

**IMPLEMENTASI *LONG SHORT TERM MEMORY* PADA PREDIKSI
HARGA SAHAM PERUSAHAAN DI BIDANG PANGAN**

SKRIPSI

OLEH:

KHOIRO AMALIA

NIM. 210601110083



**PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
2025**

**IMPLEMENTASI *LONG SHORT TERM MEMORY* PADA PREDIKSI
HARGA SAHAM PERUSAHAAN DI BIDANG PANGAN**

SKRIPSI

**Diajukan Kepada
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
untuk Memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Matematika (S.Mat)**

**Oleh
Khoiro Amalia
NIM. 210601110083**

**PROGRAM STUDI MATEMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG
2025**

**IMPLEMENTASI *LONG SHORT TERM MEMORY* PADA PREDIKSI
HARGA SAHAM PERUSAHAAN DI BIDANG PANGAN**

SKRIPSI

Oleh
Khoiro Amalia
NIM. 210601110083

Telah Disetujui untuk Diuji
Malang, 11 Maret 2025

Dosen Pembimbing I



Hisyam Fahmi, M.Kom.
NIP. 19890727 201903 1 018

Dosen Pembimbing II



Dr. Fachrud Rozi, M.Si.
NIP. 19800527 200801 1 012

Mengetahui,
Ketua Program Studi Matematika




Dr. Billy Susanti, M.Sc.
NIP. 19741129 200012 2 005

**IMPLEMENTASI *LONG SHORT TERM MEMORY* PADA PREDIKSI
HARGA SAHAM PERUSAHAAN DI BIDANG PANGAN**

SKRIPSI

**Oleh
Khoiro Amalia
NIM. 210601110083**

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Matematika (S.Mat)

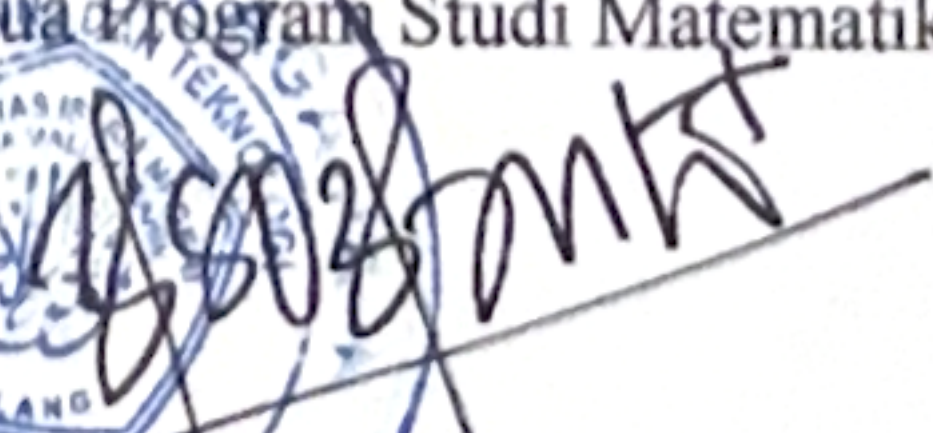
Tanggal 23 Mei 2025

Ketua Penguji : Dr. Mohammad Jamhuri, M.Si.

Anggota Penguji 1 : Ria Dhea Layla Nur Karisma, M.Si.

Anggota Penguji 2 : Hisyam Fahmi, M.Kom.

Anggota Penguji 3 : Dr. Fachrur Rozi, M.Si.

Mengetahui,
Ketua Program Studi Matematika

Dr. Eddy Susanti, M.Sc.
NIM. 09741129 200012 2 005



PERNYATAAN KEASLIAN TULISAN

Saya bertanda tangan dibawah ini

Nama : Khoiro Amalia
NIM : 210601110083
Program Studi : Matematika
Fakultas : Sains dan Teknologi
Judul Skripsi : Implementasi *Long Short Term Memory* untuk Prediksi
Harga Saham Perusahaan di Bidang Pangan

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini merupakan hasil karya sendiri, bukan pengambilan tulisan atau pemikiran orang lain yang saya akui sebagai pemikiran saya, kecuali dengan mencantumkan sumber cuplikan pada daftar rujukan di dalam terakhir. Apabila dikemudian hari terbukti skripsi ini adalah hasil jiplakan atau tiruan, maka saya bersedia menerima sanksi yang berlaku atas perbuatan tersebut.

Malang, 23 Mei 2025



Khoiro Amalia

NIM. 210601110083

MOTO

“Orang kuat bukan dia yang tak pernah jatuh, tapi dia yang terus bangkit setiap kali terjatuh”

(Ali bin Abi Thalib RA)

“Apa yang ditakdirkan untukmu takkan pernah melewatimu, dan apa yang melewatimu takkan pernah menjadi milikmu”

(Umar bin Khattab RA)

PERSEMBAHAN

Segala puji dan syukur saya panjatkan ke hadirat Allah SWT yang telah memberikan rahmat, kekuatan, dan kesabaran. Dengan penuh rasa hormat dan cinta, saya persembahkan karya ini kepada kedua orang tua saya Bapak Abdul Mukid, S.Pd dan Ibu Khalifah sumber kekuatan yang tak pernah padam.

Terimakasih atas cinta yang tak bersyarat, doa yang selalu mengiringi setiap langkah, dan pengorbanan yang tak pernah meminta balasan. Dalam setiap tetes keringat, dalam setiap malam yang kalian lalui dengan sabar, tumbuhlah semangat dalam diri penulis untuk terus melangkah. Kepada kakak tersayang Qurrotul A'yun, M.Si dan Mahsusotur Rohmania, S.Pd. terimakasih sudah menjadi panutan, teman berbagi keluh kesah, sekaligus penyemangat dalam diam.

Dukungan dan kepercayaan kalian telah banyak menguatkan penulis lebih dari yang bisa di ungkapkan. Skripsi ini mungkin hanya secarik lembaran ilmiah, tapi didalamnya tertulis setiap harap dan doa kalian yang menguatkan penulis untuk bertahan. Jika penulis bisa berdiri sampai hari ini, itu karena doa kalian yang tak pernah putus. Semoga karya ini menjadi secuil kebanggaan yang bisa saya berikan untuk membalas sedikit dari segala yang telah kalian berikan.

KATA PENGANTAR

Assalamu 'alaikum warahmatullahi wabarakatuh

Segala puji dan syukur penulis panjatkan kehadirat Allah SWT atas limpahan rahmat, taufik, dan hidayah-Nya, sehingga penulis dapat menyelesaikan penyusunan skripsi yang berjudul “Implementasi *Long Short Term Memory* pada Prediksi Harga Saham Perusahaan di Bidang Pangan”. Skripsi ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana Matematika (S.Mat) di Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Shalawat serta salam semoga senantiasa tercurahkan kepada Nabi Muhammad SAW, suri tauladan bagi seluruh umat manusia yang telah membawa cahaya ilmu dan iman ke seluruh penjuru dunia.

Penulis menyadari bahwa terselesaikannya skripsi ini tidak lepas dari bantuan, dukungan, doa, dan bimbingan dari berbagai pihak. Oleh karena itu, dengan segala kerendahan hati, penulis menyampaikan terima kasih dan penghargaan yang setinggi-tingginya kepada:

1. Prof. Dr. H. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Prof. Dr. Sri Harini, M.Si., selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Elly Susanti, S.Pd., M.Sc., selaku ketua Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Hisyam Fahmi, M.Kom. selaku dosen pembimbing I dan Dr. Fachrur Rozi, M.Si. selaku dosen pembimbing II yang telah membimbing dengan sabar, memberikan ilmu, motivasi, pengalaman, serta masukan yang sangat berarti. Semoga setiap ilmu yang diberikan menjadi amal jariyah yang terus mengalirkan pahala.
5. Dr. Mohammad Jamhuri, M.Si. selaku ketua penguji dan Ibu Ria Dhea Layla Nur Karisma, M.Si., selaku dosen penguji I yang telah meluangkan waktu, tenaga, dan pikiran untuk memberikan saran yang membangun demi kesempurnaan karya ini. Semoga setiap ilmu dan nasihat menjadi catatan amal kebaikan.

6. Seluruh dosen Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang yang telah memberikan ilmu, bimbingan, serta inspirasi yang membentuk karakter dan wawasan penulis selama menempuh pendidikan.
7. Ayah paling berjasa, Abdul Mukid, S.Pdi, sosok panutan yang tak henti menanamkan nilai-nilai tanggung jawab, kesabaran, dan keteguhan hati.
8. Ibu tercinta, Khalifah, yang doanya tak pernah putus, menjadi penopang langkah dan semangat penulis dalam melewati setiap tantangan.
9. Kakak-kakak tersayang Qurrotul A'yun, M.Si. dan Mahsusotur Rohmania, S.Pd. yang senantiasa memberikan semangat, teladan, dan cinta yang tulus.
10. Sahabat-sahabat terbaik penulis, yang telah menjadi bagian penting dalam perjalanan ini, memberi tawa, semangat, dan kekuatan di saat-saat terberat sekalipun.

Penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna. Namun dengan kerendahan hati, karya ini penulis persembahkan sebagai bentuk penghargaan atas semua dukungan dan ilmu yang telah diberikan. Semoga lembaran skripsi ini tidak hanya menjadi akhir dari sebuah perjalanan akademik, tetapi juga menjadi bukti kecil dari rasa terima kasih penulis kepada setiap insan yang senantiasa menjadi tempat berpulang dalam segala keadaan. Penulis berharap, skripsi ini dapat memberikan manfaat bagi pembaca dan menjadi pijakan awal menuju perjalanan ilmiah yang lebih luas.

Wassalamu'alaikum warahmatullahi wabarakatuh

Malang, 23 Mei 2025

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGAJUAN	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	iv
MOTO	vi
PERSEMBAHAN	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiii
DAFTAR SIMBOL	xiv
DAFTAR LAMPIRAN	xv
ABSTRAK	xvi
ABSTRACT	xvii
مستخلص البحث	xviii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	7
1.3 Tujuan Penelitian	7
1.4 Manfaat Penelitian	8
1.5 Batasan Masalah	9
1.6 Definisi Istilah	9
BAB II KAJIAN TEORI	11
2.1 Teori Pendukung	11
2.1.1 <i>Reccurent Neural Network</i>	11
2.1.2 <i>Long Short Term Memory</i>	13
2.1.3 Fungsi Aktivasi	23
2.1.4 <i>Hidden Layer</i> dan <i>Neuron Hidden</i>	26
2.1.5 <i>Batch Size</i>	27
2.1.6 <i>Epoch</i>	28
2.1.7 <i>Optimizer</i>	29
2.1.8 Analisis Teknikal	30
2.1.9 Normalisasi	32
2.1.10 <i>K-fold Cross Validation</i>	33
2.1.11 Evaluasi Model	34
2.1.12 Peramalan	37
2.1.13 Saham	38
2.2 Integrasi Nilai Keislaman dalam Perencanaan Masa Depan Melalui Prediksi	39
BAB III METODE PENELITIAN	44
3.1 Jenis Penelitian	44
3.2 Data dan Sumber Data	44
3.3 Teknik Analisis Data	46
BAB IV HASIL DAN PEMBAHASAN	51

4.2	Dataset.....	51
4.3	<i>Preprocessing</i>	53
4.3.1	<i>Ascending</i>	53
4.3.2	Ekstraksi Fitur	54
4.3.3	Normalisasi	57
4.3.4	Pembagian Data	58
4.3.5	Segmentasi	59
4.4	Konstruksi Model LSTM.....	60
4.5	Perhitungan Manual Metode LSTM	61
4.6	Validasi Model.....	86
4.7	Prediksi Harga Saham.....	88
4.9	Penerapan Prinsip Ikhtiar dan Tawakal dalam Penggunaan Metode LSTM untuk Prediksi Harga Saham.....	95
BAB V	PENUTUP.....	97
5.1	Kesimpulan	97
5.2	Saran	98
DAFTAR PUSTAKA.....		100
LAMPIRAN.....		104
RIWAYAT HIDUP		122

DAFTAR TABEL

Tabel 3.1	Data Saham PT. Indofood Sukses Makmur Tbk	45
Tabel 4.1	Dataset <i>Close Price</i>	51
Tabel 4.2	Perhitungan Nilai Untung Rugi.....	54
Tabel 4.3	Dataset <i>Close Price</i> dan Nilai RSI.....	56
Tabel 4.4	Hasil Deskriptif Dataset <i>Close Price</i> dan Nilai RSI	56
Tabel 4.5	Hasil Normalisasi Dataset <i>Close Price</i> dan Nilai RSI	58
Tabel 4.6	Hasil Pembagian Dataset <i>Close Price</i> dan Nilai RSI.....	59
Tabel 4.7	Ilustrasi Pengelompokan Data dalam Bentuk Sekuensial.....	60
Tabel 4.8	Nilai <i>forget gate</i> , <i>input gate</i> , <i>output gate</i> , dan <i>cell state</i>	85
Tabel 4.9	Hasil Nilai <i>Error</i> Menggunakan RMSE dan MAPE saat Pelatihan.....	86
Tabel 4.10	Perbandingan Data Aktual dan Hasil Prediksi <i>Close Price</i> dan RSI.....	89
Tabel 4.11	Nilai <i>Error Close Price</i> Menggunakan LSTM, MA, dan ARIMA.....	92
Tabel 4.12	Nilai <i>Error RSI</i> Menggunakan LSTM, MA, dan ARIMA	92
Tabel 4.13	Perbandingan Prediksi Saham PT Indofood dari Penelitian Terdahulu	94

DAFTAR GAMBAR

Gambar 2.1 Struktur RNN	12
Gambar 2.2 Struktur LSTM	15
Gambar 2.3 Grafik Fungsi Aktivasi Sigmoid Biner.....	24
Gambar 2.4 Grafik Fungsi Aktivasi <i>Tanh</i>	26
Gambar 2.5 Grafik <i>Overfitting</i> , Optimum, <i>Underfitting</i> dari Jumlah Epoch...	28
Gambar 2.6 <i>Time Series Cross Validation</i>	34
Gambar 3.1 QR Code Data Saham	44
Gambar 3.2 Alur Penelitian Prediksi Harga Saham dengan Metode LSTM ...	46
Gambar 4.1 Grafik Data Saham INDF.JK	52
Gambar 4.2 Grafik Perbandingan Loss dan Epoch Data Training.....	88
Gambar 4.3 Grafik Perbandingan Hasil Prediksi dan Data Aktual <i>Close Price</i>	90
Gambar 4.4 Grafik Perbandingan Hasil Prediksi dan Data Aktual Nilai RSI .	91

DAFTAR SIMBOL

ϕ_t	: <i>Forget gate</i>
k_t	: <i>Input gate</i>
c_t	: <i>Cell state</i>
ω_t	: <i>Output gate</i>
$S(x)$	: Fungsi Aktivasi sigmoid
$\tanh t$	: Fungsi aktivasi tanh

DAFTAR LAMPIRAN

Lampiran 1 Data Harga Penutupan Saham PT Indofood dan Nilai RSI	104
Lampiran 2 <i>Script Phyton</i> Prediksi Saham Menggunakan Metode <i>Long Short Term Memory</i>	115

ABSTRAK

Amalia, Khoiro. 2025. **Implementasi Long Short Term Memory untuk Prediksi Harga Saham Perusahaan di Bidang Pangan**. Skripsi. Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Hisyam Fahmi, M.Kom. (II) Dr. Fachrur Rozi, M.Si.

Kata kunci: *Long Short Term Memory* (LSTM), Prediksi Harga Saham, PT Indofood Sukses Makmur Tbk, *Time Series*, *Relative Strength Index* (RSI), RMSE, MAPE.

Long Short Term Memory (LSTM) adalah salah satu jenis jaringan saraf tiruan yang dirancang untuk mengolah data deret waktu, dengan kemampuan mengingat informasi dalam jangka panjang melalui mekanisme *memory cell* dan *gate units*. LSTM efektif dalam mengenali pola historis yang kompleks, seperti pergerakan harga saham. Penelitian ini bertujuan untuk mengevaluasi efektivitas metode LSTM dalam memprediksi harga saham PT Indofood Sukses Makmur Tbk dan mengukur tingkat akurasi prediksinya. Data yang digunakan berupa harga penutupan saham periode 2 Juni 2021 hingga 31 Januari 2025 dari *Yahoo Finance*, dengan penambahan indikator teknikal *Relative Strength Index* (RSI) sebagai fitur tambahan. Model terbaik diperoleh dengan kombinasi *hidden layer* sebanyak 4, 50 neuron, *batch size* 4, 100 *epoch*, *optimizer* Adam, serta fungsi aktivasi *tanh* dan *sigmoid*. Hasil pengujian menunjukkan nilai *error* RMSE sebesar 0,0898 dan MAPE 1,05%. Selain itu, model LSTM juga akan dibandingkan dengan metode *Moving Average* dan ARIMA, berdasarkan hasil perbandingan model LSTM memberikan hasil prediksi lebih baik dibandingkan metode *Moving Average*.

ABSTRACT

Amalia, Khoiro. 2025. **Implementation of Long Short Term Memory for Stock Price Prediction of Companies in the Food Sector.** Thesis. Mathematic Study Program, Faculty of Science and Technology, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Supervisor: (I) Hisyam Fahmi, M.Kom. (II) Dr. Fachrur Rozi, M.Si.

Keywords : Long Short Term Memory (LSTM), Stock Price Prediction, PT Indofood Sukses Makmur Tbk, Time Series, Relative Strength Index (RSI), RMSE, MAPE.

Long Short-Term Memory (LSTM) is a type of artificial neural network designed to process time series data, with the ability to remember information in the long term through the mechanism of memory cells and gate units. LSTM effectively recognizes complex historical patterns, such as stock price movements. This study aims to evaluate the effectiveness of the LSTM method in predicting the stock price of PT Indofood Sukses Makmur Tbk and measuring the accuracy of its predictions. The data used is the closing stock price period June 2, 2021, to January 31, 2025, from Yahoo Finance, with the addition of the Relative Strength Index (RSI) technical indicator as an additional feature. The best model is obtained with a combination of 4 hidden layers, 50 neurons, batch size 4, 100 epochs, Adam optimizer, and tanh and sigmoid activation functions. The test results show an error value of RMSE of 0,0898 and MAPE of 1,05%. In addition, the LSTM model is also compared with the Moving Average and ARIMA methods, where the comparison results show that the LSTM model provides better prediction performance than the Moving Average method.

مستخلص البحث

أماليا ، خويرو. 2025. تنفيذ الذاكرة طويلة المدى قصيرة المدى للتنبؤ بأسعار أسهم الشركة في قطاع الأغذية. البحث العلمي. قسم الرياضيات ، كلية العلوم والتكنولوجيا ، جامعة مولانا مالك إبراهيم الإسلامية الحكومية مالانج. المشرف: (1) هشام فهمي، الماجستير (2) فخر الرازي، الماجستير.

الكلمات المفتاحية: الذاكرة طويلة المدى قصيرة المدى ($LSTM$) ، التنبؤ بسعر السهم ، $PT Indofood$ ، $Sukses Makmur Tbk$ ، السلاسل الزمنية ، مؤشر القوة النسبية $MAPE$ ، $RMSE$ ، (RSI) .

الذاكرة طويلة المدى قصيرة المدى ($LSTM$) هي نوع من الشبكات العصبية الاصطناعية المصممة لمعالجة بيانات السلاسل الزمنية ، مع القدرة على ذكر المعلومات على المدى الطويل من خلال آليات خلايا الذاكرة ووحدات البوابة. تعتبر $LSTMs$ فعالة في التعرف على الأنماط التاريخية المعقدة ، مثل تحركات أسعار الأسهم. تهدف هذه الدراسة إلى تقييم فعالية طريقة $LSTM$ في التنبؤ بسعر سهم $PT Indofood Sukses Makmur Tbk$ وقياس دقة تنبؤاتها. البيانات المستخدمة هي في شكل أسعار إغلاق الأسهم للفترة من 2 يونيو 2021 إلى 31 يناير 2025 من *Yahoo Finance* ، مع إضافة المؤشر الفني لمؤشر القوة النسبية (RSI) كميزة إضافية. تم الحصول على أفضل نموذج بمزيج من 4 ، 100 خلية عصبية ، وحجم دفعة من 4 ، و 50 حقبة ، ومحسن آدم ، ووظائف تنشيط $tanh$ و $symmoid$. أظهرت نتائج الاختبار قيمة خطأ $RMSE$ تبلغ 0,0898 و $MAPE$ بنسبة 1,05٪. بالإضافة إلى ذلك ، ستتم أيضا مقارنة نموذج $LSTM$ بطريقتي المتوسط المتحرك و $ARIMA$ ، بناء على نتائج مقارنة نموذج $LSTM$ لتقديم نتائج تنبؤ أفضل من طريقة المتوسط المتحرك.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Semakin berkembangnya *digital* di era sekarang maka semakin berkembang pula sistem *artificial intelligence* (AI) atau kecerdasan buatan dalam berbagai bidang terutama untuk mengelola dan menganalisis data. Salah satu disiplin ilmu dalam kecerdasan buatan yang telah berkembang pesat dan mampu menangani data yang kompleks serta memberikan prediksi yang lebih akurat adalah *machine learning*. *Machine learning* adalah salah satu aplikasi dalam kecerdasan buatan (AI) yang memanfaatkan data atau informasi yang sudah ada untuk dijadikan bahan pembelajaran dengan tujuan meningkatkan kinerja atau membuat prediksi yang lebih akurat. *Machine learning* berfokus kepada pengembangan sistem yang dapat belajar secara mandiri tanpa perlu dilakukan pemrograman berulang kali (Melinda dkk., 2020). *Machine learning* dapat diartikan sebagai penerapan aplikasi komputer dan algoritma matematika dengan menggunakan metode pembelajaran berbasis data yang sudah diamati sebelumnya untuk membuat kesimpulan baru di masa depan dengan benar.

Machine learning terdiri dari berbagai macam algoritma dimana masing-masing memiliki kelebihan dan kekurangannya sendiri, diantaranya yaitu *linear regression*, *random forest*, *decision tree*, *naive bayes*, *Recurrent Neural Network* (RNN), dan lainnya. *Recurrent Neural Network* (RNN) yang juga dikenal sebagai jaringan umpan balik, merupakan salah satu jenis jaringan *neural networks* dimana didalamnya terdapat *loop* yang memungkinkan neuron-neuronnya saling mengirim koneksi umpan balik (Rizal & Soraya, 2018). RNN memiliki akurasi tinggi dalam

memprediksi data deret waktu karena jaringan pada RNN memungkinkan *output*-nya digunakan kembali sebagai *input*, yang kemudian diproses untuk menghasilkan *output* baru. Namun, RNN memiliki kelemahan karena tidak mampu mengingat dan menyimpan informasi untuk jangka waktu yang lama dengan efektif. Oleh karena itu, dikembangkanlah varian yang lebih optimal dari RNN, yaitu *Long Short Term Memory* atau LSTM.

Long Short Term Memory (LSTM) merupakan salah satu pengembangan dari RNN yang dimodifikasi dengan penambahan *memory cell* yang memungkinkan penyimpanan informasi dalam jangka waktu yang lebih panjang khususnya data yang sangat besar. *Memory cell* pada LSTM juga mampu mengatasi masalah *vanishing gradient* yang sering muncul pada RNN ketika memproses data sekuensial yang panjang (Pontoh dkk., 2022). *Vanishing gradient* terjadi ketika gradien yang digunakan untuk memperbarui bobot jaringan menjadi sangat kecil hingga mendekati nol seiring dengan banyaknya lapisan atau urutan waktu yang dilalui. LSTM tidak memiliki kekurangan yang dimiliki RNN karena LSTM memanfaatkan *memory cells* dan *gate unit* untuk mengatur informasi pada setiap masukannya. Terdapat empat *gates units* pada LSTM yaitu *forget gates*, *cell gates*, *input gates*, dan *output gates*. *Memory cells* dalam LSTM pada dasarnya digunakan untuk menyimpan informasi jarak jauh dan melacak berbagai atribut dari teks yang sedang diproses (Karpathy dkk., 2016). Metode LSTM telah banyak digunakan untuk memprediksi data deret waktu seperti harga saham perusahaan (Qotrunnada, 2022), nilai tukar IDR terhadap USD (Andrijasa dkk., 2018), nilai mata uang Bitcoin (Aldi dkk., 2018), pergerakan nilai harga emas (Owen dkk., 2022), dan lainnya.

Perekonomian suatu negara sangat dipengaruhi dengan adanya pasar modal. Selain berfungsi sebagai instrumen keuangan jangka panjang yang dapat diperdagangkan dengan durasi lebih dari satu tahun, pasar modal juga berfungsi sebagai pendanaan untuk perusahaan dan lembaga lainnya (pemerintah), serta menyediakan kesempatan investasi bagi para investor (Nurhayati & Wasilah, 2013). Pasar modal memungkinkan terjadinya transaksi jual beli saham, menyediakan sumber dana bagi bisnis, mendukung investasi, serta melibatkan berbagai derivatif dari efek atau surat berharga. Investasi di dalam pasar modal adalah kegiatan menanamkan modal pada satu atau lebih aset yang dimiliki pada masa sekarang dan umumnya memiliki jangka waktu yang panjang, dengan tujuan mendapatkan keuntungan di masa mendatang, salah satu contohnya yaitu investasi dalam bentuk saham.

Saham merupakan bukti kepemilikan modal atau dana yang disertakan dalam suatu perusahaan yang kemudian digunakan untuk mendanai aktivitas operasional perusahaan. Saat ini, dunia keuangan sangat terfokus pada pasar saham. Pasar saham menawarkan potensi keuntungan yang besar, namun juga disertai dengan risiko yang signifikan. Adanya penawaran dan permintaan atas saham menyebabkan harga saham tidak selalu stabil maupun konstan. Permintaan dan penawaran saham dipengaruhi oleh banyak hal, seperti kinerja bisnis dan industrinya, serta beberapa faktor yang menjadi variabel utamanya seperti nilai tukar mata uang, inflasi, dan suku bunga. Oleh karena itu, prediksi harga saham menjadi topik menarik bagi para pelaku pasar keuangan, investor, dan analis. Para pelaku investasi saham dituntut untuk mempelajari cara memprediksi harga saham. Pengetahuan ini penting untuk memberikan solusi dalam transaksi jual-beli saham.

guna terhindar dari kerugian. Allah berfirman dalam Q.S An-Nahl Ayat 78 yang artinya (Departemen Agama, 2015)

“Allah mengeluarkan kamu dari perut ibumu dalam keadaan tidak mengetahui sesuatu pun dan Dia menjadikan bagi kamu pendengaran, penglihatan, dan hati nurani agar kamu bersyukur”

Maksud yang terkandung dalam ayat di atas yang diintegrasikan dalam bidang keilmuan ini menunjukkan bahwa Allah SWT membekali manusia dengan sarana-prasarana untuk mendapatkan ilmu pengetahuan dan wawasan berupa pendengaran, penglihatan dan hati, dimana sebelumnya manusia dilahirkan tanpa mengetahui sesuatu apapun. Allah SWT memberikan akal pikiran kepada manusia yang memungkinkannya untuk memahami berbagai aspek kehidupan, membedakan antara kebaikan dan keburukan, serta mampu memilih sesuatu yang membawa manfaat dan yang tidak. Allah SWT menyediakan kunci-kunci pengetahuan bagi manusia, berupa pendengaran yang dapat digunakan untuk mendengar dan memahami suara, serta penglihatan yang dapat digunakan untuk melihat dan mengamati berbagai hal. Dengan adanya sarana tersebut, manusia dapat belajar dan menggunakan indra serta akal yang diberikan untuk memperoleh ilmu dan mengembangkannya. Penggunaan teknologi modern seperti LSTM dalam memprediksi harga saham adalah salah satu bentuk pengembangan ilmu pengetahuan yang diberikan Allah SWT kepada manusia, penelitian ini juga dapat dianggap sebagai ungkapan rasa syukur terhadap nikmat pengetahuan yang Allah SWT berikan (Az-Zuhaili, 2013).

Peramalan (*forecasting*) adalah seni dan ilmu untuk membuat prediksi menggunakan data historis untuk memperkirakan kejadian di masa yang akan datang. Prediksi harga saham ini sangat dibutuhkan oleh para investor dalam

pemutusan berinvestasi untuk memaksimalkan keuntungan dan meminimalkan kerugian. Para pemilik saham harus mengetahui bagaimana prediksi pergerakan harga saham agar dapat memperkirakan kapan akan menjual dan membeli kepemilikan saham. Terdapat banyak metode pada *machine learning* yang dapat digunakan untuk memprediksi data. Pada penelitian ini, penulis akan menggunakan metode LSTM untuk memprediksi harga saham perusahaan yang bergerak di bidang pangan khususnya pada PT. Indofood Sukses Makmur Tbk. Perusahaan yang bergerak di bidang pangan mengalami peningkatan penjualan dan pendapatan seiring dengan meningkatnya permintaan pangan yang merupakan kebutuhan dasar manusia. Kondisi ini berdampak positif terhadap kinerja keuangan perusahaan, sehingga menarik minat investor untuk membeli saham yang pada akhirnya berdampak pada pergerakan harga saham perusahaan tersebut. Demikian pula PT. Indofood Sukses Makmur Tbk juga mengalami kenaikan pada harga sahamnya. Perusahaan ini termasuk dalam 50 besar kapitalisasi pasar pada periode Juni 2024 menurut Bursa Efek Indonesia (BEI). Selain itu, grafik harga saham PT. Indofood Sukses Makmur Tbk cenderung mengalami peningkatan selama 5 tahun terakhir. Metode LSTM dipilih karena keunggulan metode ini dalam memproses data deret waktu seperti harga saham dimana urutan dari setiap datanya sangat penting. LSTM secara khusus mampu menangkap hubungan antara harga saham pada hari-hari sebelumnya dengan pergerakan harga di hari-hari berikutnya, dikarenakan metode ini memiliki fitur *memory cell* dan *gate units* yang membantu dalam memproses data yang sangat besar juga mengingat informasi dalam jangka waktu yang lama. Dengan keunggulan fitur yang dimiliki metode LSTM, diharapkan dapat memberikan hasil prediksi harga saham yang lebih akurat.

