

**PENERAPAN *DATA MINING* UNTUK PREDIKSI JUMLAH PRODUKSI MINYAK
SAWIT DENGAN METODE *LONG SHORT-TERM MEMORY***

SKRIPSI

SYABRINA RAMADHANI KAMAL

201401117



**PROGRAM STUDI S-1 ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN**

2024

**PENERAPAN *DATA MINING* UNTUK PREDIKSI JUMLAH PRODUKSI MINYAK
SAWIT DENGAN METODE *LONG SHORT-TERM MEMORY***

SKRIPSI

SYABRINA RAMADHANI KAMAL

201401117



**PROGRAM STUDI S-1 ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN
2024**

PERSETUJUAN

Judul : PENERAPAN *DATA MINING* UNTUK
PREDIKSI JUMLAH PRODUKSI MINYAK
SAWIT DENGAN METODE *LONG SHORT-
TERM MEMORY*

Kategori : SKRIPSI
Nama : SYABRINA RAMADHANI KAMAL
Nomor Induk Mahasiswa : 201401117
Program Studi : SARJANA (S-1) ILMU KOMPUTER
Fakultas : ILMU KOMPUTER DAN TEKNOLOGI
INFORMASI UNIVERSITAS
SUMATERA UTARA

Diluluskan di
Medan, 11 Juni 2024

Komisi Pembimbing :
Pembimbing 2

Pembimbing 1



Dewi Sartika Br.Ginting, S.Kom., M.Kom
NIP. 199005042019032023

Handrizal S.Si., M.Comp.Sc
NIP. 197706132017061001

Diketahui/Disetujui Oleh
Program Studi S-1 Ilmu
Komputer
Ketua,

Dr. Amalia ST., M.T.
NIP. 197812212014042001

PERNYATAAN**PENERAPAN *DATA MINING* UNTUK PREDIKSI JUMLAH PRODUKSI
MINYAK SAWIT DENGAN METODE *LONG SHORT-TERM MEMORY*****SKRIPSI**

Saya mengakui bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing telah disebutkan sumbernya.

Medan, 11 Juni 2024



Syabrina Ramadhani Kamal

201401117

UCAPAN TERIMA KASIH

Dengan memanjatkan puji dan syukur kepada Allah SWT atas kemurahan hati, bimbingan, berkat, dan kekuatan yang diberikannya untuk menulis skripsi ini. Terimakasih atas rahmat-Nya, yang juga senantiasa membantu, membimbing, dan mendorong saya untuk menyelesaikan skripsi yang diperlukan untuk mendapatkan gelar sarjana komputer di Program S1 Ilmu Komputer dan Teknologi Informasi di Universitas Sumatera Utara. Selain itu, marilah kita panjatkan shalawat dan salam kepada Rasulullah SAW yang telah membimbing umat manusia dari masa kegelapan menuju masa yang penuh dengan kebenaran. Penulis dengan hormat mengucapkan terima kasih kepada Mama tersayang, Yeni Susanti atas segala bentuk perjuangan, kasih sayang, dan perlindungan dengan doa-doa yang dipanjatkan untuk penulis. Dan terima kasih kepada Papa Ir. Kamaluddin, atas dukungan dan kasih sayang yang membersamai di setiap langkah penulis. Terima kasih untuk setiap dukungan yang telah diberikan hingga penulis dapat berada di titik ini.

Penulis menyampaikan rasa terima kasih yang mendalam kepada semua pihak yang telah memberikan bantuan, dukungan, dan kontribusi dalam penyusunan skripsi ini. Terimakasih penulis ucapkan kepada :

1. Bapak Dr. Muryanto Amin, S.Sos., M.Si selaku Rektor Universitas Sumatera
2. Ibu Dr. Maya Silvi Lydia B.Sc., M.Sc. selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara.
3. Ibu Dr. Amalia, S.T., M.T selaku Ketua Program Studi S1 Ilmu Komputer Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara dalam memotivasi dan memberikan arahan semasa perkuliahan.
4. Bapak Handrizal S.Si., M.Comp.Sc., sebagai Dosen Pembimbing I, telah memberikan banyak masukan, mendorong, dan mendukung penulis selama penyusunan skripsi ini.
5. Ibu Dewi Sartika Br. Ginting, S.Kom., M.Kom., sebagai dosen pembimbing II, telah memberikan banyak pengetahuan dan masukan kepada skripsi

penulis.

6. Ibu Hayatunnufus S.Kom., M.Kom selaku Dosen Pembimbing Akademik yang telah memberikan saran, motivasi dan banyak dukungan kepada penulis selama masa studi.
7. Seluruh Bapak dan Ibu Dosen Program Studi S-1 Ilmu Komputer, yang telah membimbing saya selama masa perkuliahan hingga akhir
8. Kedua orangtua penulis, Ir.Kamaluddin dan Yeni Susanti, yang selalu menyayangi, mendoakan, dan mendukung penulis di setiap aspek kehidupannya.
9. Kakak dan Abang kandung penulis Putri Aisyah Kamal dan M. Iqbal Tanjung yang terus mendoakan penulis selama proses skripsi ini.
10. M.Afif Haifan yang terus memberi semangat dan membantu penulis selama proses skripsi ini.
11. Rolin, Pia, Apis, Aslam, Wessy, Danish, Ribeng yang senantiasa selalu memberi semangat kepada penulis.
12. Teman konseran Ruri dan Hawa yang senantiasa juga selalu memberi motivasi kepada penulis.
13. Teman seperjuangan Yupra, Dinda, Rani yang selalu bersama pada saat perkuliahan dan selalu memberi semangat satu sama lain agar menyelesaikan skripsi ini tepat waktu.

Akhir kata, Terima kasih kepada Syabrina Ramadhani Kamal yang sudah berjuang dan tidak menyerah sampai sejauh ini sehingga skripsi ini dapat diselesaikan. Semoga pembaca dan masyarakat luas mendapat manfaat dari skripsi ini. Penulis berharap bahwa hasil dari penelitian ini mampu memberikan manfaat yang besar terutama kepada perusahaan dalam konteks prediksi produksi minyak sawit. Terima kasih

Medan, 11 Juni 2024

Penulis,

Syabrina Ramadhani Kamal

ABSTRAK

Produksi minyak sawit merupakan salah satu indikator penting dalam sektor agribisnis di Indonesia. Prediksi yang akurat mengenai jumlah produksi minyak sawit dapat membantu dalam proses perencanaan dan pengambilan keputusan yang lebih baik. Permasalahan yang dihadapi adalah fluktuasi jumlah tandan buah sawit yang masuk ke pabrik, hal ini menyebabkan munculnya ketidakpastian produksi minyak mentah atau CPO. Hasil produksi CPO yang berubah-ubah memberikan dampak bagi perusahaan. Penelitian ini bertujuan untuk membangun model prediksi produksi minyak sawit menggunakan metode Long Short-Term Memory (LSTM). Data yang digunakan mencakup produksi minyak sawit dari tahun 2019 hingga 2023, dengan data hingga tahun 2022 digunakan sebagai set pelatihan dan data dari tahun 2023 ke atas sebagai set pengujian. Data dinormalisasi menggunakan MinMaxScaler untuk memastikan data berada dalam rentang yang sesuai untuk pelatihan model LSTM. Model LSTM terdiri dari empat lapisan, masing-masing dengan 50 unit memori, dan dilengkapi dengan lapisan Dropout sebesar 20% untuk mencegah *overfitting*. Model ini dikompilasi menggunakan optimizer RMSprop dan fungsi kerugian Root Mean Squared Error (RMSE). Pelatihan model dilakukan selama 50 epoch dengan batch size sebesar 32. Hasil evaluasi menunjukkan bahwa model LSTM mampu memberikan prediksi produksi minyak sawit dengan nilai RMSE sebesar 0.1238 untuk total TBS terima, 0.1177 untuk total TBS olah, dan 0.1177 untuk jumlah CPO. Hasil ini menunjukkan bahwa model LSTM memiliki potensi yang baik dalam memprediksi produksi minyak sawit, namun masih memerlukan penyempurnaan lebih lanjut.

Kata Kunci: *Long Short-Term Memory (LSTM), Prediksi Produksi Minyak Sawit, MinMaxScaler, Root Mean Square Error (RMSE), Deep Learning*

*APPLICATION OF DATA MINING TO PREDICT THE AMOUNT OF PALM OIL
PRODUCTION WITH THE LONG SHORT-TERM MEMORY METHOD*

ABSTRACT

Palm oil production is one of the important indicators in the agribusiness sector in Indonesia. Accurate prediction of the amount of palm oil production can help in better planning and decision-making processes. The problem faced is the fluctuation in the number of palm fruit bunches entering the mill, which causes uncertainty in the production of crude oil or CPO. The changing results of CPO production have an impact on the company. This research aims to build a palm oil production prediction model using the Long Short-Term Memory (LSTM) method. The data used includes palm oil production from 2019 to 2023, with data up to 2022 used as the training set and data from 2023 and above as the testing set. The data was normalized using MinMaxScaler to ensure the data was within the appropriate range for LSTM model training. The LSTM model consists of four layers, each with 50 memory units, and comes with a 20% Dropout layer to prevent overfitting. The model is compiled using RMSprop optimizer and Root Mean Squared Error (RMSE) loss function. The model training was conducted for 50 epochs with a batch size of 32. The evaluation results show that the LSTM model is able to provide palm oil production prediction with RMSE values of 0.1238 for total received Palm Fruit Bunches, 0.1177 for total processed Palm Fruit Bunches, and 0.1177 for total CPO. These results show that the LSTM model has good potential in predicting palm oil production, but still requires further improvement.

Keywords: *Long Short-Term Memory (LSTM), Palm Oil Production Prediction, MinMaxScaler, Root Mean Square Error (RMSE), Deep Learning*

DAFTAR ISI

PERSETUJUAN	i
PERNYATAAN.....	ii
UCAPAN TERIMA KASIH	iii
ABSTRAK.....	v
ABSTRACT.....	vi
DAFTAR ISI	vii
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	4
1.7 Metodologi Penelitian	4
1.8 Penelitian Relevan.....	5
1.9 Sistematika Penelitian	7
BAB 2 LANDASAN TEORI	9
2.1 Prediksi.....	9
2.2 Kelapa Sawit	10
2.3 Minyak Mentah (CPO).....	11
2.4 <i>Data mining</i>	12
2.5 <i>Machine learning</i>	13
2.6 <i>Artificial Intelligence</i>	13

2.7	<i>Deep Learning</i>	14
2.8	<i>Artificial Neural Network (ANN)</i>	15
2.9	<i>Recurrent Neural Network (RNN)</i>	16
2.10	<i>Root Mean Square Error (RMSE)</i>	17
2.11	<i>Long Short-Term Memory</i>	18
2.12	Normalisasi Data	24
BAB 3 ANALISIS DAN PERANCANGAN		26
3.1	Analisis	26
3.1.2	Analisis Kebutuhan.....	27
3.2	Arsitektur Umum.....	28
3.3	Pengumpulan <i>Dataset</i>	29
3.4	<i>Preprocessing Data</i>	30
3.5	Normalisasi Data	30
3.6	Pembagian Data Set	32
3.7	Implementasi LSTM	32
3.9	Perancangan Sistem.....	35
3.9.1	<i>Flowchart</i>	35
BAB 4 HASIL DAN PEMBAHASAN.....		36
4.1	Implementasi Sistem	36
4.1.1	Spesifikasi Perangkat Keras	36
4.1.2	Spesifikasi Perangkat Lunak	36
4.2	Pengumpulan <i>Dataset</i>	37
4.3	<i>Preprocessing Data</i>	37
4.3.1	Menghapus nilai kosong	37
4.4	Pembagian Data.....	38
4.5	Konversi <i>Dataset</i>	40

4.6 Normalisasi.....	40
4.7 Pelatihan Model LSTM.....	41
4.8 Persiapan Data Uji.....	43
4.9 Implementasi LSTM	46
4.10 Evaluasi	48
4.11 Hasil Prediksi.....	49
BAB 5 PENUTUPAN	59
5.1 Kesimpulan.....	59
5.2 Saran.....	59

DAFTAR TABEL

Tabel 3. 1 Contoh Dataset	30
Tabel 3. 2 Contoh data asli sebelum dinormalisasi	30
Tabel 3. 3 Contoh nilai asli dan nilai normalisasi	31
Tabel 4. 1 Dataset	37
Tabel 4. 2 Contoh dataset TBS Terima yang dinormalisasi	40
Tabel 4. 3 Contoh dataset TBS Olah yang dinormalisasi	41
Tabel 4. 4 Contoh dataset Jumlah CPO yang dinormalisasi.....	41
Tabel 4. 5 Proses Pelatihan	47
Tabel 4. 6 Presentase Nilai Asli dan Nilai Prediksi TBS Terima	51
Tabel 4. 7 Presentase Nilai Asli dan Nilai Prediksi TBS Olah	51
Tabel 4. 8 Presentase Nilai Asli dan Nilai Prediksi Jumlah CPO.....	52

DAFTAR GAMBAR

Gambar 2. 1 Arsitektur Sistem <i>LTSM</i>	19
Gambar 2. 2 Sel Memory LSTM.....	20
Gambar 2. 3 Alur Informasi <i>Input gate</i>	21
Gambar 2. 4 Alur Informasi <i>Input gate</i>	22
Gambar 2. 5 Alur Informasi <i>Input gate</i>	23
Gambar 3. 1 Arsitektur Sistem Penelitian	28
Gambar 3. 2 <i>Flowchart</i> Sistem.....	36
Gambar 4. 1 Hasil menghapus nilai kosong.....	38
Gambar 4. 2 Kodingan Split Dataset TBS Terima	38
Gambar 4. 3 Visualisasi Hasil Split Dataset TBS Terima.....	39
Gambar 4. 4 Kodingan Split Dataset TBS Olah.....	39
Gambar 4. 5 Visualisasi Hasil Split Dataset TBS Terima.....	39
Gambar 4. 6 Kodingan Split Dataset Jumlah CPO.....	39
Gambar 4. 7 Visualisasi Hasil Split Dataset TBS Terima.....	39
Gambar 4. 8 Konversi dataset	40
Gambar 4. 9 Kodingan Pelatihan Model TBS Terima.....	42
Gambar 4. 10 Kodingan Pelatihan Model TBS Olah	43
Gambar 4. 11 Kodingan Pelatihan Model Jumlah CPO	43
Gambar 4. 12 Kodingan Reshape	43
Gambar 4. 13 Pelatihan data Uji TBS Terima	44
Gambar 4. 14 Pelatihan data Uji TBS Terima	44
Gambar 4. 15 Pelatihan data Uji TBS Terima	44
Gambar 4. 16 Persiapan Data Uji TBS Terima.....	45
Gambar 4. 17 Persiapan Data Uji TBS Olah	45
Gambar 4. 18 Persiapan Data Uji Jumlah CPO.....	45
Gambar 4. 19 Implementasi LSTM.....	47
Gambar 4. 20 Hasil Evaluasi TBS Terima.....	48
Gambar 4. 21 Hasil Evaluasi TBS Olah	48

Gambar 4. 22 Hasil Evaluasi Jumlah CPO.....	49
Gambar 4. 23 Visualisasi Data Asli dan Data Prediksi TBS Terima	49
Gambar 4. 24 Visualisasi Data Asli dan Data Prediksi TBS Terima	50
Gambar 4. 25 Visualisasi Data Asli dan Data Prediksi TBS Terima	50
Gambar 4. 26 Visualisasi Hasil Prediksi TBS Terima Tahun 2024 - 2026.....	53
Gambar 4. 27 Visualisasi Hasil Prediksi TBS Terima Tahun 2024	53
Gambar 4. 28 Visualisasi Hasil Prediksi TBS Terima Tahun 2025	54
Gambar 4. 29 Visualisasi Hasil Prediksi TBS Terima Tahun 2026	54
Gambar 4. 30 Hasil Prediksi TBS Terima untuk 3 tahun kedepan	55
Gambar 4. 31 Visualisasi Hasil Prediksi TBS Olah Tahun 2024 – 2026	55
Gambar 4. 32 Visualisasi Hasil Prediksi TBS Olah Tahun 2024	56
Gambar 4. 33 Visualisasi Hasil Prediksi TBS Olah Tahun 2025	56
Gambar 4. 34 Visualisasi Hasil Prediksi TBS Olah Tahun 2026	57
Gambar 4. 35 Hasil Prediksi TBS Olah untuk 3 tahun kedepan	57
Gambar 4. 36 Visualisasi Hasil Prediksi Jumlah CPO Tahun 2024 - 2026.....	58
Gambar 4. 37 Visualisasi Hasil Prediksi Jumlah CPO Tahun 2024	58
Gambar 4. 38 Visualisasi Hasil Prediksi Jumlah CPO Tahun 2025	59
Gambar 4. 39 Visualisasi Hasil Prediksi Jumlah CPO Tahun 2026	59
Gambar 4. 40 Hasil Prediksi TBS Terima untuk 3 tahun kedepan.....	60

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Produksi minyak sawit adalah salah satu pilar ekonomi utama bagi negara-negara produsen minyak sawit seperti Indonesia dan Malaysia. Dalam beberapa dekade terakhir, produksi minyak sawit telah meningkat secara signifikan untuk memenuhi permintaan global akan minyak nabati. Namun, fluktuasi harga komoditas, perubahan iklim, dan faktor-faktor lainnya seringkali memengaruhi produksi minyak sawit, yang menimbulkan tantangan bagi para pelaku industri dalam merencanakan produksi dan distribusi dengan efisien.

Pentingnya manajemen produksi minyak mentah atau CPO (*Crude Palm Oil*) ini juga berlaku pada perusahaan PT. Agro Indah, yang berlokasi di provinsi Riau. Perusahaan ini mengolah tandan buah sawit (TBS) menjadi minyak mentah atau CPO, yang ditentukan oleh jumlah tandan buah sawit yang masuk dan diolah dari kebun dan luar kebun, serta jumlah CPO. PT. Agro Indah menghadapi masalah seperti fluktuasi jumlah tandan buah sawit yang masuk ke pabrik, hal ini menyebabkan munculnya ketidakpastian produksi minyak mentah atau CPO. Hasil produksi CPO yang berubah-ubah memberikan dampak bagi perusahaan seperti tidak tentunya waktu perawatan peralatan dan mesin, tidak diketahuinya stok minyak CPO yang harus diperlukan, dan tidak tentunya *cashflow* perusahaan tersebut

Salah satu metode untuk memprediksi jumlah produksi minyak mentah atau CPO ini adalah dengan menggunakan teknik *data mining*. *Data mining* merupakan suatu metode untuk menggali informasi yang mungkin tidak tampak dari kumpulan data besar. *Data mining* ini membantu untuk memahami pola-pola masa depan dan karakteristik yang ada dalam data, sehingga perusahaan dapat membuat keputusan (Kamath et al., 2021). Salah satu algoritma yang dapat dimanfaatkan dalam *data mining* ini untuk memprediksi hasil produksi minyak sawit adalah algoritma *Long Short-Term Memory*. *Long Short-Term Memory* (LSTM) adalah tipe jaringan saraf tiruan yang sering digunakan dalam komputer untuk memproses data, terutama dalam konteks teks

atau deret waktu. Keistimewaan menggunakan algoritma ini adalah kemampuan mengingat informasi penting dalam jangka waktu yang panjang (Song et al., 2020).

Penelitian ini bertujuan untuk melakukan prediksi jumlah produksi minyak mentah atau CPO di PT. Agro Indah menggunakan algoritma *Long Short-Term Memory* dengan model yang tetap agar mampu memberikan hasil prediksi produksi CPO dan mendapatkan akurasi prediksi produksi CPO yang optimal. Batasan dalam penelitian ini melibatkan penggunaan data historis produksi minyak mentah atau CPO, seperti jumlah tandan buah sawit yang diterima dari kebun maupun luar kebun, serta jumlah tandan buah sawit yang diolah dari kebun maupun luar kebun dan jumlah CPO yang diperoleh dari PT. Agro Indah. Data ini mencakup periode tahun 2019 – 2023.

Selain itu penelitian ini diharapkan dapat memberikan wawasan atau pemahaman yang lebih mendalam di bidang produksi minyak mentah atau CPO. Penelitian ini juga diharapkan untuk dapat memberikan wawasan dan bantuan kepada perusahaan untuk merencanakan produksi yang lebih efisien, mengelola stok dengan lebih akurat, serta menghindari masalah stok berlebih atau kekurangan. Selain itu, diharapkan bahwa informasi yang diperoleh dari penelitian ini dapat mendukung perusahaan dalam bernegosiasi kontrak penjualan yang lebih baik, menjadwalkan perawatan peralatan dengan lebih efisien, serta mengelola *cashflow* keuangan perusahaan yang lebih baik. Perusahaan juga diharapkan dapat mengambil tindakan yang tepat untuk mengatasi faktor-faktor yang memengaruhi hasil produksi. Dengan demikian, penelitian ini diharapkan dapat meningkatkan efisiensi operasional dan mendukung pengambilan keputusan yang lebih tepat dalam berbagai aspek manajemen perkebunan kelapa sawit.

