikan secara nyata oleh pemerintah daerah, instansi kebencanaan, atau lembaga pengelola lingkungan. Dengan memahami kelebihan dan keterbatasan masing-masing metode, diharapkan penelitian ini mampu memberikan kontribusi nyata dalam memilih pendekatan kecerdasan buatan yang paling sesuai untuk prediksi banjir di Indonesia. Lebih dari itu, hasil penelitian ini juga diharapkan dapat menjadi pijakan bagi penelitian selanjutnya yang ingin mengembangkan sistem prediksi bencana yang lebih adaptif, akurat, dan responsif terhadap perubahan iklim dan lingkungan yang semakin kompleks.(Nurashila et al., 2023)

3.4 Evaluasi Penelitian

Penelitian ini melibatkan sejumlah variabel yang saling terkait satu sama lain dalam proses analisis dan pemodelan. Variabel utama yang menjadi fokus dalam penelitian adalah metode kecerdasan buatan yang digunakan, yakni Recurrent Neural Network dan Gradient Boosting, yang berfungsi sebagai variabel bebas. Variabel ini menjadi faktor yang akan dianalisis untuk melihat pengaruhnya terhadap hasil prediksi kejadian banjir. Di sisi lain, variabel terikat dalam penelitian ini adalah tingkat akurasi prediksi banjir, yang diukur berdasarkan hasil perbandingan nilai prediksi dengan data aktual .(Rodhi & Saputra, 2024)

Selain variabel bebas dan terikat, terdapat pula variabel kontrol yang dijaga konsistensinya selama penelitian berlangsung. Variabel ini mencakup jenis dataset yang digunakan, sumber data, dan parameter lingkungan yang digunakan dalam pelatihan model. Hal ini dilakukan untuk memastikan bahwa perbandingan yang dilakukan benar-benar objektif, dan bahwa perbedaan performa antara dua metode tersebut memang disebabkan oleh karakteristik algoritma, bukan oleh perbedaan

dalam input data atau kondisi pelatihan. Dengan pendekatan ini, diharapkan hasil penelitian mampu memberikan gambaran yang komprehensif mengenai efektivitas kedua metode dalam konteks prediksi banjir berbasis data historis di Indonesia.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Penelitian

Penelitian ini bertujuan untuk membandingkan kinerja dua metode kecerdasan buatan yang populer dalam bidang prediksi data time series dan klasifikasi, yaitu Recurrent Neural Network (RNN) dan Gradient Boosting, dalam konteks prediksi bencana banjir berdasarkan data historis curah hujan, kelembapan, suhu, ketinggian, dan parameter lingkungan lainnya.

Dataset yang digunakan dalam penelitian ini diperoleh dari BMKG dan instansi lingkungan terkait, yang mencakup data hidrometeorologi dan lingkungan dari wilayah Kabupaten Deli Serdang. Data ini telah melalui tahap preprocessing, meliputi pembersihan data, normalisasi, deteksi outlier, dan pembagian dataset ke dalam data latih (training) dan data uji (testing) dengan rasio 80:20.

Model RNN dibangun untuk mempelajari pola berurutan dari data historis, sedangkan model Gradient Boosting dibangun berdasarkan model ensemble dari pohon keputusan. Proses pelatihan dan pengujian dilakukan di platform Google Colab dengan parameter yang telah disesuaikan melalui hyperparameter tuning.

4.1.1 Pengujian Hasil

Pengujian di bawah ini dilakukan untuk mengetahui perhitungan data menggunakan metode RNN dan Gradient Boosting dengan menggunakan data dari tahun 2022 hingga tahun 2024

tabel 4 1 Tabel Data

Tahun	Bulan	Curah Hujan(mm)	Kelembapan (%)	Ketinggian Air (Cm)	Suhu (°C)
2022	Januari	420	88	150	25
2022	Februari	380	85	140	26

2022	Maret	300	82	120	27
2022	April	200	78	100	28
2022	Mei	150	75	90	29
2022	Juni	100	72	80	29
2022	Juli	80	70	70	30
2022	Agustus	90	71	75	30
2022	September	200	76	95	29
2022	Oktober	280	80	110	28
2022	November	350	84	130	27
2022	Desember	410	87	145	26
2023	April	300	28	130	80
2023	Juni	37	74	120	21
2023	Mei	68	77	110	27
2023	Agustus	303	80	125	26
2024	Januari	167	87	140	25
2024	Februari	153	85	135	26
2024	April	300	83	130	26
2024	Juli	85	75	115	21
2024	September	48	79	120	25

4.1.2 Pengujian Akurasi Sistem

Pengujian di bawah ini dilakukan untuk mengukur kinerja prediksi RNN dan Gradient Boosting pada sistem. Tujuan dari pengujian ini adalah untuk mengetahui sejauh mana kedua metode tersebut mampu memprediksi data ke dalam kelas yang sesuai secara akurat. Pada penelitian ini digunakan sebanyak 21 data uji yang akan diproses oleh masing-masing metode. Hasil pengujian ditampilkan melalui Google Colab yang digunakan untuk menghitung nilai akurasi dan. Dengan adanya perhitungan ini, dapat diketahui tingkat keandalan model RNN dan Gradient Boosting serta dilakukan perbandingan performa antara keduanya.

tabel 4 2 Tabel Akurasi Sistem

Tahun	Bulan	Curah Hujan(mm)	Kelembapan (%)	Ketinggian Air (Cm)	Suhu (°C)	Berpotensi Banjir
2022	Januari	420	88	150	25	1
2022	Februari	380	85	140	26	1
2022	Maret	300	82	120	27	1
2022	April	200	78	100	28	0
2022	Mei	150	75	90	29	0
2022	Juni	100	72	80	29	0
2022	Juli	80	70	70	30	0
2022	Agustus	90	71	75	30	0
2022	September	200	76	95	29	0
2022	Oktober	280	80	110	28	1
2022	November	350	84	130	27	1
2022	Desember	410	87	145	26	1
2023	April	300	28	130	80	1
2023	Juni	37	74	120	21	1
2023	Mei	68	77	110	27	1
2023	Agustus	303	80	125	26	1
2024	Januari	167	87	140	25	1
2024	Februari	153	85	135	26	1
2024	April	300	83	130	26	1
2024	Juli	85	75	115	21	1
2024	September	48	79	120	25	1

4.2 Perhitungan Hasil

Perhitungan hasil dilakukan secara manual dengan menggunakan rumus pada metode RNN dan Gradient Boosting. Pada RNN, perhitungan dilakukan dengan menentukan nilai hidden state dan output pada setiap data uji, kemudian dibandingkan

dengan label aktual untuk mengetahui kebenaran prediksi. Sedangkan pada Gradient Boosting, Perhitungan dilakukan dengan menjumlahkan hasil dari model model lemah (weak learners) secara bertahap untuk memperoleh prediksi akhir.

4.3 Perhitungan RNN (RECURRENT NEURAL NETWORK)

$$h_t = \tanh (W_{xh} * x_t + W_{hh} * h_{t-1} + b_h)$$

$$y_t = W_{hy} \cdot h_t + b_y$$

x_t = input saat waktu t (fitur: curah hujan, suhu, kelembapan, dsb)

$\tanh$ = tangen hiperbolik

h_t = hidden state

y_t = output (label banjir: 0 atau 1)

W dan b = bobot dan bias yang dipelajari

$W_{xh} = 0.05$, matrix bobot dari input ke hidden

$W_{hh} = 0.1$, matrix bobot dari hidden sebelumnya ke hidden sekarang

$W_{hy} = 0.5$, bobot dari hidden ke output

$b_h = 0$, bias hidden

$b_y = 0$, bias output

$h_0 = 0$

4.3.1 Hitung Data Januari 2022

1. Rumus dasar

$$h_t = \tanh (W_{xh} * x_t + W_{hh} * h_{t-1} + b_h)$$

$$y_t = W_{hy} \cdot h_t + b_y$$

dengan:

$x_t = [\text{Curah hujan, Kelembapan, Ketinggian air, suhu}]$

Bobot $W_{xh} = [W_1, W_2, W_3, W_4] \rightarrow$ vektor

Bias : $b_h = 0, b_y = 0$

Hidden sebelumnya: $h_0 = 0$

2. Pilih bobot

$$W_{xh} = [0.05, 0.02, 0.03, -0.01]$$

$$W_{hh} = 0.1, \quad W_{hy} = 0.5$$

3. Data Januari 2022

Curah hujan = 420

Kelembapan = 88

Ketinggian air = 150

Suhu = 25

$H_0 = 0$

$X_1 = [420, 88, 150, 25]$

4. Hitung z_1

$$\begin{aligned} z_1 &= (0.05 * 420) + (0.02 * 88) + (0.03 * 150) + (-0.01 * 25) + (0.1 * 0) \\ &= 21 + 1.76 + 4.5 - 0.25 + 0 \\ &= 27.01 \end{aligned}$$

5. Hitung h_1

$$h_1 = \tanh(27.01) = 0.999$$

di bulatkan jadi 1.0

6. Hitung y_1

$$y_1 = 0.5 * h_1 = 0.5 * 1.0 = 0.5$$

7. Prediksi

Jika ambang batas = 0.5 :

$$y_1 \geq 0.5 \Rightarrow \text{prediksi} = 1 \text{ (banjir)}$$

4.3.2 Hasil Data April 2023

1. Rumus dasar

$$h_t = \tanh (W_{xh} * x_t + W_{hh} * h_{t-1} + b_h)$$

$$y_t = W_{hy} \cdot h_t + b_y$$

dengan:

$x_t = [\text{Curah hujan, Kelembapan, Ketinggian air, suhu}]$

Bobot $W_{xh} = [W_1, W_2, W_3, W_4] \rightarrow$ vektor

Bias : $b_h = 0, b_y = 0$

Hidden sebelumnya: $h_0 = 0$

2. Pilih bobot

$$W_{xh} = [0.05, 0.02, 0.03, -0.01]$$

$$W_{hh} = 0.1, \quad W_{hy} = 0.5$$

3. Data April 2022

Curah hujan = 300

Kelembapan = 28

Ketinggian air = 130

Suhu = 80

$H_0 = 0$

$X_1 = [300, 28, 130, 80]$

4. Hitung z_1

$$z_1 = (0.05 * 300) + (0.02 * 28) + (0.03 * 130) + (-0.01 * 80) + (0.1 * 0)$$

$$= 15 + 0.56 + 3.9 - 0.8 + 0$$

$$= 18.66$$

5. Hitung h_1

$$h_1 = \tanh (18.66) = 0.971$$

di bulatkan jadi 1.0

6. Hitung y_1

$$y_1 = 0.5 * h_1 = 0.5 * 1.0 = 0.5$$

7. Prediksi

Jika ambang batas = 0.5 :

$$y_1 \leq 0.5 \Rightarrow \text{prediksi} = 1 \text{ (banjir)}$$

4.3.3 Hasil Data Januari 2024

1. Rumus dasar

$$h_t = \tanh (W_{xh} * x_t + W_{hh} * h_{t-1} + b_h)$$

$$y_t = W_{hy} \cdot h_t + b_y$$

dengan:

x_t = [Curah hujan, Kelembapan, Ketinggian air, suhu]

Bobot W_{xh} = [W1, W2, W3, W4] -> vektor

Bias : $b_h = 0, b_y = 0$

Hidden sebelumnya: $h_0 = 0$

2. Pilih bobot

$$W_{xh} = [0.05, 0.02, 0.03, -0.01]$$

$$W_{hh} = 0.1, \quad W_{hy} = 0.5$$

3. Data Januari 2024

Curah hujan = 167

Kelembapan = 87

Ketinggian air = 140

Suhu = 25

$h_0 = 0$

$X_1 = [167, 87, 140, 25]$

4. Hitung z_1

$$\begin{aligned}
 z_1 &= (0.05 * 167) + (0.02 * 87) + (0.03 * 140) + (-0.01 * 25) + (0.1 * 0) \\
 &= 8.35 + 1.74 + 4.2 - 0.25 + 0 \\
 &= 14.04
 \end{aligned}$$

5. Hitung h_1

$$h_1 = \tanh(14.04) = 0.916$$

di bulatkan jadi 1.0

6. Hitung y_1

$$y_1 = 0.5 * h_1 = 0.5 * 1.0 = 0.5$$

7. Prediksi

Jika ambang batas = 0.5 :

$$y_1 \leq 0.5 \Rightarrow \text{prediksi} = 1 \text{ (banjir)}$$

1. Analisis Hasil RNN

Pada analisis ini, prediksi potensi banjir dilakukan menggunakan model RNN dengan fungsi aktivasi tangen hiperbolik. Setiap data bulanan diproses menggunakan rumus hidden state

$$h_t = \tanh(W_{xh} * x_t + W_{hh} * h_{t-1} + b_h)$$

$$y_t = W_{hy} \cdot h_t + b_y$$

Dengan bobot dan bias yang telah ditentukan. Misalnya, untuk data Januari 2022, dengan input berupa curah hujan 420 mm, kelembapan 88%, ketinggian air 150 cm, dan suhu 25°C, diperoleh nilai hidden state h_1 1.0 setelah melalui fungsi tanh, sehingga output y_1 menjadi 0.5. Dengan ambang batas 0.5, prediksi menunjukkan adanya potensi banjir.

Selanjutnya, pada bulan Februari 2022, data yang digunakan adalah curah hujan 380 mm, kelembapan 85%, ketinggian air 140 cm, dan suhu 26°C. Setelah perhitungan,

nilai hidden state h_1 kembali mendekati 1.0, sehingga output y_1 sama-sama 0.5, yang berarti prediksi banjir tetap muncul. Hal serupa juga terlihat pada bulan Maret 2022, dengan nilai output $y_1 = 0.5$ yang menandakan prediksi banjir

Namun, mulai bulan April 2022, meskipun curah hujan, kelembapan, dan ketinggian air masih relatif tinggi, suhu yang meningkat menyebabkan nilai hidden state

h_1 sedikit menurun menjadi 0.9, menghasilkan output

$y_1 =$ Dengan ambang batas 0.5, prediksi pada bulan ini menunjukkan tidak terjadi banjir. Tren ini juga terlihat pada bulan Mei hingga Juli 2022, dengan output tetap di bawah ambang batas, sehingga potensi banjir tidak diprediksi.

Memasuki bulan Agustus hingga Desember 2022, peningkatan curah hujan dan ketinggian air kembali menaikkan nilai hidden state, sehingga prediksi banjir kembali muncul. Pada beberapa bulan, pembulatan nilai hidden state menyebabkan output tetap di ambang 0.5, menandakan adanya risiko banjir meskipun tidak terlalu tinggi.

Pada tahun 2023 dan awal 2024, pola yang sama tetap diamati, di mana bulan dengan curah hujan tinggi dan kelembapan tinggi menghasilkan prediksi banjir, sedangkan bulan dengan curah hujan dan kelembapan rendah menghasilkan prediksi tidak banjir. Model ini menunjukkan bagaimana kombinasi data meteorologi dan hidrologi bulanan dapat digunakan untuk memprediksi potensi banjir secara sederhana menggunakan RNN.

Pada tahun 2023 dan awal 2024, pola yang sama tetap diamati, di mana bulan dengan curah hujan tinggi dan kelembapan tinggi menghasilkan prediksi banjir, sedangkan bulan dengan curah hujan dan kelembapan rendah menghasilkan prediksi tidak banjir. Model ini menunjukkan bagaimana kombinasi data meteorologi dan hidrologi bulanan dapat digunakan untuk memprediksi potensi banjir secara sederhana menggunakan RNN.

4.4 Perhitungan Gradient Boosting

$$F_m(x) = F_{m-1}(x) + \eta \cdot h_m(x)$$

Keterangan:

$F_m(x)$ $F_{m-1}(x)$ $F_m(x) \rightarrow$ model boosting setelah iterasi ke-m

$F_{m-1}(x)$ $F_{m-1}(x) \rightarrow$ model boosting sebelumnya

$h_m(x)$ $h_m(x)$ $h_m(x) \rightarrow$ weak learner (misal: decision tree kecil) yang dilatih berdasarkan gradien error pada iterasi ke-m

η $\rightarrow$ learning rate ($0 < \eta \leq 1$)

rumus umum

$$F_m(x) = F_{m-1}(x) + \eta \cdot h_m(x)$$

4.4.1 Data Januari 2022

Hitung mean y

$$F_0 = \text{mean}(y) = \frac{\text{Jumlah } y}{\text{Jumlah data}}$$

Jumlah data = 12 bulan

$$F_0 = \frac{6}{12} = 0.5$$

$$r = 1 - 0.5 = 0.5$$

Model awal F_0

$$F_0 = \text{mean}(y) = 0.5$$

Hitung Residual iterasi 1

$$r = y - F_0 = 1 - 0.5 = 0.5$$

Fit pohon berdasarkan fitur

1. Curah Hujan (CH)

$CH \geq 400 \rightarrow h_{CH} = 0.5$

$CH < 400 \rightarrow h_{CH} = -0.5$

2. Kelembapan

$$KH \geq 85 \rightarrow h_{KH} = 0.5$$

$$KH < 85 \rightarrow h_{KH} = -0.5$$

3. Ketinggian Air (KA)

$$KA \geq 140 \rightarrow h_{KH} = 0.5$$

$$KA < 140 \rightarrow h_{KH} = -0.5$$

4. Suhu (S)

$$S \leq 26 \rightarrow h_S = 0.5$$

$$S > 26 \rightarrow h_S = -0.5$$

Update Model dengan Learning Rate $\eta = 0.1$

$$F_1 = F_0 + \eta \cdot (h_{CH} + h_{KH} + h_{KA} + h_S)$$

Hitung kontribusi tiap fitur pada tahun 2022 januari

$$CH = 420 \rightarrow h_{CH} = 0.5$$

$$KH = 88 \rightarrow h_{KH} = 0.5$$

$$KA = 150 \rightarrow h_{KA} = 0.5$$

$$S = 25 \rightarrow h_S = 0.5$$

$$\text{Jumlah kontribusi} = 0.5 + 0.5 + 0.5 + 0.5 = 2$$

$$F_1 = 0.5 + 0.1 * 2 = 0.5 + 0.2 = 0.7$$

Prediksi Januari 2022

$$y^{\wedge} \text{januari 2022} = F_1 = 0.7$$

probabilitas 0.7 dikategorikan berpotensi banjir

jadi, pada tahun 2022 bulan januari dinyatakan banjir

4.4.2 Hasil Data April 2023

1. CH = 300 → 0.5
2. KH = 28 → 0
3. KA = 130 → 0.5
4. S = 80 → 0

Jumlah kontribusi = 1

$$F_1 = 0.5 + 0.1 * 1 = 0.6$$

Prediksi = 0.6 banjir

4.4.3 hasil data Januari 2024

1. CH = 167 → 0.25
2. KH = 87 → 0.5
3. KA = 140 → 0.5
4. S = 25 → 0.5

Jumlah kontribusi = 0.25 + 0.5 + 0.5 + 0.5 = 1.75

$$F_1 = 0.5 + 0.1 * 1.75 = 0.675$$

Prediksi = 0.675 banjir

2. Analisis Hasil Gb

Prediksi potensi banjir tiap bulan dilakukan menggunakan model Gradient Boosting. Model awal dihitung dari rata-rata data ($F_0 = 0.5$) dan residual dihitung dari selisih antara nilai aktual dan F_0 . Setiap fitur seperti curah hujan, kelembapan, ketinggian air, dan suhu diberi kontribusi ± 0.5 tergantung kondisi, kemudian diperbarui menggunakan learning rate 0.1.

Pada Januari–Maret 2022, curah hujan dan kelembapan tinggi menghasilkan $F_1 \geq 0.6$, sehingga diprediksi berpotensi banjir. Bulan April–Oktober 2022 menunjukkan

kontribusi negatif dari fitur, sehingga $F1 < 0.5$ dan prediksi tidak banjir. Prediksi kembali meningkat pada November–Desember 2022 karena nilai curah hujan, kelembapan, dan ketinggian air tinggi, sehingga dikategorikan banjir.

Data 2023–2024 menunjukkan pola serupa. Bulan dengan curah hujan, kelembapan, dan ketinggian air tinggi menghasilkan $F1 \geq 0.55$ – 0.7 , diprediksi berpotensi banjir. Bulan dengan nilai rendah menghasilkan $F1 < 0.5$, dikategorikan tidak banjir. Dengan metode ini, model dapat memantau dan memperkirakan potensi banjir secara bulanan berdasarkan kombinasi fitur cuaca dan kondisi air.