Merujuk pada penelitian sebelumnya yang telah dilakukan oleh Qotrunnada pada tahun 2022 mengenai prediksi harga saham dengan menggunakan metode *Long Short Term Memory* serta menggunakan parameter analisis teknikal. Terdapat 7 fitur yang digunakan untuk memprediksi harga saham diantaranya *date*, *close*, *open*, *high*, *low*, dan *volume*, dengan menggunakan data saham *time series* milik perusahaan PT. Erajaya Swasembada Tbk yang didapatkan melalui *website yahoo finance* periode Desember 2011 - Desember 2021. Pada penelitian tersebut perhitungan nilai *error* yang didapatkan dengan teknikal analisis yang digunakan yaitu RSI, dengan menggunakan RMSE sebesar 12,07 dan MAPE sebesar 1,86% dari model yang paling optimal dengan parameter *batch size* sebanyak 4 dan *epoch* maksimum 100. Kemudian penelitian selanjutnya pernah dilakukan oleh Kartika dan Karmilasari pada tahun 2022 dengan metode yang sama memprediksi harga saham PT. Aneka Tambang Tbk. Terdapat 6 fitur yang diidentifikasi pada penelitian ini yaitu *Open*, *High*, *Low*, *Close*, *Adj Close*, dan *Volume*. Data saham didapatkan melalui *website yahoo finance* periode 29 September 2005 hingga 13 April 2021. Pada penelitian ini dengan menggunakan *epoch* 100 didapatkan nilai RMSE yang hampir mendekati 0 sebesar 0,3304 dan nilai R2 yang hampir mendekati nilai 1 yaitu sebesar 0,9902. Penelitian lainnya pernah dilakukan oleh Pratama dan Banowosari pada tahun 2024 dengan membandingkan metode *Extreme Gradient Boosting* (XGBOOST) dan metode *Long Short Term Memory* (LSTM) untuk memprediksi harga saham milik PT. Bank Mandiri Tbk. Pada penelitian ini diperoleh hasil bahwa metode LSTM merupakan metode terbaik dalam memprediksi harga saham ditunjukkan dengan perolehan nilai *error* dengan menggunakan R2 sebesar 98.44% dan MAPE sebesar 1.77 sedangkan metode

XGBOOST memperoleh hasil nilai *error* dengan menggunakan R2 sebesar 89.09% dan MAPE sebesar 3.21%.

Penelitian ini bertujuan untuk mengevaluasi efektivitas metode LSTM dalam memprediksi harga saham perusahaan di bidang pangan khususnya PT. Indofood Sukses Makmur Tbk serta mengidentifikasi tingkat akurasi prediksinya. Dengan demikian, diharapkan penelitian ini dapat bermanfaat bagi perkembangan ilmu pengetahuan dan teknologi serta memberikan kontribusi positif bagi masyarakat dengan membantu investor membuat keputusan yang lebih baik dan mengurangi risiko di masa depan.

1.2 Rumusan Masalah

Berdasarkan penjelasan pada latar belakang maka rumusan masalah yang akan diambil dalam penelitian ini sebagai berikut:

1. Bagaimana evaluasi performa metode *Long Short Term* untuk prediksi harga saham PT. Indofood Sukses Makmur Tbk?
2. Bagaimana hasil prediksi harga saham PT. Indofood Sukses Makmur Tbk pada periode 1 Desember 2024 - 31 Januari 2025 menggunakan metode *Long Short Term Memory*?

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah maka tujuan penelitian yang hendak dicapai adalah sebagai berikut:

1. Mengetahui hasil evaluasi performa metode *Long Short Term* untuk prediksi harga saham PT. Indofood Sukses Makmur Tbk

2. Mengetahui hasil prediksi harga saham PT. Indofood Sukses Makmur Tbk. pada periode 1 Desember 2024 - 31 Januari 2025 menggunakan metode LSTM.

1.4 Manfaat Penelitian

Berdasarkan tujuan penelitian di atas, diharapkan setelah melakukan penelitian ini dapat memberikan manfaat sebagai berikut:

1. Manfaat Teoritis

Secara teoritis, penelitian ini diharapkan dapat menambahkan ilmu pengetahuan baru terutama dalam bidang prediksi menggunakan data time series. Dengan mengimplementasikan metode *Long Short Term Memory* untuk memprediksi harga saham dapat dijadikan referensi ataupun acuan mengenai metode yang dapat digunakan bagi penelitian-penelitian selanjutnya yang ingin melakukan prediksi data time series.

2. Manfaat Praktis

Secara praktis hasil penelitian ini memberikan banyak manfaat baik kepada peneliti, pembaca, program studi, dan investor dengan bertambahnya ilmu pengetahuan dan wawasan untuk dikembangkan pada penelitian-penelitian selanjutnya, serta memberi referensi pada investor mengenai metode yang dapat digunakan untuk memprediksi harga saham ataupun data *time series* lainnya sehingga dapat memberikan gambaran mengenai harga saham kedepannya sebelum membuat keputusan investasi.

1.5 Batasan Masalah

Penelitian ini dibatasi pada penggunaan sejumlah parameter guna menjaga relevansi dan keakuratan hasil yang diperoleh. Parameter yang digunakan meliputi jumlah *hidden layer* sebanyak 4, jumlah neuron pada *hidden layer* sebesar 50, nilai *batch size* sebesar 2, 4, 16, dan 64, jumlah *epoch* maksimum sebanyak 50 dan 100, serta menggunakan *optimizer* Adam dan fungsi aktivasi tanh dan sigmoid

1.6 Definisi Istilah

Implementasi	: Penerapan sebuah sistem untuk menimbulkan dampak atau akibat terhadap sesuatu.
Prediksi	: Proses perkiraan sesuatu secara sistematis yang mungkin terjadi dimasa yang akan datang berdasarkan informasi pada masa sebelumnya.
Saham	: Suatu bukti penyertaan modal seseorang atau badan usaha dalam sebuah perusahaan.
LSTM	: Pengembangan dari algoritma RNN yang mampu memproses data deret waktu dan menyimpan informasi untuk jangka waktu yang lama.
Teknikal Analisis	: metode analisis yang digunakan untuk mengevaluasi pergerakan harga aset di pasar keuangan seperti saham, komoditas, dan forex.
Fluktuasi	: Perubahan nilai saham yang tidak menentu di pasar modal.

- Deviasi : Perbedaan antara nilai aktual dan nilai prediksi dan digunakan sebagai indikator seberapa baik model mampu memprediksi data.
- Volatilitas : Merujuk pada tingkat fluktuasi harga saham dalam periode waktu tertentu, volatilitas yang tinggi menunjukkan adanya perubahan harga yang signifikan dalam waktu singkat, sedangkan volatilitas rendah menunjukkan pergerakan harga yang lebih stabil.

BAB II

KAJIAN TEORI

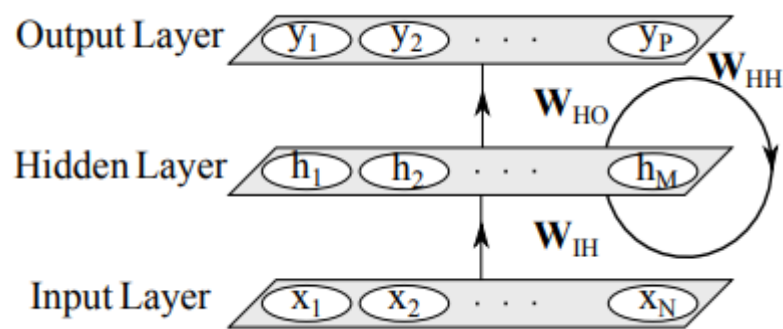
2.1 Teori Pendukung

2.1.1 *Reccurent Neural Network*

Recurrent Neural Network (RNN) merupakan salah satu tipe dari jaringan pemodelan *Neural Network*. Selain RNN, terdapat juga *Multilayer Perceptron* (MLP) dan *Feedforward Neural Network* (FNN) yang termasuk dalam kategori pemodelan *Neural Network*. RNN adalah pengembangan dari dasar pemikiran FNN yang telah banyak diimplementasikan untuk melakukan prediksi atau klasifikasi data selama bertahun-tahun. Penggunaan *neural network* ini umumnya melibatkan *input* dan *output* yang tidak saling bergantung, sehingga dapat menyebabkan penumpukan tugas pada jaringan tersebut. Namun, untuk membangun aplikasi yang mengelola data ilmiah dengan sampel data berbentuk deret waktu (*time series*), diperlukan algoritma yang mampu menangani data tersebut dengan cepat dan akurat. Oleh karena itu, dikembangkan sistem RNN untuk memproses data-data yang bersifat sekuensial atau bertahap, karena RNN dirancang khusus di mana setiap neuron dalam jaringan menerima input, melakukan operasi komputasi, dan mengeluarkan *output* pada setiap iterasi.

RNN merupakan salah satu jenis jaringan dalam *neural network* yang memiliki *loop* sebagai koneksi umpan balik di dalam strukturnya. RNN juga disebut jaringan umpan balik karena meniru cara kerja otak manusia, sehingga memungkinkan pengiriman informasi dari satu neuron ke neuron lainnya (Suyudi dkk., 2019). Perbedaan utama di antara RNN dan jaringan saraf biasa adalah bahwa RNN memiliki memori yang baik karena *output*-nya bergantung

pada komputasi sebelumnya. Selain itu, RNN memiliki kemampuan untuk menyimpan informasi dari langkah-langkah komputasi sebelumnya dan menerapkannya pada elemen-elemen berikutnya dalam urutan *input*, sehingga memungkinkan pengenalan pola data secara efektif dan menghasilkan prediksi yang akurat. Keterampilan ini membuat RNN sangat cocok untuk memodelkan urutan data. RNN mampu mengatasi urutan input yang panjang, mengelola ketergantungan jarak jauh, dan dengan mudah memperhitungkan konteks informasi pada setiap langkah waktu saat memproses urutan kata dalam teks.



Gambar 2.1 Struktur RNN

RNN terdiri dari tiga lapisan utama yaitu lapisan *input*, lapisan tersembunyi (*hidden layer*), dan lapisan *output* seperti ditunjukkan pada Gambar 2.1 (Salehinejad dkk., 2018). Lapisan *input* terdiri dari unit-unit *input* yang terhubung sepenuhnya dengan unit-unit pada lapisan tersembunyi. Unit-unit pada lapisan tersembunyi saling terhubung secara berulang sehingga memungkinkan jaringan memiliki memori internal. Lapisan tersembunyi berfungsi sebagai ruang penyimpanan informasi yang mampu mengingat dan memproses informasi dari waktu sebelumnya serta terhubung secara paralel dengan lapisan *input* maupun lapisan *output*. Semua informasi penting yang diterima akan dirangkum menjadi suatu informasi akhir dari jaringan melalui serangkaian langkah waktu. Informasi

akhir tersebut kemudian digunakan untuk menghasilkan keluaran pada lapisan output, dengan begitu RNN dapat membuat prediksi berdasarkan urutan data yang diterima sebelumnya.

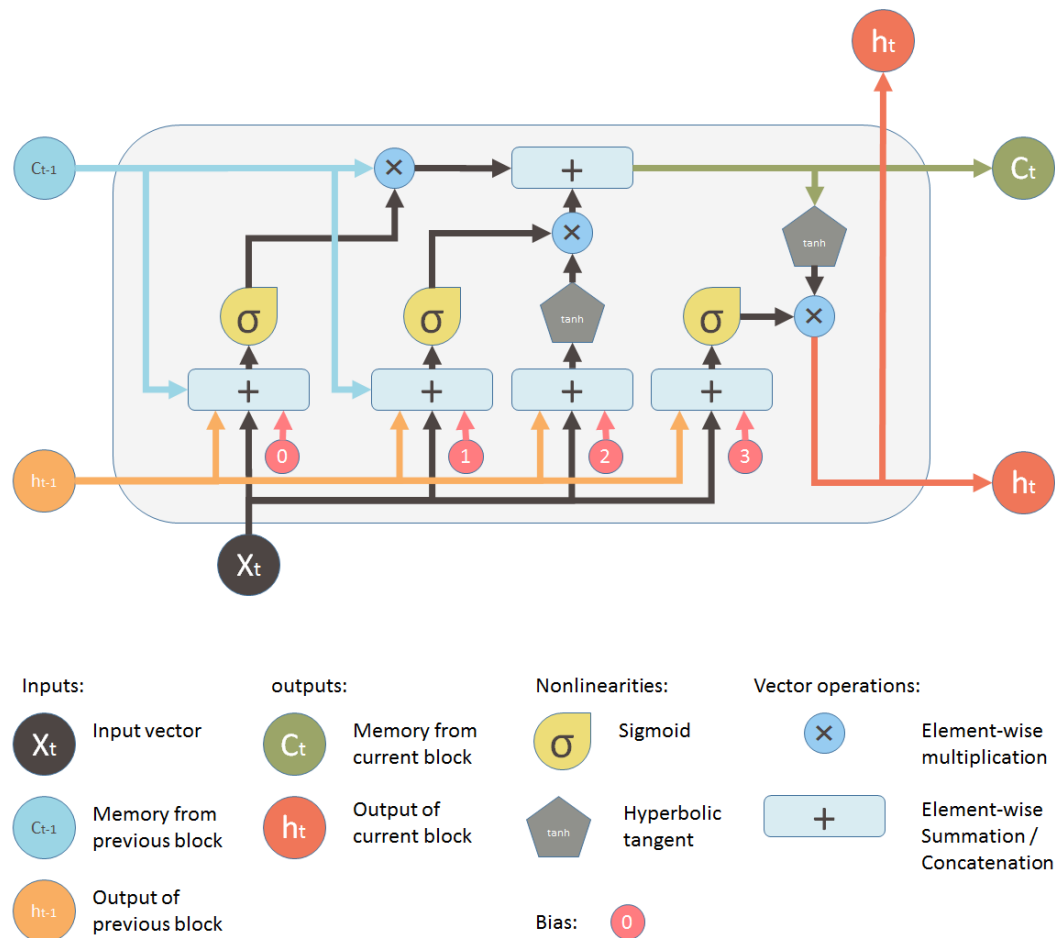
Salah satu kelemahan RNN adalah bahwa pembelajaran jangka panjang menggunakan *gradient descent* dapat menyebabkan masalah berupa menghilangnya atau meledaknya gradien (Salehinejad dkk., 2018). Selama proses pembelajaran dalam jaringan saraf tiruan, nilai gradien yang digunakan untuk memperbarui bobot jaringan terus dikalikan dengan bobot yang lebih kecil dari satu. Hal ini dilakukan untuk mencegah nilai gradien menjadi terlalu kecil atau terlalu besar, yang dapat memperlambat atau bahkan menghentikan proses pembelajaran. Salah satu solusi untuk mengatasi masalah menghilangnya atau meledaknya gradien dalam jaringan saraf tiruan adalah dengan memodifikasi arsitektur model. Ini dilakukan dengan memperkenalkan unit gerbang yang khusus dirancang untuk mempertahankan informasi dalam jangka waktu yang lebih lama. Salah satu kemajuan yang lebih efisien dari RNN adalah LSTM. LSTM dirancang untuk mengatasi masalah yang terdapat pada RNN, yaitu keterbatasan memori jangka pendek (Suyudi dkk., 2019). LSTM sering digunakan untuk memprediksi data jangka panjang karena memiliki memori yang dapat menyimpan informasi untuk digunakan kembali dalam proses perhitungan pada gerbang berikutnya.

2.1.2 Long Short Term Memory

Long Short Term Memory (LSTM) pertama kali diperkenalkan oleh Hochreiter & Schmidhuber pada tahun 1997. LSTM merupakan pengembangan

dari metode RNN yang dirancang untuk mengatasi masalah *vanishing gradient* pada RNN saat melakukan proses data sekuensial yang panjang. *Vanishing gradient* ini terjadi karena aktivasi fungsi sigmoid atau tanh yang digunakan dalam RNN, yang membuat nilai-nilai gradien menjadi sangat kecil setelah beberapa waktu. Informasi dari waktu ke waktu yang diproses dan disimpan dalam *hidden state*, akan membuat nilai-nilai gradien cenderung menjadi sangat kecil ketika gradien-gradien disalurkan kembali ke lapisan awal. Hal ini disebabkan adanya penjumlahan berulang-ulang dari produk-produk gradien yang semakin kecil. Akibatnya, lapisan awal tidak dapat diperbarui dengan efektif, sehingga informasi lama yang disimpan menjadi tidak berguna. Untuk mengatasi masalah ini, LSTM dikembangkan dengan menambahkan konsep *memory cells* yang dapat menyimpan informasi dalam waktu yang lama (Pontoh dkk., 2022).

LSTM memiliki struktur rangkaian modul jaringan syaraf yang berulang seperti RNN dengan penambahan fitur gerbang *cell (cell gate)* didalamnya. LSTM dirancang untuk mengelola memori pada setiap masukan dengan memanfaatkan *gate units* dan *memory cells*. Dengan struktur ini, LSTM dapat menghafal dan mengingat kembali informasi untuk jangka waktu yang lama, terutama untuk menangani data yang sangat besar. Salah satu keunggulan utama LSTM adalah kemampuannya untuk menyimpan informasi jarak jauh, serta melacak berbagai atribut dari teks yang sedang diproses (Karpathy dkk., 2016).



Gambar 2.2 Struktur LSTM

LSTM terdiri dari empat *gates* yang mengontrol aliran informasi dalam jaringannya yang kemudian disebut sebagai *gates units*. *Gates units* tersebut yaitu *forget gate*, *input gate*, *cell state*, dan *output gate* seperti pada Gambar 2.2 (Yan, 2016).

1. *Forget gate* (ϕ_t)

Forget gate merupakan komponen paling pertama pada diagram dimana terdapat simbol fungsi sigmoid (σ) didalamnya. *Forget gate* akan mengatur informasi mana yang akan disimpan atau harus dibuang dari *memory cell*. Informasi ini diatur menggunakan fungsi aktivasi sigmoid

yang bertujuan untuk mentransformasikan nilai antara -1 dan 1 menjadi nilai antara 0 dan 1 .

$$\text{Sigmoid}(x) = S(x) = \frac{1}{1 + e^{(-x)}} \quad (2.1)$$

Keterangan:

$S(x)$ = Nilai *output* dari fungsi sigmoid untuk input x

x = Nilai *input*

e = Bilangan *euler* dengan nilai sekitar $(2,71828)$

e^{-x} = Eksponensial dari $(-x)$

$S(x)$ = Nilai *output* dari fungsi sigmoid untuk input x

Jika nilai yang dihasilkan adalah 1 maka informasi akan disimpan dan jika nilai yang dihasilkan adalah 0 maka informasi akan dihapus (Karpathy dkk, 2016). Nilai *forget gate* dapat dihitung menggunakan Persamaan (2.2)

$$\phi_t = \sigma (W_{\phi}^T x_t + W_{h\phi}^T h_{t-1} + b_{\phi}) \quad (2.2)$$

Keterangan:

ϕ_t = *Forget gate* berbentuk vektor berdimensi m ($\mathbb{R}^m$)

W_{ϕ} = Matriks bobot untuk *input* berukuran $\mathbb{R}^{d \times m}$

$W_{h\phi}$ = Matriks bobot untuk *hidden state* berukuran $\mathbb{R}^{m \times m}$

x_t = *Input* vektor berdimensi d ($\mathbb{R}^d$)

h_{t-1} = Vektor *hidden state* berdimensi m ($\mathbb{R}^m$) dalam *timestep* sebelumnya $(t - 1)$

b_{ϕ} = Vektor bias berdimensi m ($\mathbb{R}^m$)

d = Banyaknya fitur *input*

m = Banyaknya *hidden units*

Nilai *weight* dan bias akan ditentukan secara acak oleh program dan akan terus diperbarui selama proses pelatihan (Hocreiter dkk., 1997).

Pembaruan nilai *weight* dan bias terdapat pada Persamaan (2.3), (2.4), dan (2.5)

$$W_{\phi} = W_{\phi} - \eta \cdot \nabla_{W_{\phi}} \quad (2.3)$$

$$W_{h\phi} = W_{h\phi} - \eta \cdot \nabla_{W_{h\phi}} \quad (2.4)$$

$$b_{\phi} = b_{\phi} - \eta \cdot \nabla_{b_{\phi}} \quad (2.5)$$

Keterangan:

$W_{h\phi}$	=	Bobot <i>hidden layer</i>
W_{ϕ}	=	Bobot <i>input</i>
b_{ϕ}	=	Bias
η	=	<i>Learning rate</i>
$\nabla_{W_{h\phi}}$	=	$\sum_{t=1}^T h_{t-1} \cdot (\delta_t^{\phi})^T$
$\nabla_{W_{\phi}}$	=	$\sum_{t=1}^T x_t \cdot (\delta_t^{\phi})^T$
$\nabla_{b_{\phi}}$	=	$\sum_{t=1}^T \delta_t^{\phi}$
δ_t^{ϕ}	=	$\delta_t^c \odot c_{t-1} \odot (1 - \phi_t) \odot \phi_t$
δ_t^c	=	$\delta_t^h \odot \omega_t \odot (1 - c_t \odot c_t) + \delta_{t+1}^c \odot \phi_{t+1}$
δ_t^h	=	$\partial f_t^h \odot (W_{\omega} \cdot \delta_t^{\omega})$
∂f_t^h	=	$(1 - h_t \odot h_t)$

2. *Input gate* (k_t)

Input gate terdapat pada bagian tengah diagram dengan simbol sigmoid (σ) dan *tanh* yang saling berhubungan. *Input gate* akan mengolah informasi melalui komponen (k_t) menggunakan fungsi aktivasi sigmoid yang menghasilkan nilai antara 0 dan 1. Jika nilai (k_t) mendekati 0 maka informasi baru akan dihapus dan tidak akan berkontribusi pada pembaruan *cell state* (c_t), sedangkan jika (k_t) mendekati 1 maka

informasi baru akan diizinkan untuk masuk dan berkontribusi pada pembaruan *cell state* (c_t).

Kemudian pada langkah ini juga membentuk kandidat vektor baru menggunakan fungsi aktivasi tanh untuk memastikan nilai kandidat berada dalam rentang $[-1, 1]$ agar tidak menghasilkan nilai yang terlalu besar atau terlalu kecil yang dapat mempengaruhi stabilitas pelatihan. Nilai yang mendekati (-1) dapat mempresentasikan informasi penurunan harga, sedangkan nilai mendekati 1 dapat mempresentasikan informasi kenaikan harga. Setelah itu, kedua komponen tersebut digabungkan untuk menghasilkan pembaruan-pembaruan yang akan diteruskan ke tahap berikutnya. Proses tersebut terdapat pada Persamaan (2.6) dan (2.7)

$$k_t = \sigma (W_k^T x_t + W_{hk}^T h_{t-1} + b_k) \quad (2.6)$$

$$u_t = \tanh (W_u^T x_t + W_{hu}^T h_{t-1} + b_u) \quad (2.7)$$

Keterangan:

k_t	=	<i>Input gate</i> berbentuk vektor berdimensi m ($\mathbb{R}^m$)
u_t	=	<i>Cell gate</i> berbentuk vektor berdimensi m ($\mathbb{R}^m$)
W_{hk}^T, W_{hu}^T	=	Matriks bobot untuk <i>hidden state</i> berukuran $\mathbb{R}^{m \times m}$
W_k^T, W_u^T	=	Matriks bobot untuk <i>input</i> berukuran $\mathbb{R}^{d \times m}$
x_t	=	<i>Input</i> vektor berdimensi d ($\mathbb{R}^d$)
h_{t-1}	=	Vektor <i>hidden state</i> berdimensi m ($\mathbb{R}^m$) dalam <i>timestep</i> sebelumnya ($t - 1$)
b_k, b_u	=	Vektor bias berdimensi m ($\mathbb{R}^m$)
d	=	Banyaknya fitur <i>input</i>
m	=	Banyaknya <i>hidden units</i>

Pembaruan nilai *weight* dan bias pada *input gate* terdapat pada Persamaan (2.8) sampai dengan (2.13)

$$W_k = W_k - \eta \cdot \nabla_{W_k} \quad (2.8)$$

$$W_u = W_u - \eta \cdot \nabla_{W_u} \quad (2.9)$$

$$W_{hk} = W_{hk} - \eta \cdot \nabla_{W_{hk}} \quad (2.10)$$

$$W_{hu} = W_{hu} - \eta \cdot \nabla_{W_{hu}} \quad (2.11)$$

$$b_k = b_k - \eta \cdot \nabla_{b_k} \quad (2.12)$$

$$b_u = b_u - \eta \cdot \nabla_{b_u} \quad (2.13)$$

Keterangan:

W_{hk}, W_{hu} = Bobot *hidden layer*

W_k, W_u = Bobot *input*

b_k, b_u = Bias

η = *Learning rate*

$\nabla_{W_{hk}}$ = $\sum_{t=1}^{\tau} h_{t-1} \cdot (\delta_t^k)^T$

$\nabla_{W_{hu}}$ = $\sum_{t=1}^{\tau} h_{t-1} \cdot (\delta_t^u)^T$

∇_{W_k} = $\sum_{t=1}^{\tau} x_t \cdot (\delta_t^k)^T$

∇_{W_u} = $\sum_{t=1}^{\tau} x_t \cdot (\delta_t^u)^T$

∇_{b_k} = $\sum_{t=1}^{\tau} \delta_t^k$

∇_{b_u} = $\sum_{t=1}^{\tau} \delta_t^u$

δ_t^k = $\delta_t^c \odot u_t \odot (1 - k_t) \odot k_t$

δ_t^u = $\delta_t^c \odot k_t \odot (1 - u_t) \odot u_t$

3. *Cell state* (c_t)

Cell state ditampilkan pada bagian atas diagram sebagai garis horizontal panjang yang bergerak ke kanan dari c_{t-1} ke c_t . *Cell state* akan mengganti informasi pada *memory cell* sebelumnya (c_{t-1}) dan diperbarui dengan informasi pada *memory cell* yang baru (c_t). Proses

ini dilakukan dengan menggabungkan informasi yang terdapat pada *input gate* dan *forget gate*, melalui Persamaan (2.14)

$$c_t = k_t \odot u_t + \phi_t \odot c_{t-1} \quad (2.14)$$

Keterangan:

c_t	= Nilai <i>memory cell</i> baru berbentuk vektor berdimensi $m (\mathbb{R}^m)$
ϕ_t	= Nilai <i>forget gate</i> berbentuk vektor berdimensi $m (\mathbb{R}^m)$
c_{t-1}	= Nilai <i>memory cell</i> sebelumnya berbentuk vektor berdimensi $m (\mathbb{R}^m)$
k_t	= Nilai <i>input gate</i> berbentuk vektor berdimensi $m (\mathbb{R}^m)$
u_t	= Nilai kandidat <i>cell</i> baru vektor berdimensi $m (\mathbb{R}^m)$
d	= Banyaknya fitur <i>input</i>
m	= Banyaknya <i>hidden units</i>
$\odot$	= Perkalian <i>element wise</i>

Informasi baru pada *cell state* diproses melalui fungsi *tanh* untuk membatasi nilainya ke dalam rentang $[-1,1]$ untuk menangkap informasi positif (harga saham naik) dan negatif (harga saham turun) pada *cell state*. Selanjutnya, fungsi *cell state* akan dikalikan dengan *output gate* untuk menghasilkan keluaran informasi yang relevan. Proses ini dapat dirumuskan seperti pada Persamaan (2.15) dan (2.16).

$$h_t = \tanh(c_t) \odot \omega_t \quad (2.15)$$

Dimana fungsi *tanh*:

$$\tanh(x) = 2\sigma(2x) - 1 \quad (2.16)$$

Keterangan:

h_t	= Vektor <i>hidden state</i> berdimensi $m (\mathbb{R}^m)$ dalam <i>timestep t</i>
-------	--

c_t	= Nilai <i>memory cell</i> baru berbentuk vektor berdimensi m ($\mathbb{R}^m$)
ω_t	= <i>Output gate</i> berbentuk vektor berdimensi m ($\mathbb{R}^m$)
m	= Banyaknya <i>hidden units</i>
$\odot$	= Perkalian <i>element wise</i>

4. *Ouput gate* (ω_t)

Ouput gate bagian paling akhir dari diagram yang menentukan *output* pada waktu t yaitu h_{t-1} . *Ouput gate* akan menentukan informasi mana yang akan dikeluarkan dari *memory cell* (Karpathy dkk., 2016). *Output gate* menerapkan fungsi sigmoid untuk menentukan bagian dari *cell state* yang dianggap penting untuk diteruskan sebagai *output* pada *hidden state*. Dengan memanfaatkan fungsi sigmoid, *output gate* dapat memberikan nilai antara 0 dan 1 untuk setiap elemen dalam *cell state*. Nilai 0 berarti informasi sepenuhnya diabaikan dan nilai 1 berarti informasi sepenuhnya diteruskan. Proses ini dapat dirumuskan seperti pada Persamaan (2.17)

$$\omega_t = \sigma (W_{\omega}^T x_t + W_{h\omega}^T h_{t-1} + b_{\omega}) \quad (2.17)$$

Keterangan

ω_t	= <i>Output gate</i> berbentuk vektor berdimensi m ($\mathbb{R}^m$)
h_{t-1}	= Vektor <i>hidden state</i> berdimensi h ($\mathbb{R}^h$) dalam <i>timestep</i> sebelumnya ($t - 1$)
b_{ω}	= Vektor bias berdimensi m ($\mathbb{R}^m$)
W_{ω}	= Matriks bobot untuk <i>input</i> berukuran $\mathbb{R}^{d \times m}$
$W_{h\omega}$	= Matriks bobot untuk <i>hidden state</i> berukuran $\mathbb{R}^{m \times m}$
d	= Banyaknya fitur <i>input</i>
m	= Banyaknya <i>hidden units</i>

Pembaruan nilai *weight* dan bias pada *output gate* terdapat pada Persamaan (2.18), (2.19) dan (2.20)

$$W_{\omega} = W_{\omega} - \eta \cdot \nabla_{W_{\omega}} \quad (2.18)$$

$$W_{h\omega} = W_{h\omega} - \eta \cdot \nabla_{W_{h\omega}} \quad (2.19)$$

$$b_{\omega} = b_{\omega} - \eta \cdot \nabla_{b_{\omega}} \quad (2.20)$$

Keterangan:

$W_{h\omega}$	=	Bobot <i>hidden layer</i>
W_{ω}	=	Bobot <i>input</i>
b_{ω}	=	Bias
η	=	<i>Learning rate</i>
$\nabla_{W_{h\omega}}$	=	$\sum_{t=1}^{\tau} h_{t-1} \cdot (\delta_t^{\omega})^T$
$\nabla_{W_{\omega}}$	=	$\sum_{t=1}^{\tau} x_t \cdot (\delta_t^{\omega})^T$
$\nabla_{b_{\omega}}$	=	$\sum_{t=1}^{\tau} \delta_t^{\omega}$
δ_t^{ω}	=	$\delta_t^h \odot \tanh(c_t) \odot (1 - \omega_t) \odot \omega_t$