Dalam beberapa penelitian, *Long Short-Term Memory* digunakan untuk memprediksi jumlah produksi, seperti yang dilakukan oleh Tonymah Jaelania., Mohamad Yamina , Cokorda Prapti Mahandaria pada tahun 2022 berjudul “*Machine learning* untuk Prediksi Produksi Gula Nasional” dimana penelitian ini membandingkan algoritma *Machine learning Long Short-Term Memory* dan *regresi linear* untuk memprediksi jumlah produksi gula nasional. Hasil penelitian ini membuktikan bahwa algoritma *Long Short-Term Memory* mampu menghasilkan nilai *error* lebih kecil dibanding dengan *regresi linear* yaitu LSTM menghasilkan nilai *error* data *test* 0,082% dan *error data train* 0,069%. sedangkan *error* pada metode *regresi linier* adalah 8%. Perbedaan antara penelitian yang dilakukan oleh penulis dengan

penelitian sebelumnya adalah studi kasus dan ukuran akurasi dimana penulis menggunakan nilai akurasi RMSE.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, dibutuhkan penelitian untuk memprediksi jumlah produksi minyak sawit menggunakan algoritma *Long Short-Term Memory* yang dapat membantu PT. Agro Indah untuk mengatasi permasalahan seperti fluktuasi jumlah tandan buah sawit yang masuk ke pabrik serta menyebabkan munculnya ketidakpastian produksi minyak mentah atau CPO.

1.3 Batasan Masalah

Batasan dalam penelitian ini adalah :

1. Program ini ditujukan kepada PT. Agro Indah agar mendapatkan hasil prediksi jumlah CPO dan mendapatkan akurasi prediksi CPO yang optimal.
2. Penelitian ini menerapkan algoritma *Long Short-Term Memory* (LSTM).
3. Data yang digunakan adalah seperti jumlah tandan buah sawit yang diterima dari kebun maupun luar kebun, serta jumlah tandan buah sawit yang diolah dari kebun maupun luar kebun dan jumlah CPO yang diperoleh dari PT. Agro Indah.
4. Data yang digunakan mencakup periode tahun 2019 – 2023.
5. Program pada penelitian ini menghasilkan model dan nilai evaluasi menggunakan RMSE.
6. Program akan diimplementasikan menggunakan *Google Colab* menggunakan Bahasa *python*.

1.4 Tujuan Penelitian

Penelitian ini bertujuan untuk melakukan prediksi jumlah produksi minyak mentah atau CPO di PT. Agro Indah menggunakan algoritma *Long Short-Term Memory* dengan model yang tetap agar mampu memberikan hasil prediksi produksi CPO dan mendapatkan akurasi prediksi produksi CPO yang optimal. Penelitian ini melibatkan penggunaan data sebagai faktor yang mempengaruhi produksi minyak mentah atau CPO, seperti jumlah tandan buah sawit yang diterima dari kebun maupun luar kebun, serta jumlah tandan buah sawit yang diolah dari kebun maupun luar kebun dan jumlah

CPO yang diperoleh dari PT. Agro Indah . Data ini mencakup periode tahun 2019 – 2023.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini diharapkan mampu:

1. Mampu memberikan pengetahuan yang lebih di bidang produksi minyak mentah atau CPO.
2. Membantu perusahaan untuk merencanakan produksi yang lebih efisien, mengelola stok dengan lebih akurat
3. Membantu perusahaan untuk mengelola stok dengan lebih akurat serta menghindari masalah stok berlebih atau kekurangan.
4. Mendukung perusahaan dalam bernegosiasi kontrak penjualan yang lebih baik,
5. Membantu perusahaan untuk menjadwalkan perawatan peralatan dengan lebih efisien
6. Membantu perusahaan untuk mengelola *cashflow* keuangan perusahaan yang lebih baik.
7. membantu perusahaan untuk meningkatkan efisiensi operasional dan mendukung pengambilan keputusan yang lebih tepat dalam berbagai aspek manajemen perkebunan kelapa sawit.

1.7 Metodologi Penelitian

Metode penelitian yang digunakan adalah sebagai berikut:

1. Studi Pustaka

Studi pustaka merupakan tahap dimana penelitian dimulai dengan mencari referensi dan meninjau literatur seperti buku, jurnal, *e-book*, artikel ilmiah, makalah, dan penelitian tentang prediksi jumlah produksi minyak mentah atau CPO, teknik *data mining*, dan metode *Long Short-Term Memory*.

2. Studi Lapangan

Pada tahap ini, penulis akan melakukan wawancara dan survei kepada PT. Agro Indah untuk memperoleh informasi yang dibutuhkan untuk penelitian ini.

3. Analisa dan Perancangan

Penulis mengevaluasi apa yang diperlukan untuk penelitian berdasarkan ruang lingkupnya.

3. Implementasi

Implementasi prediksi ini menggunakan bahasa pemrograman *python* menggunakan *Google Colab*. Pada tahap ini telah dilakukan perhitungan menggunakan algoritma *Long Short-Term Memory* dan perhitungan nilai akurasi menggunakan RMSE.

4. Pengujian

Pada pengujian ini performa menggunakan data historis produksi yang telah terjadi sebelumnya, pengujian dengan data pembanding, pengujian sensitivitas, pengujian kesalahan, validasi model, uji coba interaktif dan pengujian kinerja.

5. Dokumentasi

Dokumentasi adalah pendekatan yang menggunakan analisis dan penelaahan dokumen atau sumber data yang telah ada sebagai sumber informasi utama dalam penelitian. Metode ini melibatkan pengumpulan, evaluasi, dan interpretasi dokumen-dokumen yang relevan dengan topik penelitian untuk mendapatkan pemahaman yang lebih baik tentang fenomena yang diteliti.

1.8 Penelitian Relevan

1. Penelitian yang dilakukan oleh Mohammad Isa Irawan dan Thoriq Afa Faisal Muhammad pada tahun 2023 berjudul “Implementasi *Long Short-Term Memory* (LSTM) untuk Prediksi Intensitas Curah Hujan (Studi Kasus: Kabupaten Malang)” berfokus pada prediksi curah hujan di Kabupaten Malang. Studi ini menemukan bahwa Model LSTM adalah salah satu algoritma yang efektif untuk prediksi cuaca. Hasil penelitian menunjukkan bahwa nilai error yang digunakan dalam penelitian ini, yaitu RMSE dan MAE terkecil, masing-masing sebesar 0.98162 dan 0.68847, menunjukkan bahwa model tersebut lebih akurat dengan tingkat kesalahan yang lebih rendah.
2. Penelitian yang dilakukan oleh Mohamad Yamina, Toniya Jaelania dan Cokorda Prapti Mahandaria pada tahun 2022 membandingkan metode prediksi produksi

gula di Indonesia menggunakan pembelajaran mesin, *Long Short-Term Memory* (LSTM), dengan metode *regresi linier*. Penelitian ini didasarkan pada data sekunder dari studi sebelumnya dan laporan instansi terkait. Data nasional yang digunakan mencakup urutan waktu selama 52 tahun, dari 1968 hingga 2020. Hasil penelitian menunjukkan bahwa metode regresi linier memiliki tingkat kesalahan sebesar 8%. Sebaliknya, prediksi menggunakan pembelajaran mesin menunjukkan tingkat kesalahan yang lebih rendah. Metode LSTM menghasilkan *error training* sebesar 0,069% dan *error data pengujian* sebesar 0,082%.

3. Penelitian oleh Michael David, Imam Cholissodin, dan Novanto Yudistira pada tahun 2023 yang berjudul "*Prediksi Harga Cabai menggunakan Metode Long-Short Term Memory* (Studi Kasus: Kota Malang)" dilakukan karena masalah fluktuasi harga cabai yang signifikan di Kota Malang. Fluktuasi ini membuat prediksi harga cabai menjadi tugas yang menantang. Pemerintah khawatir tentang stabilisasi harga cabai agar tetap terjangkau, sehingga inflasi harga cabai di Kota Malang dapat terjaga dengan baik. Penelitian ini melibatkan beberapa tahap prediksi, termasuk pre-processing, normalisasi data, pelatihan model, dan prediksi menggunakan metode *Long Short-Term Memory* (LSTM), serta evaluasi hasil dengan *Mean Square Error* (MSE).
4. Penelitian oleh Harisatul Aulia, Muhammad Alfaris Oktavian, Arya Mulya Kusuma, Putri Patricia, Muhammad Rizky Akbar dan Abdiansah Abdiansah yang berjudul "*Prediksi Gender Berdasarkan Nama Bahasa Indonesia menggunakan Long Short-Term Memory*" berfokus pada penggunaan metode *Long Short-Term Memory* (LSTM) untuk memprediksi dan mengklasifikasikan jenis kelamin berdasarkan nama-nama orang Indonesia. Tujuan penelitian ini adalah untuk meningkatkan pemahaman ilmiah dan menemukan inovasi baru yang dapat mendukung penelitian masa depan. Penelitian ini dibatasi oleh penggunaan dataset yang hanya terdiri dari nama-nama orang Indonesia dan klasifikasi jenis kelamin yang hanya mencakup dua kategori, yaitu laki-laki dan perempuan. Penelitian yang dilakukan oleh Adhipramana Raihan Yuthadi, Imam Cholissodin, dan Edy Santoso dengan judul "*Prediksi Jumlah Kasus COVID-19 di Dunia dengan menggunakan Metode Long Short-Term Memory*" membahas tentang COVID-19, penyakit menular yang disebabkan oleh virus baru yang muncul di Wuhan, Tiongkok pada Desember 2019. Penyakit ini dengan cepat menyebar ke lebih dari 114 negara,

menyebabkan dampak serius pada sektor ekonomi. Dalam penelitian ini, peramalan jumlah kasus COVID-19 dilakukan menggunakan metode *Long Short-Term Memory* (LSTM) dengan data dari enam negara, yaitu Indonesia, China, Jepang, Singapura, Malaysia, dan Korea Selatan. Tujuan penelitian ini adalah memberikan analisis yang akurat untuk mendukung pengambilan keputusan oleh pemerintah dan pihak terkait.

1.9 Sistematika Penelitian

Adapun sistematika penulisan yang digunakan dalam penelitian ini adalah sebagai berikut :

BAB 1: Pendahuluan

Sub bab pendahuluan adalah bagian pertama dari suatu penelitian, makalah, atau laporan yang bertujuan untuk memperkenalkan pembaca dengan topik yang akan dibahas. Pendahuluan juga memberikan gambaran tentang latar belakang topik, termasuk sejarah dan kemajuan terbaru dalam bidang tersebut. Selain itu, pendahuluan juga menentukan masalah atau pertanyaan penelitian yang akan dijawab, menunjukkan betapa relevan dan pentingnya topik tersebut untuk penelitian yang akan datang. Bagian pendahuluan juga mencakup tujuan penelitian, yang dijelaskan secara jelas oleh penulis untuk menetapkan fokus dan arah penelitian serta memberikan pembaca pemahaman yang baik tentang manfaat yang diharapkan dari penelitian. Pendahuluan yang baik membantu pembaca memahami topik yang akan dibahas dan tujuan penelitian.

BAB 2: Landasan Teori

Bab landasan teori akan berfokus pada dua elemen utama: produksi minyak sawit dan metode *Long Short-Term Memory* (LSTM). Pertama, akan dibahas secara menyeluruh tentang industri minyak sawit, termasuk faktor-faktor yang memengaruhi produksi minyak sawit. Selain itu, akan ada penjelasan tentang bagaimana prediksi produksi sangat penting untuk perencanaan bisnis dan manajemen risiko. Bab ini juga akan membahas metode LSTM sebagai alat yang berguna untuk memodelkan data deret waktu. Ini akan memberikan gambaran umum tentang struktur LSTM, mekanisme kerjanya, dan keunggulannya dalam menangani pola temporal yang kompleks. Penjelasan ini akan memastikan pemahaman yang mendalam tentang subjek penelitian

dan teknik analisis yang digunakan, dan akan memberikan landasan yang kokoh untuk penelitian yang akan datang.

BAB 3: Analisis dan Perancangan

Bab ini mencakup analisis dan perancangan sistem prediksi menggunakan algoritma *Long Short-Term Memory*.

BAB 4: Implementasi dan Pengujian

Sub bab ini membahas metode implementasi dan uji sistem yang telah dirancang. Tujuan utama dari bab ini adalah untuk memberikan gambaran menyeluruh tentang proses implementasi sistem, mulai dari tahap perancangan hingga produk jadi yang siap digunakan. Di sisi lain, bagian pengujian membahas teknik yang digunakan untuk menguji kinerja dan keandalan sistem, menampilkan hasil, dan memeriksa apakah sistem memenuhi spesifikasi dan tujuan yang telah ditetapkan.

BAB 5: Kesimpulan dan Saran

Sub ini merupakan bagian penting karena merupakan bab kesimpulan dan saran, yang memberikan ringkasan singkat dari hasil dan temuan penelitian serta saran untuk penelitian lanjutan atau implikasi praktis dari temuan tersebut. Dalam bab ini, penelitian akan merangkum hasil utama penelitian, menekankan pentingnya temuan tersebut, dan memberikan rekomendasi untuk pengembangan lebih lanjut. Kesimpulan juga dapat memberikan evaluasi penting terhadap metodologi penelitian dan rekomendasi untuk meningkatkan penelitian di masa mendatang. Sub bab ini memberikan ringkasan menyeluruh dan mendalam tentang kontribusi penelitian terhadap pengetahuan saat ini dan prospek penelitian masa depan.

BAB 2

LANDASAN TEORI

2.1 Prediksi

Prediksi adalah proses memperkirakan atau meramalkan kejadian di masa depan berdasarkan data historis atau informasi yang ada. Dalam berbagai bidang, prediksi digunakan untuk membantu pengambilan keputusan yang lebih baik (Kusuma et al., 2023). Misalnya, dalam bisnis, prediksi dapat membantu dalam menentukan strategi pemasaran, perencanaan produksi, dan manajemen persediaan. Di sektor keuangan, prediksi digunakan untuk memperkirakan harga saham, nilai tukar mata uang, dan risiko investasi. Sementara itu, dalam ilmu pengetahuan dan teknologi, prediksi membantu dalam memahami tren perubahan iklim, epidemiologi penyakit, dan perkembangan teknologi.

Metode prediksi sangat beragam, mulai dari teknik statistik sederhana hingga algoritma pembelajaran mesin yang kompleks. Metode tradisional seperti *regresi linier* dan analisis deret waktu telah lama digunakan untuk melakukan prediksi berdasarkan hubungan linear antara variabel. Namun, dengan kemajuan teknologi dan ketersediaan data besar (*big data*) (David et al., n.d.-a), metode yang lebih canggih seperti jaringan saraf tiruan, khususnya *Long Short-Term Memory* (LSTM), semakin populer. LSTM, yang merupakan jenis dari jaringan saraf berulang (*Recurrent Neural Network*), sangat efektif dalam menangani data deret waktu yang memiliki pola kompleks dan hubungan jangka panjang.

Implementasi metode LSTM dalam prediksi memiliki keunggulan karena kemampuannya dalam mempertahankan informasi relevan dari data historis untuk periode waktu yang panjang. Hal ini memungkinkan LSTM untuk membuat prediksi yang lebih akurat dalam situasi di mana pola temporal sangat berpengaruh, seperti prediksi cuaca, harga saham, atau jumlah kasus penyakit. Misalnya, dalam prediksi jumlah kasus COVID-19, LSTM dapat memanfaatkan data harian kasus yang terjadi sebelumnya untuk mengidentifikasi tren dan pola penyebaran virus, sehingga memberikan estimasi yang lebih andal tentang jumlah kasus di masa mendatang.

Selain keunggulannya, prediksi dengan metode LSTM juga menghadapi tantangan, terutama dalam hal kebutuhan data yang besar dan kompleksitas perhitungan. Untuk menghasilkan model prediksi yang akurat, diperlukan dataset yang cukup besar dan representatif. Proses pelatihan model juga memerlukan sumber daya komputasi yang tinggi dan waktu yang cukup lama. Namun, seiring dengan perkembangan teknologi komputasi dan peningkatan aksesibilitas data, metode prediksi berbasis LSTM dan teknik pembelajaran mesin lainnya terus mengalami peningkatan efisiensi dan akurasi. Dengan demikian, prediksi menjadi alat yang semakin penting dan dapat diandalkan dalam berbagai aspek kehidupan dan industri..

2.2 Kelapa Sawit

Kelapa sawit (*Elaeis guineensis*) adalah tanaman tropis yang telah menjadi salah satu komoditas penting dalam industri pertanian dan minyak nabati. Tanaman ini berasal dari wilayah Afrika Barat, tetapi saat ini juga ditanam secara luas di berbagai belahan dunia dengan iklim tropis, seperti Indonesia, Malaysia, Thailand, dan Nigeria (Rahmatullah & Destia, 2020). Kelapa sawit dikenal karena buahnya yang menghasilkan minyak nabati yang memiliki berbagai kegunaan, mulai dari bahan baku untuk makanan, kosmetik, hingga bahan bakar biodiesel. Tingkat produktivitas yang tinggi dari tanaman kelapa sawit membuatnya menjadi salah satu tanaman minyak nabati yang paling efisien dan populer di dunia.

Tanaman kelapa sawit tumbuh baik di daerah dengan curah hujan yang tinggi dan suhu yang stabil, yang umumnya terjadi di daerah tropis. Perkebunan kelapa sawit biasanya terdiri dari lahan-lahan yang luas yang ditanami dengan pohon kelapa sawit. Proses pertumbuhan dan produksi kelapa sawit memerlukan waktu yang cukup lama, biasanya memakan waktu sekitar 3 hingga 4 tahun sejak tanam hingga masa panen pertama. Setelah mencapai usia produksi, tanaman kelapa sawit dapat menghasilkan buah secara terus-menerus selama lebih dari 20 tahun dengan perawatan yang tepat.

Industri kelapa sawit memiliki peran penting dalam ekonomi global, terutama bagi negara-negara produsen utama seperti Indonesia dan Malaysia. Produksi dan perdagangan minyak sawit memberikan kontribusi signifikan terhadap pendapatan negara dan lapangan kerja. Namun, industri ini juga menghadapi tantangan terkait dengan keberlanjutan lingkungan, hak asasi manusia, dan kesejahteraan sosial. Oleh karena itu, upaya-upaya untuk meningkatkan produksi secara berkelanjutan dan efisien,

sambil memperhatikan dampak lingkungan dan sosialnya, menjadi fokus penting dalam pengelolaan perkebunan kelapa sawit di masa depan..

Penggunaan *data mining* memiliki peran penting dalam mengolah dan menganalisis *dataset* yang berkaitan dengan produksi minyak sawit. *Data mining* akan digunakan untuk mengidentifikasi pola-pola yang tersembunyi dalam dataset produksi minyak sawit, seperti faktor-faktor yang memengaruhi produksi seperti kondisi cuaca, metode pertanian, dan faktor ekonomi. Metode *data mining* yang digunakan mungkin mencakup teknik-teknik seperti regresi, klasifikasi, dan pengelompokan data, yang akan membantu dalam memahami hubungan antara variabel-variabel yang relevan dengan produksi minyak sawit.

Lebih lanjut, *data mining* juga akan mempersiapkan *dataset* yang sesuai untuk dianalisis dengan menggunakan metode *Long Short-Term Memory* (LSTM). LSTM adalah salah satu teknik dalam pembelajaran mesin yang efektif dalam menangani data deret waktu. Dengan memanfaatkan *data mining*, dataset produksi minyak sawit dapat diolah sedemikian rupa sehingga sesuai dengan format yang diperlukan oleh LSTM. Selain itu, *data mining* juga dapat digunakan untuk mengidentifikasi variabel-variabel yang paling signifikan dalam memprediksi produksi minyak sawit, yang dapat menjadi masukan penting dalam pengembangan model LSTM. Dengan demikian, penggunaan *data mining* dalam penelitian ini menjadi langkah awal yang penting untuk mempersiapkan data dan informasi yang diperlukan dalam proses prediksi produksi minyak sawit dengan metode *Long Short-Term Memory*.

2.3 Minyak Mentah (CPO)

CPO atau minyak sawit mentah adalah jenis minyak kelapa sawit yang diperoleh dari pemerasan atau ekstraksi daging buah kelapa sawit. Biasanya, minyak ini berasal dari spesies kelapa sawit *Elaeis guineensis* dan belum mengalami proses pemurnian (Irawan et al., 2021).

Perlu diingat bahwa minyak sawit mentah berbeda dengan palm kernel oil atau minyak inti kelapa sawit, meskipun keduanya berasal dari buah kelapa sawit yang sama. Selain itu, perbedaan juga ada dengan minyak kelapa yang dihasilkan dari inti buah kelapa. Satu perbedaan utama dari minyak sawit mentah dibandingkan dengan jenis minyak nabati lainnya adalah tingginya kandungan beta-karoten yang memberikan

warna kemerahan pada minyak tersebut. Beta-karoten adalah senyawa awal dari vitamin A dan memiliki pigmen dengan warna dominan merah atau jingga pada buah atau sayuran.

