BAB II
TINJAUAN PUSTAKA

Tinjauan pustaka pada penelitian sangat penting bagi peneliti untuk memperoleh pemahaman yang lebih baik tentang topik penelitian. Tujuan dari tinjauan pustaka ini adalah untuk mengumpulkan dan menganalisis informasi terkait penelitian yang akan dilakukan, serta mempertimbangkan penelitian-penelitian sebelumnya yang telah dilakukan oleh para peneliti lain. Hal ini dilakukan untuk memahami perkembangan terkini terkait topik yang sedang diteliti., kesalahan atau kekurangan pada tinjauan pustaka sebelumnya, dan fokus penelitian yang berbeda. Melakukan tinjauan pustaka baru dapat membantu peneliti untuk memperoleh informasi terkini dan relevan terkait topik penelitian, serta memperbaiki kesalahan atau kekurangan pada tinjauan pustaka sebelumnya. Hal ini akan membantu peneliti untuk membuat penelitian yang lebih baik dan akurat.

2.1 Penelitian Terdahulu

Penelitian ini, mengacu pada penelitian sebelumnya sehingga memperoleh informasi dan wawasan yang dibutuhkan dalam rancangan model indikasi potensi banjir. Penelitian sebelumnya di tampilkan pada Tabel 2.1. Secara keseluruhan, referensi menunjukkan hasil yang baik dalam indikasi banjir. Dari hasil penelitian sebelumnya yang ditampilkan Tabel 2.1, dapat menjadi landasan peneliti dalam pemilihan metode yang digunakan, sehingga dengan perbedaan data *input* dengan pemecahan masalah yang berbeda dapat berpengaruh pada hasil model yang disusun dalam penelitian. Penting untuk dicatat bahwa tidak ada satu model pun yang sempurna, dan pendekatan terbaik akan bervariasi tergantung pada aplikasi spesifiknya.

Penelitian ini digunakan untuk mengetahui lokasi-lokasi yang rentan terjadi banjir pada tingkat kota atau desa, sehingga dapat menjadi instrumen untuk merumuskan strategi jangka panjang untuk *Smart Cities*. Sehingga diketahui sejak awal pada kondisi curah hujan dengan frekuensi tertentu dapat mengakibatkan banjir pada lokasi yang dimasukan dalam proses identifikasi. Hasil penelitian ini memperkuat pemahaman kecocokan model yang diusulkan dalam mengidentifikasi

daerah atau titik berpotensi banjir. Kemudian selanjutnya, dapat mengurangi dampak buruk banjir terhadap masyarakat dan lingkungan.

Tabel 2. 1 Penelitian terdahulu

No.	Judul	Peneliti	Metode dan Penggunaan	Hasil Penelitian	Perbedaan dengan Penelitian ini
1.	*SMOTE and Gaussian Noise Based Sensor Data Augmentation*	Arslan Mehmet, Metehan Guzel, Mehmet Mehmet, Suat Mehmet (2019)	LSTM dengan SMOTE dan *Gaussian Noise* untuk prediksi saja pada Suhu, Kelembaban, Cahaya, dan Udara	Peneliti mengusulkan dua algoritma untuk melakukan augmentasi data sensor, yaitu SMOTE *Regression* dan *Gaussian Noise* based algorithms	Penggunaan LSTM Forecasting Gaussian Noise pada curah hujan dan identifikasi untuk pengambilan keputusan
2	*Learning imbalanced datasets based on SMOTE and Gaussian distribution*	Tingting Pan, Junhong Zhao, Wei Wu, Jie Yang (2020)	Metode *SMOTE* dan *Gaussian* untuk *oversampling*, yang mengkombinasikan reduksi dimensi dan distribusi *Gaussian* untuk menghasilkan penambahan data sintesis yang lebih masuk akal.	Secara signifikan mengungguli pendekatan lain yang umum digunakan, terutama dalam hal meningkatkan nilai *F-measure* dan akurasi serta ketahanan terhadap *dataset* dan *classifier* yang berbeda.	Pengembangan LSTM dengan *Gaussian Noise* untuk *forecasting* curah hujan.
3.	*Performance Analysis of Isolation Forest Algorithm in Fraud Detection of Credit Card Transactions*	Waspada I, Bahtiar Nurdin, Wirawan P W, Awan Bagus D A (2020)	Metode *Isolation Forest* digunakan untuk Deteksi *Credit Card Transaction*	Hasil model *Isolation Forest* cukup baik dalam melakukan deteksi dalam transaksi. Mengusulkan pemilihan fitur, dan pengaturan hiperparameter dalam tahap pemodelan.	Pengembangan model dengan LSTM dan *Isolation forest*, untuk obyek masalah yang berbeda.
4.	*Urban Flood Forecasting Using A Hybrid Modeling Approach Based On A Deep Learning Technique*	Moon, Hyeontae, Yoon, Sunkwon, Moon, Youngil (2023)	Model LSTM dikembangkan untuk memprediksi waktu terjadinya banjir berdasarkan kejadian hujan deras	Model LSTM yang dikembangkan untuk memprediksi waktu terjadinya banjir berdasarkan variabel hujan menunjukkan tingkat akurasi yang tinggi.	Dapat menghasilkan prediksi dan *forecasting* curah hujan tinggi dalam bulanan yang berpotensi banjir.
5.	*High temporal resolution urban flood prediction using*	Zhang L, Qin H, Mao J, Cao *X*, Fu G (2023)	Metode *Attention-based* LSTM, digunakan untuk Sistem peringatan dini untuk banjir	Hasil menunjukan prediksi kedalaman air dari faktor curah hujan dalam menit pada	Dapat menghasilkan prediksi dan *forecasting* curah hujan.