Selanjutnya, hasil pada *layer output* (ω_t) akan merepresentasikan hasil prediksi akhir dari LSTM pada waktu (t) berdasarkan Persamaan (2.21)

$$o_t = f^o(W_o^T h_t + b_o) \quad (2.21)$$

Keterangan:

o_t	=	<i>Layer output</i> berbentuk vektor berdimensi m ($\mathbb{R}^m$)
h_t	=	Vektor <i>hidden state</i> berdimensi h ($\mathbb{R}^h$) pada <i>timestep</i> t
b_o	=	Vektor bias berdimensi m ($\mathbb{R}^m$)
W_o	=	Matriks bobot untuk <i>input</i> berukuran $\mathbb{R}^{d \times m}$
d	=	Banyaknya fitur <i>input</i>
m	=	Banyaknya <i>hidden units</i>

Pembaruan nilai *weight* dan bias pada *layer output* ini terdapat pada Persamaan (2.22) dan (2.23)

$$W_o = W_o - \eta \cdot \nabla_{W_o} \quad (2.22)$$

$$b_o = b_o - \eta \cdot \nabla_{b_o} \quad (2.23)$$

Keterangan:

W_o = Bobot *input*

b_o = Bias

η = *Learning rate*

∇_{W_o} = $\sum_{t=1}^{\tau} h_t \cdot (\delta_t^o)^T$

∇_{b_o} = $\sum_{t=1}^{\tau} \delta_t^o$

δ_t^o = $\partial f_t^o \odot \partial \varepsilon_{x_t}$

2.1.3 Fungsi Aktivasi

Fungsi aktivasi dalam algoritma *backpropagation* berperan sebagai sinyal untuk menentukan keluaran ke beberapa neuron lainnya. Fungsi aktivasi memiliki peranan sangat penting dalam algoritma *backpropagation* karena penggunaannya tergantung sesuai kebutuhan dan target yang diinginkan (Cindi, 2024). Fungsi aktivasi digunakan dalam jaringan saraf untuk menentukan apakah neuron akan diaktifkan atau tidak (Rachmadina, 2024). Fungsi ini berperan penting dalam memproses *input* yang diterima oleh neuron dan menentukan *output* dari suatu jaringan saraf tiruan. Fungsi aktivasi dalam jaringan saraf tiruan (*neural network*) bertujuan untuk mengubah *input* yang diterima oleh neuron menjadi *output* yang spesifik. Nilai input yang diproses melalui fungsi aktivasi kemudian akan didistribusikan ke neuron di lapisan berikutnya. Syarat fungsi aktivasi dalam *backpropagation* adalah bersifat kontinu, terdiferensial dengan

mudah dan merupakan fungsi yang tidak turun. Terdapat beberapa fungsi aktivasi yang sering digunakan dalam jaringan syaraf tiruan yaitu:

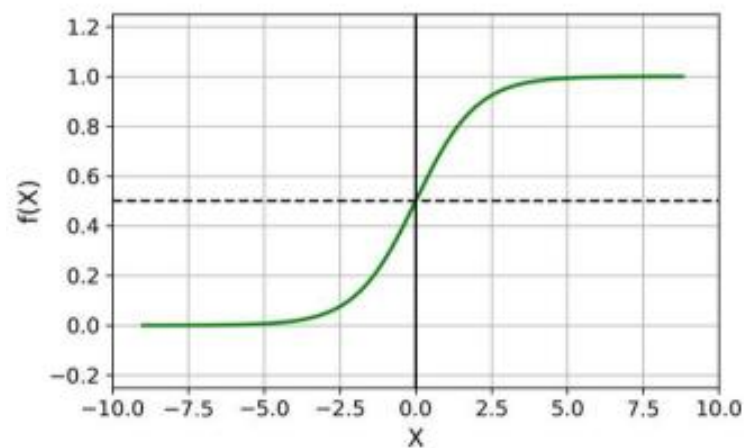
1. Sigmoid Biner

Fungsi aktivasi sigmoid biner merupakan fungsi aktivasi yang paling sering digunakan untuk jaringan syaraf tiruan yang dilatih dengan menggunakan metode *backpropagation* (Ganesha, 2017). Fungsi ini menghasilkan nilai *output* yang terletak pada interval 0 sampai 1. Contoh umum dari fungsi sigmoid ditampilkan pada Gambar 2.3 dan didefinisikan pada Persamaan (2.24)

$$S(x) = \frac{1}{1 + e^{-x}} \quad (2.24)$$

Keterangan:

$S(x)$	=	Nilai <i>output</i> dari fungsi sigmoid untuk <i>input</i> x
x	=	Nilai <i>input</i>
e	=	Bilangan <i>euler</i> dengan nilai sekitar (2,71828)
e^{-x}	=	Eksponensial dari $(-x)$



Gambar 2.3 Grafik Fungsi Aktivasi Sigmoid Biner

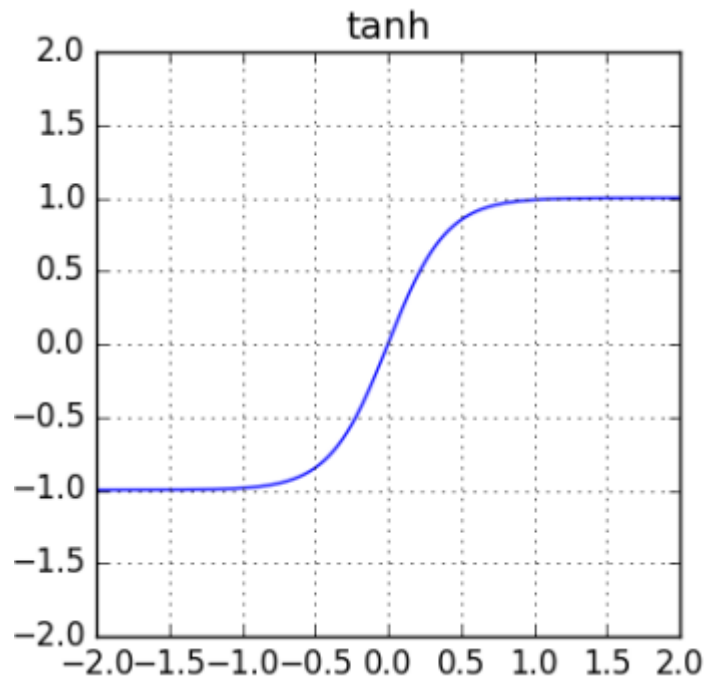
2. *Hyperbolic Tangent Function (Tanh)*

Hyperbolic Tangent Function atau yang biasa disebut fungsi aktivasi *tanh* merupakan salah satu fungsi aktivasi *non-linear* yang umum digunakan dalam jaringan saraf tiruan. Fungsi aktivasi *tanh* memiliki keunggulan dalam hal kecepatan konvergensi selama proses pelatihan model serta menghasilkan akurasi yang lebih tinggi dibandingkan dengan fungsi aktivasi sigmoid biner (Wibawa, 2016). Performa yang ditawarkan oleh fungsi aktivasi *tanh* hampir sama dengan performa klasifikasi yang dihasilkan oleh fungsi aktivasi *Relu* (Wibawa, 2016). Fungsi aktivasi *tanh* menghasilkan *output* dalam rentang nilai lebih luas dari fungsi aktivasi sigmoid biner yaitu diantara -1 hingga 1 (Rachmadina, 2024) Fungsi aktivasi *tanh* ditunjukkan pada Gambar 2.4 dan didefinisikan pada Persamaan (2.25).

$$\tanh t = \frac{e^t - e^{-t}}{e^t + e^{-t}} \quad (2.25)$$

Keterangan

$\tanh t$	=	Nilai <i>output</i> dari fungsi aktivasi <i>tanh</i> untuk <i>input</i> t
t	=	Nilai <i>input</i>
e	=	Bilangan <i>euler</i> dengan nilai sekitar (2,71828)
e^t, e^{-t}	=	Eksponensial dari $(t, -t)$



Gambar 2.4 Grafik Fungsi Aktivasi *Tanh*

2.1.4 *Hidden Layer* dan *Neuron Hidden*

Hidden layer dalam jaringan saraf tiruan merupakan lapisan yang terletak di antara *input layer* dan *output layer*. Lapisan ini tidak berinteraksi langsung dengan data *input* atau *output*, tetapi berfungsi sebagai pemroses informasi yang diterima dari lapisan *input* sebelum diteruskan ke lapisan *output*. *Hidden layer* terdiri dari neuron-neuron tersembunyi (*neuron hidden*) yang bertindak sebagai unit pemrosesan utama dalam jaringan. Setiap *neuron hidden* menerima sekumpulan *input* dari neuron di lapisan sebelumnya, melakukan transformasi melalui fungsi aktivasi, dan meneruskan hasilnya ke neuron di lapisan berikutnya. Proses ini memungkinkan jaringan saraf untuk memodelkan hubungan *non-linear* yang kompleks dalam data (Khumaidi dkk., 2022).

Hidden layer dapat terdiri dari beberapa lapisan untuk menghasilkan komposisi algoritma yang optimal dalam meminimalkan *error* pada *output*.

Penentuan jumlah *hidden layer* dan jumlah neuron dalam setiap lapisan merupakan aspek penting dalam perancangan arsitektur jaringan saraf, karena dapat mempengaruhi kinerja dan kemampuan model. Penelitian ini akan melakukan percobaan dengan beberapa nilai *hidden layer* maupun jumlah neuron untuk menentukan konfigurasi yang menghasilkan kinerja paling optimal dalam pelatihan jaringan saraf.

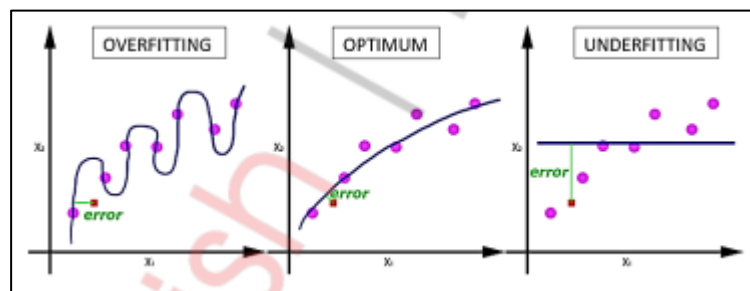
2.1.5 *Batch Size*

Batch size merupakan jumlah sampel data yang diproses oleh jaringan saraf tiruan dalam satu iterasi pelatihan. Penggunaan *batch size* bertujuan untuk mengatasi keterbatasan sumber daya komputasi, karena tidak memungkinkan untuk memproses seluruh dataset sekaligus. Oleh karena itu, dataset dibagi menjadi beberapa bagian atau kelompok agar proses pelatihan lebih efisien dan stabil (Khumaidi dkk., 2022). Selama proses pelatihan model, dataset dibagi menjadi beberapa *batch* seperti misalnya jika terdapat 100 sampel data dan *batch size* yang digunakan adalah 5, maka algoritma akan memproses 5 sampel pertama (data ke-1 hingga ke-5) dalam satu *batch*, kemudian melanjutkan ke *batch* berikutnya dengan 5 sampel berikutnya (data ke-6 hingga ke-10), dan seterusnya hingga seluruh dataset terbagi menjadi 20 *batch*. Penelitian ini akan melakukan percobaan dengan menggunakan beberapa nilai *batch size* untuk mengetahui ukuran *batch size* yang menghasilkan kinerja paling optimal dalam pelatihan jaringan saraf.

2.1.6 Epoch

Epoch merupakan satu siklus penuh dalam proses pelatihan jaringan saraf tiruan (*Neural Network*) di mana seluruh dataset digunakan untuk memperbarui bobot model dan kemudian diproses ulang dari awal. Ukuran dataset yang besar nantinya akan dibagi ke dalam beberapa *batch* karena tidak memungkinkan untuk diproses sekaligus agar pelatihan lebih efisien.

Penggunaan lebih dari satu *epoch* diperlukan karena satu kali proses pelatihan terhadap seluruh dataset belum cukup untuk menghasilkan bobot optimal. Jika jumlah *epoch* terlalu sedikit, model berisiko mengalami *underfitting*, yaitu kondisi di mana model belum mampu menangkap pola yang cukup dari data. Sebaliknya, jika jumlah *epoch* terlalu banyak, model berisiko mengalami *overfitting* yaitu kondisi di mana model terlalu menyesuaikan diri dengan data pelatihan dan kehilangan kemampuan generalisasi terhadap data baru seperti pada Gambar 2.5 (Khumaidi dkk., 2022).



Gambar 2.5 Grafik *Overfitting*, *Optimum*, *Underfitting* dari Jumlah *Epoch*

Setiap *epoch* terdiri dari beberapa iterasi dimana jumlahnya bergantung pada ukuran *batch* yang digunakan. Selama proses pelatihan, bobot model diperbarui pada setiap iterasi berdasarkan perhitungan gradien, sehingga semakin banyak *epoch* yang digunakan semakin banyak pula pembaruan yang terjadi. Oleh karena itu, pemilihan jumlah *epoch* yang optimal sangat penting dan harus

disesuaikan dengan kompleksitas model serta karakteristik data (Khumaidi dkk., 2022).

2.1.7 *Optimizer*

Optimizer merupakan algoritma yang digunakan dalam pembelajaran mesin untuk menyesuaikan bobot dan bias suatu model dengan tujuan mengurangi kesalahan atau *loss* pada data. Peran utama *optimizer* adalah memperbarui bobot dalam jaringan saraf tiruan agar model dapat menghasilkan prediksi yang lebih akurat (Suryalim dkk., 2022). Terdapat beberapa jenis *optimizer* yang umum digunakan dalam pelatihan model pembelajaran mesin diantaranya *Adaptive Moment Estimation* (Adam), *Stochastic Gradient Descent* (SGD), Adagrad, dan lainnya. Pemilihan *optimizer* yang tepat dapat mempengaruhi proses pelatihan menjadi lebih efisien dan model dapat mencapai hasil yang optimal. Penelitian ini akan menggunakan *optimizer* Adam karena merupakan salah satu metode optimasi yang sudah banyak digunakan terutama untuk melatih *deep neural network* berbagai aplikasi *deep learning* (Mahajaya dkk., 2024).

Optimizer adaptive moment estimation (Adam) mengombinasikan keunggulan dari RMSProp yaitu *learning rate decay* dan metode momentum yaitu *smoothing* untuk menyesuaikan laju pembelajaran secara adaptif pada setiap parameter model (Suryalim dkk., 2022). *Optimizer* Adam memadukan kedua persamaan dari RMSProp dan metode momentum untuk memperbarui nilai bobot dan bias pada model. *Optimizer* Adam menghitung estimasi pertama (*mean*) dan kedua (*uncentered variance*) dari gradien, kemudian menggunakan kedua estimasi tersebut untuk mengatur *learning rate* untuk setiap parameter. Persamaan

optimizer Adam ditampilkan pada Persamaan (2.26) sampai dengan Persamaan (2.30).

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad (2.26)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_{2t} \quad (2.27)$$

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \quad (2.28)$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \quad (2.29)$$

$$\theta_{t+1} = \theta_t - \frac{\alpha \hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} \quad (2.30)$$

Keterangan

m_t	=	Estimasi momen pertama (<i>mean</i>) dari gradien
v_t	=	Estimasi momen kedua (variansi tak terpusat) dari gradien
g_t	=	Gradien pada waktu- t
β_1, β_2	=	Koefisien peluruhan eksponensial untuk estimasi momen
θ	=	Parameter yang akan dioptimalkan
α	=	<i>Learning rate</i>
ϵ	=	Konstanta

2.1.8 Analisis Teknikal

Terdapat dua jenis analisis yang dapat digunakan dalam pengambilan keputusan investasi yaitu analisis fundamental dan analisis teknikal. Analisis fundamental merupakan metode yang digunakan untuk menilai suatu saham dengan mempertimbangkan berbagai aspek yang meliputi, analisis perusahaan, perekonomian internasional, dan perekonomian nasional (Syamsir, 2004). Sedangkan analisis teknikal adalah metode yang digunakan untuk menganalisis perilaku pasar dengan memanfaatkan grafik dan data historis untuk memprediksi

pergerakan di masa depan. Pendekatan ini berfokus pada pola dan tren yang terbentuk dalam harga saham serta volume perdagangan untuk mengidentifikasi peluang investasi. Jadi dapat disimpulkan bahwa analisis teknikal adalah metode yang digunakan untuk mempelajari pergerakan harga saham berdasarkan data historis dengan tujuan memprediksi tren harga saham di masa depan. Berbeda dengan analisis fundamental, yang menganalisis faktor keuangan, ekonomi, atau berita perusahaan, teknikal analisis berfokus pada pola harga dan perilaku pasar.

Analisis teknikal tidak hanya bergantung pada pola grafik harga, tetapi juga melibatkan perhitungan indikator saham menggunakan Persamaan matematik. Setiap indikator yang terdapat pada analisis teknikal dihitung berdasarkan data historis yang memberi wawasan lebih mengenai pergerakan harga dan potensi tren saham di masa depan. Dengan demikian, analisis teknikal menjadi lebih menyeluruh dengan menggabungkan pembacaan grafik dan perhitungan indikator untuk membuat keputusan investasi yang lebih tepat. Terdapat beberapa indikator pada analisis teknikal yang digunakan untuk menentukan tren pasar diantaranya *Moving Average Convergence Divergence* (MACD), *Williams %Range*, *Bollinger bands*, *Simple Moving Average* (SMA), *Relative Strength Index* (RSI), dan *fibonacci retracement*.

Relative Strength Index (RSI) adalah indikator analisis yang digunakan untuk mengukur kecepatan perubahan dalam kenaikan dan penurunan harga saham. RSI pertama kali dikembangkan oleh J. Welles Wilder pada tahun 1978. Indikator ini dapat memberikan gambaran apakah harga pasar telah mencapai kondisi *overbought* atau *oversold*. Nilai RSI berkisar antara 1 sampai 100. Jika nilai yang dihasilkan berada di atas 70 maka saham dianggap *overbought*,

sedangkan jika nilai berada di bawah 30 maka saham dianggap *oversold*. Nilai RSI dihitung seperti yang dijelaskan pada Persamaan (2.31) (Sezer dkk., 2017).

$$RSI = 100 - \left(\frac{100}{1 + RS} \right) \quad (2.31)$$

$$RS = \frac{\text{Rata - rata keuntungan}}{\text{rata - rata kerugian}} \quad (2.32)$$

2.1.9 Normalisasi

Normalisasi data adalah proses penskalaan nilai atribut sebuah dataset ke dalam rentang yang seragam tanpa mengubah distribusi relatif antar nilai atau menghilangkan informasi penting (Chandra dkk., 2022). Tujuan dari normalisasi adalah memastikan bahwa setiap fitur pada dataset dalam kondisi setara sehingga memiliki kontribusi yang seimbang dalam proses pembelajaran mesin, terutama pada algoritma yang sensitif terhadap perbedaan skala antar fitur. Normalisasi data perlu dilakukan pada tahap *preprocessing* karena algoritma seperti regresi linier dan jaringan saraf tiruan dapat dipengaruhi oleh perbedaan skala antar fitur, yang berpotensi menyebabkan model memberikan bobot lebih besar pada fitur dengan skala yang lebih tinggi (Muflih, 2021). Dengan menerapkan normalisasi, seluruh fitur disesuaikan dalam skala yang seragam, sehingga model dapat belajar lebih optimal dan menghasilkan prediksi yang lebih akurat.

Penelitian ini akan menggunakan metode *Min-Max Normalization* yaitu jenis normalisasi yang mentransformasikan linier pada data aktual (Chandra dkk., 2022). Metode ini mengubah data ke dalam skala baru pada rentang 0 hingga 1. Rumus normalisasi ditunjukkan pada Persamaan (2.33)

$$X_{norm} = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (2.33)$$

Keterangan

x	=	Data yang dinormalisasi
$\min(x)$	=	Data terkecil suatu fitur
$\max(x)$	=	Data terbesar suatu fitur

2.1.10 *K-fold Cross Validation*

Cross validation atau validasi silang adalah metode untuk mengevaluasi kinerja suatu model dengan membagi data menjadi beberapa subset, melatih model pada beberapa subset dan mengujinya pada yang lain (Bintoro dkk., 2023). *Cross validation* dikenal sebagai penaksir rotasi, dimana data di dalamnya dibagi menjadi k himpunan bagian yang seimbang. Model klasifikasi dilatih dan diuji sebanyak k kali. Pada setiap iterasi, satu subset digunakan sebagai data uji, sedangkan k subset lainnya berfungsi sebagai data latih (Nurhayati dkk., 2014).

K-fold cross validation adalah salah satu metode yang digunakan untuk membagi data menjadi data pelatihan dan data validasi. Teknik ini banyak digunakan pada penelitian karena dapat mengurangi bias dalam proses pengambilan sampel (Ridwansyah, 2022). *K-fold cross validation* bekerja dengan membagi keseluruhan data menjadi k subset data dengan ukuran yang sama besar, kemudian proses pelatihan dan pengujian dilakukan sebanyak k kali di mana setiap subset secara bergiliran digunakan sebagai data validasi sementara subset lainnya akan digunakan sebagai data latih. Dengan demikian, setiap subset data akan memperoleh kesempatan yang sama untuk berperan sebagai data latih atau data validasi seperti pada Gambar 2.6. Penggunaan teknik ini diharapkan dapat

membantu dalam memperoleh estimasi performa model yang lebih stabil dan tidak bergantung pada pembagian data tertentu.

<i>Fold 1</i>	<i>Fold 2</i>	<i>Fold 3</i>	...	<i>Fold K</i>
Test	Train	Train	...	Train
Train	Test	Train	...	Train
Train	Train	Test	...	Train
...	...	...	...	...
Train	Train	Train	...	Test

Gambar 2.6 *K-fold Cross Validation*

2.1.11 Evaluasi Model

Pengukuran tingkat kesalahan atau deviasi antara nilai yang diprediksi oleh model dengan nilai aktual merupakan langkah penting untuk mengevaluasi kinerja model dalam proses analisis data dan pemodelan. Evaluasi ini bertujuan untuk menentukan sejauh mana model mampu menghasilkan prediksi yang akurat serta memastikan bahwa model dapat digunakan secara efektif dalam pengambilan keputusan. Berbagai metode telah dikembangkan untuk mengukur tingkat kesalahan dalam prediksi, khususnya dalam bidang keuangan dan pasar saham. Beberapa metrik evaluasi yang umum digunakan antara lain *Mean Absolute Error* (MAE), *Mean Square Error* (MSE), *Root Mean Square Error* (RMSE), dan *Mean Absolute Percentage Error* (MAPE). Masing-masing metrik memiliki keunggulan dan karakteristik tersendiri dalam menilai kinerja model prediksi. Pada penelitian ini evaluasi pemodelan yang akan digunakan yaitu *Root Mean Square Error* (RMSE) dan *Mean Absolute Percentage Error* (MAPE). RMSE memberikan indikasi mengenai besarnya kesalahan absolut dalam satuan harga saham, sementara MAPE menunjukkan tingkat kesalahan dalam skala persentase yang memungkinkan perbandingan antar saham atau aset dengan nilai yang berbeda-beda. Oleh karena itu, penggunaan kombinasi metrik RMSE dan

MAPE diharapkan dapat memberikan gambaran yang lebih komprehensif mengenai kinerja model prediksi yang dikembangkan.

1. **Root Mean Square Error (RMSE)**

Root Mean Square Error (RMSE) merupakan salah satu metrik evaluasi yang digunakan untuk mengukur selisih antara nilai hasil prediksi dengan nilai aktual. Semakin besar nilai RMSE yang dihasilkan maka semakin tinggi tingkat kesalahan prediksi yang dilakukan oleh model, sehingga menunjukkan bahwa tingkat keakuratannya semakin rendah. Sebaliknya, semakin kecil nilai RMSE maka hasil prediksi yang diperoleh semakin mendekati nilai aktual, sehingga menunjukkan bahwa model memiliki tingkat akurasi yang lebih baik (Sautomo dkk., 2021). RMSE dihitung dengan mengambil akar kuadrat dari rata-rata selisih kuadrat antara nilai prediksi dan nilai aktual. Rumus perhitungan nilai *error* menggunakan RMSE diberikan pada Persamaan (2.34) (Hyndman, 2014).

$$RMSE = \sqrt{T^{-1} \sum_{t=1}^T (y_t - \hat{y}_{t|t-1})^2} \quad (2.34)$$

Keterangan:

T	=	Jumlah total data selama periode pengamatan
y_t	=	Nilai aktual pada waktu ke- t
$\hat{y}_{t t-1}$	=	Nilai prediksi pada waktu ke- t , berdasarkan data hingga waktu ke- $t - 1$.
$(y_t - \hat{y}_{t t-1})^2$	=	Kuadrat dari selisih antara nilai aktual dan nilai prediksi pada waktu ke- t
$\sum_{t=1}^T$	=	Penjumlahan seluruh <i>error</i> kuadrat dari waktu ke-1 hingga ke- T

Tidak ada standar baku yang menetapkan kriteria nilai RMSE dapat dianggap baik atau buruk secara umum. Kriteria nilai RMSE sangat bergantung pada konteks dan skala data yang digunakan (Amazon Web Service, 2025).

2. *Mean Absolute Percentage Error (MAPE)*

Mean Absolute Percentage Error (MAPE) merupakan metrik yang mengukur rata-rata kesalahan absolut dalam bentuk persentase terhadap nilai aktual. MAPE dihitung dengan mengambil rata-rata selisih absolut antara nilai aktual dan nilai prediksi kemudian dinormalisasi terhadap nilai aktualnya. Semakin rendah nilai persentase kesalahan yang dihasilkan maka tingkat akurasi hasil prediksi yang dilakukan oleh model semakin tinggi. MAPE dapat dihitung menggunakan Persamaan (2.35) (Hyndman, 2014).

$$MAPE = 100T^{-1} \sum_{t=1}^T \frac{|y_t - \hat{y}_{t|t-1}|}{|y_t|} \quad (2.35)$$

Keterangan:

T	=	Jumlah total data selama periode pengamatan
y_t	=	Nilai aktual pada waktu ke- t
$\hat{y}_{t t-1}$	=	Nilai prediksi pada waktu ke- t , berdasarkan data hingga waktu ke- $t - 1$.
$ y_t - \hat{y}_{t t-1} $	=	Nilai absolut dari nilai aktual pada waktu ke- t
$\sum_{t=1}^T$	=	Penjumlahan seluruh <i>error</i> kuadrat dari waktu ke-1 hingga ke- T

Tabel 2.1 menampilkan kriteria akurasi model berdasarkan nilai MAPE yang dihasilkan menurut (Aditya dkk., 2019).

Tabel 2.1 Kriteria Akurasi MAPE

Nilai MAPE	Akurasi Prediksi
$< 10\%$	Sangat Baik
$10\% < MAPE \leq 20\%$	Baik
$20\% < MAPE \leq 50\%$	Cukup
$> 50\%$	Buruk

2.1.12 Peramalan

Peramalan memiliki beberapa definisi berdasarkan pendapat para ahli. Menurut Stevenson dkk., (2014) menjelaskan bahwa peramalan atau *forecasting* adalah pernyataan mengenai nilai yang akan datang dari suatu variabel. Dengan memanfaatkan berbagai informasi, prediksi yang lebih akurat dapat membantu dalam pengambilan keputusan yang lebih baik. Sedangkan menurut Heizer dkk., (2015), peramalan (*forecasting*) merupakan kombinasi antara seni dan ilmu pengetahuan dalam memprediksi kejadian yang akan terjadi di masa depan. Jadi dapat disimpulkan bahwa peramalan adalah seni dan ilmu untuk membuat prediksi menggunakan data historis untuk memproyeksikan apa yang akan terjadi dimasa depan. Peramalan adalah suatu proses yang berkelanjutan, karena data baru akan terus muncul, sehingga model perlu diperbarui dan dievaluasi secara berkala. Peramalan akan melibatkan data historis dan memproyeksikannya ke masa yang akan datang dengan bantuan model matematika. Peramalan bertujuan untuk mengurangi ketidak pastian terhadap sesuatu yang akan terjadi dimasa yang akan datang. Peramalan atau *forecasting* dalam penelitian ini berarti proses untuk memprediksi bagaimana grafik harga saham PT. Indofood Sukses Makmur Tbk di masa yang akan datang.

2.1.13 Saham

Saham dapat diartikan sebagai bukti kepemilikan atau partisipasi dalam sebuah perusahaan. Ketika seseorang membeli saham perusahaan, mereka secara esensial berinvestasi dalam perusahaan tersebut. Nilai saham mencerminkan nilai kekayaan suatu perusahaan yang telah menerbitkannya dan fluktuasinya sangat dipengaruhi oleh keseimbangan antara penawaran dan permintaan di pasar saham. Dana yang diinvestasikan dari pembelian saham akan digunakan oleh manajemen perusahaan untuk mendukung operasional dan pengembangan saham. Sebagai imbalannya, investor berpotensi menerima *return* dalam bentuk *dividen*. *Dividen* merupakan laba atau pendapatan suatu perusahaan dan akan menjadi salah satu keuntungan yang diperoleh investor. Tujuan utama seorang investor dalam berinvestasi adalah untuk mendapatkan *return* yang optimal, sambil tetap mempertimbangkan risiko investasi yang harus dihadapinya.

Sebelum melakukan investasi, investor perlu memahami perubahan harga saham agar dapat membuat keputusan investasi yang bijaksana. *Return on Assets* (ROA) adalah rasio yang digunakan untuk menilai seberapa efektif perusahaan dalam menghasilkan laba dari aktivitas investasi yang dilakukan. Rasio ini memberikan gambaran tentang kemampuan perusahaan dalam memanfaatkan total asetnya untuk memperoleh keuntungan. Dengan demikian, ROA menjadi alat penting untuk mengevaluasi kinerja keuangan perusahaan dalam menghasilkan laba dari semua sumber daya yang dimiliki. Semakin tinggi nilainya maka semakin baik kinerjanya karena menandakan kepemilikan perusahaan semakin kokoh. Perhitungan ROA dapat menggunakan Persamaan (2.36)

$$\text{Return On Assets} = \frac{\text{Laba Bersih}}{\text{Total Assets}} \times 100 \quad (2.36)$$

Return on Equity (ROE) adalah parameter yang digunakan untuk mengevaluasi kemampuan perusahaan dalam menghasilkan laba bagi pemegang saham. Menurut Rinati (2016) menjelaskan bahwa ROE merupakan ukuran yang menggambarkan tingkat keuntungan perusahaan dengan cara membandingkan laba bersih yang dihasilkan dengan ekuitas rata-rata yang dimiliki oleh pemegang saham. ROE memberikan indikasi seberapa efektif perusahaan dalam menggunakan modal yang dimiliki untuk menghasilkan keuntungan. Perhitungan ROE dapat menggunakan Persamaan (2.37)

$$\text{Return On Equity} = \frac{\text{Laba Setelah Pajak}}{\text{Total Equity}} \times 100 \quad (2.37)$$

2.2 Integrasi Nilai Keislaman dalam Perencanaan Masa Depan Melalui

Prediksi

Prediksi merupakan suatu proses yang sistematis untuk memperkirakan kemungkinan kejadian di masa depan dengan memanfaatkan informasi yang diperoleh dari data masa lalu dan saat ini. Tujuan dari prediksi ini adalah untuk mengurangi kesalahan, yaitu selisih antara hasil yang sebenarnya terjadi dan hasil yang diperkirakan. Setiap makhluk di muka bumi ini tidak ada yang mengetahui apa yang akan terjadi di masa depan, namun sangat penting bagi setiap makhluk untuk mempersiapkan diri dalam menghadapi masa depannya. Allah SWT berfirman dalam QS. Al-Hasyr ayat 18 yang artinya: (Departemen Agama, 2015)

“Wahai orang-orang yang beriman, bertakwalah kepada Allah dan hendaklah setiap orang memperhatikan apa yang telah diperbuatnya untuk hari esok (akhirat). Bertakwalah kepada Allah. Sesungguhnya Allah Maha Teliti terhadap apa yang kamu kerjakan”

Disebutkan didalam Tafsir Al-Munir bahwa, wahai orang-orang yang beriman, yang meyakini dan mempercayai Allah SWT dan Rasul-Nya, laksanakanlah segala perintah Allah SWT dan jauhilah larangan-Nya. Takutlah akan hukuman-Nya dan hendaklah setiap individu memperhatikan amal yang telah disiapkan untuk Hari Kiamat. Lakukanlah muhasabah dan koreksilah diri sebelumnya nantinya dikoreksi dan dihisab. Bertakwalah kepada Allah SWT, karena tidak ada satu pun amal perbuatan yang tersembunyi dari-Nya. Allah SWT akan membalas segala amal-amal manusia baik yang kecil maupun besar, sedikit maupun banyak (Az-Zuhaili, 2013).