2.4 *Data mining*

Data mining adalah proses mengungkap pola berharga atau informasi tersembunyi dari kumpulan data yang besar atau kompleks. Tujuan utama dari *data mining* adalah menemukan pola yang bernilai, pengetahuan baru, atau informasi yang dapat dimanfaatkan untuk membuat keputusan yang lebih baik atau untuk lebih memahami fenomena yang diamati. Proses *data mining* melibatkan beberapa tahap, termasuk pemilihan data, pra-pemrosesan data, pemodelan, evaluasi, dan interpretasi hasil (Kamath et al., 2021). Tahapan-tahapan ini membentuk alur kerja yang sistematis untuk menganalisis data dengan tujuan memperoleh pemahaman yang lebih dalam atau informasi yang berguna dari data tersebut.

Salah satu teknik yang sering digunakan dalam *data mining* adalah analisis asosiasi, yang bertujuan menemukan hubungan antara item dalam kumpulan data besar, seperti data transaksi penjualan di toko ritel. Teknik ini mengidentifikasi asosiasi antara item-item yang sering muncul bersama dalam transaksi. Misalnya, analisis asosiasi dapat menunjukkan bahwa pelanggan yang membeli roti juga sering membeli mentega. Selain itu, klasifikasi adalah teknik lain yang umum dalam *data mining*, di mana model belajar dari data berlabel untuk memprediksi kategori atau kelas baru untuk data yang tidak berlabel. Contohnya, klasifikasi dapat digunakan untuk memprediksi apakah email yang masuk adalah spam atau bukan.

Selain itu, *data mining* juga melibatkan teknik klustering untuk mengelompokkan data ke dalam grup yang serupa berdasarkan atribut tertentu. Misalnya, klustering dapat digunakan untuk mengelompokkan pelanggan ke dalam segmen-segmen pasar berdasarkan perilaku belanja mereka. Teknik regresi juga sering digunakan dalam *data mining* untuk memodelkan hubungan antara variabel dependen dan independen dalam data. Teknik ini membantu memahami bagaimana perubahan dalam satu variabel memengaruhi variabel lain, dan dapat digunakan untuk membuat prediksi tentang nilai variabel dependen di masa depan berdasarkan nilai variabel independen.

Data mining memiliki aplikasi yang luas di berbagai bidang, termasuk bisnis, ilmu pengetahuan, kesehatan, keamanan, dan banyak lagi. Di bidang bisnis, *data mining* digunakan untuk analisis pasar, manajemen risiko, pemasaran yang dipersonalisasi, dan pengoptimalan proses bisnis. Di bidang ilmu pengetahuan, *data mining* digunakan untuk menemukan pola-pola dalam data eksperimental atau observasional, yang dapat membantu dalam penemuan baru atau pemahaman fenomena yang kompleks (Yunus & Akbar, 2020). Dengan demikian, *data mining* berperan penting dalam menyediakan wawasan berharga dari data yang ada, yang dapat digunakan untuk membuat keputusan yang lebih baik atau untuk mendukung penelitian dan inovasi di berbagai bidang.

2.5 *Machine learning*

Pembelajaran Mesin (ML) atau *Machine learning* adalah pendekatan dalam dunia kecerdasan buatan yang banyak dimanfaatkan untuk meniru cara manusia berpikir dan menyelesaikan masalah atau melakukan tugas secara otomatis. Secara sederhana, *Machine learning* berusaha meniru bagaimana kita, manusia, belajar dan membuat keputusan berdasarkan pengalaman. Prediksi dan klasifikasi adalah dua fungsi utama pembelajaran mesin. *Machine learning* memiliki dua aplikasi utama, yakni klasifikasi dan prediksi. Keunikan dari *Machine learning* terletak pada tahap pelatihan atau pembelajaran (training). Oleh karena itu, *Machine learning* memerlukan data pelatihan yang disebut sebagai data training (Adnan & Amelia, 2022).

Klasifikasi mirip dengan cara manusia membedakan satu hal dari yang lain, di mana mesin menggunakan teknik untuk mengelompokkan atau mengkategorikan objek berdasarkan karakteristik tertentu. Sementara itu, prediksi atau regresi digunakan oleh mesin untuk membuat perkiraan tentang hasil dari suatu data masukan berdasarkan informasi yang telah dipelajari selama proses pelatihan. Artinya, mesin belajar dari pengalaman untuk membuat perkiraan di masa depan.

2.6 *Artificial Intelligence*

Dengan bantuan pengetahuan yang disebut *artificial intelligence* (AI), komputer mampu melaksanakan tugas-tugas yang umumnya dilakukan oleh manusia dan memerlukan kecerdasan manusia. Tidak seperti kemampuan manusia, AI dapat

mengolah data dengan lebih cepat dan membuat prediksi yang lebih akurat. AI semakin terintegrasi ke dalam kehidupan sehari-hari dan menjadi perhatian investasi bagi bisnis di berbagai industri.

Artificial Intelligence (AI) atau sering disebut sebagai teknologi kecerdasan buatan, memainkan peran yang signifikan di zaman sekarang. Teknologi ini mengalami perkembangan besar yang berpengaruh pada hampir semua bidang kehidupan, terutama dalam era teknologi yang sedang berlangsung, di mana AI semakin berkembang dan menjadi lebih canggih. Saat ini, kemajuan pesat dalam teknologi telah mengubah cara orang menjalani kehidupan mereka, termasuk cara mereka berkomunikasi dan bekerja. Kecerdasan buatan menjadi sangat penting dalam era teknologi saat ini untuk menyediakan solusi inovatif dan efisien untuk berbagai masalah yang dihadapi manusia..(Farwati et al., 2023).

Artificial Intelligence (AI) digunakan untuk prediksi dengan memanfaatkan kemampuannya dalam menganalisis data secara efisien dan mengenali pola-pola kompleks. Dengan mengumpulkan dan memproses data historis serta variabel terkait, AI dapat memberikan perkiraan atau prediksi dalam berbagai bidang, seperti cuaca, keuangan, atau tren pasar saham. Pendekatan utama dalam AI untuk prediksi adalah menggunakan teknik *Machine Learning*, di mana sistem dapat belajar dari data dan meningkatkan ketepatan prediksinya seiring waktu.

2.7 Deep Learning

Deep Learning, yang termasuk dalam *machine learning*, menonjolkan pemanfaatan *neural network* yang memiliki banyak lapisan (*deep neural networks*) untuk mengolah dan memahami data. *Neural network*, sebagai model matematis yang terinspirasi dari struktur jaringan saraf manusia, menjadi dasar dari konsep ini. Keunggulan utama *Deep Learning* terletak pada kemampuannya untuk secara otomatis memahami representasi tingkat tinggi dari data melalui serangkaian lapisan pengolahan, sehingga menghilangkan kebutuhan untuk melakukan ekstraksi fitur secara manual (Muhammad Haris Diponegoro et al., 2021)

Deep Learning memiliki peran signifikan diberikan kepada lapisan-lapisan tersebut untuk membantu model memahami struktur data yang semakin kompleks. Ini memungkinkan model untuk belajar merepresentasikan fitur-fitur abstrak,

membuktikan kemanfaatannya dalam tugas-tugas seperti pengenalan gambar, pemrosesan bahasa alami, dan prediksi. Dengan demikian, *Deep Learning* membawa keunggulan dalam menangani tantangan struktur data yang semakin rumit tanpa bergantung pada ekstraksi fitur manual.

2.8 Artificial Neural Network (ANN)

Artificial Neural Network (ANN) adalah model komputasi terinspirasi dari struktur dan fungsi jaringan saraf biologis manusia. ANN terdiri dari sejumlah besar unit pemrosesan sederhana yang disebut neuron atau node, yang terorganisir dalam lapisan-lapisan. Setiap neuron menerima *input*, melakukan operasi matematika pada *input* tersebut, dan menghasilkan *output* yang kemudian disampaikan ke neuron-neuron pada lapisan berikutnya. Neuron-neuron dalam lapisan pertama disebut lapisan *input*, sementara neuron-neuron dalam lapisan terakhir disebut lapisan *output*. Lapisan-lapisan antara keduanya disebut lapisan tersembunyi, di mana transformasi dan pembelajaran terjadi (Farwati et al., 2023).

ANN digunakan untuk memodelkan hubungan kompleks antara *input* dan *output*, dan dapat digunakan untuk berbagai tugas seperti klasifikasi, regresi, pengenalan pola, dan prediksi. Keunggulan utama dari ANN adalah kemampuannya untuk belajar dari data latih, sehingga ANN dapat menyesuaikan diri dengan pola-pola yang ada dalam data dan melakukan prediksi yang akurat pada data yang belum pernah dilihat sebelumnya. Pelatihan ANN dilakukan melalui proses iteratif yang disebut pembelajaran, di mana parameter-parameter jaringan (misalnya, bobot dan bias) diubah sedemikian rupa sehingga meminimalkan kesalahan antara *output* yang dihasilkan oleh jaringan dan *output* yang diharapkan.

Terdapat beberapa jenis arsitektur ANN, termasuk jaringan *feedforward* dan jaringan rekurensi. Jaringan *feedforward* adalah jenis paling sederhana dari ANN, di mana informasi mengalir dari lapisan *input* ke lapisan *output* tanpa siklus atau umpan balik. Sebaliknya, jaringan rekurensi memiliki struktur yang lebih kompleks, dengan kemampuan untuk menangani data berurutan atau berbasis waktu, karena memiliki siklus atau umpan balik yang mengizinkan informasi untuk beredar kembali ke lapisan sebelumnya. Kedua jenis jaringan ini memiliki kelebihan dan kekurangan masing-

masing, dan dipilih tergantung pada tugas yang ingin diselesaikan dan karakteristik data yang ada.

Penerapan ANN telah merambah ke berbagai bidang, termasuk pengenalan wajah, pengenalan suara, pemrosesan bahasa alami, pengenalan tulisan tangan, pengenalan pola, dan banyak lagi. ANN telah menjadi salah satu alat yang paling kuat dalam analisis data dan kecerdasan buatan, karena kemampuannya untuk memodelkan hubungan yang rumit dalam data dan melakukan prediksi yang akurat. Dengan perkembangan teknologi dan metode pembelajaran mesin, ANN terus menjadi area penelitian yang sangat aktif, dengan upaya untuk meningkatkan kinerja, efisiensi, dan skalabilitasnya dalam berbagai aplikasi yang beragam.

2.9 Recurrent Neural Network (RNN)

Recurrent Neural Network (RNN) adalah jenis arsitektur dalam *machine learning* yang dirancang khusus untuk menangani data berurutan atau data berbasis urutan, seperti teks, suara, atau rangkaian waktu. RNN memiliki kemampuan untuk mengingat informasi sebelumnya dalam urutan data, membuatnya sangat cocok untuk tugas-tugas yang melibatkan konteks temporal. Struktur dasar dari RNN terdiri dari serangkaian unit atau sel yang terhubung satu sama lain dalam urutan *linier*, dengan setiap unit menerima *input* dari unit sebelumnya dan menghasilkan *output* yang kembali ke unit berikutnya (Muhammad Haris Diponegoro et al., 2021). Hal ini memungkinkan RNN untuk mempertahankan pemahaman tentang konteks dalam data berurutan dan membuat prediksi berbasis pada informasi sebelumnya.

Salah satu keunggulan utama dari RNN adalah kemampuannya untuk memproses data berurutan dengan panjang yang bervariasi. Hal ini membuatnya sangat berguna dalam berbagai tugas seperti pemrosesan bahasa alami, pemodelan bahasa, dan pemrosesan sinyal waktu. Namun, RNN juga memiliki beberapa kelemahan, terutama dalam menangani masalah yang melibatkan dependensi jarak jauh antara informasi dalam urutan data. Masalah ini dikenal sebagai masalah *vanishing gradients*, di mana informasi yang relevan dengan prediksi saat ini dapat hilang atau "lenyap" saat melewati urutan panjang, menghasilkan kinerja yang kurang memuaskan dalam beberapa kasus.

Guna mengatasi masalah *vanishing gradients*, berbagai varian dari RNN (*Recurrent Neural Network*) telah dikembangkan, salah satunya adalah *Long Short-*

Term Memory (LSTM). LSTM adalah jenis arsitektur RNN yang dilengkapi dengan mekanisme pintar untuk mengontrol aliran informasi dalam sel-selnya. Dengan adanya gate kontrol, LSTM dapat memilih untuk menyimpan atau mengabaikan informasi berdasarkan pada kebutuhan, memungkinkannya untuk mempertahankan informasi yang relevan dalam jangka waktu yang lama. Seiring dengan LSTM, *Gated Recurrent Unit* (GRU) adalah varian lain dari RNN yang menawarkan efisiensi yang tinggi dalam pelatihan dan penggunaan, dengan mekanisme gating yang lebih sederhana dibandingkan dengan LSTM.

Meskipun RNN memiliki kelebihan dalam menangani data berurutan, ada beberapa tantangan dalam pelatihan dan penggunaannya. Salah satu masalah umum adalah kecenderungan RNN untuk menghasilkan *output* yang berulang-ulang dalam beberapa kasus, yang dikenal sebagai masalah mode *collapse* (Cahyadi et al., 2020). Selain itu, pelatihan RNN juga bisa menjadi rumit dan memakan waktu, terutama dalam kasus *dataset* yang besar atau kompleks. Meskipun demikian, dengan pengembangan terus-menerus dalam bidang *machine learning*, RNN tetap menjadi alat yang sangat kuat dan relevan dalam pemrosesan data berurutan dan terus digunakan dalam berbagai aplikasi yang berkaitan dengan urutan data.

2.10 Root Mean Square Error (RMSE)

Root Mean Squared Error (RMSE) adalah metrik yang mengukur tingkat kesalahan dalam hasil prediksi, nilai RMSE yang lebih rendah menunjukkan bahwa prediksi model tersebut lebih akurat. Sebagai metode evaluasi, RMSE memperhitungkan perbedaan antara nilai yang diprediksi oleh model dengan nilai sebenarnya yang diobservasi (Sanjaya & Heksaputra, 2020). Dengan demikian, RMSE berfungsi sebagai alat penilaian yang membantu mengukur sejauh mana model mampu menghasilkan estimasi yang mendekati nilai yang sebenarnya, dan nilai RMSE yang rendah mengindikasikan performa prediktif yang lebih baik. Ini memiliki implikasi praktis yang signifikan, karena model dengan RMSE yang kecil cenderung memberikan prediksi yang lebih dapat diandalkan dalam berbagai aplikasi.

Keuntungan dari *Root Mean Squared Error* (RMSE) adalah dapat memberikan nilai kesalahan dalam satuan yang sama dengan data asli, sehingga lebih mudah diinterpretasikan dan rentan terhadap *outlier* lebih sedikit dibandingkan dengan MSE

karena pengaruh kuadrat pada *outlier* dikurangi oleh pengakaran. Proses penghitungan *Root Mean Squared Error* (RMSE) melibatkan tahap awal pengurangan nilai aktual dari nilai peramalan, diikuti dengan langkah pengkuadratan dan penjumlahan total seluruh hasil kuadrat. Total tersebut kemudian dibagi oleh jumlah data yang ada. Selanjutnya, hasil perhitungan ini kembali diolah dengan menghitung nilai akar kuadrat sesuai dengan formula yang diberikan. Dengan demikian, RMSE memberikan gambaran mengenai tingkat kesalahan atau deviasi antara prediksi model dengan data aktual dalam sebuah *dataset*. Berikut rumus menghitung RMSE :

$$\sqrt{\frac{1}{n} \sum_i^n (\tilde{y}_i - y_i)^2} \quad (1)$$

Keterangan :

\tilde{y}_i : Nilai hasil prediksi

y_i : Nilai aktual / Nilai sebenarnya :

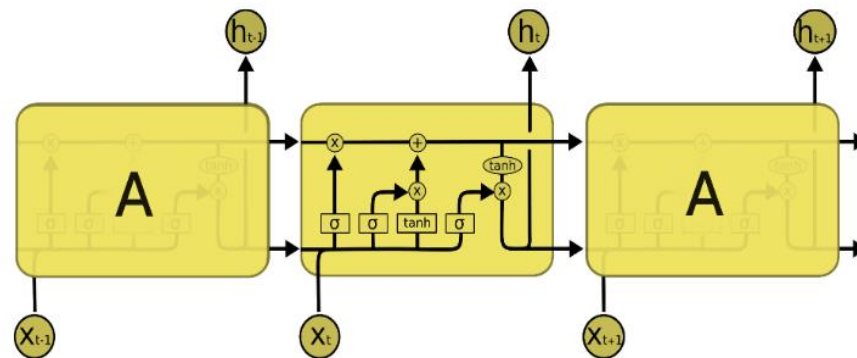
n :Jumlah *dataset*

2.11 Long Short-Term Memory

Long Short-Term Memory (LSTM) merupakan sebuah metode dari jenis jaringan saraf tiruan yang termasuk dalam kategori *Recurrent Neural Networks* (RNN). LSTM dirancang untuk mengatasi kelemahan RNN dalam mempelajari dependensi jangka panjang dalam data deret waktu. RNN standar sering mengalami masalah yang dikenal sebagai "*vanishing gradient*" saat melatih model dengan dependensi temporal yang panjang, menyebabkan hilangnya informasi penting dari awal urutan data. LSTM mengatasi masalah ini dengan menggunakan struktur memori khusus yang terdiri dari sel-sel memori dan mekanisme pengaturan berupa tiga gerbang: *input*, *output*, dan *forget* (Yuthadi et al., n.d.). Gerbang-gerbang ini memungkinkan LSTM untuk mempertahankan, memperbarui, atau menghapus informasi secara selektif, sehingga memungkinkannya untuk menangkap hubungan temporal yang panjang dengan lebih efektif.

Keunggulan LSTM membuatnya sangat berguna dalam berbagai aplikasi yang melibatkan data deret waktu atau data urutan, seperti prediksi cuaca, pemrosesan bahasa alami, analisis sentimen, dan prediksi harga saham. Dalam konteks pemrosesan bahasa alami, misalnya, LSTM dapat mengingat konteks kalimat atau paragraf yang panjang,

membantu dalam tugas-tugas seperti penerjemahan mesin dan pengenalan ucapan. Dalam prediksi harga saham, LSTM dapat menganalisis tren pasar dan pola fluktuasi harga dari data historis untuk memberikan prediksi yang lebih akurat (Insani & Sanjaya, 2022). Meskipun memerlukan sumber daya komputasi yang signifikan untuk pelatihan, kemampuan LSTM dalam menangani data urutan yang kompleks menjadikannya pilihan yang unggul dalam banyak aplikasi pembelajaran mesin. Arsitektur LSTM terdiri dari lapisan *input*, lapisan *output*, dan lapisan tersembunyi seperti terlihat pada gambar 2.1 dibawah ini:



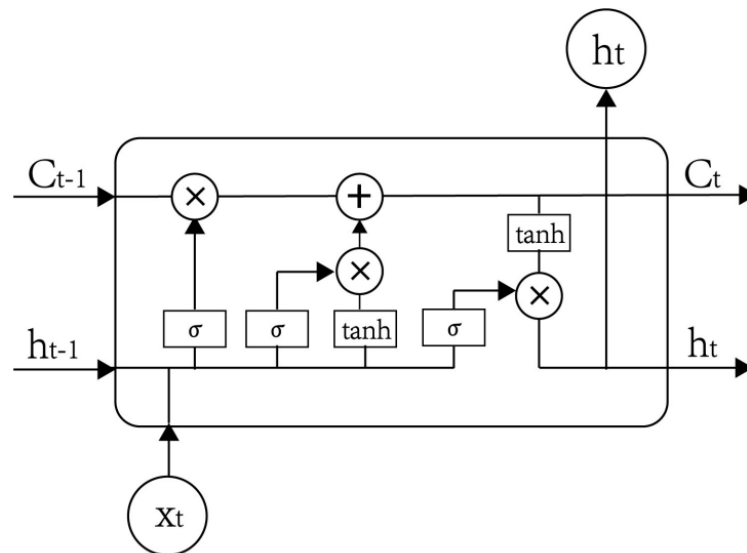
Gambar 2. 1 Arsitektur Sistem LTSM

Sumber : (Adi & Sudianto, 2022)

Struktur *Long Short-Term Memory* (LSTM) terdiri dari unit-unit yang disebut sel memori, yang merupakan elemen dasar dalam jaringan LSTM. Setiap sel memori memiliki tiga komponen utama yang dikenal sebagai gerbang: gerbang *input*, gerbang *output*, dan gerbang *forget* (Qiu et al., 2020). Gerbang *input* menentukan seberapa banyak informasi baru yang akan disimpan dalam sel memori. Gerbang *forget* menentukan informasi apa yang harus dihapus dari sel memori, dan gerbang *output* mengontrol seberapa banyak informasi dari sel memori yang akan digunakan sebagai *output* untuk tahap selanjutnya. Gerbang-gerbang ini terdiri dari jaringan saraf sigmoid yang menghasilkan nilai antara 0 dan 1, yang kemudian digunakan untuk mengatur aliran informasi dalam sel memori.