No.	Judul	Peneliti	Metode dan Penggunaan	Hasil Penelitian	Perbedaan dengan Penelitian ini
	attention-based LSTM *models*			3 lokasi berbeda yang sudah ditentukan dengan hasil yang berbeda.	
6.	Perbandingan LSTM dan GRU Untuk Memprediksi Curah Hujan	Carnegie & Chairani, (2023)	Metode LSTM digunakan untuk prediksi curah hujan.	Hasil akurasi prediksi LSTM dibandingkan dengan GRU untuk mengetahui metode yang paling optimal.	Memberikan referensi pengembangan LSTM dengan *Gaussian Noise*.
7	*Probabilistic evaluation of cultural soil heritage hazards in China from extremely imbalanced site investigation data using SMOTE-Gaussian process classification*	Chao Song, Hongzhen Peng, Ling Xu, Tengyuan Zhao, Zhiqian Guo, Wenwu Chen (2024)	SMOTE dan *Gaussian Process Classification*	Metode SMOTE-GPC terbukti efektif untuk memprediksi tingkat bahaya pada situs CSH dengan data yang tidak seimbang.	Digunakan untuk memprediksi curah hujan tinggi dengan data yang tidak seimbang.

2.2 Pengertian Banjir

Banjir didefinisikan sebagai luapan air sementara ke daratan. Banjir dapat disebabkan oleh berbagai faktor, termasuk hujan lebat, air yang mencair setelah salju, dan gelombang badai. Banjir dapat menyebabkan kerusakan yang luas pada properti dan infrastruktur, dan bahkan dapat menyebabkan hilangnya nyawa (Pradhan dkk., 2023). Perubahan iklim diperkirakan meningkatkan frekuensi dan besarnya bencana lingkungan, termasuk kondisi cuaca ekstrem. Pada saat yang sama, pusat-pusat perkotaan diproyeksikan akan terus berkembang dalam beberapa tahun ke depan, sehingga diperlukan mekanisme yang efektif untuk mengatasi ancaman bencana, khususnya banjir (Motta dkk., 2021).

2.2.1 Jenis Banjir

Pusat Kritis Kesehatan Kementerian Kesehatan RI pada tahun 2018 melaporkan, terdapat lima jenis banjir yang dapat dibedakan sebagai berikut :

a) Banjir Bandang

Jenis banjir yang sangat membahayakan karena dapat membawa berbagai objek dan menimbulkan kerusakan parah. Banjir bandang umumnya terjadi di daerah pegunungan dan dapat dipicu oleh hilangnya hutan di kawasan tersebut.

b) Banjir Air

Jenis banjir yang umum terjadi, biasanya disebabkan oleh meluapnya air sungai, danau, atau selokan. Banjir air terjadi ketika intensitas hujan sangat tinggi sehingga air tidak tertampung dan meluap.

c) Banjir Lumpur

Jenis banjir yang mirip dengan banjir bandang namun terjadi ketika banjir keluar dari dalam bumi dan mencapai daratan. Banjir lumpur mengandung bahan berbahaya dan gas yang dapat mempengaruhi kesehatan makhluk hidup.

d) Banjir Rob (Banjir Laut Air Pasang)

Jenis banjir yang terjadi akibat naiknya permukaan laut. Banjir rob biasanya terjadi di wilayah sekitar pesisir pantai.

e) Banjir Genangan

Jenis banjir yang serupa dengan banjir air, namun terjadi akibat hujan deras yang tidak tertampung.

Banjir yang disebabkan oleh hujan yang sangat deras menjadikan alasan penelitian ini dilakukan, mengacu pada teori jenis banjir, sehingga hasil prediksi yang dilakukan dapat sebagai acuan dalam pengambilan keputusan terkait kondisi curah hujan.