Ayat ini menekankan pentingnya mempersiapkan diri untuk masa depan dalam setiap aspek kehidupan dengan perbuatan baik dan perencanaan yang matang. Allah meminta manusia untuk memperhatikan setiap tindakan dan keputusan yang diambil agar tidak hanya bermanfaat di masa kini, tetapi juga memberikan dampak positif di masa mendatang baik di kehidupan dunia maupun di kehidupan akhirat. Dalam ayat ini, perintah untuk bertakwa diulang dua kali sebagai penegasan, penguatan, dan sekaligus dorongan agar manusia mengerjakan hal-hal yang bermanfaat baik di dunia maupun di akhirat.

Selain itu prinsip tawakal atau berserah diri kepada Allah setelah melakukan usaha dan perencanaan juga diajarkan dalam QS. Ali 'Imran ayat 159 yang artinya: (Departemen Agama, 2015)

"Maka, berkat rahmat Allah engkau (Nabi Muhammad) berlaku lemah lembut terhadap mereka. Seandainya engkau bersikap keras dan berhati kasar, tentulah mereka akan menjauh dari sekitarmu. Oleh karena itu, maafkanlah mereka, mohonkanlah ampunan untuk mereka, dan bermusyawarahlah dengan mereka dalam segala urusan (penting). Kemudian apabila engkau telah membulatkan tekad, maka bertawakallah kepada Allah. Sesungguhnya Allah mencintai orang-orang yang bertawakal"

Dari ayat-ayat tersebut, dapat dipahami bahwa meskipun manusia tidak dapat memprediksi masa depan dengan pasti, manusia didorong untuk membuat perencanaan yang baik, bertindak berdasarkan pengetahuan dan prinsip-prinsip Islam, dan kemudian menyerahkan hasil akhirnya kepada Allah. Prinsip-prinsip ini dapat digunakan sebagai dasar dalam membuat prediksi yang lebih bijak dan beretika.

2.3 Kajian Topik dengan Teori Pendukung

Peramalan (*forecasting*) adalah seni dan ilmu untuk membuat prediksi menggunakan data historis untuk memproyeksikan apa yang akan terjadi di masa depan. Peramalan melibatkan penggunaan data historis kemudian memproyeksikannya ke masa yang akan datang dengan memanfaatkan model matematika. Peramalan atau prediksi bertujuan untuk mengurangi ketidakpastian terhadap sesuatu yang akan terjadi di masa yang akan datang. Sebagai pelaku pasar keuangan, seorang investor perlu memahami cara memprediksi fluktuasi harga saham agar dapat membuat keputusan investasi yang tepat. Fitur-fitur yang digunakan untuk memprediksi data saham diantaranya:

1. *Date*: Data tanggal, bulan, dan tahun transaksi saham terjadi untuk melacak pergerakan harga dari waktu ke waktu
2. *Open*: Harga saham pada saat pembukaan perdagangan di hari tertentu untuk menganalisis pergerakan harga selanjutnya
3. *Close*: Harga saham pada saat penutupan perdagangan di bursa saham pada hari tertentu digunakan untuk perhitungan pada analisis teknikal

4. *High*: Harga tertinggi yang dicapai pada hari tertentu untuk mengetahui kekuatan pasar saham tersebut
5. *Low*: Harga terendah yang dicapai pada hari tertentu untuk memberikan wawasan mengenai batasan pergerakan harga
6. *Volume*: Jumlah total lembar saham yang diperdagangkan selama periode tertentu.

Long Short Term Memory (LSTM) adalah salah satu algoritma dalam *machine learning* yang dikenal memiliki tingkat akurasi tinggi dalam memprediksi data berbentuk deret waktu (*time series*). *Memory cell* pada LSTM dapat menyimpan informasi dari langkah-langkah komputasi sebelumnya untuk jangka waktu yang lama khususnya data yang sangat besar dan menerapkannya pada elemen-elemen berikutnya dalam urutan *input*, sehingga memungkinkan pengenalan pola data secara efektif dan menghasilkan prediksi yang akurat. Algoritma LSTM sering digunakan untuk memprediksi data harga saham dimana urutan dari setiap datanya sangat penting. LSTM secara khusus mampu menangkap hubungan antara harga saham pada hari-hari sebelumnya dengan pergerakan harga di hari-hari berikutnya. Yan dalam penelitiannya tahun 2016 menjelaskan bahwa fitur gerbang sel (*cell gate*) dalam LSTM adalah kunci utama dari arsitekturnya yang berfungsi untuk mengatur aliran informasi dalam jaringan. Dalam model LSTM akan dilakukan proses pada fitur *cell gate* berupa:

1. Menghitung nilai *forget gate*.
2. Menghitung nilai *input gate*.
3. Memperbarui memori yang berada pada *cell*.
4. Menghitung *output gate* dan nilai *output* akhir.

Dalam praktiknya, prediksi harga saham sering kali menjadi tantangan karena banyaknya variabel yang mempengaruhi. Dengan penerapan LSTM, prediksi harga saham dapat lebih akurat karena algoritma ini mampu menangani data yang kompleks dan memperhitungkan pola jangka panjang.

BAB III

METODE PENELITIAN

3.1 Jenis Penelitian

Penelitian ini merupakan penelitian kuantitatif karena penelitian ini berfokus pada pengolahan dan analisis data saham yang bersifat numerik. Selain itu penelitian ini juga merupakan penelitian terapan karena bertujuan untuk memberikan solusi untuk suatu permasalahan yaitu memprediksi harga saham.

3.2 Data dan Sumber Data

Data yang digunakan pada penelitian ini merupakan data sekunder berupa harga saham perusahaan PT. Indofood Sukses Makmur Tbk (INDF.JK) yang diperoleh dari situs *website Yahoo Finance*. PT. Indofood Sukses Makmur Tbk adalah perusahaan yang bergerak di bidang pengolahan pangan menjadi makanan dan minuman. Penelitian ini berfokus pada penggunaan data harga penutupan saham (*close*) pada periode 02 Juni 2021 hingga 31 Januari 2025 sebanyak 889 data. Harga penutupan saham (*close*) merupakan harga terakhir yang tercatat pada akhir sesi perdagangan dalam satu hari dan sering digunakan dalam analisis serta pelaporan karena mencerminkan kinerja saham pada hari tersebut. Data harga saham yang digunakan pada penelitian ini dapat dilihat pada Tabel 3.1. dan dapat diakses pada Gambar 3.1



Gambar 3.1 QR Code Data Saham

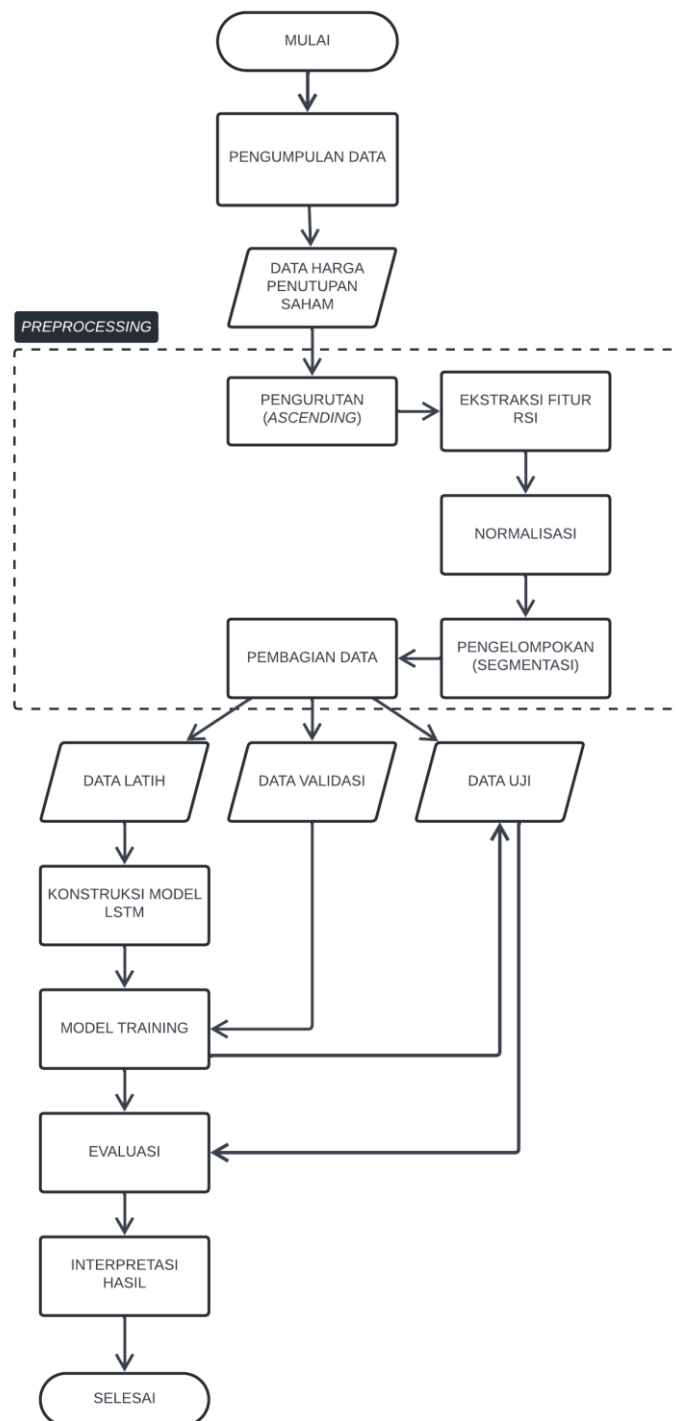
Tabel 3.1 Data Saham PT. Indofood Sukses Makmur Tbk

<i>Date</i>	<i>Close</i>
02/06/2021	6450
03/06/2021	6675
04/06/2021	6575
07/06/2021	6575
08/06/2021	6375
....	
22/01/2025	7625
23/01/2025	7500
24/01/2025	7550
30/01/2025	7625
31/01/2025	7850

Selain itu, penelitian ini juga akan menggunakan nilai *Relative Strength Index* (RSI) menjadi salah satu fitur yang juga akan diprediksi. RSI merupakan salah satu indikator analisis teknikal yang digunakan untuk mengukur tingkat kejenuhan suatu saham apakah berada dalam kondisi *overbought* atau *oversold*. Perhitungan nilai RSI terdapat pada Persamaan (2.31) dengan menggunakan data historis harga penutupan saham selama periode 14 hari sebelumnya. Jadi, fitur-fitur yang digunakan pada penelitian ini terdiri dari harga penutupan saham (*close*) dan nilai RSI.

3.3 Teknik Analisis Data

Penelitian ini akan membahas bagaimana implementasi algoritma *Long Short Term Memory* dalam memprediksi harga saham PT. Indofood Sukses Makmur Tbk. Alur penelitian yang akan dilakukan terdapat pada Gambar 3.2.



Gambar 3.2 Alur Penelitian Prediksi Harga Saham dengan Metode LSTM

1. Pengumpulan *dataset*

Pada tahapan pengumpulan data, data yang digunakan yaitu berupa data harga penutupan saham PT. Indofood Sukses Makmur Tbk yang diperoleh dari *website yahoo finance* periode waktu 02 Juni 2021 hingga 31 Januari 2025 sebanyak 889 baris data seperti pada Tabel 3.1.

2. *Preprocessing*

Langkah-langkah yang akan dilakukan pada *preprocessing* diantaranya yaitu pengurutan (*ascending*), ekstraksi fitur, normalisasi, segmentasi, pembagian data (Suyudi dkk, 2019). Proses ini bertujuan untuk mengubah data yang sebelumnya tidak terstruktur menjadi format yang sesuai dengan kebutuhan analisis dan pemrosesan lebih lanjut.

- a. Pengurutan (*Ascending*)

Mengurutkan data harga saham dimulai dari yang terdahulu atau yang paling lama hingga data terbaru.

- b. Ekstraksi fitur

Menambah fitur nilai RSI yang dapat dihitung menggunakan Persamaan (2.31).

- c. Pembagian data

Data dibagi menjadi data latih mencakup periode 02 Juni 2021 hingga 30 Oktober 2024 dan data uji mencakup periode 01 November 2024 hingga 31 Januari 2025. Setelah proses segmentasi data latih kemudian dibagi menjadi lima subset menggunakan *k-fold cross validation*. Pada setiap iterasi, satu subset digunakan sebagai data validasi sedangkan empat subset lainnya digunakan sebagai data latih. Data latih digunakan

untuk membangun dan melatih model, data validasi berfungsi untuk mengevaluasi kinerja model yang telah dikembangkan, sedangkan data uji digunakan untuk menguji model yang telah dilatih dan divalidasi dengan tujuan menilai performa akhir model serta menampilkan hasil prediksi.

d. Normalisasi

Pada tahap normalisasi data akan dilakukan penskalaan pada rentang 0 hingga 1 dengan tujuan untuk menyamaratakan data. Nilai normalisasi didapatkan dari Persamaan (2.33).

e. Pengelompokan (Segmentasi)

Data mentah dipisahkan dan dikelompokkan menjadi data yang sesuai dengan kebutuhan sistem. Pada penelitian ini akan membentuk *window* data dengan panjang rangkaian yaitu 30 hari yang akan digunakan untuk memprediksi nilai di hari ke 31.

3. Membuat konstruksi model LSTM menggunakan data latih

Membangun model LSTM dengan menggunakan serangkaian percobaan parameter. Penentuan parameter yang digunakan merujuk pada penelitian sebelumnya yaitu (Puteri, 2023).

a. *Hidden layer*

Lapisan dalam jaringan saraf yang terletak diantara lapisan *input* dan lapisan *output*. Lapisan *hidden layer* yang akan diterapkan pada penelitian ini adalah 4.

b. Neuron *hidden*

Neuron *hidden* adalah unit dasar dalam jaringan saraf yang berfungsi untuk memproses informasi. Jumlah neuron *hidden* yang akan diterapkan pada penelitian ini adalah 50.

c. Nilai *batch size*

Batch size menentukan berapa banyak contoh pelatihan yang akan digunakan untuk menghitung *gradient* dan memperbarui bobot model sebelum melakukan langkah pembaruan. Selama proses penelitian akan dilakukan beberapa percobaan nilai-nilai *batch size* untuk mengetahui nilai *batch size* mana yang akan memberikan hasil prediksi terbaik. Nilai-nilai *batch size* yang akan digunakan pada penelitian ini yaitu 4, 16, 32 dan 64.

d. *Epoch* maksimum

Epoch adalah jumlah maksimum siklus pelatihan yang diizinkan untuk model jaringan saraf dalam proses pelatihan. *Epoch* maksimum yang akan dilakukan percobaan pada penelitian ini yaitu 50 dan 100.

e. *Optimizer*

Algoritma yang digunakan dalam pelatihan model jaringan saraf untuk memperbarui bobot dan bias berdasarkan gradien dari *loss function*. *Optimizer* yang akan diterapkan pada penelitian ini adalah *optimizer Adam*.

f. Fungsi aktivasi.

Proses ini melibatkan beberapa langkah berupa:

a. Menghitung nilai *forget gate* menggunakan Persamaan (2.2)

- b. Menghitung nilai *input gate* menggunakan Persamaan (2.6) dan (2.7)
 - c. Memperbarui memori yang berada pada *cell* menggunakan Persamaan (2.14)
 - d. Menghitung *output gate* dan nilai *ouput* akhir menggunakan Persamaan (2.17) dan (2.21)
- 4. Validasi model *training* dengan data validasi.

Menerapkan model menggunakan data validasi serta menentukan nilai *batch size* dan *epoch* yang akan digunakan untuk memberikan hasil prediksi terbaik.
- 5. Mengevaluasi model

Model dievaluasi dengan menghitung nilai *error* menggunakan *Root Mean Square Error* (RMSE) dan *Mean Absolute Percentage Error* (MAPE) pada Persamaan (2.34) dan (2.35).
- 6. Menguji model menggunakan data uji

Menerapkan model terbaik pada data uji untuk menampilkan hasil prediksi yang kemudian akan dibandingkan dengan data aktual. Hasil perbandingan kemudian divisualisasikan dalam bentuk grafik dan numerik.

BAB IV

HASIL DAN PEMBAHASAN

4.2 Dataset

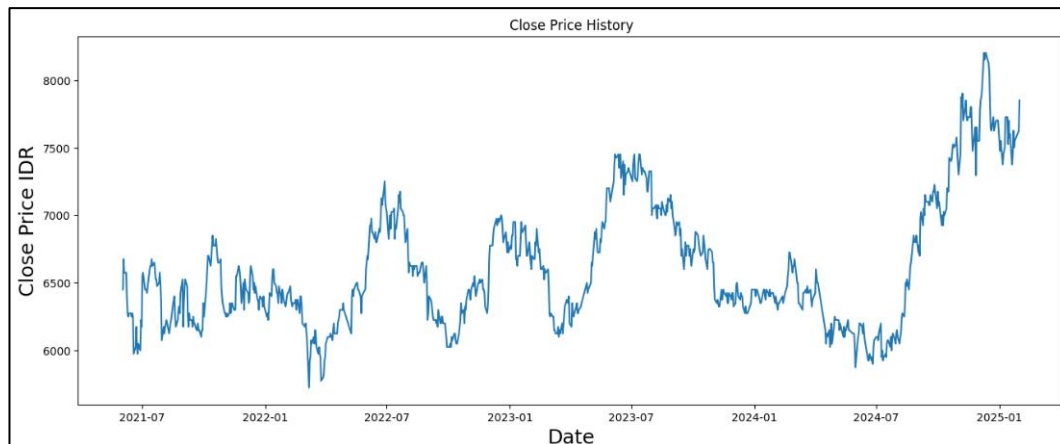
Data yang digunakan adalah data harian harga saham PT. Indofood Sukses Makmur Tbk pada periode 02 Juni 2021 hingga 31 Januari 2024 dimana jumlah data sebanyak 889 data seperti pada lampiran 1. Data diambil langsung melalui *website yahoo finance* dengan kode saham “INDF.JK” yang kemudian disimpan dalam file *excel*. Harga penutupan saham dipilih karena mencerminkan nilai akhir dari suatu saham dalam satu sesi perdagangan dan sering digunakan dalam berbagai model analisis keuangan serta prediksi pergerakan harga saham.

Struktur dataframe akan terdiri dari dua kolom utama yaitu *date* dan *close*, sebagaimana ditampilkan pada Tabel 4.1. Kolom *date* merepresentasikan informasi mengenai tanggal perdagangan, sedangkan kolom *close* menunjukkan nilai harga penutupan saham pada setiap periode yang diamati.

Tabel 4.1 Dataset *Close Price*

<i>Date</i>	<i>Close</i>
02/06/2021	6450
03/06/2021	6675
04/06/2021	6575
07/06/2021	6575
08/06/2021	6375
...	...
22/01/2025	7625
23/01/2025	7500
24/01/2025	7550
30/01/2025	7625
31/01/2025	7850

Pergerakan harga penutupan saham PT. Indofood Sukses Makmur Tbk selama 4 tahun terakhir terdapat pada grafik Gambar 4.1



Gambar 4.1 Grafik Data Saham INDF.JK

Grafik yang ditampilkan merupakan representasi historis dari pergerakan harga penutupan saham “INDF.JK” dalam satuan Rupiah (IDR). Sumbu horizontal (sumbu X) menunjukkan dimensi waktu, sedangkan sumbu vertikal (sumbu Y) menunjukkan nilai harga penutupan saham pada setiap periode yang diamati.

Berdasarkan Gambar 4.1, dapat diamati bahwa harga saham mengalami fluktuasi yang signifikan sepanjang periode yang ditampilkan. Pada tahun-tahun tertentu, pergerakan harga saham mengalami kenaikan yang cukup tajam tetapi disertai dengan penurunan dalam beberapa bulan berikutnya seperti pada pertengahan tahun 2022. Tren serupa kembali terlihat pada tahun 2023 di mana harga saham mengalami lonjakan hingga mencapai titik tertinggi baru sebelum kembali mengalami penurunan secara bertahap. Memasuki awal tahun 2024, harga saham berada dalam kondisi yang relatif rendah sebelum mengalami kenaikan tajam yang berkelanjutan hingga akhir periode observasi. Volatilitas yang cukup tinggi pada grafik ini mencerminkan adanya ketidakpastian pada fluktuasi harga

saham sehingga perlu perhatian khusus jika ingin dilakukan pengambilan keputusan investasi.

Kolom *close* yang telah dipilih kemudian didefinisikan sebagai variabel X , yang berperan sebagai variabel bebas dalam penelitian ini. Variabel X akan merepresentasikan data historis harga penutupan saham yang akan menjadi dasar dalam membangun model prediksi. Selanjutnya, pada tahap *preprocessing* data akan dibuat suatu variabel target yang disebut sebagai variabel Y . Variabel Y merupakan variabel terikat dalam penelitian ini dan berfungsi sebagai nilai yang akan diprediksi berdasarkan pola dari variabel X .

4.3 *Preprocessing*

4.3.1 *Ascending*

Tahap *ascending* merupakan proses pengurutan data berdasarkan urutan tanggal, dimulai dari tanggal terlama hingga tanggal terbaru. Urutan waktu yang benar sangat penting dalam analisis data saham, karena pergerakan harga saham dari waktu ke waktu membentuk pola yang nantinya akan digunakan dalam proses prediksi. Karena data yang digunakan dalam penelitian ini tergolong sebagai data *time series*, maka secara prinsip urutannya sudah benar sesuai dengan tanggal pencatatan. Namun diperlukan verifikasi lebih lanjut untuk menghindari kemungkinan adanya ketidakteraturan dalam susunan data akibat kesalahan dalam proses pengambilan atau penyimpanan data.

Terdapat 889 baris data yang perlu diperiksa dalam penelitian ini. Mengingat jumlah data yang cukup besar, proses pengurutan secara manual tidak

efisien dan berisiko tinggi terhadap kesalahan. Oleh karena itu, program komputer digunakan untuk memastikan bahwa data benar-benar tersusun secara kronologis.

4.3.2 Ekstraksi Fitur

Tahap ini akan menambahkan fitur baru pada *dataframe* yaitu nilai RSI yang merupakan salah satu indikator analisis teknikal untuk mengukur tingkat kejenuhan suatu saham apakah berada dalam kondisi *overbought* atau *oversold*. Pada penelitian ini, perhitungan nilai RSI menggunakan periode 14 hari berdasarkan rekomendasi dari J. Welles Wilder yang pertama kali memperkenalkan RSI, dengan begitu maka perhitungan nilai RSI dapat dimulai pada hari ke-15 karena dibutuhkan data historis selama 14 hari pertama untuk menghitung nilai rata-rata kenaikan (*average gain*) dan rata-rata penurunan (*average loss*). Perhitungan nilai RSI dapat menggunakan Persamaan (2.31), untuk memberikan gambaran yang lebih jelas berikut diberikan contoh implementasi perhitungan manual RSI:

Tabel 4.2 Perhitungan Nilai Untung Rugi

Hari ke-	Tanggal	Close	Keuntungan	Kerugian
1	02/06/2021	6.450	-	-
2	03/06/2021	6.675	225	0
3	04/06/2021	6.575	0	100
4	07/06/2021	6.575	0	0
5	08/06/2021	6.375	0	200
6	09/06/2021	6.275	0	100
7	10/06/2021	6.250	0	25
8	11/06/2021	6.275	25	0
9	14/06/2021	6.275	0	0
10	15/06/2021	6.250	0	25
11	16/06/2021	6.275	25	0
12	17/06/2021	6.250	0	25
13	18/06/2021	5.975	0	275

14	21/06/2021	6.025	50	0
15	22/06/2021	6.175	150	0
16	23/06/2021	6.025	0	150

$$\begin{aligned}
 \text{Hari ke-15} \quad : \quad RS &= \frac{\text{Rata-rata keuntungan}}{\text{rata-rata kerugian}} \\
 &= \frac{23,21}{53,57} \\
 &= 0,433
 \end{aligned}$$

$$\begin{aligned}
 RSI &= 100 - \left(\frac{100}{1+RS} \right) \\
 &= 100 - \left(\frac{100}{1+0,433} \right) \\
 &= 38,775
 \end{aligned}$$

$$\begin{aligned}
 \text{Hari ke-16} \quad : \quad RS &= \frac{\text{Rata-rata keuntungan}}{\text{rata-rata kerugian}} \\
 &= \frac{17,85}{53,57} \\
 &= 0,333
 \end{aligned}$$

$$\begin{aligned}
 RSI &= 100 - \left(\frac{100}{1+RS} \right) \\
 &= 100 - \left(\frac{100}{1+0,333} \right) \\
 &= 21,739
 \end{aligned}$$

$$\begin{aligned}
 \text{Hari ke-17} \quad : \quad RS &= \frac{\text{Rata-rata keuntungan}}{\text{rata-rata kerugian}} \\
 &= \frac{17,85}{57,14} \\
 &= 0,312
 \end{aligned}$$

$$\begin{aligned}
 RSI &= 100 - \left(\frac{100}{1+RS} \right) \\
 &= 100 - \left(\frac{100}{1+0,266} \right) \\
 &= 22,727
 \end{aligned}$$

Setelah dilakukan perhitungan nilai RSI pada seluruh data, langkah selanjutnya adalah menggabungkan nilai-nilai tersebut ke dalam *dataframe* utama RSI seperti pada Tabel 4.3 agar dapat digunakan pada analisis lebih lanjut.

Dengan adanya penambahan fitur RSI pada *dataframe*, analisis yang dilakukan pada tahap selanjutnya dapat mempertimbangkan informasi tambahan untuk meningkatkan akurasi model dalam memprediksi pergerakan harga saham.

Tabel 4.3 Dataset *Close Price* dan Nilai RSI

<i>Date</i>	<i>Close</i>	RSI
22/06/2021	6175	38,775
23/06/2021	6025	21,739
24/06/2021	5975	22,727
25/06/2021	6050	27,659
....		
22/01/2025	7625	54,687
23/01/2025	7500	48,484
24/01/2025	7550	55,737
30/01/2025	7625	55,737
31/01/2025	7850	60,869

Tahap selanjutnya adalah melakukan analisis deskriptif terhadap fitur *close* dan RSI. Analisis ini bertujuan untuk memberikan Gambaran umum mengenai karakteristik data seperti sebaran nilai, rata-rata, nilai minimum dan maksimum, serta standar deviasi. Selain itu, analisis ini juga berfungsi untuk memastikan bahwa nilai pada masing-masing fitur berjumlah sama, agar tidak mempengaruhi keakuratan hasil penelitian. Hasil analisis deskriptif pada fitur *close* dan RSI disajikan pada Tabel 4.4.

Tabel 4.4 Hasil Deskriptif Dataset *Close Price* dan Nilai RSI

	<i>Close</i>	RSI
Count	876	876
Mean	6615,66210	50,842780
Std	466,27094	14,916527
25%	6275	40,740741
50%	6475	50
75%	6875	60,618480
Min	5725	12,903226
Max	8200	90,909091

4.3.3 Normalisasi

Pada tahap ini nilai pada setiap fitur akan dinormalisasikan pada rentang 0 hingga 1 guna memastikan bahwa seluruh data yang digunakan untuk pelatihan model LSTM berada dalam skala yang seragam. Proses normalisasi ini bertujuan untuk meningkatkan stabilitas komputasi dalam algoritma pemodelan. Normalisasi juga bertujuan untuk mengurangi dominasi nilai pada fitur dengan skala yang lebih besar terhadap fitur lainnya, sehingga hasil analisis lebih akurat dan dapat diinterpretasikan dengan lebih baik. Proses normalisasi dapat dihitung dengan menggunakan Persamaan (2.33), berikut disajikan contoh perhitungan manual dalam proses normalisasi.

Close

$$X_1 = \frac{x - \min(X_1)}{\max(X_1) - \min(X_1)}$$

$$\text{Hari ke-1} : X_{11} = \frac{6025-5725}{8200-5725} = 0,173$$

$$\text{Hari ke-2} : X_{12} = \frac{6175-5725}{8200-5725} = 0,261$$

$$\text{Hari ke-3} : X_{13} = \frac{6025-5725}{8200-5725} = 0,173$$

RSI

$$X_2 = \frac{x - \min(X_2)}{\max(X_2) - \min(X_2)}$$

$$\text{Hari ke-1} : X_{21} = \frac{30,232-12,903}{90,909-12,903} = 0,222$$

$$\text{Hari ke-2} : X_{22} = \frac{38,775-12,903}{90,909-12,903} = 0,331$$

$$\text{Hari ke-3} : X_{23} = \frac{21,739-12,903}{90,909-12,903} = 0,113$$

setelah seluruh data yang digunakan selama pelatihan model LSTM melalui proses normalisasi, hasil akhir disajikan pada Tabel 4.5.