Keberadaan gerbang-gerbang ini memungkinkan LSTM untuk mempertahankan, memperbarui, atau menghapus informasi secara selektif selama proses pelatihan, sehingga memungkinkan jaringan untuk menangkap dan mempertahankan hubungan temporal yang panjang dalam data deret waktu. Selain itu, LSTM memiliki mekanisme umpan balik yang memungkinkan informasi untuk terus mengalir melalui sel memori

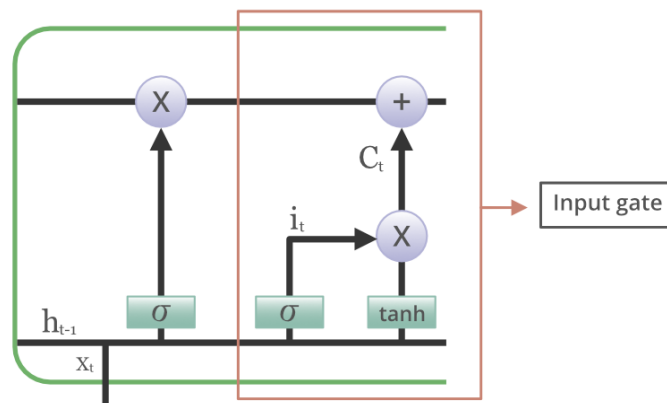
dari satu tahap ke tahap berikutnya, memastikan bahwa informasi penting dari awal urutan data tidak hilang. Struktur ini membuat LSTM sangat efektif dalam menangani masalah yang melibatkan dependensi jangka panjang, seperti pemrosesan bahasa alami, prediksi deret waktu, dan tugas-tugas lain yang memerlukan pemahaman konteks atau pola dalam data sekuensial sel *memory* dapat dilihat seperti gambar 2.2 dibawah ini:



Gambar 2. 2 Sel *Memory* LSTM

Sumber: (Qiu et al., 2020)

Lapisan tersembunyi terdiri dari sel-sel memori, dimana setiap sel memori dilengkapi dengan tiga gate, yaitu *input gate*, *forget gate*, dan *output gate*. *Input gate* dalam LSTM memiliki peran kunci dalam mengendalikan jumlah informasi baru yang akan disimpan dalam sel memori (Song et al., 2020). Dengan memanfaatkan fungsi sigmoid, *input gate* memutuskan sejauh mana setiap nilai *input* harus diperhatikan atau diizinkan untuk memasuki sel memori, memberikan kendali yang presisi terhadap integrasi informasi baru ke dalam *output*. Alur informasi *Input gate* dapat dilihat pada gambar 2.3 dibawah ini:



Gambar 2. 3 Alur Informasi *Input gate*

Sumber : *geeksforgeeks.org*

Fungsi *input gate* dalam struktur Long Short-Term Memory (LSTM) adalah untuk mengontrol aliran informasi baru ke dalam sel memori. *Input gate* menentukan seberapa banyak informasi baru yang akan disimpan dalam sel memori pada setiap langkah waktu (*time step*) dalam urutan data. *Input gate* menggunakan fungsi sigmoid untuk menghasilkan nilai antara 0 dan 1 (Liu et al., 2020). Nilai 0 menunjukkan bahwa informasi baru tidak relevan dan tidak harus disimpan, sedangkan nilai 1 menunjukkan bahwa informasi baru sangat penting dan harus disimpan sepenuhnya. Oleh karena itu, *input gate* memungkinkan LSTM untuk secara selektif menentukan seberapa banyak informasi baru yang akan dimasukkan ke dalam sel memori, yang membantu dalam mengatasi masalah dependensi jangka panjang dalam data deret waktu. Rumus dari *Input gate* dapat dilihat pada persamaan 2 berikut:

$$i_t = \sigma(W_i \cdot x_t + U_i \cdot h_{t-1} + b_i) \quad (2)$$

Dimana :

i_t : *output* dari *input gate*.

σ : fungsi aktivasi sigmoid.

W_i : bobot untuk nilai *input* pada waktu (t)

x_t : *input* pada waktu (t)

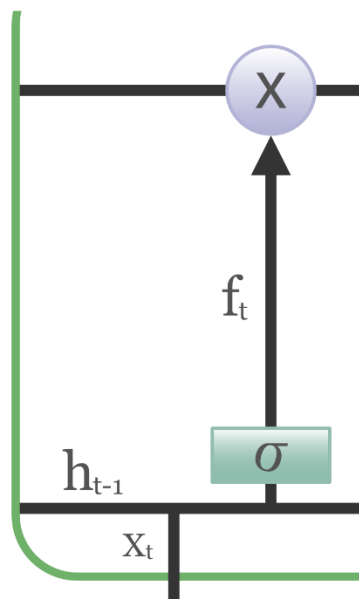
U_i : bobot untuk *hidden state* pada waktu (t-1)

h_{t-1} : *hidden state* pada waktu (t-1)

b_i : bias

Rumus ini menggambarkan bagaimana *input gate* menggunakan bobot, *input*, *hidden state* sebelumnya, dan bias untuk memutuskan informasi mana yang akan disimpan dalam *output*.

Input gate dalam LSTM memiliki peran penting dalam menghapus informasi yang tidak lagi relevan dari sel memori. Fungsi aktivasi sigmoid digunakan oleh *input gate* untuk menentukan informasi mana yang akan dihapus. *Output* dari *input gate* memiliki rentang antara 0 dan 1, di mana nilai 0 mengindikasikan bahwa sel memori sepenuhnya dilupakan, sementara nilai 1 menunjukkan bahwa sel memori sepenuhnya dipertahankan. Dengan cara ini, *input gate* membantu LSTM dalam memutuskan sejauh mana informasi lama harus diabaikan, memungkinkan penyimpanan informasi yang relevan dan penting dalam *output*. Alur informasi *input gate* dapat dilihat pada gambar 2.4 dibawah ini:



Gambar 2. 4 Alur Informasi *Input gate*

Sumber : [geeksforgeeks.org](https://www.geeksforgeeks.org/lstm-cell/)

Rumus dari *Input gate* dapat dilihat pada persamaan 3 berikut:

$$f_t = \sigma(W_f \cdot x_t + U_f \cdot h_{t-1} + b_f) \quad (3)$$

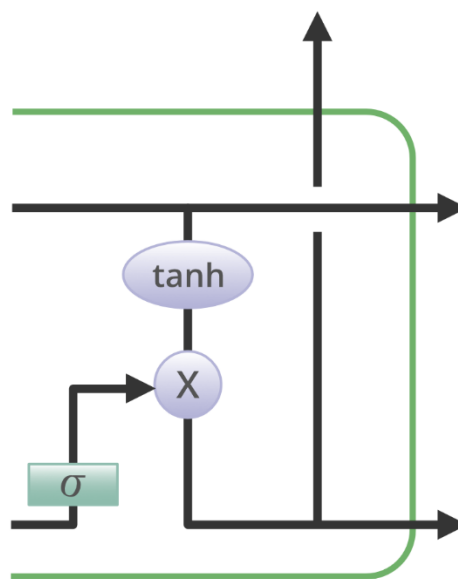
Dimana :

f_t : output dari *input gate*.

- σ : fungsi aktivasi sigmoid.
 W_f : bobot untuk nilai *input* pada waktu (t)
 x_t : *input* pada waktu (t)
 U_f : bobot untuk *hidden state* pada waktu (t-1)
 H_{t-1} : *hidden state* pada waktu (t-1)
 b_f : bias

Rumus ini mencerminkan cara *input gate* menggunakan bobot, *input*, *hidden state* sebelumnya, dan bias untuk menentukan informasi yang harus dihapus dari dalam *output*. Ini memberikan kemampuan pada LSTM untuk mengatur aliran informasi yang masuk ke dalam *output*, memastikan bahwa hanya informasi yang relevan yang dipertahankan dan digunakan dalam langkah-langkah selanjutnya dari prosesnya (Sanjaya & Heksaputra, 2020).

Input gate, atau gerbang keluar, berperan kunci dalam menghasilkan *output* dari LSTM. Fungsinya adalah mengatur seberapa banyak informasi yang akan dihasilkan dari sel memori pada suatu waktu tertentu. *Input gate* menggunakan fungsi sigmoid untuk mengontrol dampak nilai dalam sel memori terhadap *output* dari LSTM pada saat tersebut. Dengan demikian, *input gate* memungkinkan LSTM untuk mengatur keluaran sesuai dengan relevansi informasi dalam sel memori pada titik waktu yang bersangkutan. Alur informasi *input gate* dapat dilihat seperti gambar 2.5 dibawah ini:



Gambar 2. 5 Alur Informasi *Input gate*

Sumber : [geeksforgeeks.org](https://www.geeksforgeeks.org)

Output Fungsi *output* gate dalam struktur Long Short-Term Memory (LSTM) adalah untuk mengontrol seberapa banyak informasi dari sel memori yang akan digunakan sebagai *output* dari LSTM pada setiap langkah waktu (time step) dalam urutan data. *Output* gate menggunakan fungsi sigmoid untuk menghasilkan nilai antara 0 dan 1 . Nilai *output* gate tersebut akan menentukan seberapa banyak informasi dari sel memori yang akan diteruskan ke lapisan berikutnya dari jaringan saraf atau digunakan sebagai *output* dari LSTM. Ketika nilai *output* gate mendekati 1, hal ini menunjukkan bahwa informasi dari sel memori tersebut dianggap penting dan harus disertakan dalam *output*. Sebaliknya, ketika nilai *output* gate mendekati 0, informasi dari sel memori tersebut dianggap tidak relevan dan tidak harus disertakan dalam *output*. Dengan demikian, *output* gate memungkinkan LSTM untuk secara selektif mengatur aliran informasi keluar dari sel memori, membantu dalam menghasilkan *output* yang relevan dan berdasarkan konteks dari data deret waktu yang dianalisis.. Rumus dari *Input* gate dapat dilihat pada persamaan 4 berikut:

$$ot = \sigma(W_o \cdot x_t + U_o \cdot h_{t-1} + b_o) \quad (4)$$

Dimana:

ot : *output* dari *input* gate.

σ : fungsi aktivasi sigmoid

W_o : bobot untuk nilai *input* pada waktu (t)

x_t : *input* pada waktu (t).

U_o : bobot untuk *hidden state* pada waktu (t-1).

h_{t-1} : *hidden state* pada waktu (t-1)

b_o : basis

2.12 Normalisasi Data

Metode normalisasi data adalah proses yang bertujuan untuk menyelaraskan rentang nilai beberapa variabel sehingga tidak ada yang terlalu besar atau terlalu kecil, yang mempermudah analisis statistik. Ada beberapa metode normalisasi data yang umum digunakan, termasuk Min-Max normalization, Z-score Normalization, dan Decimal Scaling Normalization .

Salah satu metode yang sering digunakan adalah normalisasi Decimal Scaling, yaitu metode transformasi data yang menormalkan rentang nilai setiap atribut ke dalam skala tertentu dengan menggeser nilai desimal dari data ke arah yang diinginkan. Pendekatan ini bertujuan untuk menghasilkan data yang lebih konsisten dan seimbang. Penelitian terdahulu menunjukkan bahwa perbedaan rentang nilai antara atribut dalam proses transformasi data dapat menyebabkan atribut dengan nilai yang jauh lebih kecil menjadi tidak berfungsi sebagaimana mestinya. Oleh karena itu, data yang telah dinormalisasi menggunakan metode Decimal Scaling menunjukkan hasil yang lebih baik secara signifikan dibandingkan dengan klasifikasi yang tidak menggunakan transformasi data (Kusnaldi et al., 2022).

Dalam konteks penelitian ini, normalisasi data sangat penting karena model Long Short-Term Memory (LSTM) sangat sensitif terhadap skala dan distribusi data input. Dengan menggunakan metode Decimal Scaling, kami dapat memastikan bahwa setiap atribut dalam dataset memiliki rentang nilai yang sama, sehingga meningkatkan kinerja model dalam memprediksi produksi minyak sawit. Normalisasi data membantu dalam mempercepat konvergensi selama pelatihan model dan mengurangi risiko overfitting, yang pada akhirnya menghasilkan prediksi yang lebih akurat dan andal.

Rumus normalisasi dapat dilihat pada persamaan 5 dibawah ini:

$$XNorm = \frac{X - Xmin}{Xmax - Xmin} \quad (5)$$

Keterangan:

X = nilai asli dalam *dataset*

$XNorm$ = nilai yang dinormalisasi

$Xmin$ = nilai paling kecil dalam *dataset*

$Xmax$ = nilai paling besar dalam *dataset*.

BAB 3

ANALISIS DAN PERANCANGAN

3.1 Analisis

Analisis adalah tahap penting dalam pengembangan sistem yang bertujuan untuk menghasilkan rancangan awal dari perangkat lunak. Melalui analisis, kita mengidentifikasi dan memahami kebutuhan serta masalah yang ingin dipecahkan oleh sistem yang akan dibangun. Hasil analisis ini menjadi panduan dalam perancangan dan pengembangan sistem, memastikan bahwa sistem yang dikembangkan dapat memenuhi kebutuhan pengguna dengan baik dan memberikan solusi yang efisien terhadap masalah yang dihadapi.

3.1.1 Analisis Masalah

Masalah utama yang perlu diatasi adalah fluktuasi jumlah tandan buah sawit yang masuk ke pabrik, yang menyebabkan ketidakpastian dalam produksi minyak kelapa sawit mentah (CPO) di PT. Agro Indah. Fluktuasi ini dapat mengakibatkan gangguan dalam perencanaan produksi, manajemen stok, serta cashflow perusahaan. Selain itu, ketidakpastian dalam produksi CPO juga berdampak pada waktu perawatan peralatan dan mesin di pabrik. Selama ini, proses peramalan produksi masih dilakukan secara manual dan berdasarkan pengalaman, sehingga rentan terhadap *human error* dan kurangnya akurasi. Oleh karena itu, diperlukan solusi yang lebih efisien dalam peramalan produksi CPO, yang dapat memberikan hasil yang lebih akurat dan dapat diandalkan. Dengan adanya prediksi yang baik, PT. Agro Indah diharapkan dapat meningkatkan efisiensi operasional PT. Agro Indah, memperbaiki perencanaan produksi, mengelola stok secara lebih akurat, serta meningkatkan kepastian dalam cashflow perusahaan.

3.1.2 Analisis Kebutuhan

Analisis kebutuhan fungsional dan *non-fungsional* adalah dua komponen utama analisis kebutuhan. Kedua hal tersebut memiliki prioritas dan kebutuhan yang berbeda. Kebutuhan fungsional mengacu pada fungsi-fungsi yang harus dimiliki oleh model yang sedang dibangun atau dirancang, yang sesuai dengan tujuan yang ingin dicapai. Sedangkan kebutuhan *non-fungsional* menjelaskan tentang aspek tambahan model, seperti batasan, performa, tampilan, dan dokumentasi, yang bertujuan untuk mendukung model agar lebih baik dan stabil.

3.1.2.1 Analisis Fungsional

Kebutuhan Fungsional adalah bagian yang ditujukan sebagai tujuan dari pembuatan sistem dan kebutuhan bisa dikerjakan oleh sistem. Kebutuhan Fungsional mengenai proses yang terjadi dan yang dilakukan oleh sistem yang membangun di penelitian ini.

Kebutuhan fungsional dalam penelitian ini adalah:

1. Model harus mampu melakukan prediksi produksi minyak sawit untuk jangka waktu tiga tahun ke depan.
2. Model harus dapat mengelola dan memproses data historis produksi minyak sawit.
3. Model harus memberikan prediksi produksi minyak tahun untuk setiap hari dalam jangka waktu yang ditentukan.

3.1.2.2 Analisis *Non Fungsional*

Analisis *Non Fungsional* merupakan kebutuhan yang menjelaskan aspek tambahan dari model, seperti batasan, performa, tampilan, dan dokumentasi. Prioritasnya adalah untuk mendukung stabilitas, kinerja, dan kualitas model secara keseluruhan.

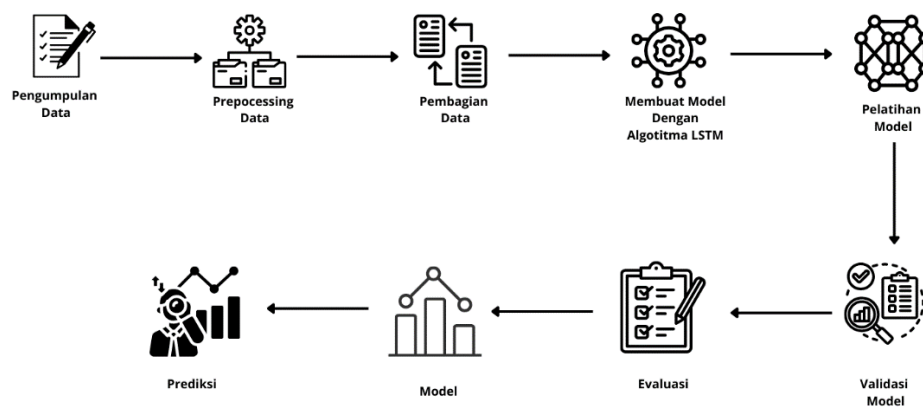
Kebutuhan *Non Fungsional* dalam penelitian ini adalah:

1. Model harus mampu melakukan prediksi dengan tingkat akurasi yang baik.

2. Waktu respon model terhadap permintaan harus kurang dari 1 detik
3. Antarmuka model harus dirancang dengan intuitif dan mudah dimengerti.
4. Model harus memiliki dokumentasi yang lengkap dan terstruktur.

3.2 Arsitektur Umum

Arsitektur sistem pada penelitian ini dapat diperinci seperti gambar 3.1 dibawah ini:



Gambar 3. 1 Arsitektur Sistem Penelitian

Proses pengembangan model prediksi produksi minyak sawit dimulai dengan pengumpulan data historis produksi dari tahun 2019 hingga 2023. Data yang terkumpul kemudian akan melalui tahap *Preprocessing*, di mana data akan dibersihkan dari nilai-nilai yang hilang atau tidak valid, serta dinormalisasi jika diperlukan. Setelah data bersih, langkah selanjutnya adalah membagi data menjadi dua bagian, yaitu data latih dan data pengujian. Selanjutnya, model akan dibuat menggunakan algoritma *Long Short-Term Memory* (LSTM), yang merupakan jenis jaringan saraf tiruan khusus untuk data deret waktu. Model akan dilatih menggunakan data latih, dengan menggunakan *Root Mean Square Error* (RMSE) sebagai fungsi kerugian (*loss function*) untuk mengukur seberapa baik model dalam memprediksi produksi minyak sawit.

Setelah melalui tahap pelatihan, model akan diuji menggunakan data pengujian yang sebelumnya tidak pernah dilihat oleh model. Hasil dari pengujian tersebut akan digunakan untuk menghitung nilai RMSE, yang akan memberikan gambaran seberapa dekat prediksi model dengan nilai aktual produksi. Jika nilai RMSE belum memenuhi

target yang ditetapkan, model akan dievaluasi kembali, dan pertimbangan akan dilakukan untuk mengoptimalkan model dengan menyesuaikan parameter atau struktur LSTM. Setelah berhasil melalui evaluasi, model siap digunakan untuk melakukan prediksi produksi minyak sawit untuk periode waktu yang ditentukan. Hasil prediksi ini akan memberikan wawasan yang penting bagi perencanaan dan pengambilan keputusan di sektor produksi minyak sawit, membantu perusahaan untuk mengantisipasi fluktuasi produksi dan mengoptimalkan kinerja operasional.