2.2.2 Faktor-Faktor Penyebab Banjir

Aldiansyah & Wardani (2023) menyebutkan bahwa, terdapat beberapa penyebab banjir di suatu wilayah. Curah hujan tinggi tidak dapat dihindari, namun hasil prediksi dapat digunakan sebagai langkah antisipasi pada daerah rawan banjir dan modifikasi cuaca (Kusuma Yoga dkk., 2022). Berikut adalah faktor- faktor penyebab banjir :

a) Curah hujan yang tinggi

Curah hujan yang tinggi dapat menyebabkan genangan air dan banjir di suatu wilayah. Hal ini disebabkan oleh lahan permukaan yang terbatas dan kemampuan sistem *drainase* yang kurang optimal.

b) Pencemaran sungai dan *drainase*

Pencemaran sungai dan *drainase* dapat menyebabkan penyumbatan dan mengurangi kapasitas sistem *drainase*. Hal ini dapat menyebabkan peningkatan risiko banjir pada musim hujan.

c) Perubahan penggunaan lahan

Perubahan penggunaan lahan dari lahan terbuka menjadi lahan permukiman atau industri dapat menyebabkan peningkatan aliran air permukaan dan mengurangi kapasitas resapan air. Hal ini dapat menyebabkan peningkatan risiko banjir.

d) Pengurangan kawasan resapan air

Pengurangan kawasan resapan air seperti hutan atau lahan basah dapat mengurangi kemampuan alam untuk menyerap air, sehingga meningkatkan risiko banjir.

e) Pembangunan infrastruktur yang tidak memperhatikan aspek *drainase*

Pembangunan infrastruktur seperti jalan raya, gedung-gedung, dan pemukiman yang tidak memperhatikan aspek *drainase* dapat menyebabkan penyempitan sungai dan *drainase*, sehingga meningkatkan risiko banjir.

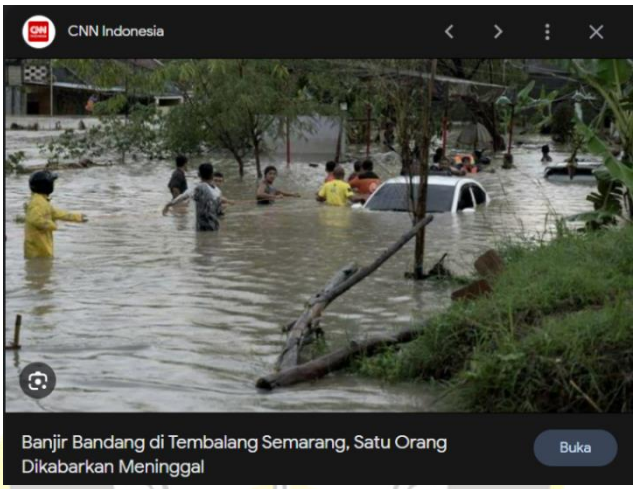
f) Kondisi topografi

Kondisi topografi seperti lereng yang curam atau dataran rendah dapat meningkatkan risiko banjir karena aliran air permukaan menjadi lebih cepat dan sulit diresap oleh tanah.

2.2.3 Daerah Rawan Banjir

Daerah rawan banjir merupakan kawasan yang rentan atau seringkali terkena peristiwa banjir (Maryati, 2018). Daerah rawan banjir dapat diklasifikasikan menjadi empat daerah, yaitu daerah pantai, daerah dataran banjir, daerah sempadan sungai, dan daerah cekungan. Klasifikasi ini dapat membantu dalam mengidentifikasi daerah yang berpotensi terkena dampak banjir dan membantu dalam perencanaan mitigasi bencana di daerah tersebut (A. Septian dkk., 2020). Daerah rawan banjir adalah suatu wilayah yang memiliki potensi untuk terkena banjir seperti ditunjukkan pada Gambar 2.1, akibat dari berbagai faktor seperti curah hujan yang tinggi, kondisi topografi yang rendah, dan sistem *drainase* yang buruk (Lee & Kim, 2021). Penentuan daerah rawan banjir adalah proses untuk menentukan wilayah atau area yang memiliki potensi terkena banjir. Proses ini

melibatkan analisis beberapa faktor yang mempengaruhi risiko banjir, seperti curah hujan, topografi, *drainase*, dan penggunaan lahan. Hasil dari penentuan daerah rawan banjir dapat digunakan sebagai dasar untuk mengambil kebijakan mitigasi dan adaptasi terhadap risiko banjir, seperti pembangunan infrastruktur yang lebih baik, pengaturan penggunaan lahan, dan peningkatan sistem peringatan dini (Setyani & Saputra, 2016).



Gambar 2. 1 Kejadian Banjir (CNN Indonesia, 2022)

2.3.4 Curah Hujan Area

Barnett (2015) menyebutkan dalam bukunya, curah hujan adalah bagian penting dari siklus air. Presipitasi adalah proses turunnya air dari atmosfer ke permukaan bumi dalam bentuk hujan. Curah hujan sangat penting bagi kehidupan di Bumi, karena menjadi sumber air untuk konsumsi, pertanian, dan berbagai kebutuhan lainnya. Kemudian, curah hujan area adalah curah hujan yang diperlukan untuk penyusunan suatu rancangan pemanfaatan air dan rancangan pengendalian banjir dengan curah hujan rata-rata di seluruh daerah yang bersangkutan, bukan curah hujan pada suatu titik tertentu. Curah hujan ini disebut curah hujan wilayah/daerah/area dan dinyatakan dalam mm (Mori, 2003).