Tabel 4.5 Hasil Normalisasi Dataset *Close Price* dan Nilai RSI

<i>Date</i>	<i>Close</i>	<i>RSI</i>
21/06/2021	0,173	0,222
22/06/2021	0,260	0,331
23/06/2021	0,173	0,113
....		
28/10/2024	0,972	0,861
29/10/2024	1	0,965
30/10/2024	0,945	0,814

4.3.4 Pembagian Data

Pada tahap pembagian data, data sebelumnya akan dibagi menjadi dua kelompok utama yaitu data latih dan data uji. Data uji mencakup periode 01 November 2024 hingga 31 Januari 2025 sesuai dengan rumusan masalah yang menjadi fokus penelitian. Dengan demikian, penelitian ini akan menggunakan dua subset data yaitu data latih dan data uji. Data latih digunakan untuk membangun dan melatih model agar dapat mengenali pola dalam data. Sementara data uji digunakan untuk menguji model yang telah dilatih dan divalidasi dengan tujuan menilai performa akhir model serta menampilkan hasil prediksi berdasarkan data yang belum pernah digunakan dalam proses pelatihan.

Karena data saham merupakan data *time series*, pembagian data tidak dilakukan dengan acak melainkan tetap mempertahankan urutan kronologisnya. Pendekatan ini bertujuan untuk memastikan bahwa model yang dikembangkan dapat belajar dari pola historis dengan baik serta mampu menghasilkan prediksi yang lebih akurat. Pembagian data secara rinci disajikan pada Tabel 4.6.

Tabel 4.6 Hasil Pembagian Dataset *Close Price* dan Nilai RSI

Satuan	Data Latih	Data Uji
Jumlah	717	59
Periode	21 Juni 2021 - 30 Oktober 2024	1 November 2024 - 31 Januari 2025.

Berdasarkan pembagian data yang telah ditentukan, data latih terdiri atas 717 data yang mencakup periode 21 Juni 2021 hingga 30 Oktober 2024. Sementara data uji terdiri atas 59 data yang mencakup periode 1 November 2024 hingga 31 Januari 2025. Selanjutnya hanya data latih yang akan digunakan dalam proses pelatihan model LSTM, sementara data uji akan digunakan pada tahap pengujian untuk mengevaluasi performa model setelah pelatihan selesai.

4.3.5 Segmentasi

Segmentasi data merupakan proses pembagian dan pengelompokan data historis ke dalam rangkaian (*window*) yang lebih kecil untuk menangkap pola pergerakan data dalam rentang waktu yang sudah ditentukan, guna meningkatkan akurasi dalam proses prediksi. Setiap kelompok data biasanya akan membentuk rangkaian sepanjang 30, 60, atau 90 mengacu pada kajian penelitian terdahulu yang sudah dilakukan. Rangkaian ini nantinya akan diambil dari data historis yang ada dan digunakan untuk memprediksi nilai target pada urutan selanjutnya. Pada penelitian ini panjang segmen yang digunakan adalah 30 data, yang berarti setiap rangkain data akan terdiri dari 30 nilai sebelumnya untuk memprediksi nilai target pada periode berikutnya.

Gambaran lebih lanjut mengenai proses segmentasi data terdapat pada Tabel 4.7, yang menyajikan ilustrasi pembentukan kelompok data berdasarkan metode *sliding window*.

Tabel 4.7 Ilustrasi Pengelompokan Data dalam Bentuk Sekuensial

x	y
$x_1, x_2, x_3, \dots, x_{30}$	x_{31}
$x_2, x_3, x_4, \dots, x_{31}$	x_{32}
$x_3, x_4, x_5, \dots, x_{32}$	x_{33}
$x_4, x_5, x_6, \dots, x_{33}$	x_{34}

4.4 Konstruksi Model LSTM

Pada tahap ini, dilakukan perancangan model LSTM dengan menginisialisasi parameter-parameter utama yang berperan dalam proses pembelajaran model. Parameter yang diterapkan meliputi jumlah *hidden layer*, jumlah neuron dalam *hidden layer*, *batch size*, jumlah *epoch* maksimum, jenis *optimizer*, serta fungsi aktivasi. Penentuan nilai parameter dapat disesuaikan dengan kebutuhan peneliti dan tidak ada aturan yang pasti dalam penentuannya, sehingga pada tahap ini akan dilakukan beberapa percobaan menggunakan nilai yang berbeda-beda guna mengetahui penggunaan nilai yang akan menghasilkan prediksi terbaik. Pemilihan nilai untuk setiap parameter ini sangat berpengaruh terhadap performa model dalam melakukan prediksi terhadap data saham yang bersifat *time series*. Penentuan *hyperparameter* yang digunakan pada penelitian ini merujuk pada penelitian sebelumnya milik (Puteri, 2023).

Nilai-nilai *hyperparameter* yang akan diuji untuk setiap parameter adalah sebagai berikut:

- a. *Hidden layer* : 4
- b. *Neuron hidden* : 50
- c. *Nilai batch size* : 4, 16, 32, dan 64
- d. *Epoch maksimum* : 50 dan 100
- e. *Optimizer* : Adam
- f. *Fungsi aktivasi* : *Tanh* dan Sigmoid

Proses pencarian kombinasi parameter terbaik dilakukan menggunakan metode *grid search*. Pada setiap percobaan kombinasi parameter, dilakukan evaluasi menggunakan teknik *k-fold cross validation* dengan lima *fold*, di mana data pelatihan dan validasi dibagi menjadi lima subset. Pada setiap iterasi, satu subset digunakan sebagai data validasi sedangkan empat subset lainnya digunakan sebagai data latih. Setiap kombinasi parameter diuji dengan melakukan pelatihan dan validasi di lima *fold* berbeda. Setiap *fold* akan dilakukan evaluasi performa model dengan menggunakan *Root Mean Squared Error* (RMSE) dan *Mean Absolute Percentage Error* (MAPE). Nilai rata-rata dari kelima *fold* digunakan untuk menilai stabilitas dan keakuratan model setiap kombinasi parameter. Kombinasi dengan nilai rata-rata RMSE terkecil akan disimpan sebagai model terbaik.

4.5 Perhitungan Manual Metode LSTM

Pada tahap ini, akan dilakukan perhitungan LSTM untuk setiap *time step*. Perhitungan ini bertujuan untuk memahami bagaimana model LSTM memproses data historis secara berulang melalui mekanisme *gate-gate* utama yang terdapat

dalam arsitektur LSTM yaitu *forget gate*, *input gate*, *cell state* dan *output gate*. Sebagai contoh, penulis mengambil sampel perhitungan pada layer ke 0 dan 1 menggunakan lima baris data pertama. Pada perhitungan ini, jumlah *time step* yang digunakan adalah 3 sedangkan jumlah *hidden unit* yang diterapkan sebanyak 2. Pada tahap awal, nilai *weight* dan bias akan ditentukan secara acak oleh program dan akan terus diperbarui secara berkala menggunakan algoritma optimasi *Adam Optimizer* dengan tujuan meminimalkan *error* dalam prediksi.

Secara matematis, perhitungan LSTM pada layer ke 0 dan 1 dapat dijelaskan dengan persamaan sebagai berikut:

Diketahui:

- Data = $\begin{bmatrix} 0,507 & 0,365 \\ 0,579 & 0,456 \\ 0,594 & 0,365 \\ 0,594 & 0,365 \\ 0,594 & 0,383 \end{bmatrix}$
- $X_1 = \begin{bmatrix} 0,507 & 0,365 \\ 0,579 & 0,456 \\ 0,594 & 0,365 \end{bmatrix}, Y_1 = [0,594 \quad 0,365]$
- $X_2 = \begin{bmatrix} 0,579 & 0,456 \\ 0,594 & 0,365 \\ 0,594 & 0,365 \end{bmatrix}, Y_2 = [0,594 \quad 0,383]$
- Layer ke-0
- $X_1 = \begin{bmatrix} 0,507 & 0,365 \\ 0,579 & 0,456 \\ 0,594 & 0,365 \end{bmatrix}$

Perhitungan pada *time step* ke-1 ($t = 1$)

Variabel	Nilai
x_1	$\begin{bmatrix} 0,507 \\ 0,365 \end{bmatrix}$
h_0	$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$
c_0	$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$

Menghitung *forget gate* menggunakan Persamaan (2.2)

Variabel	Nilai
W_{ϕ}	$\begin{bmatrix} 0,287 & 0,804 \\ 0,236 & -0,111 \end{bmatrix}$
$W_{h\phi}$	$\begin{bmatrix} 0,062 & -0,615 \\ -0,651 & -0,208 \end{bmatrix}$
b_{ϕ}	$\begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix}$

$$\begin{aligned}
 \phi_1 &= \sigma (W_{\phi}^T x_1 + W_{h\phi}^T h_0 + b_{\phi}) \\
 &= \sigma \left(\begin{bmatrix} 0,287 & 0,236 \\ 0,804 & -0,111 \end{bmatrix} \begin{bmatrix} 0,507 \\ 0,365 \end{bmatrix} + \begin{bmatrix} 0,062 & -0,615 \\ -0,615 & -0,208 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} 0,231 \\ 0,367 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} 0,709 \\ 0,318 \end{bmatrix} \right) \\
 &= \begin{bmatrix} 0,671 \\ 0,578 \end{bmatrix}
 \end{aligned}$$

Menghitung *input gate* menggunakan Persamaan (2.6) dan (2.7)

Variabel	Nilai
W_k	$\begin{bmatrix} -0,298 & 0,716 \\ -0,261 & 1,038 \end{bmatrix}$
W_u	$\begin{bmatrix} 0,281 & -0,841 \\ -0,506 & 0,083 \end{bmatrix}$
W_{hk}	$\begin{bmatrix} -0,116 & -1,073 \\ -0,132 & 0,198 \end{bmatrix}$
W_{hu}	$\begin{bmatrix} -0,219 & 0,301 \\ -0,433 & -0,265 \end{bmatrix}$
b_k	$\begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix}$
b_u	$\begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix}$

$$\begin{aligned}
k_1 &= \sigma (W_k^T x_1 + W_{hk}^T h_0 + b_k) \\
&= \sigma \left(\begin{bmatrix} -0,298 & -0,261 \\ 0,716 & 1,038 \end{bmatrix} \begin{bmatrix} 0,507 \\ 0,365 \end{bmatrix} + \begin{bmatrix} -0,116 & -0,132 \\ -1,073 & 0,198 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} -0,246 \\ 0,741 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} 0,837 \\ 1,368 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0,697 \\ 0,797 \end{bmatrix} \\
u_1 &= \sigma (W_u^T x_1 + W_{hu}^T h_0 + b_u) \\
&= \sigma \left(\begin{bmatrix} 0,281 & -0,506 \\ -0,841 & 0,083 \end{bmatrix} \begin{bmatrix} 0,507 \\ 0,365 \end{bmatrix} + \begin{bmatrix} -0,219 & -0,433 \\ 0,301 & -0,265 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} -0,042 \\ -0,396 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} -0,03 \\ -0,397 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0,492 \\ 0,402 \end{bmatrix}
\end{aligned}$$

Menghitung *cell state* menggunakan Persamaan (2.14)

Variabel	Nilai
k_1	$\begin{bmatrix} 0,697 \\ 0,797 \end{bmatrix}$
u_1	$\begin{bmatrix} 0,492 \\ 0,402 \end{bmatrix}$
ϕ_1	$\begin{bmatrix} 0,671 \\ 0,578 \end{bmatrix}$
c_0	$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$

$$\begin{aligned}
c_1 &= k_1 \odot u_1 + \phi_1 \odot c_0 \\
&= \begin{bmatrix} 0,697 \\ 0,797 \end{bmatrix} \odot \begin{bmatrix} 0,492 \\ 0,402 \end{bmatrix} + \begin{bmatrix} 0,671 \\ 0,578 \end{bmatrix} \odot \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} 0,342 \\ 0,321 \end{bmatrix}
\end{aligned}$$

Menghitung *output gate* menggunakan Persamaan (2.17)

Variabel	Nilai
W_{ω}	$\begin{bmatrix} 0,411 & 0,011 \\ -0,401 & 0,711 \end{bmatrix}$
$W_{h\omega}$	$\begin{bmatrix} -0,011 & 0,172 \\ 0,276 & -0,309 \end{bmatrix}$
b_{ω}	$\begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix}$

$$\begin{aligned}
 \omega_1 &= \sigma (W_{\omega}^T x_1 + W_{h\omega}^T h_0 + b_{\omega}) \\
 &= \sigma \left(\begin{bmatrix} 0,411 & -0,401 \\ 0,011 & 0,711 \end{bmatrix} \begin{bmatrix} 0,507 \\ 0,365 \end{bmatrix} + \begin{bmatrix} -0,011 & 0,276 \\ 0,172 & -0,309 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} 0,062 \\ 0,265 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} 0,524 \\ 0,687 \end{bmatrix} \right) \\
 &= \begin{bmatrix} 0,628 \\ 0,665 \end{bmatrix}
 \end{aligned}$$

Menghitung *hidden state* menggunakan Persamaan (2.15)

Variabel	Nilai
c_1	$\begin{bmatrix} 0,342 \\ 0,321 \end{bmatrix}$
ω_1	$\begin{bmatrix} 0,628 \\ 0,665 \end{bmatrix}$

$$\begin{aligned}
 h_1 &= \tanh(c_1) \odot \omega_1 \\
 &= \tanh \left(\begin{bmatrix} 0,342 \\ 0,321 \end{bmatrix} \right) \odot \begin{bmatrix} 0,628 \\ 0,665 \end{bmatrix} \\
 &= \begin{bmatrix} 0,658 \\ 0,621 \end{bmatrix} \odot \begin{bmatrix} 0,628 \\ 0,665 \end{bmatrix} \\
 &= \begin{bmatrix} 0,413 \\ 0,412 \end{bmatrix}
 \end{aligned}$$

Perhitungan pada *time step* ke-2 ($t = 2$)

Variabel	Nilai
x_2	$\begin{bmatrix} 0,579 \\ 0,456 \end{bmatrix}$
h_1	$\begin{bmatrix} 0,413 \\ 0,412 \end{bmatrix}$
c_1	$\begin{bmatrix} 0,342 \\ 0,321 \end{bmatrix}$

Menghitung *forget gate* menggunakan Persamaan (2.2)

Variabel	Nilai
W_ϕ	$\begin{bmatrix} 0,287 & 0,804 \\ 0,236 & -0,111 \end{bmatrix}$
$W_{h\phi}$	$\begin{bmatrix} 0,062 & -0,615 \\ -0,651 & -0,208 \end{bmatrix}$
b_ϕ	$\begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix}$

$$\begin{aligned}
 \phi_2 &= \sigma (W_\phi^T x_2 + W_{h\phi}^T h_0 + b_\phi) \\
 &= \sigma \left(\begin{bmatrix} 0,287 & 0,236 \\ 0,804 & -0,111 \end{bmatrix} \begin{bmatrix} 0,579 \\ 0,456 \end{bmatrix} + \right. \\
 &\quad \left. \begin{bmatrix} 0,062 & -0,651 \\ -0,615 & -0,208 \end{bmatrix} \begin{bmatrix} 0,413 \\ 0,412 \end{bmatrix} + \begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} 0,273 \\ 0,414 \end{bmatrix} + \begin{bmatrix} -0,242 \\ -0,339 \end{bmatrix} + \begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} 0,509 \\ 0,026 \end{bmatrix} \right) \\
 &= \begin{bmatrix} 0,624 \\ 0,506 \end{bmatrix}
 \end{aligned}$$

Menghitung *input gate* menggunakan Persamaan (2.6) dan (2.7)

Variabel	Nilai
W_k	$\begin{bmatrix} -0,298 & 0,716 \\ -0,261 & 1,038 \end{bmatrix}$
W_u	$\begin{bmatrix} 0,281 & -0,841 \\ -0,506 & 0,083 \end{bmatrix}$
W_{hk}	$\begin{bmatrix} -0,116 & -1,073 \\ -0,132 & 0,198 \end{bmatrix}$
W_{hu}	$\begin{bmatrix} -0,219 & 0,301 \\ -0,433 & -0,265 \end{bmatrix}$

b_k	$\begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix}$
b_u	$\begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix}$

$$\begin{aligned}
k_2 &= \sigma (W_k^T x_2 + W_{hk}^T h_0 + b_k) \\
&= \sigma \left(\begin{bmatrix} -0,298 & -0,261 \\ 0,716 & 1,038 \end{bmatrix} \begin{bmatrix} 0,579 \\ 0,456 \end{bmatrix} \right. \\
&\quad \left. + \begin{bmatrix} -0,116 & -0,132 \\ -1,073 & 0,198 \end{bmatrix} \begin{bmatrix} 0,413 \\ 0,412 \end{bmatrix} + \begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} -0,291 \\ 0,887 \end{bmatrix} + \begin{bmatrix} -0,102 \\ -0,361 \end{bmatrix} + \begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} 0,69 \\ 1,153 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0,665 \\ 0,761 \end{bmatrix} \\
u_2 &= \sigma (W_u^T x_2 + W_{hu}^T h_1 + b_u) \\
&= \sigma \left(\begin{bmatrix} 0,281 & -0,506 \\ -0,841 & 0,083 \end{bmatrix} \begin{bmatrix} 0,579 \\ 0,456 \end{bmatrix} + \right. \\
&\quad \left. \begin{bmatrix} -0,219 & -0,433 \\ 0,301 & -0,265 \end{bmatrix} \begin{bmatrix} 0,413 \\ 0,412 \end{bmatrix} + \begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} -0,068 \\ -0,449 \end{bmatrix} + \begin{bmatrix} -0,268 \\ 0,015 \end{bmatrix} + \begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} -0,324 \\ -0,435 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0,419 \\ 0,392 \end{bmatrix}
\end{aligned}$$

Menghitung *cell state* menggunakan Persamaan (2.14)

Variabel	Nilai
k_2	$\begin{bmatrix} 0,665 \\ 0,761 \end{bmatrix}$
u_2	$\begin{bmatrix} 0,419 \\ 0,392 \end{bmatrix}$
ϕ_2	$\begin{bmatrix} 0,624 \\ 0,506 \end{bmatrix}$
c_1	$\begin{bmatrix} 0,342 \\ 0,321 \end{bmatrix}$

$$\begin{aligned}
c_2 &= k_2 \odot u_2 + \phi_2 \odot c_1 \\
&= \begin{bmatrix} 0,665 \\ 0,761 \end{bmatrix} \odot \begin{bmatrix} 0,419 \\ 0,392 \end{bmatrix} + \begin{bmatrix} 0,624 \\ 0,506 \end{bmatrix} \odot \begin{bmatrix} 0,342 \\ 0,321 \end{bmatrix} \\
&= \begin{bmatrix} 0,278 \\ 0,298 \end{bmatrix} + \begin{bmatrix} 0,213 \\ 0,162 \end{bmatrix} \\
&= \begin{bmatrix} 0,491 \\ 0,46 \end{bmatrix}
\end{aligned}$$

Menghitung *output gate* menggunakan Persamaan (2.17)

Variabel	Nilai
W_{ω}	$\begin{bmatrix} 0,411 & 0,011 \\ -0,401 & 0,711 \end{bmatrix}$
$W_{h\omega}$	$\begin{bmatrix} -0,011 & 0,172 \\ 0,276 & -0,309 \end{bmatrix}$
b_{ω}	$\begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix}$

$$\begin{aligned}
\omega_2 &= \sigma (W_{\omega}^T x_2 + W_{h\omega}^T h_0 + b_{\omega}) \\
&= \sigma \left(\begin{bmatrix} 0,411 & -0,401 \\ 0,011 & 0,711 \end{bmatrix} \begin{bmatrix} 0,579 \\ 0,456 \end{bmatrix} + \right. \\
&\quad \left. \begin{bmatrix} -0,011 & 0,276 \\ 0,172 & -0,309 \end{bmatrix} \begin{bmatrix} 0,413 \\ 0,412 \end{bmatrix} + \begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} 0,421 \\ 0,331 \end{bmatrix} + \begin{bmatrix} 0,109 \\ -0,056 \end{bmatrix} + \begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} 0,992 \\ 0,697 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0,729 \\ 0,667 \end{bmatrix}
\end{aligned}$$

Menghitung *hidden state* menggunakan Persamaan (2.15)

Variabel	Nilai
c_2	$\begin{bmatrix} 0,491 \\ 0,46 \end{bmatrix}$
ω_2	$\begin{bmatrix} 0,729 \\ 0,667 \end{bmatrix}$

$$\begin{aligned}
h_2 &= \tanh(c_2) \odot \omega_2 \\
&= \tanh\left(\begin{bmatrix} 0,491 \\ 0,46 \end{bmatrix}\right) \odot \begin{bmatrix} 0,729 \\ 0,667 \end{bmatrix} \\
&= \begin{bmatrix} 0,911 \\ 0,861 \end{bmatrix} \odot \begin{bmatrix} 0,729 \\ 0,667 \end{bmatrix} \\
&= \begin{bmatrix} 0,664 \\ 0,574 \end{bmatrix}
\end{aligned}$$

Perhitungan pada *time step* ke-3 ($t = 3$)

Variabel	Nilai
x_3	$\begin{bmatrix} 0,594 \\ 0,365 \end{bmatrix}$
h_2	$\begin{bmatrix} 0,664 \\ 0,574 \end{bmatrix}$
c_2	$\begin{bmatrix} 0,491 \\ 0,46 \end{bmatrix}$

Menghitung *forget gate* menggunakan Persamaan (2.2)

Variabel	Nilai
W_ϕ	$\begin{bmatrix} 0,287 & 0,804 \\ 0,236 & -0,111 \end{bmatrix}$
$W_{h\phi}$	$\begin{bmatrix} 0,062 & -0,615 \\ -0,651 & -0,208 \end{bmatrix}$
b_ϕ	$\begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix}$

$$\begin{aligned}
\phi_3 &= \sigma (W_\phi^T x_3 + W_{h\phi}^T h_0 + b_\phi) \\
&= \sigma \left(\begin{bmatrix} 0,287 & 0,236 \\ 0,804 & -0,111 \end{bmatrix} \begin{bmatrix} 0,594 \\ 0,365 \end{bmatrix} + \begin{bmatrix} 0,062 & -0,651 \\ -0,615 & -0,208 \end{bmatrix} \begin{bmatrix} 0,664 \\ 0,574 \end{bmatrix} + \begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} 0,256 \\ 0,437 \end{bmatrix} + \begin{bmatrix} -0,332 \\ -0,527 \end{bmatrix} + \begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} 0,402 \\ -0,139 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0,599 \\ 0,465 \end{bmatrix}
\end{aligned}$$

Menghitung *input gate* menggunakan Persamaan (2.6) dan (2.7)

Variabel	Nilai
W_k	$\begin{bmatrix} -0,298 & 0,716 \\ -0,261 & 1,038 \end{bmatrix}$
W_u	$\begin{bmatrix} 0,281 & -0,841 \\ -0,506 & 0,083 \end{bmatrix}$
W_{hk}	$\begin{bmatrix} -0,116 & -1,073 \\ -0,132 & 0,198 \end{bmatrix}$
W_{hu}	$\begin{bmatrix} -0,219 & 0,301 \\ -0,433 & -0,265 \end{bmatrix}$
b_k	$\begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix}$
b_u	$\begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix}$

$$\begin{aligned}
 k_3 &= \sigma (W_k^T x_3 + W_{hk}^T h_0 + b_g) \\
 &= \sigma \left(\begin{bmatrix} -0,298 & -0,261 \\ 0,716 & 1,038 \end{bmatrix} \begin{bmatrix} 0,594 \\ 0,365 \end{bmatrix} + \begin{bmatrix} -0,116 & -0,132 \\ -1,073 & 0,198 \end{bmatrix} \begin{bmatrix} 0,664 \\ 0,574 \end{bmatrix} + \begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} -0,272 \\ 0,804 \end{bmatrix} + \begin{bmatrix} -0,152 \\ -0,598 \end{bmatrix} + \begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} 0,659 \\ 0,833 \end{bmatrix} \right) \\
 &= \begin{bmatrix} 0,659 \\ 0,696 \end{bmatrix} \\
 u_3 &= \sigma (W_u^T x_3 + W_{hu}^T h_2 + b_u) \\
 &= \sigma \left(\begin{bmatrix} 0,281 & -0,506 \\ -0,841 & 0,083 \end{bmatrix} \begin{bmatrix} 0,594 \\ 0,365 \end{bmatrix} + \begin{bmatrix} -0,219 & -0,433 \\ 0,301 & -0,265 \end{bmatrix} \begin{bmatrix} 0,664 \\ 0,574 \end{bmatrix} + \begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} -0,017 \\ -0,469 \end{bmatrix} + \begin{bmatrix} -0,393 \\ 0,047 \end{bmatrix} + \begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} -0,398 \\ -0,423 \end{bmatrix} \right) \\
 &= \begin{bmatrix} 0,401 \\ 0,395 \end{bmatrix}
 \end{aligned}$$

Menghitung *cell state* menggunakan Persamaan (2.14)

Variabel	Nilai
k_3	$\begin{bmatrix} 0,659 \\ 0,696 \end{bmatrix}$
u_3	$\begin{bmatrix} 0,401 \\ 0,395 \end{bmatrix}$
ϕ_3	$\begin{bmatrix} 0,599 \\ 0,465 \end{bmatrix}$
c_2	$\begin{bmatrix} 0,491 \\ 0,46 \end{bmatrix}$

$$\begin{aligned}
 c_3 &= k_3 \odot u_3 + \phi_3 \odot c_2 \\
 &= \begin{bmatrix} 0,659 \\ 0,696 \end{bmatrix} \odot \begin{bmatrix} 0,401 \\ 0,395 \end{bmatrix} + \begin{bmatrix} 0,599 \\ 0,465 \end{bmatrix} \odot \begin{bmatrix} 0,491 \\ 0,46 \end{bmatrix} \\
 &= \begin{bmatrix} 0,264 \\ 0,274 \end{bmatrix} + \begin{bmatrix} 0,294 \\ 0,213 \end{bmatrix} \\
 &= \begin{bmatrix} 0,558 \\ 0,487 \end{bmatrix}
 \end{aligned}$$

Menghitung *output gate* menggunakan Persamaan (2.17)

Variabel	Nilai
W_ω	$\begin{bmatrix} 0,411 & 0,011 \\ -0,401 & 0,711 \end{bmatrix}$
$W_{h\omega}$	$\begin{bmatrix} -0,011 & 0,172 \\ 0,276 & -0,309 \end{bmatrix}$
b_ω	$\begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix}$

$$\begin{aligned}
 \omega_3 &= \sigma (W_\omega^T x_3 + W_{h\omega}^T h_0 + b_\omega) \\
 &= \sigma \left(\begin{bmatrix} 0,411 & -0,401 \\ 0,011 & 0,711 \end{bmatrix} \begin{bmatrix} 0,594 \\ 0,365 \end{bmatrix} + \begin{bmatrix} -0,011 & 0,276 \\ 0,172 & -0,309 \end{bmatrix} \begin{bmatrix} 0,664 \\ 0,574 \end{bmatrix} + \begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} 0,097 \\ 0,266 \end{bmatrix} + \begin{bmatrix} 0,151 \\ -0,063 \end{bmatrix} + \begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} 0,71 \\ 0,625 \end{bmatrix} \right) \\
 &= \begin{bmatrix} 0,671 \\ 0,651 \end{bmatrix}
 \end{aligned}$$

Menghitung *hidden state* menggunakan Persamaan (2.15)

Variabel	Nilai
c_3	$\begin{bmatrix} 0,558 \\ 0,487 \end{bmatrix}$
ω_3	$\begin{bmatrix} 0,671 \\ 0,651 \end{bmatrix}$

$$\begin{aligned}
 h_3 &= \tanh(c_3) \odot \omega_3 \\
 &= \tanh\left(\begin{bmatrix} 0,558 \\ 0,487 \end{bmatrix}\right) \odot \begin{bmatrix} 0,671 \\ 0,651 \end{bmatrix} \\
 &= \begin{bmatrix} 1,012 \\ 0,903 \end{bmatrix} \odot \begin{bmatrix} 0,671 \\ 0,651 \end{bmatrix} \\
 &= \begin{bmatrix} 0,679 \\ 0,587 \end{bmatrix}
 \end{aligned}$$

Menghitung hasil prediksi akhir menggunakan Persamaan (2.21)

Variabel	Nilai
h_3	$\begin{bmatrix} 0,679 \\ 0,587 \end{bmatrix}$
W_y	$\begin{bmatrix} 1,585 & -0,547 \\ 0,433 & 0,169 \end{bmatrix}$
b_y	$\begin{bmatrix} -0,792 \\ 0,531 \end{bmatrix}$

$$\begin{aligned}
 y_1 &= \sigma(W_y h_3 + b_y) \\
 &= \sigma\left(\begin{bmatrix} 1,585 & -0,547 \\ 0,433 & 0,169 \end{bmatrix} \begin{bmatrix} 0,679 \\ 0,587 \end{bmatrix} + \begin{bmatrix} -0,792 \\ 0,531 \end{bmatrix}\right) \\
 &= \sigma\left(\begin{bmatrix} 0,755 \\ 0,393 \end{bmatrix} + \begin{bmatrix} -0,792 \\ 0,531 \end{bmatrix}\right) \\
 &= \sigma\left(\begin{bmatrix} -0,037 \\ 0,924 \end{bmatrix}\right) \\
 &= \begin{bmatrix} 0,491 \\ 0,715 \end{bmatrix}
 \end{aligned}$$

Nilai y_1 merupakan hasil prediksi pada *layer* ke-0. Tahap selanjutnya dengan cara yang sama akan dilakukan perhitungan pada *layer* ke-1 dengan

menggunakan data input yang baru, namun dengan parameter serta nilai bobot dan bias yang tetap sama.