3.3 Pengumpulan *Dataset*

Pengumpulan data adalah proses penghimpunan atau pengambilan data dari berbagai sumber yang relevan untuk digunakan dalam analisis atau penelitian. Data yang dikumpulkan oleh penulis pada penelitian ini dilakukan dengan cara mengumpulkan data historis produksi minyak sawit. Data ini termasuk total Tandan Buah Segar (TBS) yang diterima dari kebun dan luar kebun, total TBS yang diolah dari kebun dan luar kebun, serta jumlah minyak sawit mentah (CPO) yang dihasilkan setiap harinya. Data historis ini diperoleh dari perusahaan PT. Agro Indah serta lembaga atau instansi terkait yang memiliki data terkait produksi minyak sawit. Proses pengumpulan data dilakukan dengan cara wawancara dan izin kepada PT. Agro Indah untuk mendapatkan data produksi minyak sawit mereka. Selain itu, dilakukan juga pengumpulan data dari berbagai laporan dan dokumen resmi yang dikeluarkan oleh perusahaan. Data tersebut mencakup periode tahun 2019 hingga 2023 perharinya dimana jumlah data sebanyak 1826 data untuk memperoleh rentang waktu yang cukup luas untuk analisis. Dalam proses pengumpulan data, juga diperhatikan aspek kualitas data seperti kebenaran, ketepatan, dan kelengkapan. Data-data tersebut kemudian dianalisis dan dimasukkan ke dalam format yang sesuai untuk pengolahan lebih lanjut. Penting untuk memastikan bahwa data yang dikumpulkan memiliki integritas yang baik dan dapat diandalkan untuk digunakan dalam pembuatan model prediksi produksi minyak sawit. Pada tabel 3.1 merupakan contoh dataset pada penelitian ini:

Tabel 3. 1 Contoh Dataset

Date	Base	TOTAL_TERIMA	TOTAL_OLAH	JUMLAH_CPO	Kebun_Terima	Luar Kebun_Ter	Kebun_Olah	Luar Kebun_Olah
1/1/2019	772	773	770	154	537	236	536	234
1/2/2019	852	853	850	170	521	332	520	330
1/3/2019	818	820	816	163,2	462	358	460	356
1/4/2019	835	837	833	166,6	487	350	485	348
1/5/2019	788	790	785	157	423	367	420	365
1/6/2019	874	876	871	174,2	528	348	526	345
1/7/2019	771	778	763	152,6	411	367	410	353
1/8/2019	727	729	725	145	409	320	407	318
1/9/2019	879	880	877	175,4	504	376	502	375
1/10/2019	752	754	750	150	512	242	510	240
1/11/2019	816	818	814	162,8	476	342	474	340
1/12/2019	735	737	732	146,4	426	311	424	308
1/13/2019	738	740	736	147,2	462	278	460	276
1/14/2019	740	742	738	147,6	422	320	420	318
1/15/2019	716	718	714	142,8	462	256	460	254
1/16/2019	754	756	751	150,2	513	243	511	240
1/17/2019	699	700	697	139,4	435	265	434	263
1/18/2019	710	712	707	141,4	481	231	479	228
1/19/2019	747	749	744	148,8	503	246	500	244
1/20/2019	761	764	758	151,6	513	251	510	248

3.4 Preprocessing Data

Preprocessing data merupakan rangkaian tindakan yang diterapkan pada data mentah (*raw data*) untuk membersihkannya, mengubah formatnya, dan menyiapkannya agar dapat digunakan dalam analisis atau pemodelan. Hal ini bertujuan untuk meningkatkan kualitas data, menghilangkan noise atau gangguan, dan meningkatkan kinerja algoritma yang akan digunakan pada tahap selanjutnya.

3.5 Normalisasi Data

Normalisasi data adalah proses untuk mengubah rentang nilai dari setiap fitur dalam *dataset* sehingga semua fitur memiliki skala yang seragam. Hal ini dilakukan untuk mencegah beberapa fitur yang memiliki rentang nilai yang jauh lebih besar daripada fitur lainnya mendominasi proses pemodelan. Dengan normalisasi, setiap nilai dalam setiap fitur akan diubah sedemikian rupa sehingga berada dalam rentang antara 0 dan 1. Dimana pada penelitian ini menggunakan *Min-Max Scaling* untuk menormalisasi data.

Pada Tabel 3.2 dibawah merupakan contoh data sebelum dan sesudah normalisasi:

Tabel 3. 2 Contoh data asli sebelum dinormalisasi

Data Asli
773

853
820
837
790

Langkah – Langkah perhitungannya:

1. Tentukan nilai minimum dan nilai maksimum dalam dataset

$$X_{Min} = 773$$

$$X_{Max} = 853$$

2. Gunakan rumus MinMaxScaling dan masukkan nilai kedalam rumus tersebut

$$X_{Norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (6)$$

a. $X = 773$

$$X' = \frac{773 - 773}{853 - 773} = \frac{0}{80} = 0$$

b. $X = 853$

$$X' = \frac{853 - 773}{853 - 773} = \frac{80}{80} = 1$$

c. $X = 820$

$$X' = \frac{820 - 773}{853 - 773} = \frac{47}{80} = 0.587$$

d. $X = 837$

$$X' = \frac{837 - 773}{853 - 773} = \frac{64}{80} = 0.8$$

e. $X = 790$

$$X' = \frac{790 - 773}{853 - 773} = \frac{17}{80} = 0.212$$

Dari perhitungan tersebut didapatkan hasil seperti tabel 3.3 dibawah ini :

Tabel 3. 3 Contoh nilai asli dan nilai normalisasi

Nilai Asli	Nilai Normalisasi
773	0
853	1
820	0.587

837	0.8
790	0.212

3.6 Pembagian Data Set

Dalam penelitian ini, data uji digunakan untuk mewakili data produksi minyak sawit pada tahun 2023, sementara data latih digunakan untuk mewakili data produksi pada tahun-tahun sebelumnya, yakni dari tahun 2019 hingga 2022. Data latih merupakan kumpulan data yang mencakup semua informasi historis yang diperlukan untuk melatih model, seperti jumlah tandan buah sawit yang diterima, jumlah tandan buah sawit yang diolah, dan produksi CPO. Dengan melatih model menggunakan data latih ini, model dapat belajar dari pola-pola yang terdapat dalam data historis untuk membuat prediksi tentang produksi minyak di masa depan.

Sementara itu, data uji berperan sebagai set data independen yang digunakan untuk menguji performa model. Data uji mencakup informasi produksi minyak sawit selama tahun 2023 yang tidak pernah dilihat oleh model sebelumnya. Penggunaan data uji yang terpisah ini memungkinkan kita untuk mengevaluasi seberapa baik model mampu melakukan generalisasi terhadap data yang belum pernah dilihat sebelumnya. Dengan kata lain, kita dapat mengukur sejauh mana model mampu mengikuti pola umum dari data latih dan menerapkannya pada situasi dunia nyata yang baru. Dengan menggunakan dua set data yang terpisah ini, kita dapat memastikan bahwa model memiliki kemampuan yang baik untuk memprediksi produksi minyak di masa depan.

3.7 Implementasi LSTM

Dalam penelitian ini, model LSTM (*Long Short-Term Memory*) digunakan untuk memprediksi produksi minyak kelapa sawit. Data yang digunakan mencakup total tandan buah sawit (TBS) yang diterima dari kebun dan luar kebun, total tandan buah sawit yang diolah dari kebun dan luar kebun, serta jumlah minyak kelapa sawit (CPO) setiap harinya dari tahun 2019 hingga 2023. Sebelum membangun model, data diproses menggunakan *NumPy* untuk manipulasi data numerik dan *Pandas* untuk mengelola *dataset*. Kemudian, data dinormalisasi menggunakan *MinMaxScaler* dari *Scikit-learn*.

Proses pembuatan dan pelatihan model dilakukan dengan menggunakan *Keras*, yang menyediakan berbagai fungsi untuk membangun dan mengevaluasi model *neural network*. Model LSTM dikonfigurasi dengan beberapa lapisan LSTM yang dilengkapi dengan *dropout* untuk mencegah *overfitting*, diikuti oleh lapisan-lapisan *dense* untuk *output*. Evaluasi dilakukan menggunakan metrik RMSE (*Root Mean Squared Error*) untuk mengevaluasi akurasi prediksi model terhadap data uji. Dengan memanfaatkan berbagai pustaka ini, penulis dapat mengimplementasikan model LSTM dan menganalisis hasil prediksi dengan efisien.

3.8 Penjelasan Long Short Term Memory

Long Short-Term Memory (LSTM) adalah jenis arsitektur jaringan saraf tiruan yang termasuk dalam kategori *Recurrent Neural Networks* (RNN). LSTM dirancang untuk mengatasi masalah yang dialami oleh RNN konvensional, yaitu hilangnya informasi penting saat melibatkan data sekuensial yang panjang. Dalam konteks prediksi produksi minyak sawit, LSTM sangat berguna karena mampu memanfaatkan informasi historis dalam jangka panjang untuk menghasilkan prediksi yang lebih akurat.

Struktur dan Mekanisme LSTM:

1. Sel Memori (*Memory Cell*): Sel memori adalah inti dari LSTM yang menyimpan informasi dalam jangka waktu yang panjang. Ini memungkinkan LSTM untuk mempertahankan konteks historis yang relevan selama proses prediksi.
2. Gerbang *Input* (*Input Gate*): Gerbang ini mengatur seberapa banyak informasi dari *input* saat ini yang akan disimpan dalam sel memori. Ini membantu LSTM untuk memutuskan informasi mana yang penting dan mana yang tidak.
3. Gerbang Lupa (*Forget Gate*): Gerbang ini menentukan informasi apa yang perlu dihapus dari sel memori. Dengan demikian, LSTM dapat menghilangkan informasi yang tidak lagi relevan untuk prediksi di masa mendatang.
4. Gerbang *Output* (*Output Gate*): Gerbang ini mengontrol informasi mana dari sel memori yang akan digunakan untuk menghasilkan *output* pada waktu tertentu. Ini membantu LSTM dalam menghasilkan prediksi yang didasarkan pada informasi historis yang disimpan dalam sel memori.

Pada penelitian prediksi produksi minyak sawit, langkah-langkah implementasi LSTM meliputi beberapa tahap penting sebagai berikut:

1. *Preprocessing* Data:

Normalisasi Data

Data historis produksi minyak sawit dinormalisasi menggunakan *MinMaxScaler* agar semua fitur berada dalam rentang nilai yang sama, biasanya antara 0 sampai 1. Normalisasi ini membantu mempercepat konvergensi model dan mencegah dominasi fitur dengan skala yang lebih besar.

Pembentukan Data *Input* dan *Output*

Data pelatihan dibentuk menjadi sekuens waktu yang panjangnya ditentukan. Setiap sekuens digunakan sebagai *input* untuk memprediksi nilai produksi pada hari berikutnya.

2. Pembentukan Model LSTM:

Lapisan LSTM

Model dibangun dengan beberapa lapisan LSTM untuk menangkap pola temporal dalam data produksi minyak sawit. Setiap lapisan terdiri dari sejumlah unit memori dan dilengkapi dengan lapisan *dropout* untuk mengurangi *overfitting*.

Lapisan *Dense*

Lapisan terakhir adalah lapisan *dense* dengan satu unit yang menghasilkan prediksi produksi minyak sawit.

3. Kompilasi Model:

Model LSTM dikompilasi menggunakan *optimizer* seperti '*RMSprop*' dan fungsi kerugian '*mean_squared_error*'. *Optimizer* ini membantu dalam memperbarui bobot jaringan untuk meminimalkan kesalahan prediksi.

4. Pelatihan Model:

Model dilatih menggunakan data pelatihan menggunakan 50,100 *epoch* dan 32,64 *batch size*. Dari empat kali percobaan maka didapat lah hasil bahea 100 *epoch* dan 32 *batch size* menghasilkan error terendah. Pelatihan ini membantu model untuk

belajar pola-pola dalam data historis dan menyesuaikan bobotnya untuk meningkatkan akurasi prediksi.

5. Evaluasi Model:

Setelah pelatihan, model dievaluasi menggunakan metrik seperti *Root Mean Squared Error* (RMSE) guna mengukur Tingkat akurasi model dengan membandingkan nilai prediksi pada model dan nilai sebenarnya yang ada pada dataset. RMSE memberikan gambaran tentang rata-rata kesalahan prediksi dan membantu dalam menilai kinerja model.

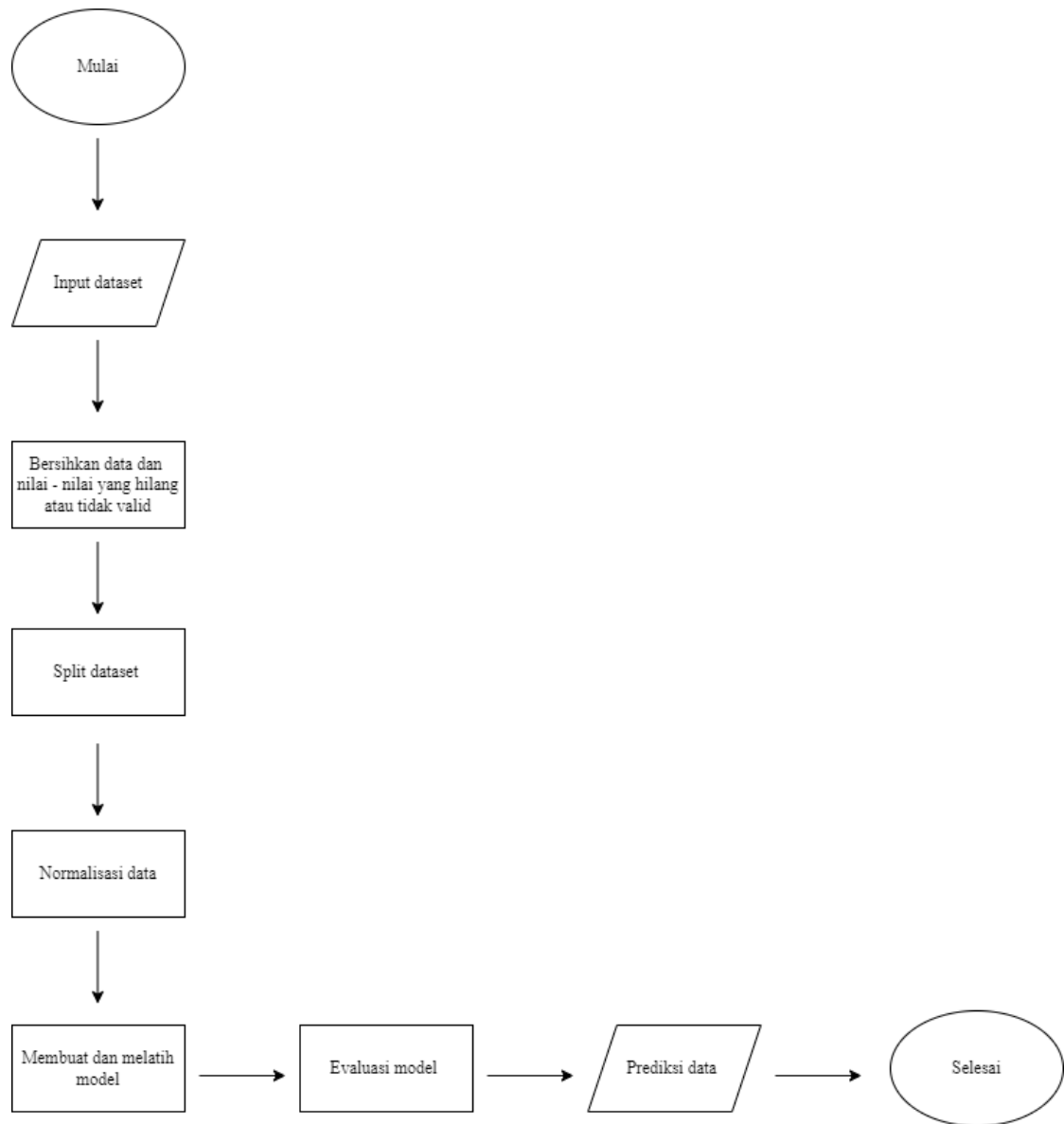
6. Prediksi:

Data uji yang tidak terlihat sebelumnya diproses dengan cara yang sama seperti data pelatihan, dan digunakan untuk menghasilkan prediksi. Prediksi ini kemudian dinormalisasi kembali ke skala aslinya untuk interpretasi yang lebih mudah

3.8 Perancangan Sistem

3.8.1 *Flowchart*

Perancangan sistem dalam penelitian ini akan direpresentasikan dalam bentuk diagram alir atau *Flowchart*. *Flowchart* ini bertujuan untuk memberikan gambaran visual mengenai alur kerja atau proses yang sedang diidentifikasi atau dirancang seperti gambar 3.2 dibawah ini:



Gambar 3. 2 *Flowchart* Sistem

BAB 4

HASIL DAN PEMBAHASAN

Bab implementasi dan pengujian dalam konteks prediksi produksi minyak sawit menggunakan metode *Long Short-Term Memory* membahas langkah-langkah yang dilakukan untuk menerapkan model *Long Short-Term Memory* dalam memprediksi.

4.1 Implementasi Sistem

4.1.1 Spesifikasi Perangkat Keras

Spesifikasi perangkat lunak adalah dokumen yang berisi rincian tentang kebutuhan dan fungsionalitas yang diharapkan dari sebuah perangkat lunak. Spesifikasi ini menjadi landasan bagi pengembang perangkat lunak untuk merancang, mengembangkan, dan menguji produk yang memenuhi kebutuhan pengguna dengan baik. Untuk dapat menerapkan dan menguji implementasi *Long Short-Term Memory* dalam memprediksi jumlah produksi minyak sawit, diperlukan perangkat keras dengan spesifikasi sebagai berikut.:

1. *Processor Intel® Core™ i5-8250U Processor 1.6 GHz*
2. *Memory RAM 8 GB*
3. *SSD dengan kapasitas 512 GB*
4. *NVIDIA® GeForce® MX350 2GB GDDR5 Intel® UHD Graphics*

Untuk mendukung persyaratan perangkat keras yang diperlukan, penulis juga memanfaatkan layanan cloud gratis dari Google yang disebut *Google Colab*. Dalam *Google Colab*.

4.1.2 Spesifikasi Perangkat Lunak

Pada penelitian ini penulis menggunakan beberapa spesifikasi perangkat lunak yang akan dijelaskan dibawah ini:

1. Sistem Operasi *Windows 11 Home 64 bit operating system*
2. *Google Colab*
3. *Python 3*
4. *Library: numpy, matplotlib, pandas, math, sklearn, SGD*

4.2 Pengumpulan Dataset

Tahap pertama adalah mengumpulkan *dataset*, dimana penulis mengumpulkan data perharinya selama lima tahun terakhir yaitu periode 2019 – 2023. *Dataset* pada penelitian ini dapat dilihat pada tabel 4.1 dibawah :

Tabel 4. 1 Dataset

Date	Base	TOTAL_TERIMA	TOTAL_OLAH	JUMLAH_CPO	Kebun_Terima	Luar Kebun_Ter	Kebun_Olah	Luar Kebun_Olah
1/1/2019	772	773	770	154	537	236	536	234
1/2/2019	852	853	850	170	521	332	520	330
1/3/2019	818	820	816	163,2	462	358	460	356
1/4/2019	835	837	833	166,6	487	350	485	348
1/5/2019	788	790	785	157	423	367	420	365
1/6/2019	874	876	871	174,2	528	348	526	345
1/7/2019	771	778	763	152,6	411	367	410	353
1/8/2019	727	729	725	145	409	320	407	318
1/9/2019	879	880	877	175,4	504	376	502	375
1/10/2019	752	754	750	150	512	242	510	240
1/11/2019	816	818	814	162,8	476	342	474	340
1/12/2019	735	737	732	146,4	426	311	424	308
1/13/2019	738	740	736	147,2	462	278	460	276
1/14/2019	740	742	738	147,6	422	320	420	318
1/15/2019	716	718	714	142,8	462	256	460	254
1/16/2019	754	756	751	150,2	513	243	511	240
1/17/2019	699	700	697	139,4	435	265	434	263
1/18/2019	710	712	707	141,4	481	231	479	228
1/19/2019	747	749	744	148,8	503	246	500	244
1/20/2019	761	764	758	151,6	513	251	510	248

4.3 Preprocessing Data

4.3.1 Menghapus nilai kosong

Tahap menghapus nilai kosong dalam algoritma *Long Short-Term Memory* sangat penting untuk mengetahui kualitas data yang akan digunakan untuk melatih dan menguji model. Jika terdapat kolom dengan jumlah nilai kosong yang tinggi, hal ini dapat menjadi masalah karena nilai yang hilang dapat mengurangi akurasi dan kinerja model. Oleh karena itu, langkah selanjutnya adalah mengambil tindakan yang sesuai, seperti menghapus baris atau kolom yang mengandung nilai kosong, menggantikan nilai-nilai kosong dengan nilai lain, atau bahkan mencari data tambahan untuk mengisi nilai kosong tersebut. dengan menjaga konsistensi dalam data dan memastikan bahwa nilai kosong ditangani dengan benar, kita dapat memastikan bahwa *dataset* yang digunakan

untuk melatih model LSTM bersih dan siap digunakan. Pada *dataset* penulis tidak ada ditemukannya nilai yang kosong sehingga tidak perlu untuk menghapus nilai tersebut. Gambar 4.1 hasil menghapus nilai kosong tertera dibawah ini :

```
Total missing values in each column:
Date          0
Base          0
TOTAL_TERIMA  0
TOTAL_OLAH    0
JUMLAH_CPO    0
Kebun_Terima  0
Luar Kebun_Terima  0
Kebun_Olah    0
Luar Kebun_Olah  0
dtype: int64
```