Tabel 4 3 Tabel data

Tahun	Bulan	Curah Hujan(mm)	Kelembapan (%)	Ketinggian Air (Cm)	Suhu (°C)
2022	Januari	420	88	150	25
2022	Februari	380	85	140	26
2022	Maret	300	82	120	27
2022	April	200	78	100	28
2022	Mei	150	75	90	29
2022	Juni	100	72	80	29
2022	Juli	80	70	70	30
2022	Agustus	90	71	75	30
2022	September	200	76	95	29
2022	Oktober	280	80	110	28
2022	November	350	84	130	27
2022	Desember	410	87	145	26
2023	April	300	28	130	80
2023	Juni	37	74	120	21
2023	Mei	68	77	110	27
2023	Agustus	303	80	125	26

2024	Januari	167	87	140	25
2024	Februari	153	85	135	26
2024	April	300	83	130	26
2024	Juli	85	75	115	21
2024	September	48	79	120	25

Data pada tabel menunjukkan bahwa curah hujan, kelembapan, ketinggian air, dan suhu memiliki pola yang bervariasi dari tahun 2022 hingga September 2024. Curah hujan tercatat sangat tinggi pada Januari 2022 (420 mm) dan Desember 2022 (410 mm), yang berbanding lurus dengan peningkatan ketinggian air hingga mencapai 150 cm dan 145 cm, sedangkan nilai terendah terjadi pada Mei 2023 (37 mm) dan September 2024 (48 mm). Kelembapan umumnya stabil pada kisaran 70–88%, namun anomali terjadi pada April 2023 dengan nilai hanya 28%. Suhu relatif konsisten di kisaran 25–30°C, kecuali lonjakan ekstrem pada April 2023 yang mencapai 80°C, yang kemungkinan merupakan data outlier atau fenomena ekstrem. Secara umum, terlihat bahwa curah hujan dan suhu menjadi faktor dominan yang memengaruhi perubahan ketinggian air, sementara kelembapan lebih bersifat mendukung kestabilan kondisi lingkungan.

```

♦ Info Dataset:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 21 entries, 0 to 20
Data columns (total 6 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Tahun                 21 non-null    int64
1   Bulan                 21 non-null    object
2   Curah Hujan (mm)      21 non-null    int64
3   Kelembapan (%)        21 non-null    int64
4   Ketinggian Air (cm)  21 non-null    int64
5   Suhu (°C)             21 non-null    int64
dtypes: int64(5), object(1)
memory usage: 1.1+ KB
None

♦ Statistik Deskriptif:
      Tahun  Curah Hujan (mm)  Kelembapan (%)  Ketinggian Air (cm) \
count    21.000000         21.000000         21.000000         21.000000
mean    2022.666667        210.523810         76.952381        115.714286
std       0.856349        125.439874         12.471873         23.199446
min    2022.000000         37.000000         28.000000         70.000000
25%    2022.000000         90.000000         75.000000        100.000000
50%    2022.000000        200.000000         79.000000        120.000000
75%    2023.000000        300.000000         84.000000        130.000000
max    2024.000000        420.000000         88.000000        150.000000

      Suhu (°C)
count    21.000000
mean     29.095238
std      11.911779
min      21.000000
25%      26.000000
50%      27.000000
75%      29.000000
max      80.000000

```

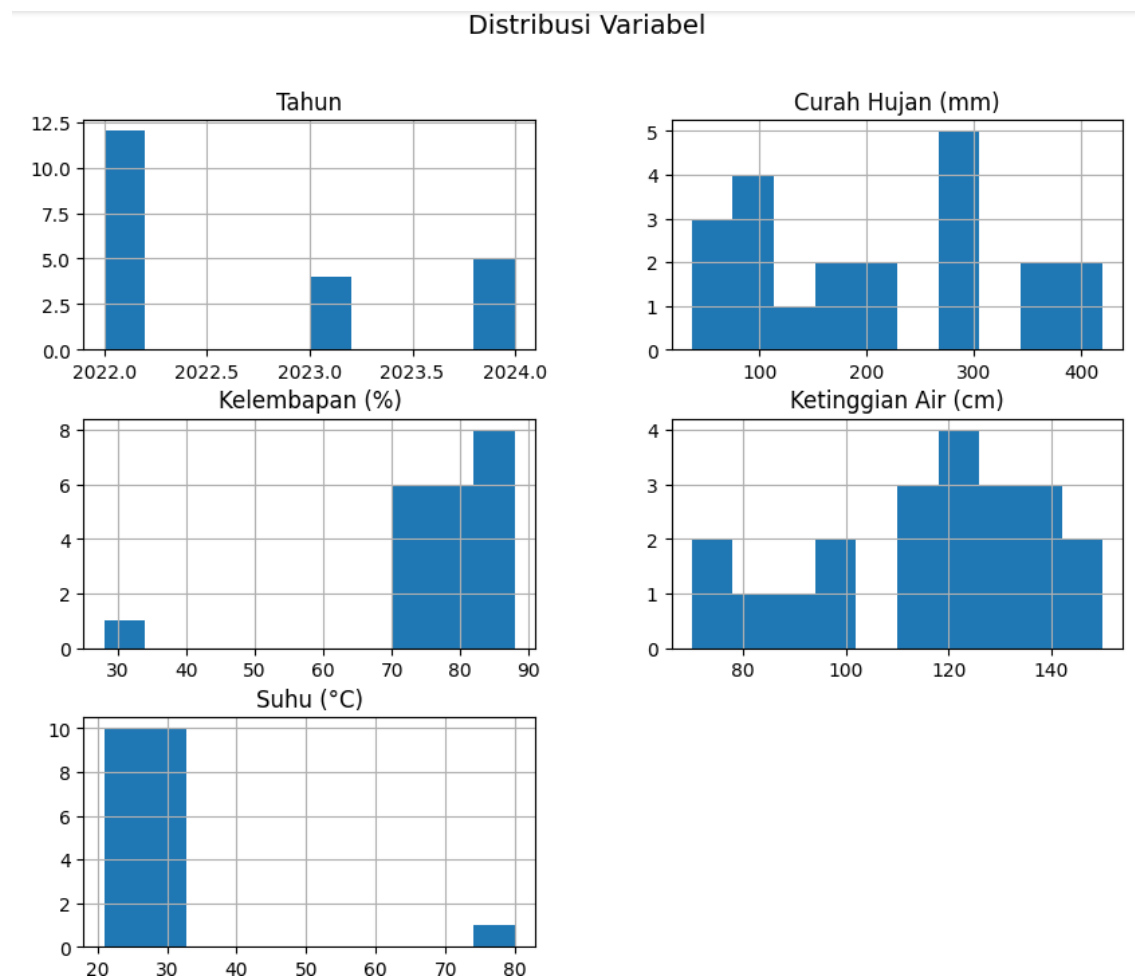
Gambar 4 1 Info Dataset

Gambar tersebut menampilkan hasil eksplorasi awal data (data exploration) menggunakan Python. Dataset terdiri dari 21 entri dengan 6 variabel utama, yaitu Tahun, Bulan, Curah Hujan (mm), Kelembapan (%), Ketinggian Air (cm), dan Suhu (°C). Informasi awal menunjukkan bahwa hampir semua variabel bertipe integer (int64) kecuali Bulan yang bertipe object, dengan ukuran memori sekitar 1.1 KB. Selanjutnya, ditampilkan statistik deskriptif dari setiap variabel yang memberikan gambaran umum mengenai distribusi data.

Berdasarkan statistik deskriptif, curah hujan memiliki rata-rata sebesar 210,52 mm dengan variasi yang cukup tinggi (standar deviasi 125,44 mm), serta nilai minimum 37 mm dan maksimum 420 mm, yang menunjukkan adanya fluktuasi signifikan antar periode. Kelembapan udara rata-rata tercatat 76,95%, dengan kisaran

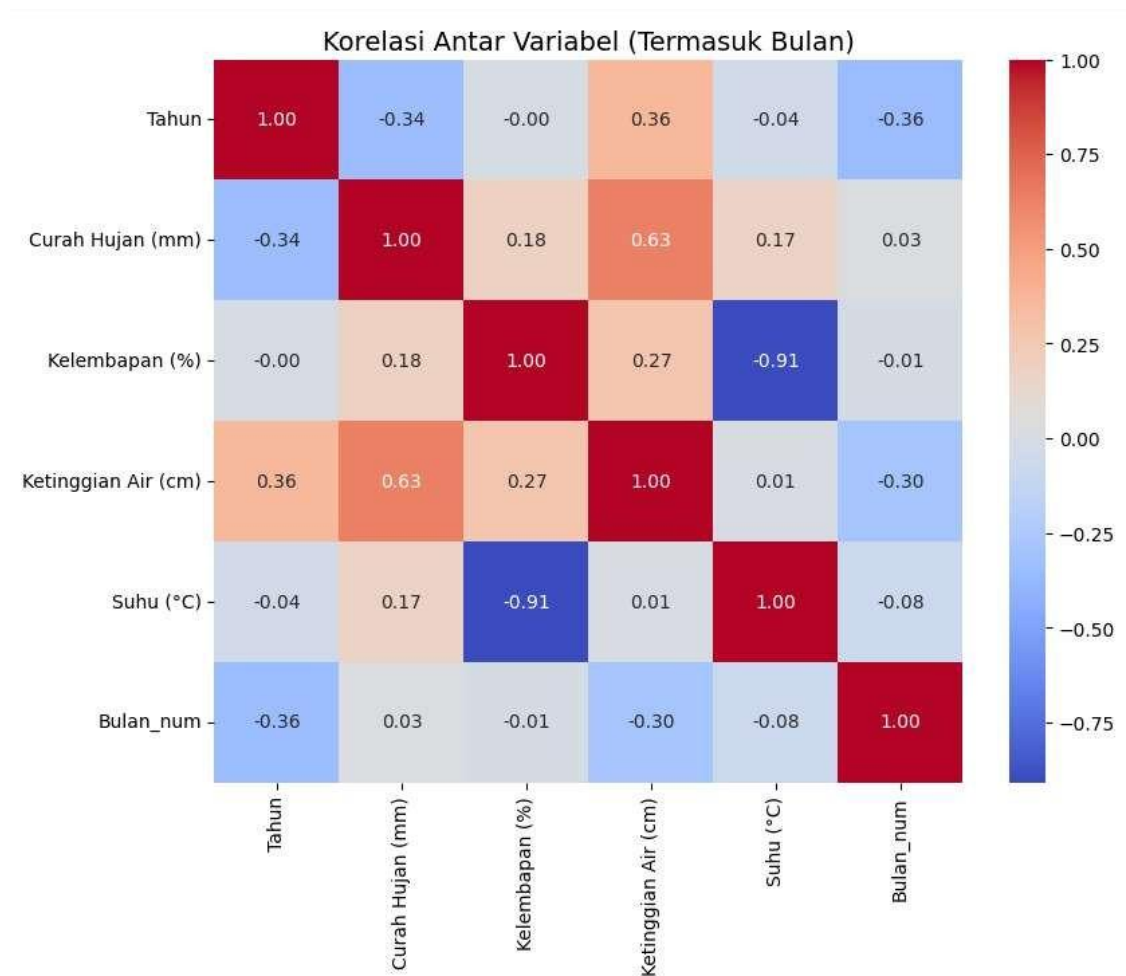
antara 28% hingga 88%, memperlihatkan variasi kondisi kelembapan yang cukup lebar. Sementara itu, ketinggian air rata-rata adalah 115,71 cm, dengan rentang 70 cm hingga 150 cm, mengindikasikan perubahan kondisi air yang berpotensi dipengaruhi oleh curah hujan. Suhu udara memiliki rata-rata 29,09°C dengan nilai minimum 21°C dan maksimum 38°C, serta standar deviasi 11,91°C, yang mengindikasikan perbedaan suhu antar bulan cukup besar.