Perhitungan curah hujan area ini harus diperkirakan dari beberapa titik pengamatan curah hujan, tujuan dari perhitungan curah hujan area ini adalah untuk menghitung curah hujan maksimum rata-rata harian dari data yang ada (Mori, 2003). Tabel 2.2 memperlihatkan bahwa intensitas curah hujan tersebut menunjukkan kategori keadaan curah hujan berdasarkan intensitas curah hujan

dalam satuan milimeter (mm) dalam rentang waktu 1 jam dan 24 jam. Kategori curah hujan yang terdapat dalam tabel tersebut meliputi hujan sangat ringan, hujan ringan, hujan normal, hujan lebat, dan hujan sangat lebat. Intensitas curah hujan dalam kategori tersebut berbeda-beda, dengan rentang intensitas yang berbeda juga untuk waktu 1 jam dan 24 jam. Tabel tersebut dapat digunakan sebagai acuan untuk memahami intensitas curah hujan dalam suatu wilayah dan membantu dalam melakukan prediksi banjir. Dalam pernyataan resmi BMKG menyebutkan bahwa Hujan lebat hingga sangat lebat disertai kilat/petir yang berlangsung terus menerus pada Sabtu, 6 Februari 2021 pukul 02.00 hingga 05.30 WIB dapat menyebabkan genangan di beberapa titik di wilayah Kota Semarang. Sehingga kondisi hujan lebat dan sangat lebat dapat digunakan sebagai faktor penyebab banjir dalam kategori curah hujan tinggi (Sukasno, 2021).

Tabel 2. 2 Keadaan curah hujan dan intensitas curah hujan

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

Sumber : Buku Hidrologi (Soewarno, 1995)

Perhitungan curah hujan area dapat dilakukan dengan menggunakan Metode Rata-Rata Aljabar, di mana nilai curah hujan area diperoleh dari rata-rata aritmatika dari data curah hujan yang tercatat di stasiun-stasiun pengukuran di dalam dan di sekitar wilayah yang dianalisis ditunjukkan pada persamaan 2.1.

$$\bar{R} = \frac{1}{n}(R_1 + R_2 + \ldots + R_n) \qquad (2.1)$$

dengan $\bar{R}$ menyatakan curah hujan area (mm), n mewakili jumlah titik-titik (pos-pos) pengamatan, $R_1, R_2 \ldots, R_n$ adalah curah hujan di tiap titik pengamatan (mm).

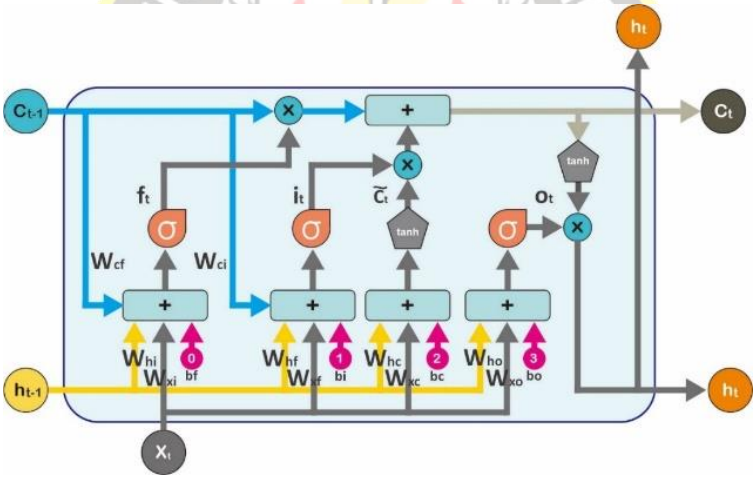
Metode rata-rata aljabar dapat digunakan dengan beberapa persyaratan antara lain

a) Sesuai untuk kawasan-kawasan yang datar (rata).

b) Sesuai untuk DAS-DAS (Daerah Aliran Sungai) dengan jumlah penakar hujan yang besar yang didistribusikan secara merata pada lokasi-lokasi yang mewakili.

2.3 *Long Short Therm Memory* (LSTM)

LSTM diperkenalkan sebagai varian khusus dari RNNs yang memiliki kemampuan untuk menyimpan informasi selama periode waktu yang lebih panjang. Hal ini membuatnya lebih unggul dalam menangani tugas-tugas yang membutuhkan memori jangka panjang (Goodfellow dkk., 2016). Pendekatan *lookback* dalam penelitian ini mencakup penggunaan informasi dari langkah waktu sebelumnya dalam perhitungan saat ini. Dengan menggunakan mekanisme *lookback*, jaringan dapat mengatasi tantangan yang melibatkan data berurutan dengan mempertahankan dan memanfaatkan informasi kontekstual dari langkah waktu sebelumnya (Hochreiter dan Schmidhuber 1997). Persamaan 2.2 - 2.6 secara umum memberikan gambaran langkah-langkah penting dalam model LSTM.