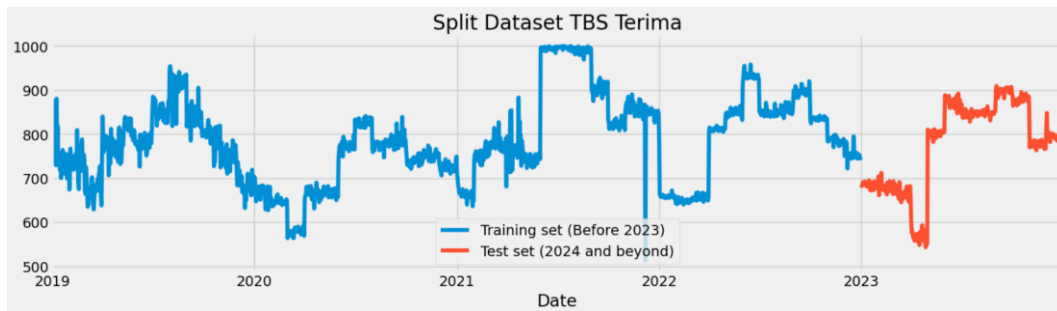
Gambar 4. 1 Hasil menghapus nilai kosong

4.4 Pembagian Data

Pada langkah ini, dataset produksi minyak kelapa sawit dibagi menjadi dua bagian: *training set* dan *test set*. *Training set* terdiri dari data produksi minyak kelapa sawit dari tahun-tahun sebelum 2023, yang digunakan untuk melatih model prediksi. Sedangkan *test set* terdiri dari data produksi mulai tahun 2023, yang akan digunakan untuk menguji seberapa baik model dapat memprediksi data baru yang belum pernah dilihat sebelumnya. Dengan pembagian ini, kita dapat memastikan bahwa model yang kita latih telah mampu menangkap pola-pola yang ada dalam data historis, serta mengevaluasi kinerja model secara objektif menggunakan data yang independen. Pada penelitian ini di prediksi jumlah tbs terima, tbs olah dan jumlah cpo, maka splitnya juga ada tiga . *split dataset* dapat dilihat pada gambar 4.2, 4.3, 4.4, 4.5, 4.6 dan 4.7 dibawah ini :

```
#Split TBS Terima
training_set = dataset.iloc[:len(dataset.loc[:'2022']), 2:3].values
test_set = dataset.iloc[len(dataset.loc[:'2023']):, 2:3].values
```

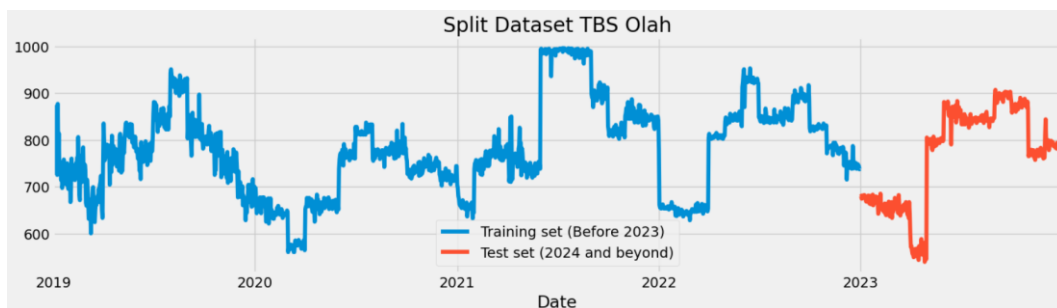
Gambar 4. 2 Kodingan Split Dataset TBS Terima



Gambar 4. 3 Visualisasi Hasil Split Dataset TBS Terima

```
#split dataset TBS OLAH
training_set2 = dataset.iloc[:len(dataset.loc[:'2022']), 3:4].values
test_set2 = dataset.iloc[len(dataset.loc[:'2023']):, 3:4].values
```

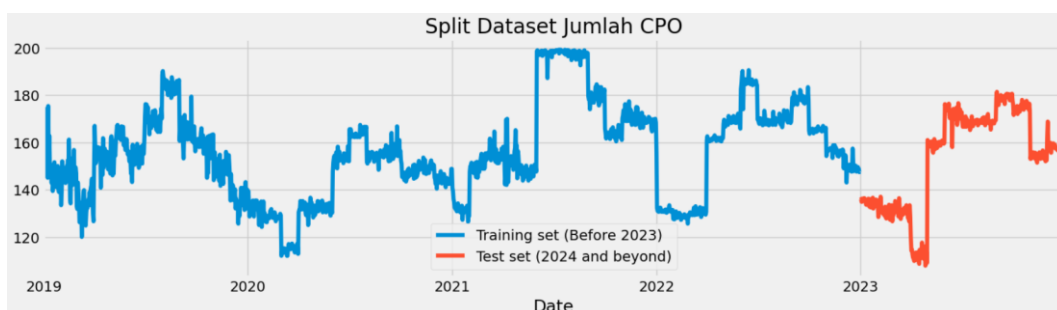
Gambar 4. 4 Kodingan Split Dataset TBS Olah



Gambar 4. 5 Visualisasi Hasil Split Dataset TBS Olah

```
#split dataset JUMLAH CPO
training_set3 = dataset.iloc[:len(dataset.loc[:'2022']), 4:5].values
test_set3 = dataset.iloc[len(dataset.loc[:'2023']):, 4:5].values
```

Gambar 4. 6 Kodingan Split Dataset Jumlah CPO



Gambar 4. 7 Visualisasi Hasil Split Dataset Jumlah CPO

4.5 Konversi *Dataset*

Pada langkah ini, dilakukan konversi kolom 'tanggal' dalam *dataset* menjadi tipe data *datetime* menggunakan fungsi `pd.to_datetime()`. Tipe data *datetime* sangat berguna karena memungkinkan kita untuk dengan mudah bekerja dengan data tanggal dan waktu dalam analisis data. Setelah itu, kolom 'Date' dijadikan sebagai indeks *DataFrame* dengan menggunakan fungsi `set_index('Date', inplace=True)`. Dengan mengatur kolom 'Date' sebagai indeks, kita dapat dengan mudah mengakses data berdasarkan tanggal, yang akan sangat berguna saat melakukan analisis data berbasis waktu, seperti peramalan atau analisis tren. Konversi *dataset* dapat dilihat pada gambar 4.8 dibawah ini:

```
# Untuk mengonversi kolom 'tanggal' menjadi tipe data datetime
dataset['Date'] = pd.to_datetime(dataset['Date'])

# mengatur kolom 'Date' sebagai indeks DataFrame
dataset.set_index('Date', inplace=True)
```

Gambar 4. 8 Konversi *dataset*

4.6 Normalisasi

Normalisasi dalam LSTM adalah untuk mengubah rentang nilai-nilai *input* menjadi rentang yang lebih kecil, umumnya antara 0 sampai 1, agar lebih sesuai dengan karakteristik aktivasi fungsi sigmoid yang umum digunakan dalam LSTM. Aktivasi sigmoid beroperasi lebih efisien ketika *input* berada dalam rentang yang dekat dengan 0 sampai 1. Pada langkah ini, dilakukan normalisasi *dataset* menggunakan *Min-Max Scaling*. Contoh *dataset* yang dinormalisasi dapat dilihat pada tabel 4.1, 4.2 dan 4.3 dibawah ini :

Tabel 4. 2 Contoh dataset TBS Terima yang dinormalisasi

<i>Dataset</i> TBS Terima sebelum di normalisasi	<i>Dataset</i> TBS Terima Sesudah di normalisasi
773	0.562
853	0.875
820	0.746
837	0.812
790	0.628
876	0.964

778	0.582
729	0.390
880	0.980
754	0.488

Tabel 4. 3 Contoh dataset TBS Olah yang dinormalisasi

<i>Dataset TBS Olah sebelum di normalisasi</i>	<i>Dataset TBS Olah Sesudah di normalisasi</i>
770	0.604
850	0.889
816	0.768
833	0.829
785	0.658
871	0.964
763	0.580
725	0.444
877	0.985
750	0.533

Tabel 4. 4 Contoh dataset Jumlah CPO yang dinormalisasi

<i>Dataset Jumlah CPO sebelum di normalisasi</i>	<i>Dataset Jumlah CPO Sesudah di normalisasi</i>
154	0.605
170	0.889
163.2	0.764
166.6	0.832
157	0.658
174.2	0.942
152.6	0.580
145	0.444
175.4	0.948
150	0.513

4.7 Pelatihan Model LSTM

Tahap ini adalah tahap persiapan pelatihan model *Long Short-Term Memory* (LSTM), yang berfokus pada pembentukan *dataset* pelatihan yang sesuai untuk model LSTM. Proses ini bertujuan untuk membuat data *input* (X_{train}) dan target *output*

(*y_train*) yang digunakan oleh model LSTM untuk belajar dan memahami pola urutan waktu dalam data yang telah dinormalisasi.

Fungsinya adalah sebagai berikut:

1. Pembuatan jendela waktu (*Windowing*):

Dalam langkah ini, kita membentuk jendela waktu sepanjang 60 hari untuk setiap titik data yang akan diprediksi. Ini berarti bahwa untuk setiap titik data yang ingin diprediksi, model menggunakan 60 nilai sebelumnya sebagai *input*. Baris `X_train.append(training_set_scaled[i-60:i,0])` mengumpulkan 60 nilai sebelumnya dari titik data saat ini dan menambahkannya ke dalam *X_train* sebagai satu set *input*. Baris `y_train.append(training_set_scaled[i, 0])` mengumpulkan nilai yang akan diprediksi (nilai pada hari ke-61) dan menambahkannya ke dalam *y_train* sebagai target *output*.

2. Membentuk *dataset* untuk pelatihan:

Setelah mengumpulkan semua set input dan target output, mereka dikonversi menjadi *array numpy* dengan `X_train, y_train = np.array(X_train), np.array(y_train)`. Konversi ini penting karena model LSTM, yang diimplementasikan dengan *framework* seperti *TensorFlow* atau *Keras*, mengharapkan input dalam bentuk *array numpy*.

Tahap ini penting untuk memastikan bahwa model LSTM dapat mempelajari hubungan temporal dan pola yang ada dalam data urutan waktu. Dengan membentuk pasangan data (*input-output*) dari jendela waktu yang terdiri dari 60 hari sebelumnya dan nilai yang akan diprediksi pada hari ke-61, model dapat memahami dan memprediksi nilai di masa depan berdasarkan pola yang ditemukan dalam data historis. Pelatihan model ini dapat dilihat pada gambar 4.9, 4.10, dan 4.11 dibawah ini :

```
# Persiapan pelatihan model
X_train = []
y_train = []
for i in range(60, len(training_set_scaled)):
    X_train.append(training_set_scaled[i-60:i, 0])
    y_train.append(training_set_scaled[i, 0])
X_train, y_train = np.array(X_train), np.array(y_train)
```

Gambar 4. 9 Kodingan Pelatihan Model TBS Terima

```

X_train2 = []
y_train2= []
for i in range(60, len(training_set_scaled2)):
    X_train2.append(training_set_scaled2[i-60:i, 0])
    y_train2.append(training_set_scaled2[i, 0])
X_train2, y_train2 = np.array(X_train2), np.array(y_train2)

```

Gambar 4. 10 Kodingan Pelatihan Model TBS Olah

```

X_train3 = []
y_train3= []
for i in range(60, len(training_set_scaled3)):
    X_train3.append(training_set_scaled3[i-60:i, 0])
    y_train3.append(training_set_scaled3[i, 0])
X_train3, y_train3 = np.array(X_train3), np.array(y_train3)

```

Gambar 4. 11 Kodingan Pelatihan Model Jumlah CPO

Selanjutnya adalah *reshape*, tahap ini adalah tahap persiapan akhir untuk data *input* sebelum digunakan dalam pelatihan model *Long Short-Term Memory* (LSTM). Pada tahap ini, kita mengubah bentuk (*reshape*) dari *dataset input* (*X_train*) agar sesuai dengan struktur yang diharapkan oleh model LSTM. Kodingan untuk *reshape* dapat dilihat pada gambar 4.12 dibawah ini :

```

X_train = np.reshape(X_train, (X_train.shape[0],X_train.shape[1],1))

```

Gambar 4. 12 Kodingan *Reshape*

4.8 Persiapan Data Uji

Persiapan data uji bertujuan untuk mempersiapkan set data uji dengan cara yang serupa dengan set data pelatihan. Ini memastikan bahwa model dapat memprediksi nilai dengan mempertimbangkan data historis yang relevan. Pada tahap ini, data historis yang digunakan untuk pelatihan dan data uji digabungkan menjadi satu kesatuan. Penggabungan ini dilakukan untuk memastikan bahwa data uji memiliki akses ke data historis yang cukup, yang diperlukan untuk membuat prediksi yang akurat. dipilih sejumlah data dari *dataset_total* untuk membentuk *input* bagi model. Data yang dipilih adalah sebanyak panjang set data uji (*test_set*) ditambah 60 entri sebelumnya. Penambahan 60 entri ini dilakukan karena model LSTM membutuhkan sejumlah nilai

historis untuk membuat prediksi (dalam hal ini, 60 nilai sebelumnya). Data yang dipilih kemudian diubah bentuknya menjadi array dua dimensi dengan satu kolom. Perubahan bentuk ini penting untuk memastikan bahwa data sesuai dengan format yang diperlukan oleh *scaler* dan model LSTM. Data yang telah diubah bentuknya kemudian dinormalisasi menggunakan *scaler* yang telah digunakan sebelumnya untuk menormalisasi data pelatihan. Secara keseluruhan, tahap ini mempersiapkan data uji dengan cara yang memungkinkan model LSTM untuk menggunakan data historis yang relevan dalam membuat prediksi. Dengan menggabungkan data pelatihan dan data uji, memilih data yang tepat, mengubah bentuk data, dan melakukan normalisasi, model dapat diberikan *input* yang konsisten dengan proses pelatihan, yang pada akhirnya meningkatkan keakuratan prediksi model terhadap produksi minyak sawit. Pelatihan data uji dapat dilihat pada gambar 4.13, 4.14 dan 4.15 dibawah ini:

```
dataset_total = pd.concat((dataset["TOTAL_TERIMA"][:'2022'],dataset["TOTAL_TERIMA"]['2023:']),axis=0)
inputs = dataset_total[len(dataset_total)-len(test_set) - 60:].values
inputs = inputs.reshape(-1,1)
inputs = sc.transform(inputs)
```

Gambar 4. 13 Pelatihan data Uji TBS Terima

```
dataset_total2 = pd.concat((dataset["TOTAL_OLAH"][:'2022'],dataset["TOTAL_OLAH"]['2023:']),axis=0)
inputs2 = dataset_total2[len(dataset_total2)-len(test_set2) - 60:].values
inputs2 = inputs2.reshape(-1,1)
inputs2 = sc2.transform(inputs2)
```

Gambar 4. 14 Pelatihan data Uji TBS Olah

```
dataset_total3 = pd.concat((dataset["JUMLAH_CPO"][:'2022'],dataset["JUMLAH_CPO"]['2023:']),axis=0)
inputs3 = dataset_total3[len(dataset_total3)-len(test_set2) - 60:].values
inputs3 = inputs3.reshape(-1,1)
inputs3 = sc3.transform(inputs3)
```

Gambar 4. 15 Pelatihan data Uji Jumlah CPO

Langkah selanjutnya adalah persiapan data uji Pada tahap ini, penulis membuat list X_{test} yang akan berisi data *input* untuk model dari set data uji. Setiap entri dalam X_{test} adalah sekuens sepanjang 60 hari dari data historis. Dengan menggunakan *loop*, penulis menambahkan sekuens data tersebut dari *inputs* (yang sudah dinormalisasi) ke dalam X_{test} . Artinya, setiap data pada X_{test} berisi data 60 hari sebelumnya yang digunakan untuk membuat prediksi untuk hari ke-61. Data dalam X_{test} kemudian diubah bentuknya menjadi *array* tiga dimensi dengan bentuk (jumlah sampel, panjang sekuens, jumlah fitur). Dalam hal ini, $X_{test}.shape[0]$ adalah jumlah sampel (jumlah sekuens yang ada), $X_{test}.shape[1]$ adalah panjang sekuens (60), dan 1 adalah jumlah fitur (satu fitur per hari, yaitu total produksi minyak sawit). Perubahan bentuk ini

penting karena model LSTM mengharapkan *input* dalam bentuk tiga dimensi. Setelah *X_test* disiapkan dan diubah bentuknya, penulis menggunakan model LSTM yang telah dilatih (*regressor*) untuk memprediksi nilai produksi minyak sawit. Model akan memberikan *output* prediksi berdasarkan data *input* yang diberikan. Prediksi yang dihasilkan oleh model masih dalam bentuk yang dinormalisasi (karena *input* yang digunakan adalah data yang sudah dinormalisasi). Oleh karena itu, penulis perlu mengembalikan skala prediksi tersebut ke skala aslinya menggunakan *sc.inverse_transform*. Dengan demikian, nilai prediksi akan berada pada skala yang sama dengan data asli produksi minyak sawit. Kodingan persiapan data uji dapat dilihat pada gambar 4.16, 4.17 dan 4.18 dibawah ini :

```
X_test = []
for i in range(60, len(inputs)):
    X_test.append(inputs[i-60:i, 0])
X_test = np.array(X_test)
X_test = np.reshape(X_test, (X_test.shape[0], X_test.shape[1], 1))
predicted_stock_price = regressor.predict(X_test)
predicted_stock_price = sc.inverse_transform(predicted_stock_price)
```

Gambar 4. 16 Persiapan Data Uji TBS Terima

```
X_test2 = []
for i in range(60, len(inputs2)):
    X_test2.append(inputs2[i-60:i, 0])
X_test2 = np.array(X_test2)
X_test2 = np.reshape(X_test2, (X_test2.shape[0], X_test2.shape[1], 1))
predicted_stock_price2 = regressor2.predict(X_test2)
predicted_stock_price2 = sc2.inverse_transform(predicted_stock_price2)
```

Gambar 4. 17 Persiapan Data Uji TBS Olah

```
X_test3 = []
for i in range(60, len(inputs3)):
    X_test3.append(inputs3[i-60:i, 0])
X_test3 = np.array(X_test3)
X_test3 = np.reshape(X_test3, (X_test3.shape[0], X_test3.shape[1], 1))
predicted_stock_price3 = regressor3.predict(X_test3)
predicted_stock_price3 = sc3.inverse_transform(predicted_stock_price3)
```

Gambar 4. 18 Persiapan Data Uji Jumlah CPO

4.9 Implementasi LSTM

Arsitektur LSTM yang digunakan dalam penelitian ini terdiri dari beberapa lapisan yang dirancang untuk memprediksi produksi minyak sawit dengan akurasi tinggi. Model ini dimulai dengan membuat sebuah objek *Sequential* dari *Keras*, yang digunakan untuk membangun jaringan saraf berlapis. Pada penelitian ini penulis juga melakukan *fine-tuning* menggunakan *hyperparameter* berikut, yang diambil berdasarkan rekomendasi untuk LSTM:

1. *Epoch* : 50
2. *Batch size* : 32

Pada penelitian ini penulis menggunakan empat lapisan LSTM karena model ini perlu menangkap informasi temporal yang kompleks dari data sejarah produksi minyak sawit. LSTM memiliki kemampuan untuk mengingat informasi jangka panjang dari *sekuens* data, yang sangat berguna dalam memodelkan tren dan pola dalam data waktu. Dengan menggunakan beberapa lapisan LSTM, model dapat mempelajari representasi yang semakin abstrak dan kompleks dari data *input*.

Penulis juga menetapkan *return_sequences=True* pada tiga lapisan LSTM pertama untuk mempertahankan urutan *output* dari setiap lapisan LSTM, yang diperlukan ketika menggunakan beberapa lapisan LSTM berurutan. Ini memungkinkan informasi untuk mengalir dari satu lapisan LSTM ke lapisan LSTM berikutnya dengan urutan yang benar. Digunakan lapisan *dropout* dengan rate 20% di setiap lapisan LSTM untuk mencegah *overfitting*. *Overfitting* dapat terjadi ketika model terlalu memperhatikan detail-detail kecil dalam data pelatihan yang mungkin tidak umum atau tidak relevan dalam data baru. Dengan menerapkan *dropout*, penulis secara efektif mematikan sebagian unit dalam setiap lapisan selama proses pelatihan, sehingga memaksa model untuk belajar dengan cara yang lebih umum dan mencegahnya mengandalkan ketergantungan yang terlalu kuat pada data pelatihan. Lalu lapisan terakhir dalam model ini adalah lapisan *dense* dengan 1 unit, yang digunakan untuk menghasilkan satu *output*, yaitu nilai prediksi produksi minyak sawit.

Setelah membangun arsitektur model, tahap berikutnya adalah kompilasi model. Pada tahap ini, penulis menggunakan *optimizer* '*RMSprop*' dan fungsi kerugian '*mean_squared_error*' untuk mengoptimalkan model. Fungsi kerugian ini cocok untuk

masalah regresi seperti prediksi produksi minyak sawit. Penulis memilih *optimizer* 'RMSprop' karena *optimizer* ini sering kali cocok untuk masalah regresi dan telah terbukti efektif dalam melatih jaringan saraf dalam banyak kasus. Fungsi kerugian '*mean_squared_error*' dipilih karena masalah ini adalah masalah regresi di mana kami mencoba meminimalkan kesalahan prediksi dari nilai aktual. Langkah terakhir dalam proses ini adalah melatih (*fit*) model menggunakan data pelatihan yang telah diproses.