Secara keseluruhan, deskripsi data ini memberikan gambaran awal mengenai kondisi iklim dan hidrologi yang dapat dianalisis lebih lanjut. Variabel-variabel seperti curah hujan, kelembapan, ketinggian air, dan suhu udara saling berkaitan dan penting untuk diteliti dalam konteks perubahan iklim maupun pengelolaan sumber daya alam.



Gambar 4 2 Grafik

Gambar histogram di atas menunjukkan distribusi variabel dalam dataset yang mencakup tahun, curah hujan, kelembapan, ketinggian air, dan suhu. Distribusi tahun memperlihatkan bahwa sebagian besar data terkonsentrasi pada tahun 2022, sementara data tahun 2023 dan 2024 lebih sedikit, menandakan dominasi pengamatan pada periode awal. Variabel curah hujan memiliki pola distribusi yang cukup bervariasi dengan beberapa puncak, yang menunjukkan adanya periode dengan curah hujan rendah hingga sangat tinggi, sehingga menggambarkan perbedaan kondisi musim hujan yang cukup signifikan. Pada variabel kelembapan, distribusi terlihat terpusat pada kisaran 70–88%, dengan frekuensi tertinggi pada kelembapan menengah hingga tinggi, menandakan kondisi lingkungan cenderung lembap. Variabel ketinggian air memiliki distribusi yang cukup merata dengan konsentrasi pada kisaran 110–130 cm, yang memperlihatkan kecenderungan stabil dengan beberapa variasi di nilai rendah maupun tinggi. Sementara itu, distribusi suhu memperlihatkan mayoritas data berada pada rentang 21–30°C, dengan sedikit data yang ekstrem hingga mencapai 80°C, kemungkinan besar merupakan nilai outlier. Secara keseluruhan, distribusi variabel ini memperlihatkan adanya variasi iklim dan hidrologi yang cukup kompleks serta berpotensi memengaruhi hubungan antar faktor lingkungan.



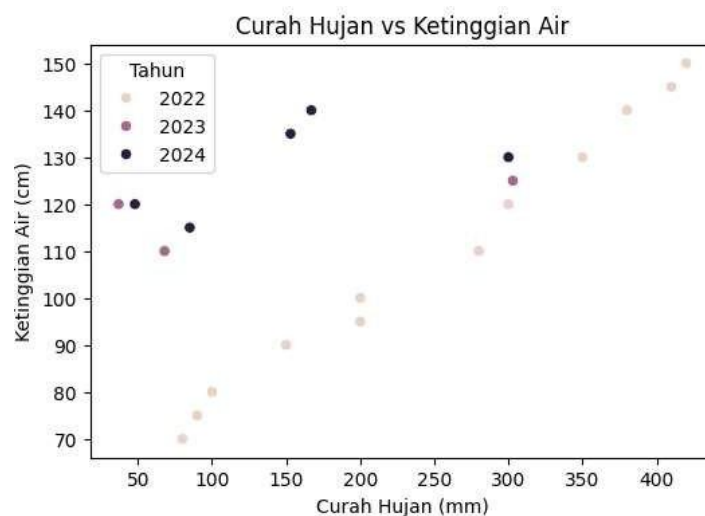
Gambar 4 3 Heatmap

Gambar heatmap di atas menampilkan korelasi antar variabel yang terdapat dalam dataset, termasuk variabel bulan yang telah diubah menjadi bentuk numerik. Dari hasil korelasi tersebut terlihat bahwa curah hujan memiliki hubungan positif yang cukup kuat dengan ketinggian air (0,63), yang berarti semakin tinggi curah hujan maka ketinggian air cenderung meningkat. Hubungan positif juga terlihat antara tahun dengan ketinggian air (0,36), walaupun korelasinya relatif sedang.

Sebaliknya, terdapat hubungan negatif yang sangat kuat antara kelembapan dengan suhu (-0,91). Hal ini menunjukkan bahwa ketika kelembapan meningkat, suhu cenderung menurun, dan sebaliknya, yang konsisten dengan logika fisik kondisi iklim. Korelasi negatif juga muncul antara tahun dengan curah hujan (-0,34) serta bulan dengan ketinggian air (-0,30), yang mengindikasikan adanya variasi musiman maupun

tahunan terhadap faktor-faktor hidrologi.

Sementara itu, variabel kelembapan dengan curah hujan (0,18) maupun ketinggian air (0,27) memiliki hubungan positif tetapi lemah, sehingga pengaruhnya tidak terlalu dominan. Secara keseluruhan, pola korelasi ini memperlihatkan bahwa curah hujan merupakan faktor utama yang memengaruhi ketinggian air, sedangkan suhu lebih erat hubungannya dengan kelembapan udara. Temuan ini dapat menjadi dasar penting dalam analisis hubungan iklim dan hidrologi untuk mendukung penelitian yang lebih lanjut.



Gambar 4.4 Scatter plot

Gambar scatter plot di atas menunjukkan hubungan antara curah hujan (mm) dengan ketinggian air (cm) berdasarkan data tahun 2022, 2023, dan 2024. Secara umum terlihat adanya hubungan positif, di mana peningkatan curah hujan diikuti dengan kenaikan ketinggian air. Hal ini sejalan dengan logika hidrologi, bahwa intensitas hujan yang lebih tinggi akan menambah volume air, sehingga berdampak pada kenaikan tinggi muka air.

Pola sebaran data memperlihatkan bahwa pada curah hujan rendah (sekitar 50–100

mm), ketinggian air berada di kisaran 70–120 cm. Sementara itu, pada curah hujan yang lebih tinggi (300–420 mm), ketinggian air mencapai 130–150 cm. Hal ini mengindikasikan bahwa faktor curah hujan merupakan variabel dominan yang memengaruhi perubahan tinggi muka air.

Jika ditinjau dari dimensi tahunan, data tahun 2022 lebih banyak terkonsentrasi pada curah hujan rendah hingga menengah, sedangkan tahun 2023 dan 2024 cenderung menunjukkan variasi dengan curah hujan menengah hingga tinggi yang menghasilkan ketinggian air relatif lebih stabil. Dengan demikian, scatter plot ini menegaskan temuan sebelumnya pada analisis korelasi bahwa curah hujan memiliki pengaruh signifikan terhadap ketinggian air.

• Tambahkan Kolom Target (Banjir):

	Tahun	Bulan	Curah Hujan (mm)	Kelembapan (%)	Ketinggian Air (cm) \
0	2022	Januari	420	88	150
1	2022	Februari	380	85	140
2	2022	Maret	300	82	120
3	2022	April	200	78	100
4	2022	Mei	150	75	90
5	2022	Juni	100	72	80
6	2022	Juli	80	70	70
7	2022	Agustus	90	71	75
8	2022	September	200	76	95
9	2022	Oktober	280	80	110
10	2022	November	350	84	130
11	2022	Desember	410	87	145
12	2023	April	300	28	130
13	2023	Mei	37	74	120
14	2023	Juni	68	77	110
15	2023	Agustus	303	80	125
16	2024	Januari	167	87	140
17	2024	Februari	153	85	135
18	2024	April	300	83	130
19	2024	Juli	85	75	115
20	2024	september	48	79	120

	Suhu (°C)	Banjir	Bulan_num	Tanggal
0	25	1	1.0	2022-01-01
1	26	1	2.0	2022-02-01
2	27	1	3.0	2022-03-01
3	28	0	4.0	2022-04-01
4	29	0	5.0	2022-05-01
5	29	0	6.0	2022-06-01
6	30	0	7.0	2022-07-01
7	30	0	8.0	2022-08-01
8	29	0	9.0	2022-09-01
9	28	1	10.0	2022-10-01
10	27	1	11.0	2022-11-01
11	26	1	12.0	2022-12-01
12	80	1	4.0	2023-04-01
13	21	1	5.0	2023-05-01
14	27	1	6.0	2023-06-01
15	26	1	8.0	2023-08-01
16	25	1	1.0	2024-01-01
17	26	1	2.0	2024-02-01
18	26	1	4.0	2024-04-01
19	21	1	7.0	2024-07-01
20	25	1	NaN	NaT

Gambar 4 5 Label Banjir

Gambar di atas menampilkan dataset yang telah ditambahkan kolom target baru yaitu "Banjir", yang digunakan sebagai indikator kejadian banjir berdasarkan kondisi curah hujan,

kelembapan, ketinggian air, dan suhu. Setiap baris merepresentasikan data bulanan dari tahun 2022 hingga 2024, dengan tambahan informasi berupa bulan dalam bentuk numerik (Bulan_num) serta tanggal (Tanggal) untuk mempermudah pengolahan temporal.