Gambar 2. 2 Ilustrasi Algoritma LSTM (Shiri M Farhad dkk., 2023)

$$f_t = \sigma(x_t W_{xf} + h_{t-1} W_{hf} + c_{t-1} W_{cf} + b_f) \qquad (2.2)$$

dengan :

σ = fungsi *sigmoid*.

x_t = *input* saat ini pada sel LSTM.

W_{xf} = matriks bobot masukan untuk gerbang lupakan.

h_{t-1} = keadaan tersembunyi sebelumnya pada sel LSTM.

W_{hf} = matriks bobot tersembunyi untuk gerbang lupakan.

c_{t-1} = sel memori sebelumnya pada sel LSTM.

W_{cf} = matriks bobot sel memori untuk gerbang lupakan.

b_f = nilai bias untuk gerbang lupakan.

Gerbang lupakan (f_t) seperti ditunjukkan pada persamaan 2.2 dalam sel LSTM berperan dalam menyeleksi informasi yang akan dipertahankan atau dihapus dari sel memori. Fungsi sigmoid (σ) mengonversi *input* ke rentang [0,1]. Komponen utama meliputi *input* saat ini (x_t), keadaan tersembunyi sebelumnya (h_{t-1}), dan sel memori sebelumnya (c_{t-1}). Matriks bobot W_{xf}, W_{hf}, dan W_{cf}, serta bias b_f, diterapkan pada komponen-komponen tersebut. Hubungan matriks tersebut menentukan proporsi informasi yang dipertahankan, memungkinkan LSTM untuk menyimpan informasi relevan jangka panjang dan menghapus yang tidak penting, meningkatkan efisiensi dalam pemrosesan data sekuensial.

$$i_t = \sigma(x_t W_{xi} + h_{t-1} W_{hi} + c_{t-1} W_{ci} + b_i) \qquad (2.3)$$

dengan :

σ = fungsi *sigmoid*.

x_t = *input* saat ini pada sel LSTM.

W_{xi} = Matriks bobot masukan.

h_{t-1} = keadaan tersembunyi sebelumnya pada sel LSTM.

W_{hi} = Matriks bobot tersembunyi.

c_{t-1} = sel memori sebelumnya pada sel LSTM.

W_{ci} = Matriks bobot sel memori.

b_i = Nilai bias.

Gerbang masukan (*input gate*) dan gerbang lupakan (*forget gate*) dalam sel LSTM memiliki peran yang saling melengkapi. Gerbang masukan (i_t) seperti pada persamaan 2.3 mengatur seberapa banyak informasi baru yang akan disimpan ke sel memori, sementara gerbang lupakan menentukan informasi mana yang akan dipertahankan atau dihapus. Keduanya menggunakan *input* saat ini, keadaan tersembunyi sebelumnya, sel memori sebelumnya, serta matriks bobot dan bias yang sesuai untuk menghasilkan nilai dalam rentang [0, 1] yang mengontrol pembaruan sel memori. Kombinasi kerja gerbang masukan dan lupakan

memungkinkan LSTM menyimpan informasi relevan jangka panjang dan menghapus yang tidak penting, meningkatkan efisiensi dalam pemrosesan data sekuensial.

$$o_t = \sigma(x_t W_{xo} + h_{t-1} W_{ho} + c_t W_{co} + b_o) \qquad (2.4)$$

dengan :

σ = fungsi *sigmoid*.

x_t = *input* saat ini pada sel LSTM.

W_{xo} = matriks bobot masukan untuk gerbang keluaran.

h_{t-1} = keadaan tersembunyi sebelumnya pada sel LSTM.

W_{ho} = matriks bobot tersembunyi untuk gerbang keluaran.

c_t = sel memori (*cell state*).

W_{co} = matriks bobot sel memori untuk gerbang keluaran.

b_o = nilai bias untuk gerbang keluaran.

Persamaan 2.4 menunjukkan perbedaan mendasar antara gerbang masukan (*input gate*) dan lupakan (*forget gate*) dengan gerbang keluaran (*output gate*) yang di representasikan sebagai (o_t) dalam sel LSTM terletak pada fungsi dan komponen yang digunakan. Gerbang masukan dan lupakan bertanggung jawab atas pembaruan sel memori, menggunakan *input* saat ini, keadaan tersembunyi sebelumnya, dan sel memori sebelumnya, serta matriks bobot dan bias terkait. Di sisi lain, gerbang keluaran menentukan informasi yang akan dikirim ke keadaan tersembunyi, memanfaatkan *input* saat ini, keadaan tersembunyi sebelumnya, dan sel memori saat ini, dengan matriks bobot dan bias yang berbeda. Gerbang masukan dan lupakan berfokus pada pembaruan memori, sedangkan gerbang keluaran fokus pada menentukan output dari sel LSTM.

$$c_t = f_t c_{t-1} + i_t \, tanh(\, x_t W_{xc} + h_{t-1} W_{hc} + b_c \,) \qquad (2.5)$$

dengan :

f_t = gerbang lupakan / *forget gate*

c_{t-1} = sel memori sebelumnya pada sel LSTM.

i_t = Gerbang masukan (*input gate*)

$tanh$ = fungsi tangen hiperbolik.

x_t = *input* saat ini pada sel LSTM.