- **Layer Ke-1**

- $X_2 = \begin{bmatrix} 0,579 & 0,456 \\ 0,594 & 0,365 \\ 0,594 & 0,365 \end{bmatrix}$

Perhitungan pada *time step* ke-1 ($t = 1$)

Variabel	Nilai
x_1	$\begin{bmatrix} 0,579 \\ 0,456 \end{bmatrix}$
h_0	$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$
c_0	$\begin{bmatrix} 0 \\ 0 \end{bmatrix}$

Menghitung *forget gate* menggunakan Persamaan (2.2)

Variabel	Nilai
W_ϕ	$\begin{bmatrix} 0,287 & 0,804 \\ 0,236 & -0,111 \end{bmatrix}$
$W_{h\phi}$	$\begin{bmatrix} 0,062 & -0,615 \\ -0,651 & -0,208 \end{bmatrix}$
b_ϕ	$\begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix}$

$$\begin{aligned}
 \phi_1 &= \sigma (W_\phi^T x_1 + W_{h\phi}^T h_0 + b_\phi) \\
 &= \sigma \left(\begin{bmatrix} 0,287 & 0,236 \\ 0,804 & -0,111 \end{bmatrix} \begin{bmatrix} 0,579 \\ 0,456 \end{bmatrix} + \begin{bmatrix} 0,062 & -0,651 \\ -0,615 & -0,208 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} 0,273 \\ 0,414 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} 0,751 \\ 0,365 \end{bmatrix} \right) \\
 &= \begin{bmatrix} 0,679 \\ 0,591 \end{bmatrix}
 \end{aligned}$$

Menghitung *input gate* menggunakan Persamaan (2.6) dan (2.7)

Variabel	Nilai
W_k	$\begin{bmatrix} -0,298 & 0,716 \\ -0,261 & 1,038 \end{bmatrix}$
W_u	$\begin{bmatrix} 0,281 & -0,841 \\ -0,506 & 0,083 \end{bmatrix}$
W_{hk}	$\begin{bmatrix} -0,116 & -1,073 \\ -0,132 & 0,198 \end{bmatrix}$
W_{hu}	$\begin{bmatrix} -0,219 & 0,301 \\ -0,433 & -0,265 \end{bmatrix}$
b_k	$\begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix}$
b_u	$\begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix}$

$$\begin{aligned}
 k_1 &= \sigma (W_k^T x_1 + W_{hk}^T h_0 + b_g) \\
 &= \sigma \left(\begin{bmatrix} -0,298 & -0,261 \\ 0,716 & 1,038 \end{bmatrix} \begin{bmatrix} 0,579 \\ 0,456 \end{bmatrix} + \begin{bmatrix} -0,116 & -0,132 \\ -1,073 & 0,198 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} -0,291 \\ 0,887 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} 0,792 \\ 1,514 \end{bmatrix} \right) \\
 &= \begin{bmatrix} 0,688 \\ 0,819 \end{bmatrix}
 \end{aligned}$$

$$\begin{aligned}
 u_1 &= \sigma (W_u^T x_1 + W_{hu}^T h_0 + b_u) \\
 &= \sigma \left(\begin{bmatrix} 0,281 & -0,506 \\ -0,841 & 0,083 \end{bmatrix} \begin{bmatrix} 0,579 \\ 0,456 \end{bmatrix} + \begin{bmatrix} -0,219 & -0,433 \\ 0,301 & -0,265 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} -0,068 \\ -0,449 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} -0,056 \\ -0,45 \end{bmatrix} \right) \\
 &= \begin{bmatrix} 0,486 \\ 0,389 \end{bmatrix}
 \end{aligned}$$

Menghitung *cell state* menggunakan Persamaan (2.14)

Variabel	Nilai
k_1	$\begin{bmatrix} 0,688 \\ 0,819 \end{bmatrix}$
u_1	$\begin{bmatrix} 0,486 \\ 0,389 \end{bmatrix}$
ϕ_1	$\begin{bmatrix} 0,679 \\ 0,591 \end{bmatrix}$

$$\begin{aligned}
 c_1 &= k_1 \odot u_1 + \phi_1 \odot c_0 \\
 &= \begin{bmatrix} 0,688 \\ 0,819 \end{bmatrix} \odot \begin{bmatrix} 0,486 \\ 0,389 \end{bmatrix} + \begin{bmatrix} 0,679 \\ 0,591 \end{bmatrix} \odot \begin{bmatrix} 0 \\ 0 \end{bmatrix} \\
 &= \begin{bmatrix} 0,334 \\ 0,318 \end{bmatrix}
 \end{aligned}$$

Menghitung *output gate* menggunakan Persamaan (2.17)

Variabel	Nilai
W_ω	$\begin{bmatrix} 0,411 & 0,011 \\ -0,401 & 0,711 \end{bmatrix}$
$W_{h\omega}$	$\begin{bmatrix} -0,011 & 0,172 \\ 0,276 & -0,309 \end{bmatrix}$
b_ω	$\begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix}$

$$\begin{aligned}
 \omega_1 &= \sigma (W_\omega^T x_1 + W_{h\omega}^T h_0 + b_\omega) \\
 &= \sigma \left(\begin{bmatrix} 0,411 & -0,401 \\ 0,011 & 0,711 \end{bmatrix} \begin{bmatrix} 0,579 \\ 0,456 \end{bmatrix} + \begin{bmatrix} -0,011 & 0,276 \\ 0,172 & -0,309 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} 0,055 \\ 0,331 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix} \right) \\
 &= \sigma \left(\begin{bmatrix} 0,517 \\ 0,753 \end{bmatrix} \right) \\
 &= \begin{bmatrix} 0,626 \\ 0,679 \end{bmatrix}
 \end{aligned}$$

Menghitung *hidden state* menggunakan Persamaan (2.15)

Variabel	Nilai
c_1	$\begin{bmatrix} 0,334 \\ 0,318 \end{bmatrix}$
ω_1	$\begin{bmatrix} 0,626 \\ 0,679 \end{bmatrix}$

$$\begin{aligned}
 h_1 &= \tanh(c_1) \odot \omega_1 \\
 &= \tanh\left(\begin{bmatrix} 0,334 \\ 0,318 \end{bmatrix}\right) \odot \begin{bmatrix} 0,626 \\ 0,679 \end{bmatrix} \\
 &= \begin{bmatrix} 0,644 \\ 0,615 \end{bmatrix} \odot \begin{bmatrix} 0,626 \\ 0,679 \end{bmatrix} \\
 &= \begin{bmatrix} 0,403 \\ 0,417 \end{bmatrix}
 \end{aligned}$$

Perhitungan pada *time step* ke-2 ($t = 2$)

Variabel	Nilai
x_2	$\begin{bmatrix} 0,594 \\ 0,365 \end{bmatrix}$
h_1	$\begin{bmatrix} 0,403 \\ 0,417 \end{bmatrix}$
c_1	$\begin{bmatrix} 0,334 \\ 0,318 \end{bmatrix}$

Menghitung *forget gate* menggunakan Persamaan (2.2)

Variabel	Nilai
W_ϕ	$\begin{bmatrix} 0,287 & 0,804 \\ 0,236 & -0,111 \end{bmatrix}$
$W_{h\phi}$	$\begin{bmatrix} 0,062 & -0,615 \\ -0,651 & -0,208 \end{bmatrix}$
b_ϕ	$\begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix}$

$$\begin{aligned}
 \phi_2 &= \sigma (W_\phi^T x_2 + W_{h\phi}^T h_0 + b_\phi) \\
 &= \sigma \left(\begin{bmatrix} 0,287 & 0,236 \\ 0,804 & -0,111 \end{bmatrix} \begin{bmatrix} 0,594 \\ 0,365 \end{bmatrix} + \right. \\
 &\quad \left. \begin{bmatrix} 0,062 & -0,651 \\ -0,615 & -0,208 \end{bmatrix} \begin{bmatrix} 0,403 \\ 0,417 \end{bmatrix} + \begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix} \right)
 \end{aligned}$$

$$\begin{aligned}
&= \sigma \left(\begin{bmatrix} 0,256 \\ 0,437 \end{bmatrix} + \begin{bmatrix} -0,246 \\ -0,334 \end{bmatrix} + \begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} 0,488 \\ 0,054 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0,619 \\ 0,513 \end{bmatrix}
\end{aligned}$$

Menghitung *input gate* menggunakan Persamaan (2.6) dan (2.7)

Variabel	Nilai
W_k	$\begin{bmatrix} -0,298 & 0,716 \\ -0,261 & 1,038 \end{bmatrix}$
W_u	$\begin{bmatrix} 0,281 & -0,841 \\ -0,506 & 0,083 \end{bmatrix}$
W_{hk}	$\begin{bmatrix} -0,116 & -1,073 \\ -0,132 & 0,198 \end{bmatrix}$
W_{hu}	$\begin{bmatrix} -0,219 & 0,301 \\ -0,433 & -0,265 \end{bmatrix}$
b_k	$\begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix}$
b_u	$\begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix}$

$$\begin{aligned}
k_2 &= \sigma (W_k^T x_2 + W_{hk}^T h_0 + b_k) \\
&= \sigma \left(\begin{bmatrix} -0,298 & -0,261 \\ 0,716 & 1,038 \end{bmatrix} \begin{bmatrix} 0,594 \\ 0,365 \end{bmatrix} \right. \\
&\quad \left. + \begin{bmatrix} -0,116 & -0,132 \\ -1,073 & 0,198 \end{bmatrix} \begin{bmatrix} 0,403 \\ 0,417 \end{bmatrix} + \begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} -0,272 \\ 0,804 \end{bmatrix} + \begin{bmatrix} -0,101 \\ -0,349 \end{bmatrix} + \begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} 0,71 \\ 1,082 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0,671 \\ 0,746 \end{bmatrix} \\
u_2 &= \sigma (W_u^T x_2 + W_{hu}^T h_1 + b_u) \\
&= \sigma \left(\begin{bmatrix} 0,281 & -0,506 \\ -0,841 & 0,083 \end{bmatrix} \begin{bmatrix} 0,594 \\ 0,365 \end{bmatrix} + \right. \\
&\quad \left. \begin{bmatrix} -0,219 & -0,433 \\ 0,301 & -0,265 \end{bmatrix} \begin{bmatrix} 0,403 \\ 0,417 \end{bmatrix} + \begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} -0,017 \\ -0,469 \end{bmatrix} + \begin{bmatrix} -0,268 \\ 0,011 \end{bmatrix} + \begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix} \right)
\end{aligned}$$

$$\begin{aligned}
&= \sigma \left(\begin{bmatrix} -0,273 \\ -0,459 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0,432 \\ 0,387 \end{bmatrix}
\end{aligned}$$

Menghitung *cell state* menggunakan Persamaan (2.14)

Variabel	Nilai
k_2	$\begin{bmatrix} 0,671 \\ 0,746 \end{bmatrix}$
u_2	$\begin{bmatrix} 0,432 \\ 0,387 \end{bmatrix}$
ϕ_2	$\begin{bmatrix} 0,619 \\ 0,513 \end{bmatrix}$
c_1	$\begin{bmatrix} 0,334 \\ 0,318 \end{bmatrix}$

$$\begin{aligned}
c_2 &= k_2 \odot u_2 + \phi_2 \odot c_1 \\
&= \begin{bmatrix} 0,671 \\ 0,746 \end{bmatrix} \odot \begin{bmatrix} 0,432 \\ 0,387 \end{bmatrix} + \begin{bmatrix} 0,619 \\ 0,513 \end{bmatrix} \odot \begin{bmatrix} 0,334 \\ 0,318 \end{bmatrix} \\
&= \begin{bmatrix} 0,289 \\ 0,288 \end{bmatrix} + \begin{bmatrix} 0,206 \\ 0,163 \end{bmatrix} \\
&= \begin{bmatrix} 0,495 \\ 0,451 \end{bmatrix}
\end{aligned}$$

Menghitung *output gate* menggunakan Persamaan (2.17)

Variabel	Nilai
W_ω	$\begin{bmatrix} 0,411 & 0,011 \\ -0,401 & 0,711 \end{bmatrix}$
$W_{h\omega}$	$\begin{bmatrix} -0,011 & 0,172 \\ 0,276 & -0,309 \end{bmatrix}$
b_ω	$\begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix}$

$$\begin{aligned}
\omega_2 &= \sigma (W_\omega^T x_2 + W_{h\omega}^T h_1 + b_\omega) \\
&= \sigma \left(\begin{bmatrix} 0,411 & -0,401 \\ 0,011 & 0,711 \end{bmatrix} \begin{bmatrix} 0,594 \\ 0,365 \end{bmatrix} + \right. \\
&\quad \left. \begin{bmatrix} -0,011 & 0,276 \\ 0,172 & -0,309 \end{bmatrix} \begin{bmatrix} 0,403 \\ 0,417 \end{bmatrix} + \begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix} \right)
\end{aligned}$$

$$\begin{aligned}
&= \sigma \left(\begin{bmatrix} 0,097 \\ 0,266 \end{bmatrix} + \begin{bmatrix} 0,111 \\ -0,059 \end{bmatrix} + \begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} 0,67 \\ 0,629 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0,661 \\ 0,652 \end{bmatrix}
\end{aligned}$$

Menghitung *hidden state* menggunakan Persamaan (2.15)

Variabel	Nilai
c_2	$\begin{bmatrix} 0,495 \\ 0,451 \end{bmatrix}$
ω_2	$\begin{bmatrix} 0,661 \\ 0,652 \end{bmatrix}$

$$\begin{aligned}
h_2 &= \tanh(c_2) \odot \omega_2 \\
&= \tanh\left(\begin{bmatrix} 0,495 \\ 0,451 \end{bmatrix}\right) \odot \begin{bmatrix} 0,661 \\ 0,652 \end{bmatrix} \\
&= \begin{bmatrix} 1,443 \\ 1,393 \end{bmatrix} \odot \begin{bmatrix} 0,661 \\ 0,652 \end{bmatrix} \\
&= \begin{bmatrix} 0,953 \\ 0,908 \end{bmatrix}
\end{aligned}$$

Perhitungan pada *time step* ke-3 ($t = 3$)

Variabel	Nilai
x_3	$\begin{bmatrix} 0,594 \\ 0,365 \end{bmatrix}$
h_2	$\begin{bmatrix} 0,953 \\ 0,908 \end{bmatrix}$
c_2	$\begin{bmatrix} 0,495 \\ 0,451 \end{bmatrix}$

Menghitung *forget gate* menggunakan Persamaan (2.2)

Variabel	Nilai
W_ϕ	$\begin{bmatrix} 0,287 & 0,804 \\ 0,236 & -0,111 \end{bmatrix}$
$W_{h\phi}$	$\begin{bmatrix} 0,062 & -0,615 \\ -0,651 & -0,208 \end{bmatrix}$
b_ϕ	$\begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix}$

$$\begin{aligned}
\phi_3 &= \sigma (W_{\phi}^T x_3 + W_{h\phi}^T h_0 + b_{\phi}) \\
&= \sigma \left(\begin{bmatrix} 0,287 & 0,236 \\ 0,804 & -0,111 \end{bmatrix} \begin{bmatrix} 0,594 \\ 0,365 \end{bmatrix} + \right. \\
&\quad \left. \begin{bmatrix} 0,062 & -0,651 \\ -0,615 & -0,208 \end{bmatrix} \begin{bmatrix} 0,953 \\ 0,908 \end{bmatrix} + \begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} 0,256 \\ 0,437 \end{bmatrix} + \begin{bmatrix} -0,532 \\ -0,774 \end{bmatrix} + \begin{bmatrix} 0,478 \\ -0,049 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} 0,202 \\ -0,386 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0,551 \\ 0,404 \end{bmatrix}
\end{aligned}$$

Menghitung *input gate* menggunakan Persamaan (2.6) dan (2.7)

Variabel	Nilai
W_k	$\begin{bmatrix} -0,298 & 0,716 \\ -0,261 & 1,038 \end{bmatrix}$
W_u	$\begin{bmatrix} 0,281 & -0,841 \\ -0,506 & 0,083 \end{bmatrix}$
W_{hk}	$\begin{bmatrix} -0,116 & -1,073 \\ -0,132 & 0,198 \end{bmatrix}$
W_{hu}	$\begin{bmatrix} -0,219 & 0,301 \\ -0,433 & -0,265 \end{bmatrix}$
b_k	$\begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix}$
b_u	$\begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix}$

$$\begin{aligned}
k_3 &= \sigma (W_k^T x_3 + W_{hk}^T h_0 + b_k) \\
&= \sigma \left(\begin{bmatrix} -0,298 & -0,261 \\ 0,716 & 1,038 \end{bmatrix} \begin{bmatrix} 0,594 \\ 0,365 \end{bmatrix} \right. \\
&\quad \left. + \begin{bmatrix} -0,116 & -0,132 \\ -1,073 & 0,198 \end{bmatrix} \begin{bmatrix} 0,953 \\ 0,908 \end{bmatrix} + \begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} -0,272 \\ 0,804 \end{bmatrix} + \begin{bmatrix} -0,231 \\ -0,842 \end{bmatrix} + \begin{bmatrix} 1,083 \\ 0,627 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} 0,58 \\ 0,589 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0,641 \\ 0,643 \end{bmatrix}
\end{aligned}$$

$$\begin{aligned}
u_3 &= \sigma (W_u^T x_3 + W_{hu}^T h_2 + b_u) \\
&= \sigma \left(\begin{bmatrix} 0,281 & -0,506 \\ -0,841 & 0,083 \end{bmatrix} \begin{bmatrix} 0,594 \\ 0,365 \end{bmatrix} + \right. \\
&\quad \left. \begin{bmatrix} -0,219 & -0,433 \\ 0,301 & -0,265 \end{bmatrix} \begin{bmatrix} 0,953 \\ 0,908 \end{bmatrix} + \begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} -0,017 \\ -0,469 \end{bmatrix} + \begin{bmatrix} -0,601 \\ 0,046 \end{bmatrix} + \begin{bmatrix} 0,012 \\ -0,001 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} -0,606 \\ -0,424 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0,352 \\ 0,395 \end{bmatrix}
\end{aligned}$$

Menghitung *cell state* menggunakan Persamaan (2.14)

Variabel	Nilai
k_3	$\begin{bmatrix} 0,641 \\ 0,643 \end{bmatrix}$
u_3	$\begin{bmatrix} 0,352 \\ 0,395 \end{bmatrix}$
ϕ_3	$\begin{bmatrix} 0,551 \\ 0,404 \end{bmatrix}$
c_2	$\begin{bmatrix} 0,495 \\ 0,451 \end{bmatrix}$

$$\begin{aligned}
c_3 &= k_3 \odot u_3 + \phi_3 \odot c_2 \\
&= \begin{bmatrix} 0,641 \\ 0,643 \end{bmatrix} \odot \begin{bmatrix} 0,352 \\ 0,395 \end{bmatrix} + \begin{bmatrix} 0,551 \\ 0,404 \end{bmatrix} \odot \begin{bmatrix} 0,495 \\ 0,451 \end{bmatrix} \\
&= \begin{bmatrix} 0,225 \\ 0,253 \end{bmatrix} + \begin{bmatrix} 0,272 \\ 0,182 \end{bmatrix} \\
&= \begin{bmatrix} 0,497 \\ 0,435 \end{bmatrix}
\end{aligned}$$

Menghitung *output gate* menggunakan Persamaan (2.17)

Variabel	Nilai
W_ω	$\begin{bmatrix} 0,411 & 0,011 \\ -0,401 & 0,711 \end{bmatrix}$
$W_{h\omega}$	$\begin{bmatrix} -0,011 & 0,172 \\ 0,276 & -0,309 \end{bmatrix}$
b_ω	$\begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix}$

$$\begin{aligned}
\omega_3 &= \sigma (W_{\omega}^T x_3 + W_{h\omega}^T h_2 + b_{\omega}) \\
&= \sigma \left(\begin{bmatrix} 0,411 & -0,401 \\ 0,011 & 0,711 \end{bmatrix} \begin{bmatrix} 0,594 \\ 0,365 \end{bmatrix} + \begin{bmatrix} -0,011 & 0,276 \\ 0,172 & -0,309 \end{bmatrix} \begin{bmatrix} 0,953 \\ 0,908 \end{bmatrix} + \begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} 0,097 \\ 0,266 \end{bmatrix} + \begin{bmatrix} 0,241 \\ -0,116 \end{bmatrix} + \begin{bmatrix} 0,462 \\ 0,422 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} 0,8 \\ 0,572 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0,689 \\ 0,639 \end{bmatrix}
\end{aligned}$$

Menghitung *hidden state* menggunakan Persamaan (2.15)

Variabel	Nilai
c_3	$\begin{bmatrix} 0,497 \\ 0,435 \end{bmatrix}$
ω_3	$\begin{bmatrix} 0,689 \\ 0,639 \end{bmatrix}$

$$\begin{aligned}
h_3 &= \tanh(c_3) \odot \omega_3 \\
&= \tanh \left(\begin{bmatrix} 0,497 \\ 0,435 \end{bmatrix} \right) \odot \begin{bmatrix} 0,689 \\ 0,639 \end{bmatrix} \\
&= \begin{bmatrix} 0,919 \\ 0,818 \end{bmatrix} \odot \begin{bmatrix} 0,689 \\ 0,639 \end{bmatrix} \\
&= \begin{bmatrix} 0,633 \\ 0,522 \end{bmatrix}
\end{aligned}$$

Menghitung hasil prediksi akhir menggunakan Persamaan (2.21)

Variabel	Nilai
h_3	$\begin{bmatrix} 0,633 \\ 0,522 \end{bmatrix}$
W_y	$\begin{bmatrix} 1,585 & -0,547 \\ 0,433 & 0,169 \end{bmatrix}$
b_y	$\begin{bmatrix} -0,792 \\ 0,531 \end{bmatrix}$

$$\begin{aligned}
y_1 &= \sigma (W_y h_3 + b_y) \\
&= \sigma \left(\begin{bmatrix} 1,585 & -0,547 \\ 0,433 & 0,169 \end{bmatrix} \begin{bmatrix} 0,633 \\ 0,522 \end{bmatrix} + \begin{bmatrix} -0,792 \\ 0,531 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} 0,717 \\ 0,362 \end{bmatrix} + \begin{bmatrix} -0,792 \\ 0,531 \end{bmatrix} \right) \\
&= \sigma \left(\begin{bmatrix} -0,075 \\ 0,893 \end{bmatrix} \right) \\
&= \begin{bmatrix} 0,481 \\ 0,709 \end{bmatrix}
\end{aligned}$$

Menghitung nilai *error* menggunakan RMSE dan MAPE pada salah satu fitur

Variabel	Nilai
Y	$\begin{bmatrix} 0,594 \\ 0,594 \end{bmatrix}$
$\widehat{y}$	$\begin{bmatrix} 0,491 \\ 0,481 \end{bmatrix}$
T	2
n	2

$$\begin{aligned}
\text{RMSE} &= \sqrt{T^{-1} \sum_{t=1}^T (Y_t - \widehat{y}_{t|t-1})^2} \\
&= \sqrt{\frac{1}{2} ((0,594 - 0,491)^2 + (0,594 - 0,481)^2)} \\
&= \sqrt{\frac{1}{2} ((0,103)^2 + (0,113)^2)} \\
&= \sqrt{\frac{1}{2} (0,011 + 0,012)} \\
&= \sqrt{\frac{1}{2} (0,023)} \\
&= \sqrt{0,0115} \\
&= 0,107
\end{aligned}$$

$$\begin{aligned}
\text{MAPE} &= 100 T^{-1} \sum_{i=1}^T \frac{|Y_t - \widehat{y}_{t|t-1}|}{|Y_t|} \\
&= 100 \times \frac{1}{2} \left(\frac{|0,594 - 0,491|}{|0,594|} + \frac{|0,594 - 0,481|}{|0,594|} \right) \\
&= 100 \times \frac{1}{2} \left(\frac{0,103}{0,594} + \frac{0,113}{0,594} \right) \\
&= 100 \times \frac{1}{2} (0,173 + 0,191)
\end{aligned}$$

$$\begin{aligned}
&= 100 \times \frac{1}{2} (0,364) \\
&= 100 \times 0,182 \\
&= 18,2\%
\end{aligned}$$

Langkah-langkah perhitungan yang telah diberikan sebelumnya diterapkan pada setiap langkah waktu (*time step*). Setiap kali model menerima data masukan baru, nilai *cell state* (c_t) dan *hidden state* (h_t) diperbarui sesuai dengan mekanisme *forget gate*, *input gate*, dan *output gate*. Nilai *cell state* (c_t) berfungsi sebagai memori jangka panjang yang menyimpan informasi penting dari data sebelumnya, sedangkan *hidden state* (h_t) digunakan sebagai *output* yang akan diteruskan ke langkah waktu berikutnya. Hasil dari setiap perhitungan akan melalui fungsi aktivasi untuk menyesuaikan nilai keluaran ke dalam rentang tertentu. Fungsi aktivasi *tanh* digunakan untuk menyesuaikan nilai pada rentang -1 hingga 1, sedangkan sigmoid pada rentang 0-1.

Bobot dan bias model selama proses pelatihan akan diperbarui dengan bantuan *optimizer* setelah seluruh dataset telah diiterasi sesuai dengan jumlah epoch yang telah ditentukan. *Optimizer* berperan dalam menyesuaikan bobot model agar *error* pada hasil prediksi dapat diminimalkan. Proses ini akan terus berlanjut hingga diperoleh model terbaik yang sesuai dengan parameter yang telah ditentukan. Setelah proses pelatihan selesai, model yang telah dilatih akan digunakan dalam proses pengujian untuk memperoleh nilai prediksi atau nilai target (y). Hasil prediksi selanjutnya dievaluasi dengan menghitung tingkat menggunakan metode RMSE dan MAPE.

Hasil *forget gate*, *input gate*, *output gate*, dan *cell state* dengan menggunakan data sebenarnya yang digunakan pada penelitian terdapat pada Tabel 4.8.

Tabel 4.8 Nilai *Forget Gate*, *Input Gate*, *Output Gate*, dan *Cell State*

	<i>Forget Gate</i>	<i>Input Gate</i>	<i>Output Gate</i>	<i>Cell State</i>
<i>Timestep 0</i>	$\begin{bmatrix} 0,73 \\ 0,72 \\ 0,79 \\ \vdots \\ 0,71 \\ 0,72 \end{bmatrix}$	$\begin{bmatrix} 0,51 \\ 0,52 \\ 0,72 \\ \vdots \\ 0,44 \\ 0,51 \end{bmatrix}$	$\begin{bmatrix} 0,53 \\ 0,59 \\ 0,45 \\ \vdots \\ 0,39 \\ 0,57 \end{bmatrix}$	$\begin{bmatrix} -0,07 \\ 0,19 \\ -0,37 \\ \vdots \\ -0,11 \\ -0,16 \end{bmatrix}$
<i>Timestep 1</i>	$\begin{bmatrix} 0,74 \\ 0,69 \\ 0,81 \\ \vdots \\ 0,64 \\ 0,67 \end{bmatrix}$	$\begin{bmatrix} 0,46 \\ 0,42 \\ 0,81 \\ \vdots \\ 0,31 \\ 0,41 \end{bmatrix}$	$\begin{bmatrix} 0,51 \\ 0,62 \\ 0,33 \\ \vdots \\ 0,29 \\ 0,52 \end{bmatrix}$	$\begin{bmatrix} -0,16 \\ 0,17 \\ -0,91 \\ \vdots \\ -0,11 \\ -0,18 \end{bmatrix}$
<i>Timestep 2</i>	$\begin{bmatrix} 0,72 \\ 0,66 \\ 0,81 \\ \vdots \\ 0,58 \\ 0,64 \end{bmatrix}$	$\begin{bmatrix} 0,41 \\ 0,39 \\ 0,81 \\ \vdots \\ 0,24 \\ 0,35 \end{bmatrix}$	$\begin{bmatrix} 0,47 \\ 0,59 \\ 0,28 \\ \vdots \\ 0,25 \\ 0,51 \end{bmatrix}$	$\begin{bmatrix} -0,23 \\ 0,14 \\ -1,41 \\ \vdots \\ -0,07 \\ -0,17 \end{bmatrix}$
.....				
<i>Timestep 28</i>	$\begin{bmatrix} 0,64 \\ 0,65 \\ 0,85 \\ \vdots \\ 0,41 \\ 0,62 \end{bmatrix}$	$\begin{bmatrix} 0,39 \\ 0,33 \\ 0,88 \\ \vdots \\ 0,11 \\ 0,26 \end{bmatrix}$	$\begin{bmatrix} 0,41 \\ 0,41 \\ 0,26 \\ \vdots \\ 0,12 \\ 0,43 \end{bmatrix}$	$\begin{bmatrix} -1,09 \\ 2,41 \\ -4,18 \\ \vdots \\ 2,33 \\ -8,31 \end{bmatrix}$
<i>Timestep 29</i>	$\begin{bmatrix} 0,63 \\ 0,65 \\ 0,85 \\ \vdots \\ 0,41 \\ 0,61 \end{bmatrix}$	$\begin{bmatrix} 0,38 \\ 0,33 \\ 0,88 \\ \vdots \\ 0,11 \\ 0,26 \end{bmatrix}$	$\begin{bmatrix} 0,39 \\ 0,41 \\ 0,27 \\ \vdots \\ 0,12 \\ 0,43 \end{bmatrix}$	$\begin{bmatrix} -0,11 \\ 0,24 \\ -4,35 \\ \vdots \\ 0,01 \\ -0,01 \end{bmatrix}$
<i>Timestep 30</i>	$\begin{bmatrix} 0,74 \\ 0,52 \\ 0,73 \\ \vdots \\ 0,69 \\ 0,72 \end{bmatrix}$	$\begin{bmatrix} 0,51 \\ 0,52 \\ 0,73 \\ \vdots \\ 0,44 \\ 0,51 \end{bmatrix}$	$\begin{bmatrix} 0,53 \\ 0,61 \\ 0,45 \\ \vdots \\ 0,38 \\ 0,58 \end{bmatrix}$	$\begin{bmatrix} -0,08 \\ 0,21 \\ -0,41 \\ \vdots \\ -0,11 \\ -0,18 \end{bmatrix}$

4.6 Validasi Model

Setelah dilakukan proses pelatihan kombinasi hyperparameter yang divariasikan melalui teknik *grid search*. Setiap percobaan kombinasi *hyperparameter* dilakukan evaluasi menggunakan teknik *time series split* dengan lima *fold*. Selama proses pelatihan, performa model dievaluasi dengan menggunakan dua metrik yaitu *Root Mean Squared Error* (RMSE) dan *Mean Absolute Percentage Error* (MAPE). Evaluasi ini bertujuan untuk menentukan performa model terbaik dalam melakukan prediksi serta mengidentifikasi apakah model yang digunakan sudah optimal atau masih memerlukan perbaikan. Nilai rata-rata dari kedua metrik tersebut dihitung dari kelima *fold* untuk menilai stabilitas dan keakuratan model terhadap setiap kombinasi parameter. Kombinasi yang menghasilkan nilai rata-rata RMSE dan MAPE paling rendah dianggap sebagai kombinasi terbaik dan disimpan menjadi model terbaik. Hasil perhitungan nilai *error* pada penelitian ini dengan menggunakan kedua metode tersebut disajikan dalam Tabel 4.9.