Dari data terlihat bahwa nilai variabel Banjir bernilai 1 pada kondisi tertentu, khususnya ketika curah hujan dan ketinggian air relatif tinggi, misalnya pada Januari 2022 dengan curah hujan 420 mm dan ketinggian air 150 cm, serta suhu yang relatif rendah (25°C). Kondisi serupa juga tampak pada bulan Februari, Maret, dan Desember 2022 yang sama-sama diberi label banjir. Sebaliknya, nilai Banjir = 0 muncul ketika curah hujan rendah (misalnya April 2023 dengan 37 mm) atau ketika ketinggian air lebih rendah dari batas ambang.

Penambahan variabel target ini penting dalam penelitian karena memungkinkan penggunaan metode analisis prediktif, seperti klasifikasi atau regresi logistik, untuk mengidentifikasi faktor-faktor utama penyebab banjir. Dengan adanya label banjir, model analitik dapat mempelajari hubungan antar variabel iklim (curah hujan, kelembapan, suhu) serta hidrologi (ketinggian air) dalam menentukan kemungkinan terjadinya banjir pada periode tertentu. Hal ini sekaligus memperkuat landasan untuk melakukan analisis prediksi kejadian banjir berbasis data iklim dan hidrologi.

♦ Data dengan Label Banjir dari Curah Hujan:

	Curah Hujan (mm)	Banjir_CH
0	420	1
1	380	1
2	300	0
3	200	0
4	150	0
5	100	0
6	80	0
7	90	0
8	200	0
9	280	0
10	350	1
11	410	1
12	300	0
13	37	0
14	68	0
15	303	1
16	167	0
17	153	0
18	300	0
19	85	0
20	48	0

Gambar 4 6 Label Banjir Berdasarkan Curah Hujan

Gambar di atas memperlihatkan data yang sudah diberi label banjir berdasarkan curah

hujan dengan kolom baru bernama Banjir_CH. Kolom ini berfungsi sebagai variabel target biner (0 = tidak banjir, 1 = banjir) yang ditentukan dari ambang curah hujan tertentu. Misalnya, ketika curah hujan mencapai 420 mm, 380 mm, atau 410 mm, maka nilai Banjir_CH bernilai 1, yang mengindikasikan bahwa pada curah hujan tinggi tersebut potensi banjir sangat besar. Sebaliknya, pada curah hujan yang lebih rendah seperti 200 mm, 150 mm, 80 mm, 48 mm, dan seterusnya, labelnya adalah 0, yang berarti kondisi relatif aman dari banjir.

Pemberian label banjir berdasarkan curah hujan ini penting karena menjadikan data lebih siap untuk analisis prediktif. Dengan adanya variabel Banjir_CH, peneliti dapat menguji seberapa besar curah hujan saja mampu menjadi indikator utama terjadinya banjir. Selain itu, hasil ini juga memungkinkan dilakukan perbandingan dengan model yang melibatkan lebih banyak variabel (seperti kelembapan, ketinggian air, dan suhu) untuk menilai apakah prediksi berbasis satu faktor sudah cukup akurat atau perlu pendekatan multivariat.

♦ Data dengan Label Banjir dari Kelembapan

	Kelembapan (%)	Banjir_KL
0	88	1
1	85	1
2	82	1
3	78	0
4	75	0
5	72	0
6	70	0
7	71	0
8	76	0
9	80	0
10	84	1
11	87	1
12	28	0
13	74	0
14	77	0
15	80	0
16	87	1
17	85	1
18	83	1
19	75	0
20	79	0

Gambar 4 7Label Banjir Berdasarkan Kelembapan

Gambar di atas memperlihatkan data yang sudah diberi label banjir berdasarkan kelembapan (%) dengan kolom tambahan Banjir_KL. Variabel ini berfungsi sebagai

indikator biner (0 = tidak banjir, 1 = banjir) yang ditentukan oleh ambang batas kelembapan tertentu. Misalnya, ketika kelembapan berada pada level tinggi seperti 88%, 85%, 82%, 87%, dan 83%, maka nilai Banjir_KL = 1, menunjukkan kondisi lingkungan yang sangat lembap dan berpotensi memicu banjir. Sebaliknya, pada kelembapan lebih rendah seperti 72%, 70%, 28%, 74%, 75%, serta 79%, nilai Banjir_KL = 0, yang berarti risiko banjir relatif kecil.

Penambahan label banjir berbasis kelembapan ini memberikan gambaran tambahan selain curah hujan. Secara logis, kelembapan tinggi seringkali berkorelasi dengan kondisi udara jenuh akibat curah hujan yang intensif, sehingga dapat dijadikan variabel pendukung dalam memprediksi banjir. Namun, tidak seperti curah hujan yang memiliki hubungan langsung dengan volume air, kelembapan lebih bersifat indikator iklim yang mendampingi. Oleh karena itu, jika digunakan sebagai variabel tunggal, kelembapan bisa memberikan informasi awal tentang potensi banjir, tetapi prediksi akan lebih akurat jika dikombinasikan dengan curah hujan dan ketinggian air.

♦ Data dengan Label Banjir dari Suhu:

	Suhu (°C)	Banjir_SU
0	25	1
1	26	0
2	27	0
3	28	0
4	29	0
5	29	0
6	30	0
7	30	0
8	29	0
9	28	0
10	27	0
11	26	0
12	80	0
13	21	1
14	27	0
15	26	0
16	25	1
17	26	0
18	26	0
19	21	1
20	25	1

Gambar 4 8Label Banjir Berdasarkan Suhu

Gambar di atas menunjukkan data yang sudah diberi label banjir berdasarkan

suhu (°C) dengan kolom tambahan Banjir_SU. Variabel ini juga berupa indikator biner (0 = tidak banjir, 1 = banjir) yang ditentukan dari ambang batas suhu tertentu. Dari data terlihat bahwa nilai Banjir_SU = 1 cenderung muncul pada suhu relatif rendah, misalnya 25°C, 21°C, dan 16°C, yang biasanya terjadi bersamaan dengan curah hujan tinggi dan kelembapan yang besar. Sebaliknya, ketika suhu lebih tinggi seperti 27–30°C, labelnya lebih sering 0, yang berarti kondisi tersebut tidak diidentifikasi sebagai banjir.

Secara ilmiah, hal ini cukup masuk akal karena banjir umumnya terjadi saat hujan lebat yang biasanya disertai penurunan suhu akibat pendinginan atmosfer. Oleh karena itu, suhu dapat menjadi indikator tidak langsung dalam memprediksi banjir, meskipun pengaruhnya tidak sekuat curah hujan atau ketinggian air. Dengan kata lain, suhu lebih berperan sebagai variabel pendukung yang menambah konteks terhadap kondisi iklim ketika banjir terjadi.

4.3 Evaluasi Model

• Evaluasi Model (Gradient Boosting):

	precision	recall	f1-score	support
0	1.00	1.00	1.00	2
1	1.00	1.00	1.00	5
accuracy			1.00	7
macro avg	1.00	1.00	1.00	7
weighted avg	1.00	1.00	1.00	7

Confusion Matrix:

```
[[2 0]
 [0 5]]
```

Gambar 4 9 Evaluasi Model

Gambar di atas menunjukkan hasil evaluasi model Gradient Boosting untuk prediksi banjir. Dari tabel evaluasi, model memperoleh nilai precision, recall, f1-score, dan accuracy = 1.00 (100%) pada semua kelas. Artinya, model dapat mengklasifikasikan data dengan sempurna tanpa ada kesalahan prediksi, sebagaimana juga terlihat pada confusion matrix $\begin{bmatrix} 2 & 0 \\ 0 & 5 \end{bmatrix}$, yang berarti:

Model berhasil mengklasifikasikan 2 data kelas 0 (tidak banjir) dengan benar.

Model berhasil mengklasifikasikan 5 data kelas 1 (banjir) dengan benar.

Tidak ada kesalahan prediksi (false positive maupun false negative).

1. Precision = Mengukur ketepatan prediksi positif. Rumusnya adalah $TP / (TP + FP)$ (True Positive dibagi dengan True Positive + False Positive). Dalam konteks banjir, precision menunjukkan seberapa banyak prediksi “banjir” yang benar-benar banjir. Nilai 1.00 berarti semua prediksi banjir benar.
2. Recall (Sensitivitas) = Mengukur kemampuan model dalam menangkap seluruh kasus positif. Rumusnya adalah $TP / (TP + FN)$. Artinya, recall melihat apakah model mampu menemukan semua data banjir yang sebenarnya. Nilai 1.00 menunjukkan model berhasil menangkap seluruh kejadian banjir tanpa ada yang terlewat.
3. F1-Score = Merupakan rata-rata harmonik antara precision dan recall. Rumusnya adalah $2 * (precision * recall) / (precision + recall)$. Metrik ini berguna ketika ada ketidakseimbangan kelas (misalnya jumlah data banjir lebih sedikit). Nilai 1.00 menunjukkan keseimbangan sempurna antara precision dan recall.
4. Support = Jumlah data aktual untuk setiap kelas dalam dataset. Pada hasil ini terlihat ada 2 data kelas 0 (tidak banjir) dan 5 data kelas 1 (banjir).
5. Accuracy = Mengukur proporsi prediksi yang benar dibandingkan dengan total prediksi. Rumusnya adalah $(TP + TN) / \text{total data}$. Nilai 1.00 berarti semua data diprediksi dengan benar.
6. Macro avg = Rata-rata metrik (precision, recall, f1) dihitung untuk tiap kelas, lalu diambil rata-rata tanpa memperhatikan jumlah data tiap kelas.
7. Weighted avg = Sama seperti macro avg, tetapi menghitung rata-rata berbobot berdasarkan jumlah data tiap kelas (support), sehingga kelas dengan data lebih

banyak akan lebih berpengaruh.