W_{xc} = matriks bobot sel memori untuk gerbang masukan.

h_{t-1} = keadaan tersembunyi sebelumnya pada sel LSTM.

W_{hc} = matriks bobot sel memori untuk gerbang tersembunyi.

b_c = nilai bias sel memori.

Sel memori (c_t) adalah komponen kunci sel LSTM, menyimpan informasi jangka panjang. Pembaruannya diatur gerbang masukan (i_t) dan lupakan (f_t). Gerbang lupakan menentukan informasi (c_{t-1}) yang dipertahankan/dilupakan (nilai f_t 0-1). Gerbang masukan menentukan informasi baru yang disimpan (nilai i_t 0-1). (c_t) dihitung dari informasi dipertahankan (f_t * c_{t-1}) dan informasi baru. Dengan koordinasi gerbang, (c_t) menyimpan dan memperbarui informasi dalam sel LSTM.

$$h_t = o_t \, tanh\,(c_t) \qquad (2.6)$$

dengan :

o_t = Gerbang keluaran (*output gate*).

$tanh$ = fungsi tangen hiperbolik.

c_t = sel memori (*cell state*).

Persamaan 2.6 merepresentasikan h_t sebagai keadaan tersembunyi saat ini dari sel LSTM, yang akan digunakan sebagai keluaran dan juga *input* untuk langkah selanjutnya. Dengan menggunakan persamaan 2.2 - 2.6 model LSTM dapat mengatur aliran informasi dengan cermat dan mengingat informasi yang penting dalam rangkaian waktu yang panjang dan melakukan prediksi waktu yang akan datang.

2.4 *Gaussian Noise* Distribusi Normal

Carl Friedrich Gauss pertama kali mendeskripsikan distribusi normal (atau *Gaussian*) dengan dua parameter, yaitu *mean* (μ) dan standar deviasi (σ), yang menentukan bentuk dan lokasi kurva dalam distribusi data pada tahun 1809 dalam konteks kesalahan pengukuran di astronomi. Selama abad ke-19, distribusi ini diterapkan secara luas dalam area probabilitas dan statistik yang sedang berkembang (Brereton, 2014). Distribusi total curah hujan cenderung mendekati distribusi normal pada skala yang lebih besar, seperti bulanan atau tahunan, karena

tiga alasan utama. Pertama, berdasarkan Teorema Limit Pusat, penjumlahan banyak variabel acak akan menghasilkan distribusi normal. Kedua, total curah hujan pada skala besar merupakan akumulasi banyak kejadian harian, sehingga efek "pemusatan" ini berlaku. Ketiga, fluktuasi curah hujan harian di iklim lembab cenderung lebih acak dan terdistribusi secara merata, juga mendekati pola normal. Kombinasi ketiga faktor ini menyebabkan distribusi total curah hujan pada skala besar mendekati kurva *Gaussian* (Wilks, 1995).

Distribusi normal direpresentasikan dengan sumbu horizontal mewakili suatu pengukuran, seperti tinggi pria, dan sumbu vertikal mewakili probabilitas atau frekuensi. Distribusi *Gaussian* (atau normal) dicirikan oleh tiga parameter: *mean* (pusat distribusi), luas (1 untuk *Probability Density Function* (PDF) atau mewakili total probabilitas dari semua nilai pengukuran), dan lebar atau variabilitas (sering diukur pada setengah tinggi atau diekspresikan sebagai deviasi standar). Selanjutnya dasar distribusi *Gaussian* dituliskan dalam persamaan 2.7 s.d 2.9.

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}}\, e^{-\frac{1}{2}\frac{(x-\mu)^2}{\sigma}} \qquad (2.7)$$

$$noise = \mu_n + \sigma_n * random\ number \qquad (2.8)$$

$$noisy\ data = original\ data + noise \qquad (2.9)$$

dengan :

$\mu =$ nilai rata-rata dari distribusi *Gaussian* secara keseluruhan

$\sigma =$ nilai rata-rata dari variansi dari distribusi *Gaussian* secara keseluruhan

$\mu_n =$ nilai rata-rata dari distribusi *Gaussian* yang akan ditambahkan ke data asli

$\sigma_n =$ nilai standar deviasi dari *Gaussian* yang akan ditambahkan ke data asli

Fungsi densitas probabilitas normal atau *Gaussian*, dapat digunakan untuk memodelkan variabel acak kontinu yang mengikuti distribusi normal. Dalam persamaan ini, $f(x)$ adalah fungsi densitas probabilitas, μ adalah nilai rata-rata (*mean*), σ adalah standar deviasi, dan *e* adalah eksponensial. Penting diketahui bahwa persamaan 2.7 merupakan tahapan awal untuk melihat rentang batas atas dan batas bawah yang cocok karena tujuannya adalah menambahkan *noise* pada data dengan tidak menghilangkan karakteristik data. Persamaan 2.8 menunjukkan cara untuk menghitung *noise* atau gangguan yang ditambahkan pada data asli. Dalam