Tabel 4.9 Hasil Nilai *Error* Menggunakan RMSE dan MAPE saat Pelatihan

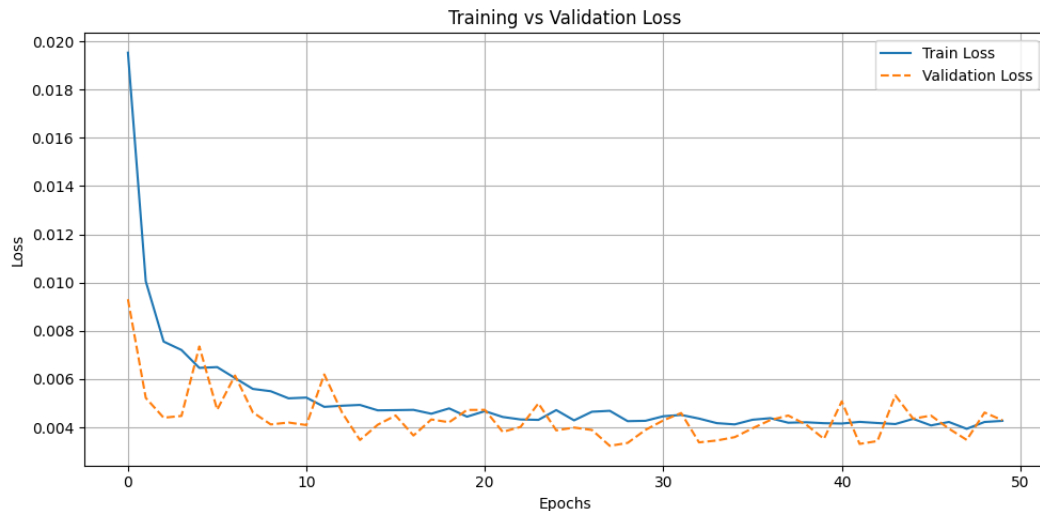
<i>Batch</i>	<i>Epoch</i>	RMSE	MAPE
4	50	0,1119	1,32%
4	100	0,0898	1,05%
16	50	0,1041	1,21%
16	100	0,1020	1,18%
32	50	0,1113	1,29%
32	100	0,1037	1,20%
64	50	0,1244	1,46%
64	100	0,1119	1,28%

Berdasarkan Tabel 4.9, model yang menampilkan performa paling optimal diperoleh dengan penggunaan parameter-parameter berikut:

- a. *Hidden layer* : 4
- b. *Neuron hidden* : 50
- c. *Nilai batch size* : 4
- d. *Epoch maksimum* : 100
- e. *Optimizer* : Adam
- f. *Fungsi aktivasi* : Tanh and Sigmoid

Hasil perhitungan nilai *error* menggunakan kombinasi parameter diatas menunjukkan nilai RMSE sebesar 0,0898 dan MAPE 1,05%. Berdasarkan kriteria yang sudah dijelaskan pada Bab II, nilai MAPE sebesar 1,05% termasuk dalam kategori sangat baik (MAPE <10%) sehingga tingkat kesalahan model dalam melakukan prediksi relatif kecil. Dengan demikian, model diharapkan mampu menghasilkan peramalan dengan tingkat akurasi yang tinggi pada data uji.

Selain itu ditampilkan juga grafik perbandingan antara nilai *loss* serta *epoch* pada data latih dan data validasi untuk mendukung analisis performa model. Grafik perbandingan antara nilai *loss* serta *epoch* pada data latih dan data validasi menggambarkan bagaimana nilai *loss* mengalami perubahan seiring dengan bertambahnya jumlah *epoch* selama proses pelatihan. Pola penurunan nilai *loss* pada grafik menunjukkan bahwa model semakin mampu meminimalkan kesalahan prediksi seiring bertambahnya iterasi pelatihan. Grafik perbandingan antara nilai *loss* serta *epoch* pada data latih dan data validasi ditampilkan pada Gambar 4.2



Gambar 4.2 Grafik Perbandingan *Train* dan *Validation Loss*

Gambar 4.2 menunjukkan grafik perbandingan antara nilai *loss* serta *epoch* pada data latih dan data validasi. Berdasarkan grafik, pola *loss* terlihat menurun secara signifikan hingga sekitar *epoch* ke-30 menunjukkan bahwa model mulai mampu mempelajari pola pada data dengan baik. Setelah *epoch* ke-30, baik *train loss* maupun *validation loss* cenderung stabil dan tidak menunjukkan perubahan yang drastis hingga akhir proses pelatihan. Meski terdapat fluktuasi kecil pada *validation loss*, hal tersebut masih dalam batas yang wajar dan tidak ditemukan indikasi *overfitting* karena nilai *validation loss* tetap berada dalam kisaran yang sebanding dengan *train loss*. Dengan demikian, jumlah *epoch* yang digunakan dapat dikatakan sudah efektif untuk mencapai hasil pelatihan yang optimal.

4.7 Prediksi Harga Saham

Setelah diperoleh model terbaik berdasarkan evaluasi terhadap serangkaian percobaan dengan berbagai nilai parameter, tahap selanjutnya adalah menerapkan model pada data uji. Data uji yang digunakan dalam penelitian ini merupakan data harga saham pada periode 1 November 2024 sampai dengan 31 Januari 2025.

Sebelum diimplementasikan pada model, data uji terlebih dahulu melalui tahap *preprocessing* yang meliputi proses penyortiran data, pengurutan dengan *ascending*, serta ekstraksi fitur untuk memperoleh nilai indikator RSI. Tujuan dari penerapan ini adalah untuk memperoleh hasil prediksi harga saham dan nilai RSI berdasarkan pola yang telah dipelajari oleh model.

Data uji yang sudah melalui tahap *preprocessing* kemudian akan diterapkan pada model terbaik yang sudah dilatih menggunakan data latih untuk menampilkan hasil prediksi pada data uji. Hasil prediksi yang telah diperoleh dari model kemudian akan divisualisasikan dalam bentuk numerik dan grafik. Visualisasi tersebut bertujuan untuk menampilkan perbandingan antara data aktual dan hasil prediksi harga saham selama periode iterasi. Visualisasi yang ditampilkan akan membantu memberikan gambaran mengenai kemampuan model dalam mengikuti pola pergerakan harga saham yang sebenarnya. Perbandingan antara hasil prediksi dan data aktual pada data uji terdapat pada tabel 4.10.

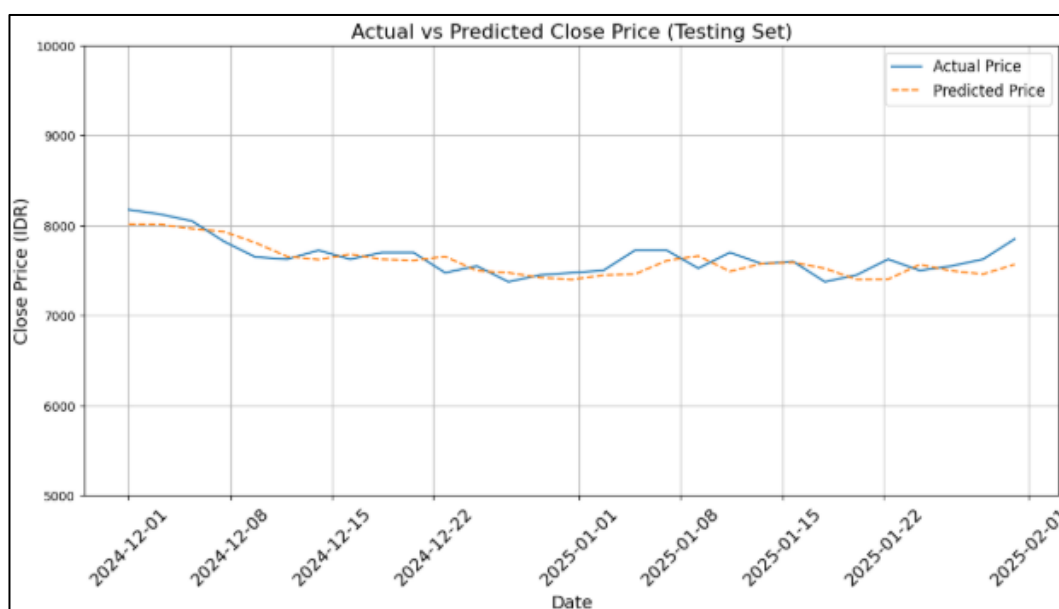
Tabel 4.10 Perbandingan Data Aktual dan Hasil Prediksi *Close Price* dan RSI

Tanggal	Data Validasi		Prediksi (y)	
	Close	RSI	Close	RSI
02/12/2024	7550	40,38	7650	45,12
03/12/2024	7775	51,51	7650	46,12
04/12/2024	7850	54,38	7750	48,01
05/12/2024	7875	54,38	7780	48,01
06/12/2024	7925	55,68	7800	49,32
....				
22/01/2025	7625	54,68	7700	50,12
23/01/2025	7500	48,48	7600	44,51
24/01/2025	7550	55,73	7650	50,81
30/01/2025	7625	55,73	7700	50,81
31/01/2025	7850	60,86	7750	55,70

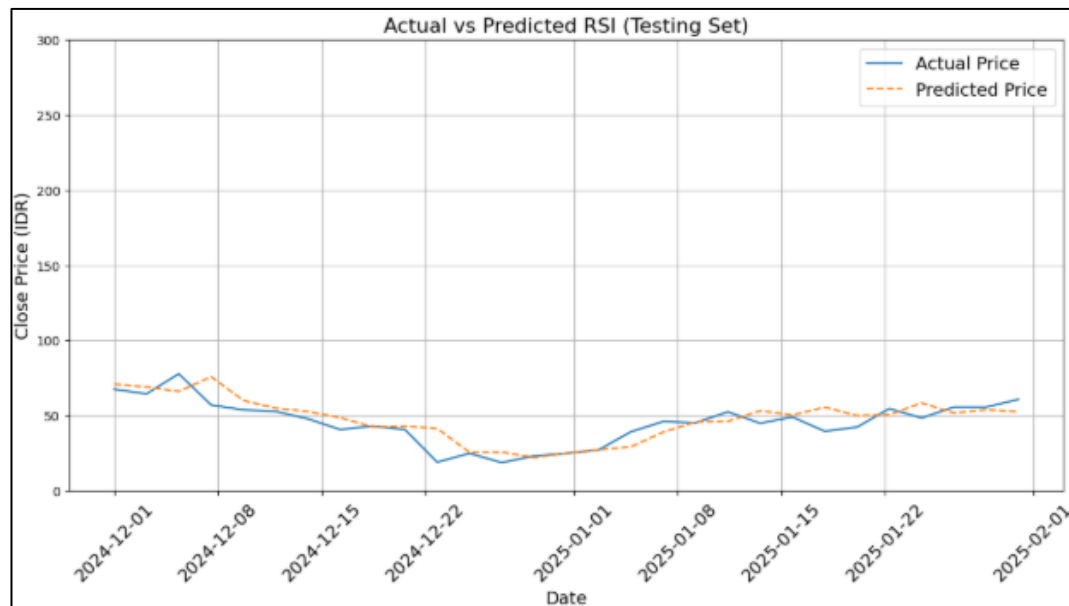
Tabel 4.10 menampilkan perbandingan antara data aktual dan hasil prediksi harga penutupan saham dan nilai indikator RSI pada periode 1 Desember 2024

hingga 31 Januari 2025. Jika berdasarkan tabel, model mampu menghasilkan prediksi yang akurat dengan tingkat kesalahan yang rendah. Hal ini dibuktikan dengan nilai *loss* yang kecil di antara data aktual dan hasil prediksi pada kedua fitur. Untuk memberikan gambaran yang lebih jelas mengenai kemampuan model dalam mengikuti pergerakan harga saham pada seluruh data, hasil prediksi akan ditampilkan dalam bentuk grafik seperti pada Gambar 4.3 dan Gambar 4.4.

Gambar 4.3 menampilkan perbandingan antara data aktual dan hasil prediksi dari harga penutupan saham yang diperoleh dari model LSTM menggunakan data uji. Sedangkan Gambar 4.4 menampilkan perbandingan antara data aktual dan hasil prediksi dari nilai RSI. Kurva berwarna biru merepresentasikan data aktual pada periode pengujian, sedangkan kurva merah menunjukkan hasil prediksi yang dihasilkan oleh model dalam memperkirakan pergerakan harga saham.



Gambar 4.3 Grafik Perbandingan Hasil Prediksi dan Data Aktual *Close Price*



Gambar 4.4 Grafik Perbandingan Hasil Prediksi dan Data Aktual Nilai RSI

Berdasarkan grafik yang ditampilkan pada Gambar 4.3 dan 4.4, model LSTM yang diterapkan untuk memprediksi data menunjukkan pola yang serupa dengan data aktual tetapi belum sepenuhnya dapat mengikuti nilai aktual secara tepat. Garis prediksi secara umum dapat mengikuti tren garis aktual dengan baik meskipun masih terdapat sedikit perbedaan nilai pada beberapa titik. Hal ini menunjukkan bahwa model LSTM yang diterapkan mampu mempelajari pola historis data harga saham dan merepresentasikannya pada data yang belum pernah dilihat sebelumnya. Selain itu, kemampuan model dalam menangkap pergerakan RSI menunjukkan bahwa LSTM tidak hanya efektif dalam memprediksi data harga, tetapi juga dapat digunakan untuk memodelkan indikator teknikal lainnya yang memiliki sifat *time series*. Untuk memperoleh penilaian yang lebih akurat terhadap performa model, selanjutnya akan dilakukan perhitungan nilai *error* pada data uji.

4.8 Evaluasi Nilai *Error*

Hasil prediksi model pada data uji kemudian dievaluasi dengan menggunakan dua metrik utama yaitu RMSE dan MAPE. Nilai RMSE menunjukkan rata-rata kesalahan prediksi dalam satuan harga saham, sedangkan MAPE menunjukkan rata-rata kesalahan dalam bentuk persentase terhadap nilai aktual. Model LSTM pada penelitian ini menghasilkan nilai RMSE sebesar 0,0898 dan MAPE sebesar 1,05%.

Sebagai perbandingan, dilakukan prediksi pada data yang sama dengan menggunakan metode *Moving Average* dan ARIMA. Selanjutnya, hasil prediksi akan dilakukan perhitungan nilai *error* menggunakan metrik RMSE dan MAPE untuk masing-masing metode. Hasil perhitungan nilai *error* dari ketiga metode yaitu *Moving Average*, ARIMA, dan LSTM kemudian dibandingkan untuk mengevaluasi apakah model LSTM telah memenuhi kriteria sebagai model yang baik untuk digunakan pada prediksi harga saham. Hasil perhitungan nilai *error* pada ketiga metode ditampilkan pada Tabel 4.10 dan 4.11.

Tabel 4.11 Nilai *Error Close Price* Menggunakan LSTM, MA, dan ARIMA

Metode Prediksi	RMSE	MAPE
LSTM	0,0898	1,05%
<i>Moving Average</i>	0,1651	1,81%
ARIMA	0,0757	0,73%

Tabel 4.12 Nilai *Error RSI* Menggunakan LSTM, MA, dan ARIMA

Metode Prediksi	RMSE	MAPE
LSTM	0,6452	10,97%
<i>Moving Average</i>	0,3469	6,78%
ARIMA	0,3617	7,13%

Berdasarkan hasil evaluasi terhadap model prediksi dengan dua fitur saham yaitu *close price* dan RSI pada Tabel 4.10 dan 4.11, performa model *Long Short Term Memory* (LSTM) dalam memprediksi harga penutupan (*close price*) menunjukkan nilai RMSE sebesar 0,0898 dan MAPE sebesar 1,05%. Meskipun nilai tersebut menunjukkan tingkat kesalahan yang rendah, performa LSTM masih tidak sebaik model ARIMA yang menghasilkan nilai *error* lebih kecil yaitu 0,0757 dan 0,73%. Akan tetapi, performa LSTM lebih baik dibandingkan metode *Moving Average* dengan nilai *error* yang dihasilkan yaitu RMSE 0,1651 dan MAPE 1,81%. Pada prediksi nilai RSI, model LSTM menghasilkan nilai tertinggi diantara ketiga metode yaitu sebesar RMSE 0,6452 dan MAPE 10,97%. Nilai *error* yang cukup tinggi khususnya pada prediksi RSI, dimungkinkan karena RSI bukan merupakan harga saham aktual, melainkan indikator teknikal yang merupakan hasil turunan dari *close price*. Berdasarkan hasil perbandingan pada Tabel 4.10 dan 4.11 menunjukkan bahwa meskipun LSTM memiliki kemampuan dalam menangkap pola kompleks pada data deret waktu, performanya belum mampu mengguli model ARIMA maupun *Moving Average* dalam memprediksi data saham milik PT Indofood Sukses Makmur Tbk terutama dalam memprediksi indikator RSI.

Selain itu, untuk memberikan gambaran yang lebih menyeluruh terhadap kinerja model yang dikembangkan dalam penelitian, dilakukan pula perbandingan dengan beberapa hasil penelitian sebelumnya yang juga melakukan prediksi terhadap harga saham PT Indofood Sukses Makmur Tbk. Perbandingan ini bertujuan untuk mengevaluasi sejauh mana model yang digunakan dalam penelitian dapat menghasilkan prediksi yang lebih baik dibandingkan dengan model atau

pendekatan lain yang telah diterapkan oleh peneliti terdahulu. Hasil perbandingan ditampilkan pada Tabel 4.13.

Berdasarkan hasil perbandingan dengan beberapa penelitian terdahulu pada Tabel 4.13, model LSTM yang digunakan dalam penelitian ini dengan input fitur *close price* dan RSI pada periode 2 Juni 2021 hingga 31 Januari 2025 menghasilkan nilai RMSE sebesar 0,0898 dan MAPE 1,05%. Nilai MAPE 1,05% menunjukkan bahwa model memiliki tingkat kesalahan prediksi yang lebih rendah dibandingkan model *Hibrida Singular Spectrum Analysis (SSA) – Autoregressive Integrated Moving Average (ARIMA)* milik Dewanti, dkk (2024) serta *Autoregressive Integrated Moving Average (ARIMA)* dan *Grey Theory* milik Sitompul, dkk (2024). Namun, penelitian yang dikembangkan oleh Risnandar, dkk (2025) memberikan model LSTM yang lebih baik ditunjukkan dengan nilai RMSE yang dihasilkan lebih kecil yaitu 0,0316 dibandingkan dengan nilai RMSE pada penelitian ini.

Tabel 4.13 Perbandingan Prediksi Saham PT Indofood dari Penelitian Terdahulu

Metode	Fitur	Periode	RMSE	MAPE
Model <i>Hibrida Singular Spectrum Analysis (SSA) – Autoregressive Integrated Moving Average (ARIMA)</i> (Dewanti dkk., 2024)	<i>Close Price</i>	01 Agustus – 29 Desember 2023	-	1,54%
<i>Autoregressive Integrated Moving Average (ARIMA)</i> (Sitompul dkk, 2024)	<i>Close Price</i>	Januari 2018 – Juni 2023	-	4,61%
<i>Grey Theory</i> (Sitompul dkk, 2024)	<i>Close Price</i>	Januari 2018 – Juni 2023	-	13,2%
<i>Long Short Term Memory (LSTM)</i> (Risnandar dkk., 2025)	<i>Open, High, Low, Close, Volume,</i>	1 Desember 2014 – 6	0,0316	-

	EMA_4, EMA_9, EMA_18, RSI, MACD, Signal line, Close_Lag1, Close_Lag2, Close_Lag3.	Desember 2024		
<i>Long Short Term Memory</i> (LSTM)	<i>Close Price</i> dan RSI	02 Juni 2021 – 31 Januari 2025	0,0898	1,05%

4.9 Penerapan Prinsip Ikhtiar dan Tawakal dalam Penggunaan Metode

LSTM untuk Prediksi Harga Saham

Dalam tafsir Al-Munir (Az-Zuhaili, 2013) mengenai Q.S. Al-Hasyr ayat 18, Allah mengingatkan manusia untuk memperhatikan setiap amal maupun perbuatan yang dilakukan saat ini sebagai persiapan untuk masa depan terutama di hari akhir. Allah menegaskan pentingnya sikap muhasabah, yaitu melakukan intropeksi agar setiap tindakan yang diambil bermanfaat dan tidak sia-sia. Prinsip ini mengajarkan manusia untuk memiliki perencanaan yang matang dalam menghadapi masa depan termasuk dalam urusan duniawi, sehingga tidak hanya berorientasi pada keuntungan jangka pendek, tetapi juga berdampak baik dalam jangka panjang. Dalam konteks penelitian ini perencanaan yang dilakukan yakni dengan memprediksi informasi yang akan terjadi di masa depan dengan memanfaatkan data ataupun informasi historis sebelumnya. Dengan melakukan perencanaan ini menunjukkan bahwa manusia memanfaatkan bekal akal pikiran yang diberikan Allah SWT untuk membuat persiapan yang matang di masa yang akan datang. Dalam konteks penelitian ini, manusia memanfaatkan akal pikiran dengan

mengimplementasikan metode *Long Short Term Memory* untuk memprediksi harga saham sehingga menghasilkan hasil prediksi baik dalam bentuk tabel angka ataupun grafik yang dapat memberikan gambaran prediksi harga selanjutnya. Setelah melakukan persiapan yang matang, manusia diajarkan untuk bertawakal atau menyerahkan hasilnya kepada Allah SWT. Prinsip tawakal yang dijelaskan dalam QS. Ali-Imran ayat 159 menekankan agar manusia menyerahkan hasil dari usaha dan perencanaannya kepada Allah SWT, setelah melalui proses usaha yang sungguh-sungguh. Sebagaimana hasil prediksi yang diperoleh dengan pengimplementasian metode LSTM bukanlah hasil yang akurat dibuktikan dengan masih adanya nilai *error* yang dihasilkan. Oleh karena itu, pentinglah bagi manusia untuk menyerahkan hasilnya kepada Allah SWT meskipun sudah melakukan usaha dan perencanaan yang matang.

Prinsip-prinsip ini sangat relevan dalam konteks teknologi prediksi, seperti penggunaan *Long Short Term Memory* (LSTM) untuk memprediksi pergerakan harga saham di masa depan. Meskipun manusia tidak mengetahui secara pasti apa yang akan terjadi, LSTM memberikan kemampuan untuk meminimalkan ketidakpastian dengan memanfaatkan data masa lalu. Namun sebagai umat beriman, hasil akhir dari prediksi ini tetap dipercayakan kepada Allah SWT. Dengan demikian, prediksi melalui pengimplementasian teknologi tidak hanya bertujuan untuk menghindari kerugian atau kesalahan di masa depan, tetapi juga menjadi salah satu cara manusia untuk melaksanakan ikhtiar sebagaimana yang dianjurkan Allah SWT.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan pembahasan tersebut, dapat disimpulkan beberapa hasil yang didapatkan dalam penelitian ini yaitu:

1. Model LSTM yang digunakan dalam penelitian ini untuk memprediksi harga saham milik PT Indofood Sukses Makmur Tbk dengan dua fitur yaitu *close price* dan RSI, menghasilkan nilai RMSE sebesar 0,0898 dan 0,6452 serta nilai MAPE sebesar 1,05% dan 10,97% untuk periode prediksi 2 Juni 2021 hingga 31 Januari 2025. Berdasarkan hasil evaluasi serta perbandingan dengan metode ARIMA, *Moving Average* dan penelitian sebelumnya pada pembahasan, dapat disimpulkan bahwa model LSTM dalam penelitian ini belum mampu mengungguli metode statistik lainnya, khususnya dalam memprediksi indikator teknikal seperti RSI. Namun, model pada penelitian ini tetap menunjukkan potensi yang baik untuk pengembangan lebih lanjut, khususnya jika dilakukan peningkatan pada pemilihan fitur, eksplorasi parameter model yang lebih optimal, serta pemanfaatan data historis yang lebih panjang. Model terbaik dalam menghasilkan prediksi pada penelitian ini terbentuk pada penggunaan parameter yang meliputi *hidden layer* sebanyak 4 lapisan, *batch size* sebesar 4, *epoch* 100, jumlah neuron pada *hidden layer* sebanyak 50, menggunakan algoritma optimasi *Adam optimizer*, serta menggunakan fungsi aktivasi *tanh* dan *sigmoid*.

2. Hasil prediksi harga saham milik PT. Indofood Sukses Makmur Tbk pada periode 1 Desember 2024 sampai dengan 31 Januari 2025 dengan menggunakan metode LSTM menghasilkan nilai *error* RMSE 0,0898 dan MAPE 1,05%. Nilai *error* tersebut menunjukkan bahwa model LSTM memiliki kemampuan yang baik dalam memprediksi harga saham dengan tingkat kesalahan yang rendah. Garis prediksi LSTM secara umum mampu mengikuti tren harga aktual meskipun masih terdapat sedikit perbedaan nilai pada beberapa titik. Selama periode pengamatan, harga saham berada dalam kondisi pasar yang stabil ditandai dengan fluktuasi harga yang rendah dan pergerakan harga yang relatif mendatar. Periode harga saham yang stabil ini dapat menjadi peluang bagi perusahaan melakukan evaluasi terhadap strategi pertumbuhan perusahaan, seperti pengembangan produk atau perluasan pasar, agar mampu meningkatkan nilai perusahaan dalam jangka panjang.

5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, diperoleh hasil prediksi harga saham dan nilai RSI dengan menggunakan metode LSTM. Terdapat beberapa aspek yang dapat ditingkatkan guna memperoleh hasil prediksi yang lebih akurat dengan metode LSTM. Oleh karena itu, penulis memberikan beberapa rekomendasi untuk pengembangan penelitian selanjutnya yaitu dengan melakukan lebih banyak percobaan dalam penyesuaian parameter model, penggunaan data dengan periode waktu yang lebih panjang, penggunaan fitur yang lebih beragam serta dapat mengeksplorasi kombinasi metode LSTM dengan pendekatan lain seperti

SARIMA, GRU atau metode lainnya agar model lebih responsif terhadap volatilitas pasar saham dan dapat menghasilkan hasil prediksi yang lebih akurat dengan nilai *error* yang lebih kecil.

DAFTAR PUSTAKA

- Aditya, F., Devianto, D., & Maiyastri. (2019). *Peramalan Harga Emas Indonesia Menggunakan Metode Fuzzy Time Series Klasik*. (Vol. 8). Jurnal Matematika. UNAND
- Aldi, M. W. P., Jondri, & Aditsania, A. (2018). Analisis Dan Implementasi Long Short Term Memory Neural Network Untuk Prediksi Harga Bitcoin. *E-Proceeding of Engineering* , 5, 3548–3555.
- Amazon Web Service. (2025). *Amazon SageMaker AI: Developer Guide*.
- Andrijasa, M., Hidayat, H., & Sari, W. (2018). Prediksi Nilai Tukar Mata Uang Idr Terhadap Usd Dengan Teknik Deep Learning Menggunakan Model Recurrent Neural Network. *Prosiding Seminar Nasional Hasil Penelitian 2018*, 3.
- Az-Zuhaili. (2013). *Tafsir Al-Munir*.
- Bintoro, P., Ratnasari, Wihadjo, E., Putri, I. P., & Asari, A. (2023). *Pengantar Machine Learning*. PT Mafy Media Literasi Indonesia, Sumatera Barat
- Chandra, R., Chaudhary, K., & Kumar, A. (2022). *Comparison of Data Normalization for Wine Classification Using K-NN Algorithm*. International Journal of Informatics and Information Systems, University of Fiji, Lautoka
- Cindi. (2024). *Perbandingan Fungsi Aktivasi Terhadap Kinerja Algoritma Neural Network Pada Klasifikasi Data Diabetes*. Skripsi. Universitas Sulawesi Barat
- Departemen Agama, R. (2015). *Al-Quran Terjemahan, Bandung: CV Darus Sunnah*.
- Dewanti, R. T., Zukhronah, E., & Sulandari, W. (2024). *Peramalan Harga Saham Pt Indofood Sukses Makmur Tbk Menggunakan Model Hibrida Singular Spectrum Analysis (Ssa) – Autoregressive Integrated Moving Average (ARIMA)*. Jurnal Gaussian. Universitas Sebelas Maret.
- Dwi Julianti, P., Suprpta, I., & Muhammadiyah Jakarta, S. (2020). *Pengaruh Faktor Fundamental Terhadap Return Saham*. Jurnal Akuntansi. STIE Muhammadiyah Jakarta. <http://ejournal.stiemj.ac.id/index.php/akuntansi>
- Ganesha, D. (2017). *Perbandingan Fungsi Aktivasi Sigmoid Biner Dan Sigmoid Bipolar Pada Algoritma Backpropagation Untuk Klasifikasi Tingkat Preeklampsia*. Skripsi. UIN Sultan Syarif Kasim Riau Pekanbaru.

- Heizer, Jay, & Barry, R. (2015). *Manajemen Operasi, Manajemen Keberlangsungan dan Rantai Pasokan, Edisi 11, Alih Bahasa : Hirson Kurnia, Ratna Saraswati dan David Wijaya*.
- Hyndman, R. J. (2014). *Forecasting: Principles & Practice*. Buku. OTexts
- Karpathy, A., Johnson, J., & Fei-Fei, L. (2016). *Visualizing And Understanding Recurrent Networks*. Journal Computer Science. Stanford University.
- Kartika, S. D. & K. (2022). *Implementasi Long Short term Memory pada Prediksi Harga Saham PT Aneka Tambang Tbk*. Jurnal Ilmiah Komputasi. Universitas Gunadarma.
- Khumaidi, A., & Nirmala, I. A. (2022). *Algoritma Long Short Term Memory Dengan Hyperparameter Tuning: Prediksi Penjualan Produk*.
- Mahajaya, N. S., Ayu, P. D. W., & Huizen, R. R. (2024). *Pengaruh Optimizer Adam, AdamW, SGD, dan LAMB terhadap Model Vision Transformer pada Klasifikasi Penyakit Paru-paru*. Jurnal Informatika. Institut Teknologi dan Bisnis STIKOM Bali.
- Melinda, V., Primartha, R., Wijaya, A., & Jambak, M. I. (2020). *Advances in Intelligent Systems Research* (Vol. 172). Journal International. Sriwijaya International Conference on Information Technology and Its Applications
- Muflih, G. Z. (2021). *Penentuan Parameter Pembelajaran Jaringan Saraf Tiruan Backpropagation dan Pengaruhnya Terhadap Proses Pelatihan*. Jurnal Riset Teknologi Informasi dan Komputer. Universitas Ma'arif NU Kebumen.
- Nurfadilah, A., Budi, W., Kurniati, E., & Suhaedi, D. (2022). *Penerapan Metode Moving Average untuk Prediksi Indeks Harga Konsumen*.
- Nurhayati, Sri, & Wasilah. (2013). *Akuntansi Syariah di Indonesia, Edisi 5*, Jakarta: Salemba Empat
- Owen, M., Vincent, V., Ambarita, R. B., & Indra, E. (2022). Implementasi Metode Long Short Term Memory Untuk Memprediksi Pergerakan Nilai Harga Emas. *Jurnal Teknik Informasi Dan Komputer*, 5.
- 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 (Switzerland), 12(12). <https://doi.org/10.3390/app12125771>
- Puteri, D. I. (2023). *Implementasi Long Short Term Memory (LSTM) dan Bidirectional Long Short Term Memory (BiLSTM) Dalam Prediksi Harga Saham Syariah*. Jurnal Ilmiah Matematika. Universitas Padjajaran. Sumedang,

- Qotrunnada, F. (2022). *Implementasi Long Short Term Memory pada Optimalisasi Prediksi harga Saham Menggunakan Parameter Analisis Teknikal*. Skripsi. UIN Maulana Malik Ibrahim Malang.
- Rachmadina, D. A. (2024). *Penerapan Metode Backpropagation Neural Network (Bnn) Untuk Memprediksi Harga Minyak Dunia*. 1–23.
- Ridwansyah, T. (2022). *Implementasi Text Mining Terhadap Analisis Sentimen Masyarakat Dunia Di Twitter Terhadap Kota Medan Menggunakan K-Fold Cross Validation Dan Naïve Bayes Classifier*. Kajian Ilmiah Informatika dan Komputer. Universitas Budi Darma. Sumatera Utara.
- Risnandar, A. F., Supriadi, F., & Junaedi, D. I. (2025). *Prediksi Harga Saham Indofood Menggunakan Long Short-Term Memory*. 9, 4694–4701.
- Rizal, A. A., & Soraya, S. (2018). *Multi Time Steps Prediction dengan Recurrent Neural Network Long Short Term Memory*. MATRIK : Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer, 18(1), 115–124. <https://doi.org/10.30812/matrik.v18i1.344>
- Salehinejad, H., Sankar, S., Barfett, J., Colak, E., & Valaee, S. (2018). *Recent Advances in Recurrent Neural Networks*. Journal International. Cornell University
- Sautomo, S., & Pardede, H. F. (2021). *Prediksi Belanja Pemerintah Indonesia Menggunakan Long Short-Term Memory (LSTM)*. Jurnal Rekaya Sistem dan Teknologi Informasi. Vol.5 No.1 99-106.
- Sezer, O. B., Ozbayoglu, M., & Dogdu, E. (2017). A Deep Neural-Network Based Stock Trading System Based on Evolutionary Optimized Technical Analysis Parameters. *Procedia Computer Science*, 114, 473–480. <https://doi.org/10.1016/j.procs.2017.09.031>
- Sitompul, Y. A., & Sitepu, S. (2024). *Penerapan Metode Arima Dan Grey Theory Dalam Meramalkan Harga Saham Pt Indofood Sukses Makmur Tbk*. Leibniz: Jurnal Matematika. Universitas Sumatera Utara Vol. 4. Hal 12-22
- Stevenson, William, J., & Sum, C. C. (2014). *Manajemen Operasi Perspektif 110 Asia*.
- Suryalim, H. (2022). *Analisis Optimizer pada Convolutional Neural Network untuk Meningkatkan Akurasi Pengenalan Wajah*. Skripsi.
- Suyudi, Abdul Dwiyanto, M., Djamal, E. C., Maspupah Jurusan Informatika, A., & Sains dan Informatika Universitas Jenderal Achmad Yani Cimahi, F. (2019). *Prediksi Harga Saham menggunakan Metode Recurrent Neural Network*. In *Seminar Nasional Aplikasi Teknologi Informasi (SNATi)*.