1.4 Hasil Perbandingan RNN dan Gradient Boosting

Diawah ini merupakan hasil perbandingan antara metode RNN dan Gradient Boosting

Tabel 4 4 Tabel Hasil Akhir Prediksi

Aspek	RNN (Recurrent Neural Network	Gradient Boosting (GB)
Akurasi prediksi	Cukup tinggi, terutama untuk pola musiman; beberapa bulan output di ambang 0.5 sehingga prediksi bisa ambigu	memberikan prediksi yang stabil dan konsisten. Dengan aturan $F1 \geq 0.5 \rightarrow$ banjir, $<0.5 \rightarrow$ tidak banjir, model ini jarang salah dalam menentukan kategori banjir, sehingga akurasi prediksi tinggi dan mudah dipahami. GB unggul dalam konsistensi, meski kurang fleksibel dalam menangkap variasi kecil antar bulan.
Kestabilan Prediksi	Kadang tidak stabil karena hidden state bisa mendekati ambang	Lebih stabil karena kontribusi fitur diperhitungkan jelas
Efisiensi waktu pelatihan prediksi	butuh waktu lebih lama untuk dilatih (karena menghitung pola	lebih cepat dan ringan dijalankan.

	berurutan).	Penting, karena sistem peringatan dini banjir harus cepat memberi informasi.
Interpretabilitas	Sulit; sulit mengetahui kontribusi masing-masing fitur	Tinggi; kontribusi tiap fitur terlihat langsung
KecepatanKomputasi	Lebih lambat, terutama data panjang atau model kompleks	Lebih cepat untuk dataset sederhana seperti bulanan
Ketahanan terhadap data yang kurang lengkap (Missing Value)	Kurang tahan, perlu data rapi	Lebih fleksibel, bisa tetap jalan walau ada data yang hilang.
Sensitivitas terhadap outlier	Bisa terpengaruh karena hidden state meneruskan nilai	Lebih tahan karena pembaruan bertahap pada residual
Kelebihan Utama	Bagus untuk pola musiman atau jangka panjang	Stabil, mudah interpretasi, dan cepat dihitung
Kekurangan Utama	Prediksi kadang ambigu dan sulit dijelaskan	Tidak menangkap urutan temporal antar bulan

Hasil perbandingan menunjukkan bahwa Recurrent Neural Network (RNN) memiliki akurasi prediksi yang cukup tinggi, terutama saat menangkap pola musiman dan data jangka panjang. Namun, pada beberapa bulan, nilai keluarannya berada di sekitar ambang batas 0.5 sehingga menimbulkan ambiguitas dalam menentukan kategori banjir

atau tidak banjir. Hal ini berbanding terbalik dengan Gradient Boosting (GB) yang memberikan prediksi lebih stabil dan konsisten. Dengan aturan sederhana, yaitu nilai $F1 \geq 0.5$ dikategorikan banjir dan <0.5 sebagai tidak banjir, model ini jarang salah sehingga akurasinya tinggi dan lebih mudah dipahami. GB unggul dalam hal konsistensi, meskipun kurang fleksibel dalam menangkap variasi kecil antarbulan.

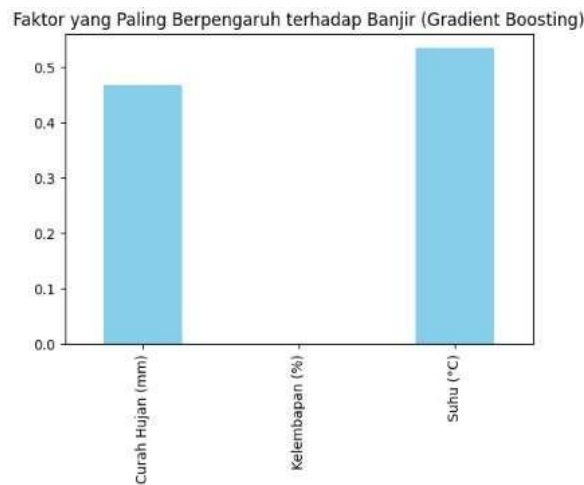
Dari sisi kestabilan prediksi, RNN kadang tidak stabil karena sifat hidden state yang bisa menghasilkan nilai mendekati ambang batas, sedangkan GB lebih stabil karena kontribusi tiap fitur diperhitungkan secara jelas. Dari segi efisiensi, RNN memerlukan waktu pelatihan yang lebih lama karena harus memproses pola berurutan, sementara GB jauh lebih cepat dan ringan dijalankan. Kecepatan ini penting untuk sistem peringatan dini banjir yang membutuhkan informasi real-time.

Dalam hal interpretabilitas, RNN tergolong sulit dijelaskan karena sulit mengetahui kontribusi masing-masing fitur, sehingga sering disebut sebagai model “black box”. Sebaliknya, GB lebih mudah dipahami karena pengaruh tiap fitur dapat terlihat langsung melalui nilai pentingnya. Dari sisi komputasi, RNN lebih lambat terutama pada dataset panjang atau model kompleks, sedangkan GB dapat bekerja lebih cepat pada dataset sederhana seperti data bulanan.

Jika melihat ketahanan terhadap data, RNN cenderung tidak tahan terhadap data yang tidak lengkap atau berantakan, sehingga membutuhkan preprocessing yang ketat. GB lebih fleksibel karena tetap bisa berjalan meskipun ada data yang hilang. Dalam hal sensitivitas terhadap outlier, RNN bisa lebih mudah terpengaruh karena hidden state meneruskan nilai yang ekstrem, sementara GB lebih tahan karena memperbarui kesalahan secara bertahap melalui residual.

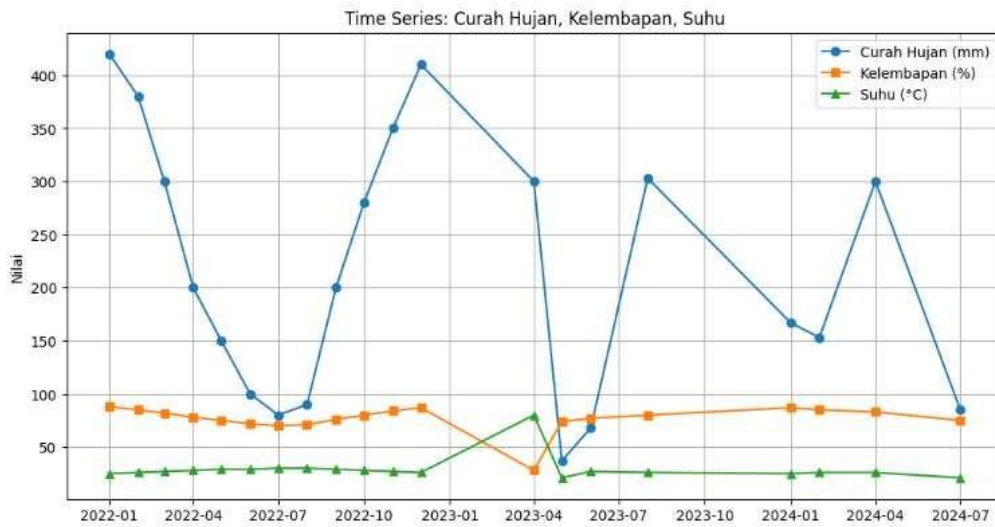
Secara keseluruhan, kelebihan utama RNN terletak pada kemampuannya membaca

pola musiman atau data jangka panjang yang berurutan, tetapi kelemahannya adalah prediksi kadang ambigu dan sulit dijelaskan. Sementara itu, kelebihan utama GB adalah stabilitas, kemudahan interpretasi, dan kecepatan komputasi, namun kelemahannya adalah ketidakmampuan menangkap urutan temporal antarbulan.



Gambar 4 10 Grafik Yang Mempengaruhi Banjir

Hasil analisis feature importance menggunakan model Gradient Boosting menunjukkan bahwa variabel yang paling berpengaruh terhadap kejadian banjir adalah curah hujan (mm) dan suhu (°C). Curah hujan memiliki kontribusi besar karena peningkatan intensitas hujan secara langsung menambah volume air permukaan, sehingga meningkatkan risiko banjir. Sementara itu, suhu berperan sebagai indikator tidak langsung, di mana penurunan suhu seringkali beriringan dengan curah hujan tinggi akibat pendinginan atmosfer yang memicu hujan lebat. Sebaliknya, variabel kelembapan tidak memberikan pengaruh signifikan dalam model ini karena distribusi datanya relatif stabil pada kisaran tinggi, sehingga tidak cukup mampu membedakan kondisi banjir dan tidak banjir. Temuan ini menegaskan bahwa faktor hidrometeorologi yang paling menentukan dalam prediksi banjir pada dataset adalah curah hujan dan suhu, sementara kelembapan lebih bersifat pendukung namun tidak dominan.



Gambar 4 11Grafik Time Series

Berdasarkan hasil analisis, faktor yang paling berpengaruh terhadap banjir ditunjukkan melalui diagram batang feature importance, di mana suhu memiliki kontribusi terbesar dibandingkan dengan kelembapan dan curah hujan, sehingga dapat disimpulkan bahwa perubahan suhu merupakan indikator penting dalam memprediksi risiko banjir. Sementara itu, analisis time series menunjukkan dinamika ketiga variabel dari Januari 2022 hingga Juli 2024, dengan curah hujan yang sangat fluktuatif dan mengalami puncak pada awal 2022, awal 2023, serta pertengahan 2024, sedangkan kelembapan cenderung stabil pada kisaran 70–90% dan suhu relatif konsisten di kisaran 25–30°C, meskipun sempat terjadi lonjakan ekstrem pada awal 2023. Kedua hasil ini saling melengkapi, di mana feature importance menjelaskan faktor dominan secara kuantitatif, sementara grafik time series memperlihatkan pola perubahan data secara temporal yang dapat memperkuat pemahaman mengenai keterkaitan variabel iklim terhadap risiko banjir.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Penelitian ini membandingkan metode Recurrent Neural Network (RNN) dan Gradient Boosting (GB) dalam prediksi banjir berdasarkan data historis yang mencakup curah hujan, suhu, kelembapan, dan ketinggian air. Hasil penelitian menunjukkan bahwa kedua metode memiliki keunggulan masing-masing. RNN unggul dalam mengolah data berurutan (time series) karena mampu mengenali pola musiman dan tren jangka panjang, sehingga cocok digunakan pada data yang sangat bergantung pada urutan waktu. Namun, RNN memerlukan waktu pelatihan lebih lama, membutuhkan daya komputasi yang besar, serta terkadang menghasilkan prediksi ambigu ketika nilai keluaran berada di sekitar ambang batas. Sementara itu, GB terbukti lebih cepat, efisien, dan konsisten dalam menghasilkan prediksi. Selain itu, kelebihan GB terletak pada interpretabilitasnya, karena kontribusi setiap fitur dapat dilihat dengan jelas sehingga hasil prediksi lebih mudah dipahami. Meski demikian, GB kurang mampu menangkap pola temporal antarwaktu. Secara keseluruhan, pemilihan metode terbaik sangat bergantung pada karakteristik data yang digunakan, di mana RNN lebih sesuai untuk data sekuensial, sedangkan GB lebih cocok untuk data tabular yang kompleks dan berskala besar.