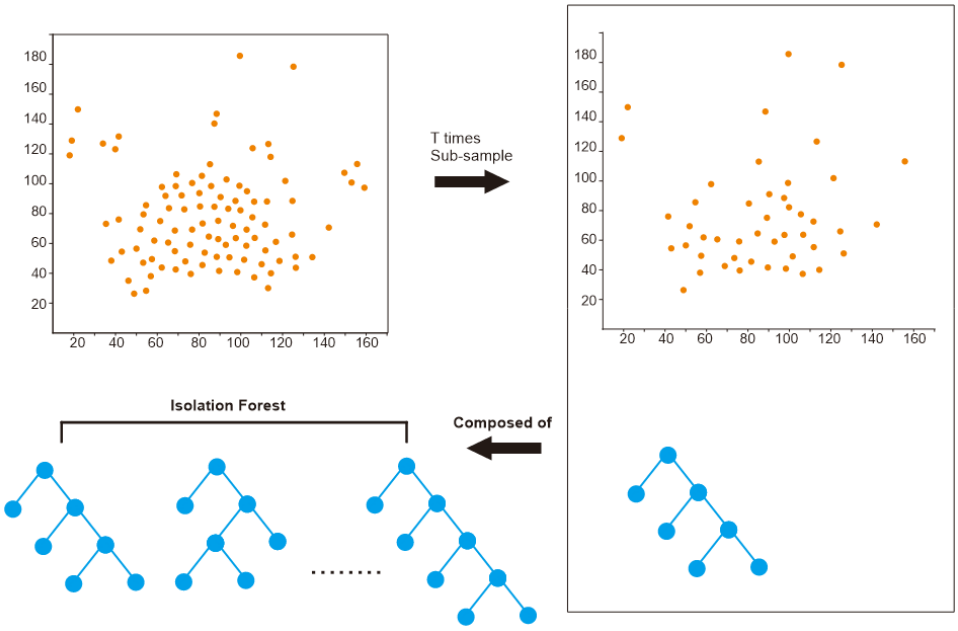
persamaan 2.8, *noise* adalah nilai *noise* yang ditambahkan pada data asli, μ_n adalah nilai rata-rata *noise*, dan σ_n adalah standar deviasi *noise*. *Random number* adalah bilangan acak yang digunakan untuk menghasilkan *noise*. Persamaan 2.9 menunjukkan cara untuk menghitung data yang telah terkontaminasi oleh *noise*. Dalam persamaan ini, *noisy data* adalah data yang telah terkontaminasi oleh *noise*, dan aktual data adalah data asli sebelum ditambahkan *noise*. Dengan menggunakan persamaan 2.7-2.9, dapat sebagai dasar data yang dihasilkan terkontaminasi oleh *Gaussian noise*. Dengan persamaan 2.7 - 2.9, *Gaussian noise* ditambahkan pada proses augmentasi data. Data asli diubah dengan menggunakan generator angka acak yang mengikuti distribusi normal standar dengan rata-rata 0 dan variasi yang berbeda. Melalui transformasi *linear*, *mean* dan standar deviasi dari angka-angka tersebut disesuaikan. Hasil transformasi tersebut kemudian ditambahkan ke data asli untuk menghasilkan data yang diberi *Gaussian noise* (Wang & Qi, 2023).

2.5 *Isolation Forest* (IF)

Isolation Forest adalah sebuah algoritma *machine learning* yang digunakan untuk deteksi *outlier*. Algoritma ini dikembangkan oleh Fei Tony Liu, Kai Ming Ting, dan Zhi-Hua Zhou pada tahun 2008. *Isolation Forest* merupakan algoritma untuk pengelompokan data dengan mengisolasi *outlier* yang jarang muncul ke dalam klaster yang berbeda dari data normal. Dalam hal ini, *outlier* dianggap sebagai data yang jarang dan berbeda dari mayoritas data normal (Stanton dkk., 2012). Dengan melihat distribusi berdasarkan nilai atau *scoring*, dapat memberikan dampak positif secara keseluruhan. Sedikit sensitivitas terhadap *outlier* di antara estimator dapat membantu dalam mendeteksi anomali, sementara terlalu banyak sensitivitas tidak memberikan manfaat yang signifikan (Dhouib dkk., 2023).

Isolation Forest (IF), dibangun berdasarkan *tree* keputusan seperti ditunjukan pada Gambar 2.3. Algoritma tidak memerlukan label yang telah ditentukan sebelumnya, sehingga IF merupakan model yang tidak diawasi (*unsupervised*). *Isolation Forest* dibangun berdasarkan fakta bahwa anomali adalah titik-titik data yang "sedikit dan berbeda" (Iivari, 2022). Dalam *Isolation Forest*, data yang diambil secara acak dari sampel diproses dalam struktur *tree* berdasarkan fitur-fitur yang dipilih secara acak. Sampel-sampel yang bergerak lebih dalam ke dalam *tree* kemungkinan lebih kecil menjadi anomali karena membutuhkan lebih banyak

pemisahan untuk mengisolasi mereka. Demikian pula, sampel-sampel yang berakhir di cabang-cabang yang lebih pendek menunjukkan adanya *anomaly* karena lebih mudah bagi *tree* untuk memisahkan mereka dari observasi lainnya (Liu dkk., 2008).