Proses pelatihan ini dilakukan sebanyak 4 kali dimana menggunakan 50 dan 100 *epoch* dengan *batch size* sebesar 32 dan 64. Hasil dari proses pelatihan ini dapat dilihat pada tabel dibawah :

Tabel 4.5 Proses Pelatihan

Jumlah <i>Epoch</i>	Jumlah <i>Batch Size</i>	Loss	Evaluasi (RMSE)
50	32	0.0225	0.1383
100	32	0.0257	0.1381
50	64	0.0383	0.1279
100	64	0.0298	0.1707

Pada tabel diatas dapat dilihat bahwa dari 4 kali percobaan didapat hasil terbaik yaitu pada 50 *epoch* dan 32 *batch size*. Berdasarkan percobaan tersebut pula penulis memilih jumlah *epoch* dan *batch size* tersebut di penelitian ini. Pengaturan ini membantu model untuk belajar dari data secara bertahap dan memperbaiki bobotnya untuk meningkatkan akurasi prediksi. Implementasi LSTM dapat dilihat pada gambar 4.19 dibawah ini :

```
# The LSTM architecture
regressor = Sequential()
# First LSTM layer with Dropout regularisation
regressor.add(LSTM(units=50, return_sequences=True, input_shape=(X_train.shape[1],1)))
regressor.add(Dropout(0.2))
# Second LSTM layer
regressor.add(LSTM(units=50, return_sequences=True))
regressor.add(Dropout(0.2))
# Third LSTM layer
regressor.add(LSTM(units=50, return_sequences=True))
regressor.add(Dropout(0.2))
# Fourth LSTM layer
regressor.add(LSTM(units=50))
regressor.add(Dropout(0.2))
# The output layer
regressor.add(Dense(units=1))

# Compiling the RNN
regressor.compile(optimizer='rmsprop', loss='mean_squared_error')
# Fitting to the training set
regressor.fit(X_train,y_train,epochs=50,batch_size=32)
```

Gambar 4. 19 Implementasi LSTM

4.10 Evaluasi

Dalam penelitian ini, diperlukan evaluasi terhadap model yang telah dibuat, tahap evaluasi dengan menggunakan *Root Mean Squared Error* (RMSE) bertujuan untuk mengukur seberapa baik model LSTM yang telah dilatih dalam memprediksi produksi minyak sawit. Penilaian ini penting karena memberikan gambaran tentang seberapa dekat nilai prediksi yang dihasilkan oleh model dengan nilai sebenarnya dari data yang belum pernah dilihat sebelumnya. Hasil evaluasi RMSE untuk prediksi produksi minyak sawit seperti gambar 4.20, 4.21 dan 4.22 dibawah ini :

```
return_rmse(test_set_scaled1,predicted_stock_price_d)

RMSE : 0.12384899092714552.
```

```
# Evaluating our model
print ("RMSE data train")
return_rmse(train_set_scaled_3,predicted_stock_price3_d)

RMSE data train
RMSE : 1.0366143568363992.
```

Gambar 4. 20 Hasil Evaluasi TBS Terima

```
return_rmse(test_set_scaled2,predicted_stock_price_d)

RMSE : 0.11770094237784422.
```

```
# Evaluating our model
print ("RMSE data train")
return_rmse(train_set_scaled_1,predicted_stock_price_d)

RMSE data train
RMSE : 0.20408227279500232.
```

Gambar 4. 21 Hasil Evaluasi TBS Olah

```
return_rmse(test_set_scaled3,predicted_stock_price_d)

RMSE : 0.11770094237784419.
```

```
# Evaluating our model
print ("RMSE data train")
return_rmse(train_set_scaled_2,predicted_stock_price2_d)
```

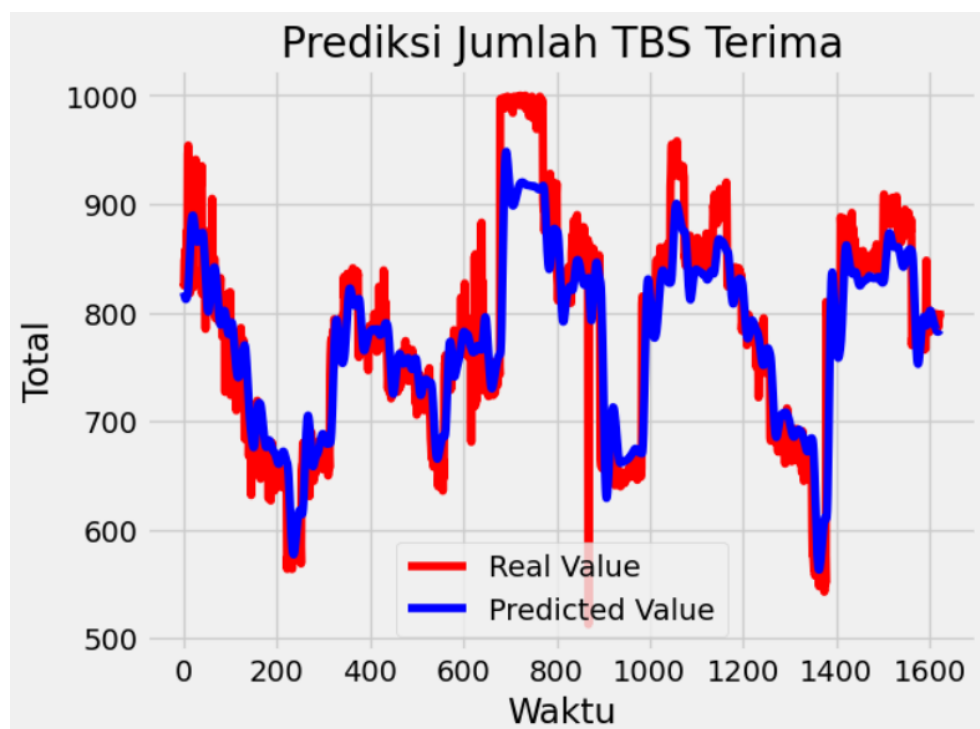
RMSE data train
RMSE : 1.176585284076752.

Gambar 4. 22 Hasil Evaluasi Jumlah CPO

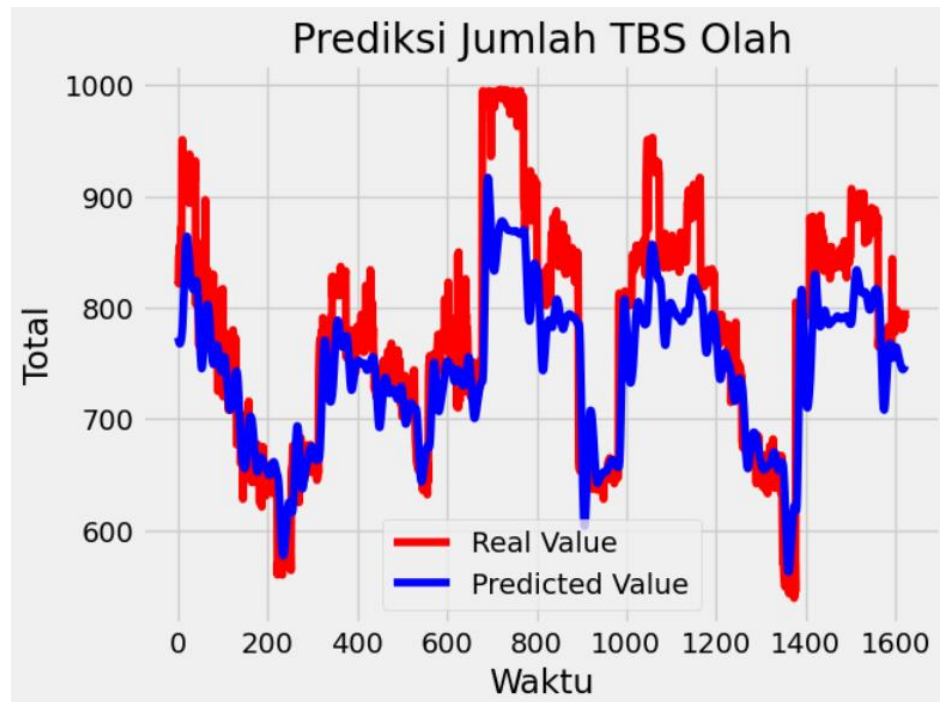
Dapat dilihat pada gambar diatas bahwa evaluasi ini menghasilkan nilai yang baik untuk memprediksi, karena sesuai teori RMSE bahwa nilai evaluasi RMSE yang baik adalah nilai yang mendekati 0.

4.11 Hasil Prediksi

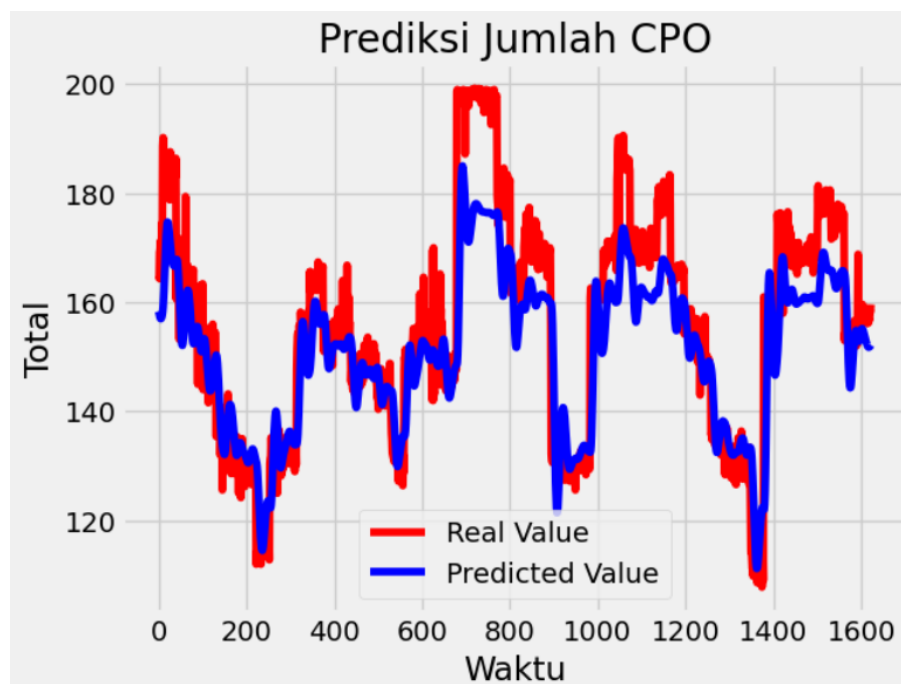
Setelah melatih model menggunakan data latih dan data uji, maka model akan memberi prediksinya. Hasil prediksi dapat dilihat pada gambar 4.23, 4.24 dan 4.25 dibawah ini :



Gambar 4. 23 Visualisasi Data Asli dan Data Prediksi TBS Terima



Gambar 4. 24 Visualisasi Data Asli dan Data Prediksi TBS Olah



Gambar 4. 25 Visualisasi Data Asli dan Data Prediksi Jumlah CPO

Dimana pada gambar tersebut terdapat dua garis yaitu garis merah dan garis biru. Garis merah merupakan nilai asli dari *Dataset* dan garis biru merupakan nilai prediksinya. Dapat dilihat dari grafik tersebut bahwa nilai asli dan nilai prediksi tidak

terlalu jauh. Bisa juga untuk melihat presentase yang ada dari setiap nilai asli dan nilai prediksi seperti tabel 4.4, 4.5 dan 4.6 dibawah ini :

Tabel 4. 6 Presentase Nilai Asli dan Nilai Prediksi TBS Terima

Total Terima Asli	Prediksi Total Terima	Presentase
827.0	809.27	97.86%
825.0	807.77	97.91%
848.0	806.10	95.06%
851.0	804.69	94.56%
858.0	804.03	93.71%
825.0	804.56	97.52%
849.0	805.86	94.92%
875.0	807.64	92.30%
852.0	810.30	95.11%
946.0	813.70	86.01%

Rata-rata dari presentase nilai asli dan nilai prediksinya pada TBS terima diperoleh 96,22%

Rata-rata Persentase Total Terima TBS (data train) diperoleh 94.08%

Tabel 4. 7 Presentase Nilai Asli dan Nilai Prediksi TBS Olah

Total Olah Asli	Prediksi Total Olah	Presentase
824.0	809.48	98.24 %
822.0	807.93	98.29 %
846.0	806.10	95,28%
858.0	804.30	94.85%
856.0	802.95	93,80%
822.0	802.43	97,62%
847.0	802.50	94,75%
872.0	803.12	92,10%
850.0	804.57	94,66%
943.0	806,75	85.55%

Rata-rata dari presentase nilai asli dan nilai prediksinya pada TBS terima diperoleh 95,79%

Rata-rata Persentase Total Olah (data train) diperoleh 93.96%

Tabel 4. 8 Presentase Nilai Asli dan Nilai Prediksi Jumlah CPO

Jumlah CPO Asli	Prediksi Jumlah CPO	Presentase
164,8	169.09	97,39%
164,4	168.66	97.40%
169,2	168.21	99.42%
171,2	167,79	97.81%
164,4	167,25	98.26%
169,4	167,20	98.70%
174,4	167,27	95.92%
170,0	166,53	98.55%
188,6	167,95	89,05%

Rata-rata dari presentase nilai asli dan nilai prediksinya pada Jumlah CPO diperoleh 95.80%.

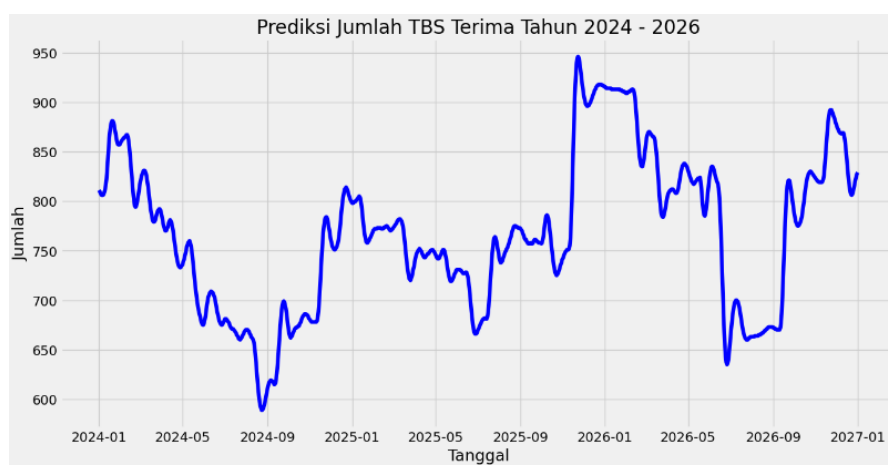
Rata-rata Persentase Total CPO (data train) diperoleh 93.21%

Hasil presentase yang diperoleh dari proses ini adalah metrik yang menggambarkan seberapa akurat model kita dalam memprediksi nilai. Presentase ini didapatkan dari perhitungan persentase keakuratan prediksi terhadap nilai asli.

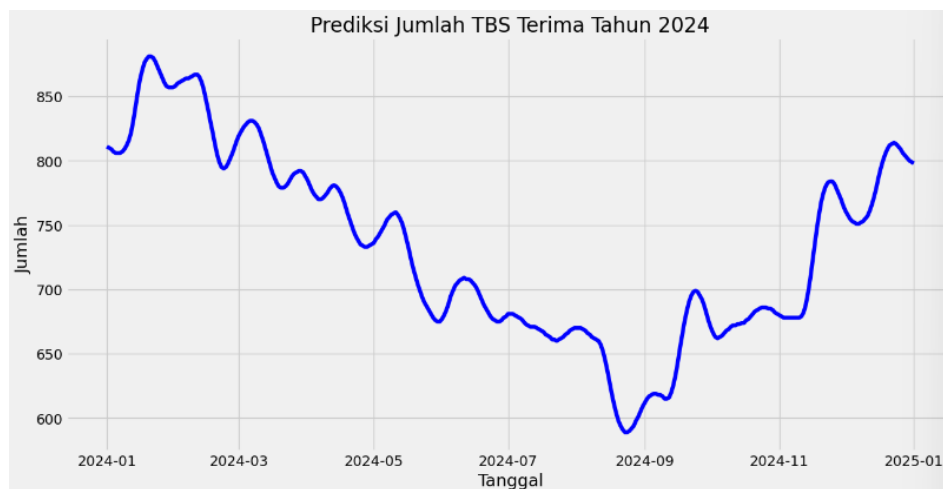
Proses perhitungan persentase akurasi prediksi dimulai dengan mengubah data asli (*test_set*) dan data prediksi (*predicted_stock_price*) ke dalam bentuk kolom untuk memudahkan pengolahan. Selanjutnya, dihitung perbedaan absolut antara nilai asli dan nilai prediksi untuk mendapatkan nilai *error*, yang mencerminkan seberapa besar kesalahan prediksi. Persentase akurasi kemudian dihitung dengan mengurangi rasio error terhadap nilai asli dari 1, lalu mengalikannya dengan 100 untuk mengubahnya menjadi persentase. Langkah ini menunjukkan seberapa dekat nilai prediksi dengan nilai asli. Untuk memastikan tidak ada nilai negatif dalam persentase akurasi, semua nilai negatif diubah menjadi nol. Setelah itu, kolom nilai asli, nilai prediksi, dan

persentase akurasi digabungkan menjadi satu array dua dimensi. Hasil akhirnya ditampilkan dalam format tabel yang menunjukkan nilai asli, nilai prediksi, dan persentase akurasi, sehingga memberikan gambaran yang jelas tentang seberapa akurat prediksi terhadap data asli.

Gambar 4.26, 4.27, 4.28 dan 4.29 dibawah ini merupakan grafik prediksi dari jumlah TBS Terimah untuk periode 3 tahun kedepan. Dapat dilihat bahwa pada grafik dibawah terjadi penurunan jumlah TBS terima pada bulan 9 tahun 2024 dan puncak terbanyak TBS terima terjadi pada bulan 12 tahun 2026.

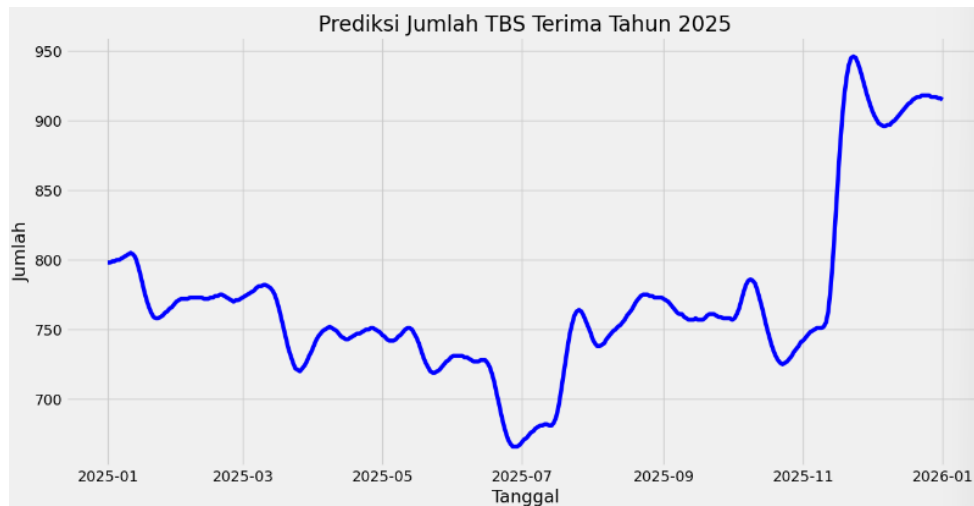


Gambar 4. 26 Visualisasi Hasil Prediksi TBS Terima Tahun 2024 - 2026



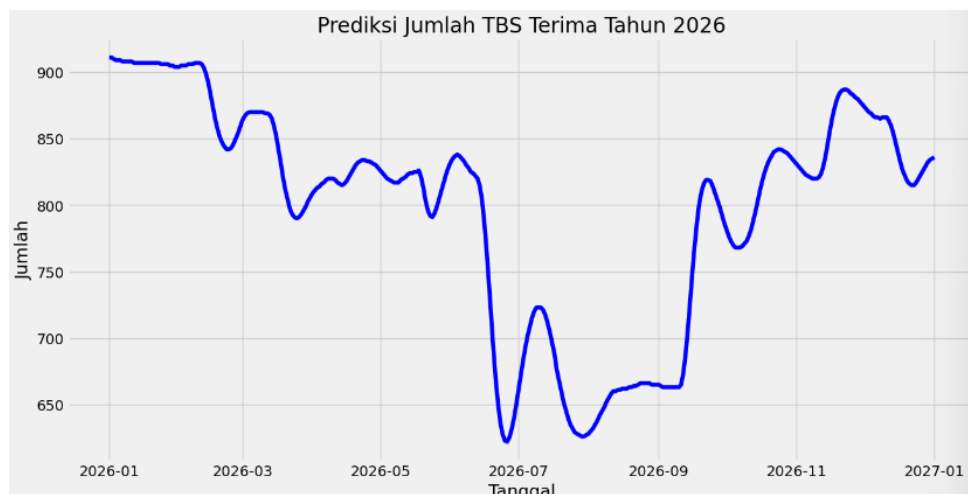
Gambar 4. 27 Visualisasi Hasil Prediksi TBS Terima Tahun 2024

Pada gambar diatas dapat dilihat bahwa terjadi penurunan jumlah TBS terima pada bulan September 2024, dimana mencapai nilai 600 ton.