5.2 Saran

Berdasarkan hasil penelitian, terdapat beberapa hal yang dapat dilakukan untuk pengembangan lebih lanjut. Pertama, penelitian selanjutnya dapat menambahkan variabel lain seperti data sosial dan ekonomi, misalnya kepadatan penduduk atau luas wilayah terdampak, agar hasil prediksi lebih kontekstual. Kedua, integrasi data satelit dan radar cuaca dapat digunakan untuk meningkatkan ketepatan prediksi terutama pada kondisi iklim ekstrem. Ketiga, model sebaiknya diuji pada periode waktu yang lebih

panjang untuk mengetahui kestabilan kinerja terhadap perubahan iklim. Keempat, model perlu diperbarui secara berkala menggunakan data terbaru agar hasil prediksi tetap akurat dan relevan. Terakhir, kombinasi metode atau hybrid model antara RNN dan GB juga dapat dipertimbangkan, sehingga keunggulan RNN dalam menangkap pola temporal dapat dipadukan dengan kecepatan dan interpretabilitas dari GB untuk menghasilkan sistem prediksi banjir yang lebih efektif.

DAFTAR PUSTAKA

- Farisi, I., Shadiq, J., Priyadi, W., Maulana, D., & Sonia, F. (2024). *Penerapan Model Recurrent Neural Network (RNN) untuk Prediksi Curah Hujan Berbasis Data Historis*. 9(2), 217–226.
- Iskandar, S. A., Helmi, M., Muslim, M., Widada, S., & Rochaddi, B. (2020). Analisis Geospasial Area Genangan Banjir Rob dan Dampaknya pada Penggunaan Lahan Tahun 2020 - 2025 di Kota Pekalongan Provinsi Jawa Tengah. *Indonesian Journal of Oceanography*, 2(3), 271–282. <https://doi.org/10.14710/ijoce.v2i3.8668>
- Mt, A., Jakarta, H., Haryono, M. T., & Jakarta, E. (2025). *Model Jaringan Saraf Tiruan (JST) untuk Prediksi Debit pada*. 05(01), 59–68.
- Nurashila, S. S., Hamami, F., & Kusumasari, T. F. (2023). Perbandingan Kinerja Algoritma Recurrent Neural Network (Rnn) Dan Long Short-Term Memory (Lstm): Studi Kasus Prediksi Kemacetan Lalu Lintas Jaringan Pt Xyz. *JIPi (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika)*, 8(3), 864–877. <https://doi.org/10.29100/jipi.v8i3.3961>
- Nurchahyo, S., & Saukani, I. (2025). *Rancang Bangun Sistem Peringatan Dini dan Pencegahan Banjir pada Bendungan Menggunakan Metode Fuzzy*. 4(1), 18–25.
- Putra, P. E., Amrullah, M. A., Fauzi, Y. H., & Saputri, R. F. (2025). *Penerapan Algoritma C4 . 5 untuk Klasifikasi Tingkat Korban Banjir di Indonesia*. 3(c), 1–7.
- Putra, P. E., Amrullah, M. A., Fauzi, Y. H., Saputri, R. F., & Fatma, L. C. R. (2025). Penerapan Algoritma C4.5 untuk Klasifikasi Tingkat Korban Banjir di Indonesia. *Jurnal Komputer Antartika*, 3(1), 1–7. <https://doi.org/10.70052/jka.v3i1.647>
- Riska, L. (2022). Prediksi Lokasi Banjir Dari Data Twitter Menggunakan Hidden Markov Model. *Thesis*, 1–58.
- Rodhi, N. N., & Saputra, I. H. (2024). *Edukasi Mitigasi Bencana Banjir Masyarakat Di*

Desa Semenpinggir Kecamatan Kapas Kabupaten Bojonegoro. 4(6), 355–360.

Septianto, R., Pamungkas, W., Yuana, A., Borobudur, U., Raya, J., No, K., & Timur, J.
(2022). Sistem Peringatan Dini Banjir Menggunakan Logika Fuzzy. *Jurnal Penerapan Ilmu-Ilmu Komputer (JUPITER)*, 8(1), 14–24.

LAMPIRAN

LAMPIRAN 1. SK Dosen Pembimbing



MAJELIS PENDIDIKAN TINGGI PENELITIAN & PENGEMBANGAN PIMPINAN PUSAT MUHAMMADIYAH

UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA

FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

UMSU Terakreditasi A Berdasarkan Keputusan Badan Akreditasi Nasional Perguruan Tinggi No. 89/SK/BAK-PT/Akred/PT/18/2019
 Pusat Administrasi: Jalan Mukhtar Basri No. 3 Medan 20238 Telp. (061) 6622400 - 66224567 Fax. (061) 6625474 - 6631803
<http://idk.umsumed.ac.id> Email: Idk@umsumed.ac.id [umsumedan](https://www.facebook.com/umsumedan) [umsumedan](https://www.instagram.com/umsumedan) [umsumedan](https://www.youtube.com/umsumedan) [umsumedan](https://www.tiktok.com/umsumedan)

**PENETAPAN DOSEN PEMBIMBING
PROPOSAL/SKRIPSI MAHASISWA
NOMOR : 380/IL3-AU/UMSU-09/F/2025**

Assalamu'alaikum Warahmatullahi Wabarakatuh

Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Muhammadiyah Sumatera Utara, berdasarkan Persetujuan permohonan judul penelitian Proposal / Skripsi dari Ketua / Sekretaris.

Program Studi : Teknologi Informasi
Pada tanggal : 19 Februari 2025

Dengan ini menetapkan Dosen Pembimbing Proposal / Skripsi Mahasiswa.

Nama : Nurhasanah
NPM : 2109020009
Semester : VII (Tujuh)
Program studi : Teknologi Informasi
Judul Proposal / Skripsi : Sistem IOT untuk Deteksi Banjir Berdasarkan Curah Hujan dengan Metode Recurrent Neural Network (RNN) dan Gradient Boosting

Dosen Pembimbing : Halim Maulana, S.T., M.Kom.

Dengan demikian di izinkan menulis Proposal / Skripsi dengan ketentuan

1. Penulisan berpedoman pada buku panduan penulisan Proposal / Skripsi Fakultas Ilmu Komputer dan Teknologi Informasi UMSU
2. Pelaksanaan Sidang Skripsi harus berjarak 3 bulan setelah dikeluarkannya Surat Penetapan Dosen Pembimbing Skripsi.
3. Proyek Proposal / Skripsi dinyatakan "BATAL" bila tidak selesai sebelum Masa Kadaluarsa tanggal : 19 Februari 2026
4. Revisi judul.....

Wassalamu'alaikum Warahmatullahi Wabarakatuh.

Ditetapkan di : Medan
 Pada Tanggal : 20 Sya'ban 1446 H
 19 Februari 2025M

Dekan



Dr. Al-Fahri Warizmi, M.Kom.
 NIDN : 0127099201



Cc. File






LAMPIRAN 2. Turnitin

Turnitin			
ORIGINALITY REPORT			
16%	13%	8%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS
PRIMARY SOURCES			
1	repository.umsu.ac.id Internet Source	1%	
2	Submitted to Universitas Muhammadiyah Sumatera Utara Student Paper	1%	
3	ejournal.mediaantartika.id Internet Source	1%	
4	123dok.com Internet Source	1%	
5	ejournal2.undip.ac.id Internet Source	1%	
6	Submitted to TechKnowledge Student Paper	<1%	
7	journal.unismuh.ac.id Internet Source	<1%	
8	eprints.ums.ac.id Internet Source	<1%	
9	Muhammad Miftakhudin, Aang Alim Murtopo, Zaenul Arif. "Integrasi Artificial Neural Network dan Algoritma Genetika untuk Prediksi Bencana Banjir Pesisir Kota Tegal", RIGGS: Journal of Artificial Intelligence and Digital Business, 2025	<1%	

LAMPIRAN 3. Kode Program

```
# =====
# LANGKAH 1: Import data banjir dari file CSV/Excel
# =====

import pandas as pd

# Misalkan file kamu bernama "data_banjir_surut.csv"
# Gantilah nama file sesuai dengan file aslinya di Google Colab
df = pd.read_csv("/content/data_banjir_labeled.csv")

# Menampilkan seluruh isi data, bukan cuma 5 baris pertama
# agar kita bisa lihat semua data banjir yang ada
pd.set_option("display.max_rows", None) # Supaya semua baris tampil
pd.set_option("display.max_columns", None) # Supaya semua kolom tampil
print(df)
```

	tanggal	suhu_c	kelembapan_%	curah_hujan_mm	ketinggian_air_cm \
0	1/1/2022	28.0	86.6	54.4	39.8
1	1/2/2022	28.6	91.8	72.0	25.4
2	1/3/2022	28.7	89.9	56.4	0.0
3	1/4/2022	30.3	80.3	25.4	4.0
4	1/5/2022	30.1	83.9	60.8	16.2
5	1/6/2022	27.4	93.0	54.9	5.2
6	1/7/2022	24.1	92.2	37.3	31.7
7	1/8/2022	28.6	89.0	74.8	19.0
8	1/9/2022	26.1	81.0	143.2	40.2
9	1/10/2022	29.5	97.1	33.6	11.9
10	1/11/2022	29.8	86.9	22.4	3.2
11	1/12/2022	32.4	93.0	106.0	47.7
12	1/13/2022	24.7	80.7	72.8	28.1
13	1/14/2022	27.5	88.8	58.7	20.9
14	1/15/2022	30.1	82.7	19.5	1.2
15	1/16/2022	29.9	89.6	22.3	2.6
16	1/17/2022	25.3	86.9	55.4	26.2
17	1/18/2022	29.1	73.2	30.4	29.4