Gambar 2. 3 Pemisahan kumpulan data untuk mengisolasi *anomaly* dan data normal (Chen dkk., 2023)

Algoritma *Isolation Forest* digunakan untuk menghitung faktor skala yang digunakan dalam normalisasi tinggi pohon isolasi seperti dinyatakan dalam persamaan 2.10.

$$c(n) = 2H(n-1) - \frac{2(n-1)}{n} \qquad (2.10)$$

dengan :

n = Jumlah langkah yang diperlukan hingga titik data terisolasi sepenuhnya.

$H(n\text{-}1)$ = fungsi harmonik dengan argument $(n\text{-}1)$, untuk perhitungan konstanta normalisasi dengan menggunakan rumus $\ln(n\text{-}1) + 0.5772156649$.

$2(n\text{-}1)/n$ = faktor koreksi yang digunakan untuk menghitung kombinasi tanpa urutan.

Fungsi *H(n-1)* adalah fungsi harmonik dengan argumen *(n-1)*, yang digunakan untuk menghitung konstanta normalisasi. Faktor koreksi *2(n-1)/n* digunakan untuk

memperhitungkan kombinasi tanpa urutan dalam perhitungan. Tingkat skor *anomaly* $s\ (x, n)$ pada titik data *x* dalam data set dengan jumlah data *n* digunakan sebagai pohon keputusan dinyatakan dalam persamaan 2.11.

$$s\ (x, n) = 2^{-\frac{E\ (h(x))}{c(n)}} \qquad (2.11)$$

dengan :

$h(x)$ = Tinggi pohon (jumlah langkah) yang diperlukan untuk mengisolasi titik data *x* dalam pohon isolasi.

$c(n)$ = Faktor skala yang digunakan untuk normalisasi tinggi pohon $h(x)$ berdasarkan jumlah data *n*.

$E(h(x))$ = Panjang jalur yang diharapkan (*expected path length*) titik data *x*, yaitu rata-rata tinggi pohon $h(x)$ dari pohon-pohon isolasi dalam hutan.

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

2.6 Validasi Model

Hasil dari model yang telah di implementasikan, dilakukan validasi untuk mengetahui tingkat akurasi model dalam penerapan model prediksi pada penelitian ini. Berikut merupakan metode yang digunakan untuk mengukur validasi model diantaranya :

2.6.1 *Mean Squared Error* (MSE)

MSE digunakan untuk mengukur ketepatan model LSTM dalam menghasilkan prediksi data sekuensial. MSE adalah metrik yang digunakan untuk mengevaluasi performa model prediksi. Metrik ini menghitung rata-rata dari selisih kuadrat antara nilai prediksi dan nilai sebenarnya. Semakin rendah MSE, semakin baik performa model tersebut. MSE memperhitungkan baik bias (perbedaan antara

prediksi rata-rata dengan nilai sebenarnya) maupun varians (variabilitas prediksi). Dengan demikian, MSE memberikan ukuran yang komprehensif terhadap akurasi dan presisi model prediksi sesuai dalam persamaan 2.12.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \tilde{Y}_i)^2 \qquad (2.12)$$

dengan :

$\tilde{Y}_i$ = Nilai hasil prediksi

Y_i = Nilai aktual / Nilai sebenarnya

n = Jumlah data

Jumlah kuadrat selisih antara nilai aktual (Y_i) dan nilai prediksi ($\tilde{Y}_i$), dibagi dengan jumlah data (n). Semakin kecil nilai MSE, semakin akurat model dalam memprediksi nilai sebenarnya. MSE sering digunakan untuk mengevaluasi model regresi, di mana tujuannya adalah meminimalkan perbedaan antara nilai prediksi dan nilai aktual.

2.6.2 *Root Mean Squared Error* (RMSE)

RMSE dihitung dari akar kuadrat kesalahan kuadrat rata-rata. Semakin rendah nilai RMSE, menunjukkan bahwa metode estimasi kesalahan pengukuran semakin akurat. RMSE sering digunakan untuk menentukan apakah metode prediksi dapat diandalkan atau tidak. Untuk menghitung RMSE, nilai aktual (Y_i) dan nilai prediksi ($\tilde{Y}_i$) dibandingkan, ditunjukkan pada persamaan 2.13.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \tilde{Y}_i)^2} \qquad (2.13)$$

dengan :

$\tilde{Y}_i$ = Nilai hasil prediksi

Y_i = Nilai aktual / Nilai sebenarnya

n = Jumlah data

RMSE berguna karena sensitif terhadap kesalahan besar dan memberikan ukuran keakuratan prediksi model secara keseluruhan. Hal ini membuat RMSE berguna untuk membandingkan kinerja model yang berbeda atau model yang sama pada *dataset* yang berbeda.