Gambar 4. 28 Visualisasi Hasil Prediksi TBS Terima Tahun 2025

Pada gambar diatas dapat dilihat bahwa puncak jumlah TBS terima terdapat pada bulan desember mencapai nilai 950 ton dan terendah pada bulan juli mencapai nilai 700 ton .



Gambar 4. 29 Visualisasi Hasil Prediksi TBS Terima Tahun 2026

Pada gambar diatas dapat dilihat terjadi puncak jumlah TBS terima pada bulan februari mencapai nilai 900 ton dan terendah dibulan juli mencapai nilai 650 ton.

Hasil prediksi TBS Terima untuk periode 3 tahun kedepan dapat dilihat pada gambar 4.30 dibawah ini:

```

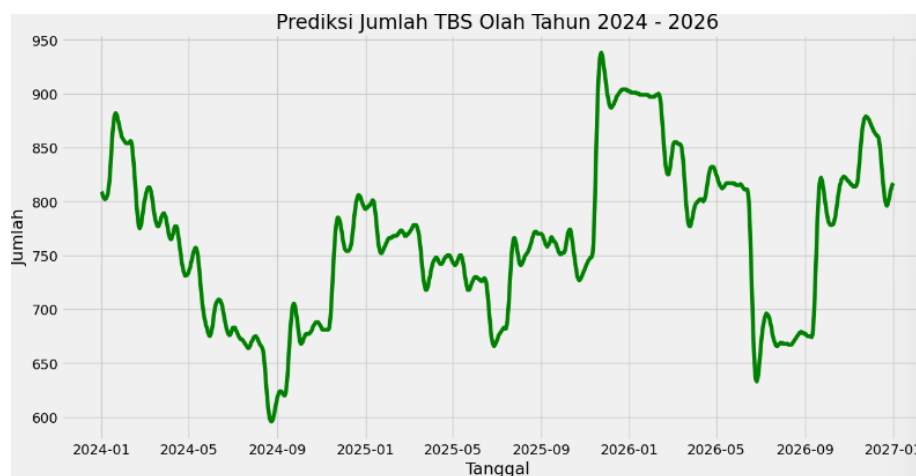
51/51 [=====] - 2s 46ms/step
Prediksi Jumlah TOTAL_TERIMA untuk 3 tahun ke depan (per hari):
      Predicted_TOTAL_TERIMA
2024-01-01      811.0
2024-01-02      810.0
2024-01-03      809.0
2024-01-04      807.0
2024-01-05      806.0
...
2026-12-28      820.0
2026-12-29      823.0
2026-12-30      826.0
2026-12-31      828.0
2027-01-01      829.0

[1097 rows x 1 columns]

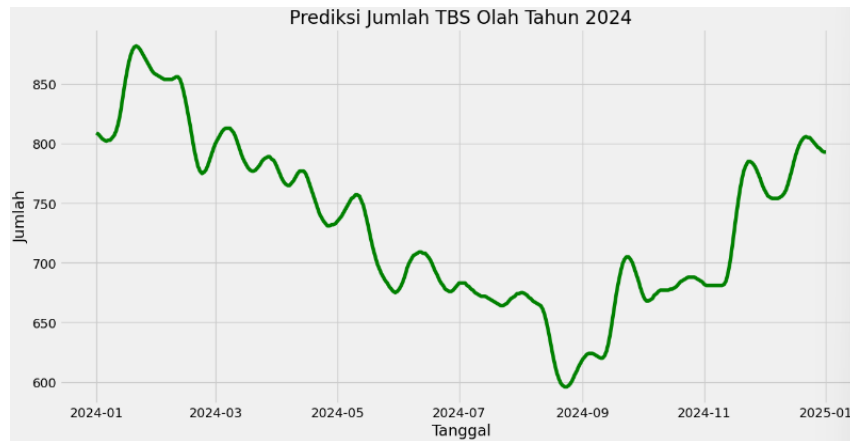
```

Gambar 4. 30 Hasil Prediksi TBS Terima untuk 3 tahun kedepan

Gambar 4.31, 4.32, 4.33 dan 4.34 dibawah ini merupakan grafik hasil prediksi Jumlah TBS Olah untuk 3 tahun kedepan, dapat dilihat bahwa pada awal 2024 hingga bulan september terjadi penurunan TBS yang akan diolah dan perlahan naik lagi hingga awal tahun 2025.

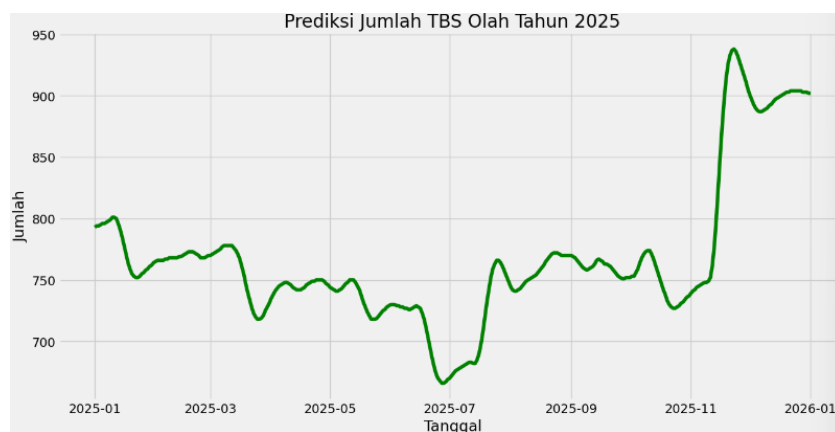


Gambar 4. 31 Visualisasi Hasil Prediksi TBS Olah Tahun 2024 – 2026



Gambar 4. 32 Visualisasi Hasil Prediksi TBS Olah Tahun 2024

Pada gambar diatas dapat dilihat bahwa jumlah TBS olah tertinggi terdapat pada bulan Februari mencapai nilai 850 ton dan terendah di bulan September mencapai nilai 600 ton.



Gambar 4. 33 Visualisasi Hasil Prediksi TBS Olah Tahun 2025

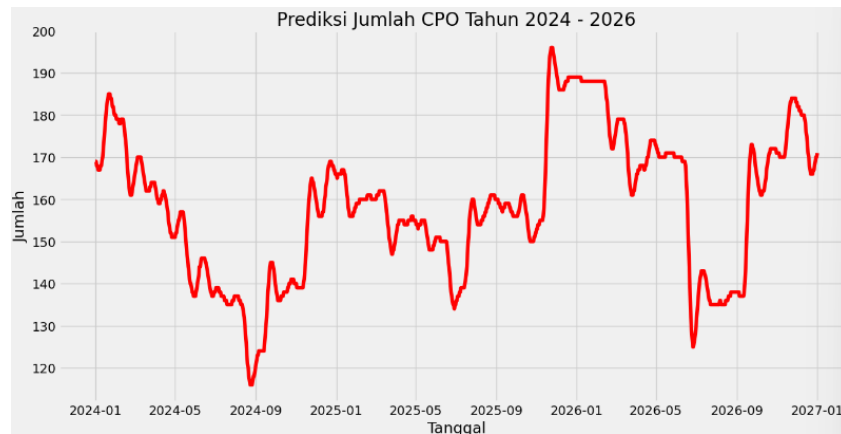
Pada gambar 4.33 diatas dapat dilihat bahwa jumlah TBS olah tertinggi terdapat di bulan desember dengan nilai hampir mencapai 950 ton dan terendah di bulan juli dengan nilai dibawah 700 ton.



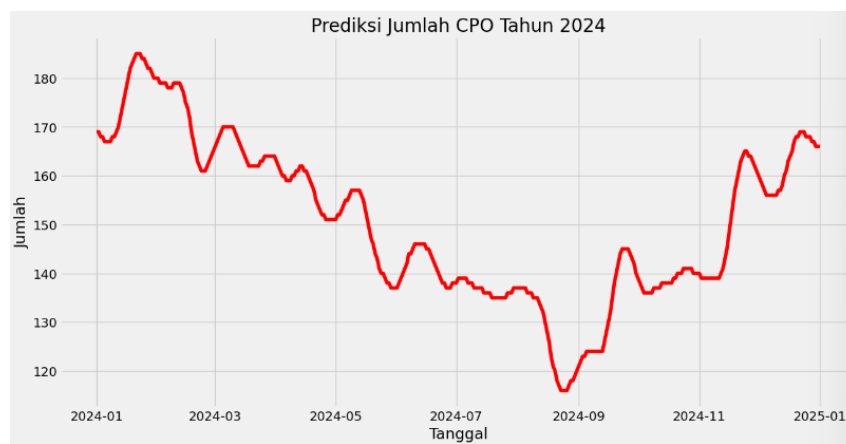
Hasil prediksi TBS Olah untuk periode 3 tahun kedepan dapat dilihat pada gambar 4.35 dibawah ini:

Gambar 4. 35 Hasil Prediksi TBS Olah untuk 3 tahun kedepan

Gambar 4.36, 4.37, 4.38 dan 4.39 dibawah ini merupakan grafik hasil prediksi Jumlah CPO untuk tiga tahun kedepan dimana dapat dilihat bahwa terjadi jumlah CPO tertinggi mencapai diatas 180 ton dan terendah mencapai dibawah 120 ton.



Gambar 4. 36 Visualisasi Hasil Prediksi Jumlah CPO Tahun 2024 - 2026



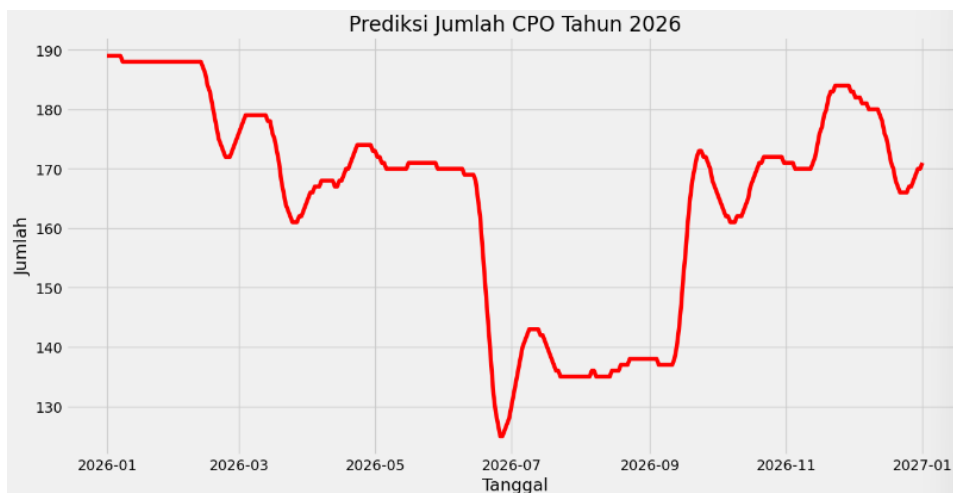
Gambar 4. 37 Visualisasi Hasil Prediksi Jumlah CPO Tahun 2024

Pada gambar 4.37 di atas dapat dilihat bahwa jumlah CPO tertinggi terdapat pada bulan Februari mencapai nilai di atas 180 ton dan terendah di bulan September mencapai nilai 120 ton.



Gambar 4. 38 Visualisasi Hasil Prediksi Jumlah CPO Tahun 2025

Pada 4.38 gambar diatas dapat dilihat bahwa jumlah CPO tertinggi terdapat di bulan Desember dengan nilai mencapai lebih dari 190 ton dan terendah di bulan juli dengan nilai dibawah 140 ton.



Gambar 4. 39 Visualisasi Hasil Prediksi Jumlah CPO Tahun 2026

Pada gambar 4.39 diatas dapat dilihat bahwa jumlah CPO tertinggi terdapat di bulan januari dengan nilai 190 ton dan terendah di bulan juli dengan nilai dibawah 130 ton.

Hasil prediksi Jumlah CPO untuk periode 3 tahun kedepan dapat dilihat pada gambar 4.40 dibawah ini:

```
51/51 [=====] - 2s 31ms/step
Prediksi Jumlah TOTAL_OLAH untuk 3 tahun ke depan (per hari):
      Predicted_JUMLAH_CPO
2024-01-01          169.0
2024-01-02          169.0
2024-01-03          168.0
2024-01-04          168.0
2024-01-05          167.0
...                ...
2026-12-28          168.0
2026-12-29          169.0
2026-12-30          170.0
2026-12-31          170.0
2027-01-01          171.0

[1097 rows x 1 columns]
```

Gambar 4. 40 Hasil Prediksi TBS Terima untuk 3 tahun kedepan

BAB 5

PENUTUPAN

5.1 Kesimpulan

Berdasarkan hasil dari implementasi LSTM dalam melakukan prediksi produksi minyak sawit, diperoleh kesimpulan berikut:

1. Dalam memprediksi jumlah produksi sawit menggunakan *metode Long Short-Term Memory* dengan menggunakan *Hyperparameter* sebesar 50 *epoch* dan *batch size* 32 menghasilkan nilai prediksi yang baik. *Hyperparameter* ini menghasilkan model yang seimbang antara akurasi, efisiensi, dan kemampuan generalisasi, sehingga memberikan hasil prediksi yang baik untuk produksi minyak sawit.
2. Hasil evaluasi menggunakan nilai RMSE dengan nilai RMSE untuk total TBS terima sebesar 0.1238, total TBS olah sebesar 0.1177 dan jumlah CPO sebesar 0.1177. Dimana hasil ini menunjukkan bahwa model LSTM memiliki tingkat akurasi yang baik dalam memprediksi produksi minyak sawit karena pada evaluasi RMSE nilai terbaik adalah nilai yang mendekati 0,
3. Hasil dari penelitian ini merupakan model dan hasil berupa nilai prediksi untuk tiga tahun kedepan

5.2 Saran

Saran yang diberikan untuk penelitian selanjutnya adalah sebagai berikut:

1. Memperbanyak jumlah *dataset* agar model mendapatkan hasil training yang lebih baik sehingga mendapatkan hasil evaluasi yang lebih baik pula.
2. Disarankan untuk melakukan *hyperparameter tuning* untuk perbandingan akurasi yang lebih

3. Melakukan perbandingan dengan metode prediksi lainnya untuk melihat keunggulan dan kelemahan LSTM dalam konteks prediksi
4. Disarankan penelitian selanjutnya dapat mengimplementasikannya kedalam *website*.

DAFTAR PUSTAKA

- Adi, R. M. S., & Sudioanto, S. (2022). Prediksi Harga Komoditas Pangan Menggunakan Algoritma Long Short-Term Memory (LSTM). *Building of Informatics, Technology and Science (BITS)*, 4(2). <https://doi.org/10.47065/bits.v4i2.2229>
- Adnan, F., & Amelia, I. (2022). *Implementasi Voice Recognition Berbasis Machine Learning. 11.*
- Cahyadi, R., Damayanti, A., & Aryadani, D. (2020). *RECURRENT NEURAL NETWORK (RNN) DENGAN LONG SHORT TERM MEMORY (LSTM) UNTUK ANALISIS SENTIMEN DATA INSTAGRAM. 5(1).*
- David, M., Cholissodin, I., & Yudistira, N. (n.d.-a). *Prediksi Harga Cabai menggunakan Metode Long-Short Term Memory (Case Study: Kota Malang).*
- David, M., Cholissodin, I., & Yudistira, N. (n.d.-b). *Prediksi Harga Cabai menggunakan Metode Long-Short Term Memory (Case Study: Kota Malang).*
- Deep Learning | Introduction to Long Short Term Memory.* (n.d.). Retrieved December 22, 2023, from <https://www.geeksforgeeks.org/deep-learning-introduction-to-long-short-term-memory/>
- Faisal Muhammad, T. A., & Irawan, M. I. (2023). Implementasi Long Short-Term Memory (LSTM) untuk Prediksi Intensitas Curah Hujan (Studi Kasus: Kabupaten Malang). *Jurnal Sains dan Seni ITS*, 12(1), A34–A39. <https://doi.org/10.12962/j23373520.v12i1.106892>
- Farwati, M., Salsabila, I. T., Navira, K. R., & Sutabri, T. (2023). *ANALISA PENGARUH TEKNOLOGI ARTIFICIAL INTELLIGENCE (AI) DALAM KEHIDUPAN SEHARI-HARI. 11.*

- Insani, F., & Sanjaya, S. (2022). *Implementasi Long Short Term Memory Neural Network Untuk Prediksi Indeks Harga Perdagangan Besar*.
- Irawan, F., Sumijan, S., & Yuhandri, Y. (2021). Prediksi Tingkat Produksi Buah Kelapa Sawit dengan Metode Single Moving Average. *Jurnal Informasi dan Teknologi*, 251–256. <https://doi.org/10.37034/jidt.v3i4.162>
- Jaelani, T. (2022). Machine Learning untuk Prediksi Produksi Gula Nasional. *JMPM (Jurnal Material dan Proses Manufaktur)*, 6(1). <https://doi.org/10.18196/jmpm.v6i1.14897>
- Kamath, P., Patil, P., S, S., Sushma, & S, S. (2021). Crop yield forecasting using data mining. *Global Transitions Proceedings*, 2(2), 402–407. <https://doi.org/10.1016/j.gltp.2021.08.008>
- Kusnaldi, M. R., Gulo, T., & Aripin, S. (2022). Penerapan Normalisasi Data Dalam Mengelompokkan Data Mahasiswa Dengan Menggunakan Metode K-Means Untuk Menentukan Prioritas Bantuan Uang Kuliah Tunggal. *Journal of Computer System and Informatics (JoSYC)*, 3(4), 330–338. <https://doi.org/10.47065/josyc.v3i4.2112>
- Kusuma, A. M., Aulia, H., Oktavian, M. A., Akbar, M. R., Patricia, P., & Abdiansah, A. (2023). Prediksi Gender Berdasarkan Nama Bahasa Indonesia Menggunakan Long Short Term Memory. *Jurnal Teknik Informatika dan Sistem Informasi*, 9(2). <https://doi.org/10.28932/jutisi.v9i2.6404>
- Liu, W., Liu, W. D., & Gu, J. (2020). Forecasting oil production using ensemble empirical model decomposition based Long Short-Term Memory neural network. *Journal of Petroleum Science and Engineering*, 189, 107013. <https://doi.org/10.1016/j.petrol.2020.107013>

- Muhammad Haris Diponegoro, Sri Suning Kusumawardani, & Indriana Hidayah. (2021). Tinjauan Pustaka Sistematis: Implementasi Metode Deep Learning pada Prediksi Kinerja Murid. *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*, 10(2), 131–138. <https://doi.org/10.22146/jnteti.v10i2.1417>
- Pontoh, R. S., Toharudin, T., Ruchjana, B. N., Gumelar, F., Putri, F. A., Agisya, M. N., & Caraka, R. E. (2022). Jakarta Pandemic to Endemic Transition: Forecasting COVID-19 Using NNAR and LSTM. *Applied Sciences*, 12(12), 5771. <https://doi.org/10.3390/app12125771>
- Qiu, J., Wang, B., & Zhou, C. (2020). Forecasting stock prices with long-short term memory neural network based on attention mechanism. *PLOS ONE*, 15(1), e0227222. <https://doi.org/10.1371/journal.pone.0227222>
- Rahmatullah, S., & Destia, D. (2018). PREDIKSI ALOKASI JUMLAH PRODUKSI MINYAK SAWIT DENGAN METODE REGRESI LINIER BERGANDA PADA PT. PALM LAMPUNG PERSADA. *Jurnal Informasi dan Komputer*, 6(2), 61–69. <https://doi.org/10.35959/jik.v6i2.114>
- Sanjaya, F. I., & Heksaputra, D. (2020). Prediksi Rerata Harga Beras Tingkat Grosir Indonesia dengan Long Short Term Memory. *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, 7(2), 163–174. <https://doi.org/10.35957/jatisi.v7i2.388>
- Song, X., Liu, Y., Xue, L., Wang, J., Zhang, J., Wang, J., Jiang, L., & Cheng, Z. (2020). Time-series well performance prediction based on Long Short-Term Memory (LSTM) neural network model. *Journal of Petroleum Science and Engineering*, 186, 106682. <https://doi.org/10.1016/j.petrol.2019.106682>
- Yunus, A., & Akbar, M. (2019). DATA MINING UNTUK MEMPREDIKSI HASIL PRODUKSI BUAH SAWIT PADA PT BUMI SAWIT SUKSES (BSS) MENGGUNAKAN METODE K-NEAREST NEIGHBOR.

Yuthadi, A. R., Cholissodin, I., & Santoso, E. (n.d.). *Prediksi Jumlah Kasus COVID-19 di Dunia dengan menggunakan Metode Long Short-Term Memory.*