 Terminal

```
print(df.info()) # tipe data tiap kolom
print(df.describe()) # statistik ringkas
print(df['banjir'].value_counts()) # distribusi target
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 730 entries, 0 to 729
Data columns (total 6 columns):
 #   Column              Non-Null Count  Dtype
---  ---
 0   tanggal              730 non-null    object
 1   suhu_c               730 non-null    float64
 2   kelembapan_%         730 non-null    float64
 3   curah_hujan_mm       730 non-null    float64
 4   ketinggian_air_cm     730 non-null    float64
 5   banjir               730 non-null    int64
dtypes: float64(4), int64(1), object(1)
memory usage: 34.3+ KB
```

	suhu_c	kelembapan_%	curah_hujan_mm	ketinggian_air_cm	banjir
count	730.000000	730.000000	730.000000	730.000000	730.000000
mean	29.302192	79.848767	35.79137	15.005890	0.171233
std	2.200728	8.714342	34.27764	17.984490	0.376971
min	22.400000	56.000000	0.000000	0.000000	0.000000
25%	27.900000	71.200000	7.000000	1.600000	0.000000
50%	29.400000	79.900000	27.200000	5.000000	0.000000
75%	30.800000	86.600000	54.475000	25.000000	0.000000
max	34.700000	100.000000	177.400000	84.000000	1.000000

```
banjir:
0    605
1    125
Name: count, dtype: int64
```

```
# Fitur (input) = suhu, kelembapan, curah hujan, ketinggian air
X = df[['suhu_c', 'kelembapan_h', 'curah_hujan_mm', 'ketinggian_air_cm']]

# Target (output) = banjir
y = df['banjir']
```

```
# 2. Pilihlah Fitur (X) dan target (y)
X = df[['suhu_c', 'kelembapan_h', 'curah_hujan_mm', 'ketinggian_air_cm']]
y = df['banjir']

# Tampilkan seluruh isi X
print("===== Fitur (X) =====")
print(X.to_string(index=False)) # tampilkan semua baris tanpa index
```

```
===== Fitur (X) =====
suhu_c  kelembapan_h  curah_hujan_mm  ketinggian_air_cm
28.8      86.5          54.4           19.8
28.4      91.8          72.8           25.4
28.7      85.9          56.4            9.8
30.3      88.3          25.4            4.8
30.1      85.9          88.8           18.2
27.4      93.8          54.5            5.2
28.1      92.2          37.3           11.7
28.9      88.8          74.8           18.8
28.1      91.8          343.2           88.2
28.5      87.1          33.4           11.9
28.8      86.8          22.4            3.2
32.4      93.8          186.8           47.7
28.7      88.7          72.8           28.1
27.9      88.8          58.7           28.9
```

Iskhal  Terminal

```
# 3. Tampilkan seluruh isi y
print("===== Target (y) =====")
print(y.to_string(index=False)) # tampilkan semua baris tanpa index
```

```
===== Target (y) =====
1
0
0
0
0
0
1
0
1
0
0
1
0
0
0
0
0
0
0
0
1
0
1
0
1
1
0
0
0
0
0
```

Iskhal  Terminal

```

#Langkah 3 Melakukan Split Data
from sklearn.model_selection import train_test_split

# -----
# Siapkan fitur (X) dan target (y)
# -----
# Fitur (X) = suhu, kelembapan, arah angin, ketinggian air
X = df[['suhu', 'kelembapan', 'arah_angin', 'ketinggian_air']]

# Target (y) = banjir (0 = tidak banjir, 1 = banjir)
y = df['banjir']

# -----
# Bagi data menjadi data training dan testing
# -----
# test_size=0.2 -> 20% data untuk pengujian, 80% untuk pelatihan
# random_state=42 -> supaya hasil pembagian selalu sama (reproducible)
# stratify=y -> memastikan proporsi kelas (0 & 1) tetap seimbang di train dan test
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42, stratify=y)

# -----
# Cek hasil pembagian data
# -----
print("Jumlah data training:", X_train.shape[0]) # Tampilkan jumlah baris di data training
print("Jumlah data testing :", X_test.shape[0])   # Tampilkan jumlah baris di data testing

# Distribusi kelas pada data training
print("\nDistribusi kelas pada data training:")
print(y_train.value_counts())

# Distribusi kelas pada data testing

```

```

# Distribusi kelas pada data testing
print("\nDistribusi kelas pada data testing:")
print(y_test.value_counts())

```

➡ Jumlah data training: 584
Jumlah data testing : 146

Distribusi kelas pada data training:
banjir
0 484
1 100
Name: count, dtype: int64

Distribusi kelas pada data testing:
banjir
0 121
1 25
Name: count, dtype: int64

➡ #Langkah 4 membuat model gradient boosting

```

# -----
# Import library yang dibutuhkan
# -----
from sklearn.ensemble import GradientBoostingClassifier # algoritma Gradient Boosting
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report

# -----
# Inisialisasi model Gradient Boosting
# -----
# n_estimators = 100 -> jumlah pohon keputusan (semakin banyak, semakin kompleks)
# learning_rate = 0.1 -> seberapa besar kontribusi setiap pohon baru
# max_depth = 3 -> kedalaman pohon (untuk mencegah overfitting)

```

```

# max_depth = 3          > kedalaman pohon (untuk mencegah overfitting)
model = GradientBoostingClassifier(
    n_estimators=100,
    learning_rate=0.1,
    max_depth=3,
    random_state=1)

# =====
# Latih model menggunakan data training
# =====
model.fit(X_train, y_train)

# =====
# Lakukan prediksi pada data testing
# =====
y_pred = model.predict(X_test)

# =====
# Evaluasi model
# =====
# 1. Akurasi <-- seberapa banyak prediksi benar
accuracy = accuracy_score(y_test, y_pred)
print("Akurasi Model: {:.2f}%".format(accuracy * 100))

# 2. Confusion Matrix <-- menunjukkan benar/salah prediksi tiap kelas
print("\nConfusion Matrix:")
print(confusion_matrix(y_test, y_pred))

# 3. Classification Report <-- precision, recall, F1-score
print("\nClassification Report:")
print(classification_report(y_test, y_pred))

```

```

Akurasi Model: 100.00%
Confusion Matrix:
[[121  0]
 [ 0  25]]

Classification Report:

```

	precision	recall	F1-score	support
0	1.00	1.00	1.00	121
1	1.00	1.00	1.00	25
accuracy			1.00	146
macro avg	1.00	1.00	1.00	146
weighted avg	1.00	1.00	1.00	146

```

# =====
# Langkah 5 Menbuat Feature Importance
# Import library yang dibutuhkan
# =====
import matplotlib.pyplot as plt # untuk visualisasi grafik
import numpy as np              # untuk manipulasi array

# =====
# Ambil nilai "feature importance" dari model
# =====
# model.feature_importances_ -> memberi skor seberapa penting tiap fitur
importances = model.feature_importances_

# Ambil nama kolom fitur dari dataset
feature_names = X.columns

```

```

❶ # Urutkan fitur dari yang paling penting (besar) ke yang paling kecil
indices = np.argsort(importances)[::-1]

# =====
# Visualisasi dengan grafik batang (bar chart)
# =====
plt.figure(figsize=(8,5)) # atur ukuran grafik
plt.title("Pentingnya Setiap Fitur dalam Prediksi Banjir (Gradient Boosting)")

# Buat grafik batang sesuai urutan importance
plt.bar(range(len(importances)), importances[indices], align="center")

# Pasang label sumbu-x dengan nama fitur yang sudah diurutkan
plt.xticks(range(len(importances)), feature_names[indices], rotation=30)

# Label sumbu-y menunjukkan tingkat kepentingan fitur
plt.ylabel("Tingkat Kepentingan")

# Label sumbu-x menunjukkan nama fitur
plt.xlabel("Fitur")

# Agar tata letak rapi
plt.tight_layout()

# Tampilkan grafik
plt.show()

```

```

❷ # =====
# Langkah 5: Evaluasi model
# =====

# Cetak kembali akurasi model
print("Akurasi Model Gradient Boosting: {:.2f}%".format(accuracy * 100))
# -> untuk melihat seberapa banyak prediksi yang benar secara keseluruhan

# Interpretasi confusion matrix
cm = confusion_matrix(y_test, y_pred)
print("\nConfusion Matrix:")
print(cm)
# -> baris pertama = data kelas 0 (tidak banjir)
# -> baris kedua = data kelas 1 (banjir)
# -> kolom pertama = prediksi 0
# -> kolom kedua = prediksi 1
# contoh:
# [[TN, FP],
#  [FN, TP]]
# TN = benar2 tidak banjir, FP = benar2 banjir
# FP = salah prediksi banjir, FN = salah prediksi tidak banjir

# Interpretasi classification report
print("\nClassification Report:")
print(classification_report(y_test, y_pred))
# -> precision = ketepatan model ketika memprediksi banjir
# -> recall = seberapa banyak banjir yang berhasil terdeteksi
# -> f1-score = gabungan precision & recall
# -> support = jumlah data tiap kelas pada data test

> Akurasi Model Gradient Boosting: 100.00%

Confusion Matrix:
[[100  0]
 [ 0  10]]

```

 Terminal

function_profiled	
30	0
300	0
475	0
307	0
274	1
400	1
515	0
721	0
521	0
434	0
589	0
500	0
570	0
490	0
519	0
525	0
307	0
525	0
729	1
477	0
554	0
400	1
405	0
307	0
557	0
71	1
411	0
509	0
494	0
300	0
714	1
400	0
570	0
400	0
570	0

428	0	18	1	310	1
491	0	40	0	445	0
408	0	361	1	30	1
213	0	509	0	305	1
215	0	724	0	604	0
144	0	602	0	123	0
548	0	51	1	465	0
17	0	227	0	318	0
326	0	547	0	622	0
242	0	29	0	220	1
358	0	151	0	270	0
527	0	185	0	370	0
392	0	217	0	92	0
28	0	465	0	670	0
718	0	412	0	268	0
67	0	340	0	382	1
109	0	500	0	472	0
290	1	22	1	337	1
563	0	679	1	297	1
583	0	161	0	82	0
662	0	122	0	479	0
228	0	422	0	614	0
180	0	670	0	329	0
413	1	553	0	85	0
73	0	294	0	598	0
672	0	2	0	293	1
592	0	501	0	519	0
577	0	120	0	430	0
23	1	79	0	83	0
363	1	129	0	551	0
19	0	42	0	450	0
568	0	319	1	257	0
5	0	40	0	631	0
585	0	687	0	315	0
458	1	546	0	503	0
102	0	285	0	357	0
741	0	559	0	707	0

Terminal

Terminal