- Syamsir, H. (2004). *Solusi Investasi Di Bursa Saham Indonesia Pendekatan Analisa Teknikal Melalui Studi Kasus Riil Dengan Dilengkapi Formulasi Metastock*. Jakarta: PT Elex Media Komputindo.
- Wibawa, M. S. (2016). Pengaruh Fungsi Aktivasi, Optimisasi dan Jumlah Epoch Terhadap Performa Jaringan Saraf Tiruan. *Jurnal Sistem Dan Informatika*.
- Yan, S. (2016). *Diambil kembali dari Medium: <https://shiyen.medium.com/understanding-lstm-and-its-diagrams37e2f46f1714>*.

LAMPIRAN

Lampiran 1 Data Harga Penutupan Saham PT Indofood dan Nilai RSI

<i>Date</i>	<i>Close</i>	<i>RSI</i>	<i>Date</i>	<i>Close</i>	<i>RSI</i>
2021-06-21	6025	30,23256	2021-08-16	6350	43,47826
2021-06-22	6175	38,77551	2021-08-18	6400	42,22222
2021-06-23	6025	21,73913	2021-08-19	6250	39,58333
2021-06-24	5975	22,72727	2021-08-20	6175	41,30435
2021-06-25	6050	27,65957	2021-08-23	6225	58,33333
2021-06-28	6000	31,70732	2021-08-24	6300	57,14286
2021-06-29	6225	47,82609	2021-08-25	6325	61,76471
2021-06-30	6175	46,80851	2021-08-26	6275	55,88235
2021-07-01	6550	59,01639	2021-08-27	6425	61,53846
2021-07-02	6575	59,67742	2021-08-30	6525	64,28571
2021-07-05	6450	56,06061	2021-08-31	6175	50,9434
2021-07-06	6450	55,38462	2021-09-01	6350	57,62712
2021-07-07	6450	56,25	2021-09-02	6450	58,33333
2021-07-08	6425	66,66667	2021-09-03	6525	59,67742
2021-07-09	6475	66,66667	2021-09-06	6475	54,23729
2021-07-12	6600	66,03774	2021-09-07	6250	45,45455
2021-07-13	6625	75	2021-09-08	6175	47,61905
2021-07-14	6625	78,26087	2021-09-09	6275	53,125
2021-07-15	6675	77,77778	2021-09-10	6225	50
2021-07-16	6625	77,77778	2021-09-13	6225	47,54098
2021-07-19	6650	72,97297	2021-09-14	6175	45,16129
2021-07-21	6525	67,5	2021-09-15	6250	49,20635
2021-07-22	6525	48	2021-09-16	6200	42,37288
2021-07-23	6475	42,30769	2021-09-17	6200	38,18182
2021-07-26	6500	54,54545	2021-09-20	6150	48,83721
2021-07-27	6575	60	2021-09-21	6150	38,88889
2021-07-28	6500	53,57143	2021-09-22	6200	35,29412
2021-07-29	6375	46,875	2021-09-23	6150	27,27273
2021-07-30	6075	30,95238	2021-09-24	6150	29,03226
2021-08-02	6175	29,26829	2021-09-27	6100	37,5
2021-08-03	6125	26,19048	2021-09-28	6150	47,82609
2021-08-04	6175	29,54545	2021-09-29	6150	36,84211
2021-08-05	6200	27,90698	2021-09-30	6350	60
2021-08-06	6225	30,95238	2021-10-01	6250	51,72414
2021-08-09	6150	27,27273	2021-10-04	6425	64,70588
2021-08-10	6125	30	2021-10-05	6500	64,70588
2021-08-12	6200	34,88372	2021-10-06	6600	72,22222
2021-08-13	6225	38,09524	2021-10-07	6700	75

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2021-10-08	6700	78,94737
2021-10-11	6625	73,17073
2021-10-12	6675	73,17073
2021-10-13	6775	79,06977
2021-10-14	6850	80,43478
2021-10-15	6775	78,7234
2021-10-18	6775	77,77778
2021-10-19	6825	78,7234
2021-10-21	6700	65,90909
2021-10-22	6650	69,04762
2021-10-25	6650	62,85714
2021-10-26	6675	60,60606
2021-10-27	6475	43,24324
2021-10-28	6400	33,33333
2021-10-29	6350	31,57895
2021-11-01	6275	31,57895
2021-11-02	6275	27,77778
2021-11-03	6250	18,18182
2021-11-04	6275	12,90323
2021-11-05	6250	13,7931
2021-11-08	6275	16,66667
2021-11-09	6350	19,35484
2021-11-10	6275	20,68966
2021-11-11	6325	27,58621
2021-11-12	6350	30
2021-11-15	6300	25,80645
2021-11-16	6300	34,78261
2021-11-17	6300	40
2021-11-18	6550	64,28571
2021-11-19	6550	72
2021-11-22	6625	75
2021-11-23	6600	75
2021-11-24	6550	68,96552
2021-11-25	6475	64,51613
2021-11-26	6350	54,28571
2021-11-29	6500	57,89474
2021-11-30	6300	51,16279
2021-12-01	6525	58
2021-12-02	6475	54,90196
2021-12-03	6450	56
2021-12-06	6425	54,90196
2021-12-07	6350	51,85185

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2021-12-08	6375	42,22222
2021-12-09	6550	50
2021-12-10	6625	50
2021-12-13	6550	48,14815
2021-12-14	6450	46,42857
2021-12-15	6500	50,90909
2021-12-16	6425	52,83019
2021-12-17	6475	48,97959
2021-12-20	6375	53,33333
2021-12-21	6375	41,66667
2021-12-22	6300	40,54054
2021-12-23	6375	46,15385
2021-12-24	6400	48,71795
2021-12-27	6375	51,35135
2021-12-28	6325	47,36842
2021-12-29	6400	41,17647
2021-12-30	6325	32,35294
2022-01-03	6250	32,35294
2022-01-04	6275	38,70968
2022-01-05	6225	32,25806
2022-01-06	6300	41,93548
2022-01-07	6425	47,05882
2022-01-10	6400	51,6129
2022-01-11	6525	58,33333
2022-01-12	6600	66,66667
2022-01-13	6600	63,63636
2022-01-14	6500	55,55556
2022-01-17	6475	55,55556
2022-01-18	6425	55,55556
2022-01-19	6425	51,51515
2022-01-20	6350	51,51515
2022-01-21	6475	62,85714
2022-01-24	6400	56,75676
2022-01-25	6350	56,75676
2022-01-26	6450	57,89474
2022-01-27	6425	50
2022-01-28	6375	48,57143
2022-01-31	6325	37,5
2022-02-02	6400	37,5
2022-02-03	6425	39,39394
2022-02-04	6425	44,82759
2022-02-07	6475	50

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2022-02-08	6375	46,875
2022-02-09	6350	45,45455
2022-02-10	6325	48,3871
2022-02-11	6350	40,74074
2022-02-14	6350	45,83333
2022-02-15	6375	52,17391
2022-02-16	6300	36,36364
2022-02-17	6375	45,83333
2022-02-18	6375	50
2022-02-21	6275	45,83333
2022-02-22	6400	50
2022-02-23	6400	48
2022-02-24	6325	42,85714
2022-02-25	6200	32,25806
2022-03-01	6175	35,71429
2022-03-02	6200	39,28571
2022-03-04	6075	34,375
2022-03-07	5725	22,22222
2022-03-08	5925	33,96226
2022-03-09	5950	33,96226
2022-03-10	6075	41,81818
2022-03-11	6050	37,73585
2022-03-14	6100	40
2022-03-15	6050	41,50943
2022-03-16	6150	40,38462
2022-03-17	6125	39,62264
2022-03-18	6025	38,88889
2022-03-21	5975	41,17647
2022-03-22	6025	44,23077
2022-03-23	5975	41,50943
2022-03-24	5950	44,89796
2022-03-25	5775	52,38095
2022-03-28	5800	42,85714
2022-03-29	5825	42,85714
2022-03-30	5925	41,17647
2022-03-31	5950	44,11765
2022-04-01	6050	47,22222
2022-04-04	6100	52,77778
2022-04-05	6100	46,875
2022-04-06	6100	48,3871
2022-04-07	6100	55,55556
2022-04-08	6125	61,53846

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2022-04-11	6075	53,84615
2022-04-12	6150	62,96296
2022-04-13	6200	67,85714
2022-04-14	6125	79,16667
2022-04-18	6125	78,26087
2022-04-19	6200	80
2022-04-20	6225	77,27273
2022-04-21	6250	77,27273
2022-04-22	6300	75
2022-04-25	6300	72,22222
2022-04-26	6300	72,22222
2022-04-27	6350	75
2022-04-28	6300	68,18182
2022-05-09	6125	50
2022-05-10	6400	67,56757
2022-05-11	6450	66,66667
2022-05-12	6400	61,11111
2022-05-13	6450	68,57143
2022-05-17	6500	70,27027
2022-05-18	6500	67,64706
2022-05-19	6450	62,85714
2022-05-20	6450	61,76471
2022-05-23	6400	55,88235
2022-05-24	6275	48,71795
2022-05-25	6400	54,54545
2022-05-27	6425	53,48837
2022-05-30	6450	57,14286
2022-05-31	6600	73,17073
2022-06-02	6700	67,64706
2022-06-03	6675	63,63636
2022-06-06	6925	75,60976
2022-06-07	6925	74,35897
2022-06-08	6975	74,35897
2022-06-09	6875	67,44186
2022-06-10	6875	70,73171
2022-06-13	6825	67,44186
2022-06-14	6875	72,09302
2022-06-15	6800	75,60976
2022-06-16	6800	72,22222
2022-06-17	6825	72,22222
2022-06-20	6900	73,68421
2022-06-21	6875	66,66667

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2022-06-22	6975	66,66667
2022-06-23	7125	73,68421
2022-06-24	7075	60
2022-06-27	7200	65,71429
2022-06-28	7250	65,71429
2022-06-29	7100	62,16216
2022-06-30	7050	58,97436
2022-07-01	7025	60,52632
2022-07-04	6825	47,72727
2022-07-05	6925	55,55556
2022-07-06	7000	58,33333
2022-07-07	6900	52,94118
2022-07-08	7025	54,71698
2022-07-11	7025	55,76923
2022-07-12	7050	53,06122
2022-07-13	6825	38,46154
2022-07-14	6925	44,44444
2022-07-15	6900	38
2022-07-18	7050	42,59259
2022-07-19	7150	51,92308
2022-07-20	7100	51,92308
2022-07-21	7175	55,55556
2022-07-22	7050	58,82353
2022-07-25	7025	54,16667
2022-07-26	7000	50
2022-07-27	7000	54,7619
2022-07-28	6900	43,90244
2022-07-29	6800	40
2022-08-01	6900	43,75
2022-08-02	6700	44,68085
2022-08-03	6575	35,41667
2022-08-04	6650	40
2022-08-05	6625	31,11111
2022-08-08	6625	24,39024
2022-08-09	6550	23,80952
2022-08-10	6625	23,80952
2022-08-11	6600	26,31579
2022-08-12	6625	28,94737
2022-08-15	6625	29,72973
2022-08-16	6550	27,5
2022-08-18	6575	32,43243
2022-08-19	6575	36,36364

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2022-08-22	6650	34,375
2022-08-23	6650	45,83333
2022-08-24	6650	57,89474
2022-08-25	6575	42,10526
2022-08-26	6500	38,09524
2022-08-29	6625	50
2022-08-30	6500	46,42857
2022-08-31	6225	27,77778
2022-09-01	6275	32,43243
2022-09-02	6400	39,02439
2022-09-05	6375	38,09524
2022-09-06	6350	40
2022-09-07	6300	36,58537
2022-09-08	6250	34,88372
2022-09-09	6225	29,26829
2022-09-12	6225	29,26829
2022-09-13	6200	28,57143
2022-09-14	6225	32,5
2022-09-15	6175	33,33333
2022-09-16	6300	33,33333
2022-09-19	6200	34,21053
2022-09-20	6200	48,14815
2022-09-21	6250	48,14815
2022-09-22	6250	36,36364
2022-09-23	6200	34,78261
2022-09-26	6200	36,36364
2022-09-27	6150	36,36364
2022-09-28	6100	36,36364
2022-09-29	6025	33,33333
2022-09-30	6025	33,33333
2022-10-03	6025	34,78261
2022-10-04	6050	34,78261
2022-10-05	6025	36,36364
2022-10-06	6100	30
2022-10-07	6075	35,29412
2022-10-10	6125	42,10526
2022-10-11	6125	35,29412
2022-10-12	6075	31,57895
2022-10-13	6050	33,33333
2022-10-14	6050	33,33333
2022-10-17	6125	47,36842
2022-10-18	6175	57,89474

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2022-10-19	6225	72,22222
2022-10-20	6350	78,26087
2022-10-21	6275	69,23077
2022-10-24	6300	69,23077
2022-10-25	6200	62,06897
2022-10-26	6325	64,51613
2022-10-27	6300	64,51613
2022-10-28	6375	65,625
2022-10-31	6450	68,57143
2022-11-01	6450	72,72727
2022-11-02	6425	72,72727
2022-11-03	6375	68,57143
2022-11-04	6450	68,57143
2022-11-07	6500	68,57143
2022-11-08	6475	64,70588
2022-11-09	6550	62,5
2022-11-10	6475	62,5
2022-11-11	6400	55,88235
2022-11-14	6475	66,66667
2022-11-15	6500	62,06897
2022-11-16	6500	64,28571
2022-11-17	6525	61,53846
2022-11-18	6500	54,16667
2022-11-21	6525	56
2022-11-22	6450	51,85185
2022-11-23	6450	56
2022-11-24	6425	47,82609
2022-11-25	6325	36
2022-11-28	6275	34,61538
2022-11-29	6325	32
2022-11-30	6450	48,14815
2022-12-01	6675	66,66667
2022-12-02	6775	67,64706
2022-12-05	6775	66,66667
2022-12-06	6775	66,66667
2022-12-07	6850	68,57143
2022-12-08	6900	72,22222
2022-12-09	6925	72,22222
2022-12-12	6975	80
2022-12-13	6925	75,67568
2022-12-14	6950	78,37838
2022-12-15	6975	88,23529

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2022-12-16	6950	90,90909
2022-12-19	7000	90,90909
2022-12-20	6975	86,2069
2022-12-21	6900	69,56522
2022-12-22	6800	52,17391
2022-12-23	6825	54,16667
2022-12-26	6875	57,69231
2022-12-27	6800	46,15385
2022-12-28	6725	37,03704
2022-12-29	6800	41,37931
2022-12-30	6725	33,33333
2023-01-02	6775	40
2023-01-03	6750	36,66667
2023-01-04	6850	42,42424
2023-01-05	6850	43,75
2023-01-06	6950	47,05882
2023-01-09	6950	48,48485
2023-01-10	6675	39,02439
2023-01-11	6675	43,24324
2023-01-12	6625	39,47368
2023-01-13	6700	41,02564
2023-01-16	6700	44,44444
2023-01-17	6825	55,26316
2023-01-18	6950	57,5
2023-01-19	6875	57,5
2023-01-20	6875	55,26316
2023-01-24	6925	58,97436
2023-01-25	6800	47,5
2023-01-26	6700	43,18182
2023-01-27	6700	37,5
2023-01-30	6800	43,18182
2023-01-31	6725	52,77778
2023-02-01	6650	48,71795
2023-02-02	6600	48,71795
2023-02-03	6700	50
2023-02-06	6675	48,78049
2023-02-07	6675	41,66667
2023-02-08	6800	41,66667
2023-02-09	6775	44,11765
2023-02-10	6900	51,28205
2023-02-13	6725	40,90909
2023-02-14	6750	47,5

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2023-02-15	6675	48,71795
2023-02-16	6600	45,2381
2023-02-17	6600	39,47368
2023-02-20	6625	44,44444
2023-02-21	6525	43,24324
2023-02-22	6600	50
2023-02-23	6575	42,85714
2023-02-24	6575	44,11765
2023-02-27	6600	45,71429
2023-02-28	6475	31,42857
2023-03-01	6275	26,19048
2023-03-02	6250	15,78947
2023-03-03	6275	21,875
2023-03-06	6250	18,75
2023-03-07	6250	20,68966
2023-03-08	6150	20
2023-03-09	6150	20
2023-03-10	6125	16,66667
2023-03-13	6125	19,23077
2023-03-14	6175	16
2023-03-15	6100	14,81481
2023-03-16	6150	20,68966
2023-03-17	6125	17,24138
2023-03-20	6200	29,62963
2023-03-21	6125	36,36364
2023-03-24	6325	55,17241
2023-03-27	6375	56,66667
2023-03-28	6375	58,62069
2023-03-29	6350	56,66667
2023-03-30	6400	67,85714
2023-03-31	6200	52,77778
2023-04-03	6175	52,77778
2023-04-04	6350	60,46512
2023-04-05	6275	54,54545
2023-04-06	6250	57,14286
2023-04-10	6325	58,13953
2023-04-11	6350	60,46512
2023-04-12	6275	53,48837
2023-04-13	6275	57,5
2023-04-14	6300	48,48485
2023-04-17	6325	46,875
2023-04-18	6350	48,48485

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2023-04-26	6500	57,89474
2023-04-27	6425	51,28205
2023-04-28	6450	65,625
2023-05-02	6500	69,69697
2023-05-03	6650	68,75
2023-05-04	6625	73,33333
2023-05-05	6700	78,125
2023-05-08	6875	80,55556
2023-05-09	6825	75,67568
2023-05-10	6900	83,78378
2023-05-11	6800	75,60976
2023-05-12	6725	69,76744
2023-05-15	6725	69,04762
2023-05-16	6825	71,11111
2023-05-17	6800	65
2023-05-19	6950	74,4186
2023-05-22	6900	70,45455
2023-05-23	6950	70,45455
2023-05-24	6950	65,78947
2023-05-25	7100	72,09302
2023-05-26	7200	72,72727
2023-05-29	7200	67,56757
2023-05-30	7150	67,56757
2023-05-31	7100	61,11111
2023-06-05	7250	73,68421
2023-06-06	7375	82,5
2023-06-07	7450	83,72093
2023-06-08	7425	80
2023-06-09	7425	82,05128
2023-06-12	7450	79,41176
2023-06-13	7350	75
2023-06-14	7425	75,67568
2023-06-15	7450	76,31579
2023-06-16	7275	58,97436
2023-06-19	7400	60
2023-06-20	7150	48
2023-06-21	7375	57,89474
2023-06-22	7225	54,09836
2023-06-23	7300	51,72414
2023-06-26	7325	48,14815
2023-06-27	7350	46,15385
2023-07-03	7250	43,63636

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2023-07-04	7375	48,33333
2023-07-05	7425	49,18033
2023-07-06	7450	53,44828
2023-07-07	7275	45,16129
2023-07-10	7250	43,54839
2023-07-11	7300	50,87719
2023-07-12	7400	50
2023-07-13	7450	62,5
2023-07-14	7450	53,84615
2023-07-17	7300	53,84615
2023-07-18	7350	52,63158
2023-07-20	7325	50
2023-07-21	7325	48,64865
2023-07-24	7275	51,42857
2023-07-25	7175	38,23529
2023-07-26	7175	34,375
2023-07-27	7250	38,23529
2023-07-28	7325	53,33333
2023-07-31	7325	55,17241
2023-08-01	7000	35
2023-08-02	7050	31,57895
2023-08-03	7050	27,77778
2023-08-04	7050	27,77778
2023-08-07	7075	35,48387
2023-08-08	7000	28,125
2023-08-09	6975	28,125
2023-08-10	7075	36,11111
2023-08-11	7050	37,14286
2023-08-14	7050	41,93548
2023-08-15	7000	39,39394
2023-08-16	7100	41,17647
2023-08-18	7050	33,33333
2023-08-21	7000	31,42857
2023-08-22	7000	50
2023-08-23	7075	52,17391
2023-08-24	7025	48
2023-08-25	7125	55,17241
2023-08-28	7100	51,72414
2023-08-29	7150	60,71429
2023-08-30	7050	54,83871
2023-08-31	7100	51,72414
2023-09-01	7000	46,875

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2023-09-04	6900	41,66667
2023-09-05	6850	41,66667
2023-09-06	6875	36,36364
2023-09-07	6950	44,11765
2023-09-08	6925	45,45455
2023-09-11	6950	47,05882
2023-09-12	6850	37,14286
2023-09-13	6900	42,85714
2023-09-14	6700	28,20513
2023-09-15	6750	32,5
2023-09-18	6600	25
2023-09-19	6675	32,55814
2023-09-20	6800	36,95652
2023-09-21	6700	36,95652
2023-09-22	6775	44,44444
2023-09-25	6775	46,51163
2023-09-26	6700	42,22222
2023-09-27	6750	40,90909
2023-09-29	6625	37,5
2023-10-02	6750	42,30769
2023-10-03	6725	44,89796
2023-10-04	6750	43,75
2023-10-05	6875	57,77778
2023-10-06	6875	55,81395
2023-10-09	6850	63,15789
2023-10-10	6800	56,75676
2023-10-11	6750	47,05882
2023-10-12	6725	51,6129
2023-10-13	6700	44,82759
2023-10-16	6725	46,66667
2023-10-17	6850	59,375
2023-10-18	6800	53,125
2023-10-19	6750	58,62069
2023-10-20	6675	44,44444
2023-10-23	6600	41,37931
2023-10-24	6725	48,48485
2023-10-25	6750	41,37931
2023-10-26	6750	41,37931
2023-10-27	6750	42,85714
2023-10-30	6725	44,44444
2023-10-31	6650	42,85714
2023-11-01	6650	44,44444

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2023-11-02	6475	36,36364
2023-11-03	6375	30,55556
2023-11-06	6350	18,75
2023-11-07	6375	22,58065
2023-11-08	6350	23,33333
2023-11-09	6325	25
2023-11-10	6325	28
2023-11-13	6450	28
2023-11-14	6375	22,22222
2023-11-15	6425	27,58621
2023-11-16	6450	30
2023-11-17	6425	30
2023-11-20	6400	32,14286
2023-11-21	6425	34,48276
2023-11-22	6350	40
2023-11-23	6350	47,61905
2023-11-24	6425	56,52174
2023-11-27	6400	52,17391
2023-11-28	6375	52,17391
2023-11-29	6425	58,33333
2023-11-30	6425	58,33333
2023-12-01	6325	39,13043
2023-12-04	6350	47,61905
2023-12-05	6475	54,16667
2023-12-06	6500	54,16667
2023-12-07	6450	52
2023-12-08	6400	50
2023-12-11	6375	46,15385
2023-12-12	6425	56
2023-12-13	6400	53,84615
2023-12-14	6400	47,82609
2023-12-15	6325	44
2023-12-18	6275	42,30769
2023-12-19	6325	42,30769
2023-12-20	6275	39,28571
2023-12-21	6275	45,83333
2023-12-22	6275	43,47826
2023-12-27	6350	38,09524
2023-12-28	6450	45,83333
2023-12-29	6450	50
2024-01-02	6450	55
2024-01-03	6400	52,38095

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2024-01-04	6450	52,38095
2024-01-05	6450	55
2024-01-08	6400	50
2024-01-09	6375	55
2024-01-10	6350	57,89474
2024-01-11	6350	52,94118
2024-01-12	6375	62,5
2024-01-15	6450	68,42105
2024-01-16	6450	68,42105
2024-01-17	6400	55,55556
2024-01-18	6375	40
2024-01-19	6450	50
2024-01-22	6425	47,36842
2024-01-23	6400	50
2024-01-24	6450	50
2024-01-25	6400	45
2024-01-26	6425	52,63158
2024-01-29	6425	55,55556
2024-01-30	6400	55,55556
2024-01-31	6375	52,63158
2024-02-01	6350	47,36842
2024-02-02	6400	44,44444
2024-02-05	6300	36,36364
2024-02-06	6350	45,45455
2024-02-07	6350	47,61905
2024-02-12	6400	45
2024-02-13	6350	42,85714
2024-02-15	6400	50
2024-02-16	6425	47,61905
2024-02-19	6475	57,14286
2024-02-20	6550	60,86957
2024-02-21	6600	64
2024-02-22	6725	72,41379
2024-02-23	6725	75
2024-02-26	6625	67,74194
2024-02-27	6575	61,29032
2024-02-28	6625	72,41379
2024-02-29	6625	70,37037
2024-03-01	6675	72,41379
2024-03-04	6550	59,375
2024-03-05	6500	59,375
2024-03-06	6500	56,66667

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2024-03-07	6350	45,71429
2024-03-08	6350	42,42424
2024-03-13	6300	34,375
2024-03-14	6425	40
2024-03-15	6425	30
2024-03-18	6450	32,25806
2024-03-19	6425	35,71429
2024-03-20	6475	42,85714
2024-03-21	6450	37,03704
2024-03-22	6425	35,71429
2024-03-25	6450	33,33333
2024-03-26	6375	36
2024-03-27	6325	36
2024-03-28	6375	40,74074
2024-04-01	6425	56,52174
2024-04-02	6600	66,66667
2024-04-03	6550	66,66667
2024-04-04	6500	55,55556
2024-04-05	6500	55,55556
2024-04-16	6150	35
2024-04-17	6050	32,55814
2024-04-18	6125	34,09091
2024-04-19	6100	34,09091
2024-04-22	6150	37,77778
2024-04-23	6025	32,65306
2024-04-24	6050	36,17021
2024-04-25	6200	45,09804
2024-04-26	6050	38,18182
2024-04-29	6175	41,37931
2024-04-30	6250	37,03704
2024-05-02	6225	37,73585
2024-05-03	6225	39,21569
2024-05-06	6225	39,21569
2024-05-07	6150	50
2024-05-08	6200	57,89474
2024-05-13	6100	48,71795
2024-05-14	6175	53,65854
2024-05-15	6100	47,61905
2024-05-16	6150	56,41026
2024-05-17	6150	55,26316
2024-05-20	6225	51,42857
2024-05-21	6150	56,25

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2024-05-22	6150	48,14815
2024-05-27	6125	40
2024-05-28	6125	41,66667
2024-05-29	6125	41,66667
2024-05-30	6050	37,03704
2024-05-31	5875	32,25806
2024-06-03	6050	41,66667
2024-06-04	6100	50
2024-06-05	6125	46,875
2024-06-06	6200	56,25
2024-06-07	6200	53,33333
2024-06-10	6175	51,6129
2024-06-11	6125	43,33333
2024-06-12	6200	53,33333
2024-06-13	6125	48,48485
2024-06-14	6075	47,05882
2024-06-19	5925	40
2024-06-20	5925	40
2024-06-21	5975	46,15385
2024-06-24	5925	52,94118
2024-06-25	5950	42,85714
2024-06-26	5900	35,71429
2024-06-27	6025	43,75
2024-06-28	6075	41,93548
2024-07-01	6100	43,75
2024-07-02	6100	45,16129
2024-07-03	6100	48,27586
2024-07-04	6075	40,74074
2024-07-05	6125	50
2024-07-08	6200	59,25926
2024-07-09	5950	51,6129
2024-07-10	6000	54,54545
2024-07-11	5925	47,05882
2024-07-12	5950	51,51515
2024-07-15	5975	51,51515
2024-07-16	5950	53,125
2024-07-17	6050	51,6129
2024-07-18	6075	50
2024-07-19	6075	48,27586
2024-07-22	6025	45,16129
2024-07-23	6100	50
2024-07-24	6000	45,94595

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2024-07-25	6100	48,71795
2024-07-26	6125	45,94595
2024-07-29	6075	58,62069
2024-07-30	6050	53,57143
2024-07-31	6125	64,28571
2024-08-01	6150	64,28571
2024-08-02	6100	58,62069
2024-08-05	6050	56,66667
2024-08-06	6100	53,57143
2024-08-07	6125	53,57143
2024-08-08	6225	59,375
2024-08-09	6275	65,625
2024-08-12	6250	60
2024-08-13	6325	72,41379
2024-08-14	6500	75
2024-08-15	6475	71,875
2024-08-16	6525	78,125
2024-08-19	6450	73,52941
2024-08-20	6525	73,52941
2024-08-21	6625	75,67568
2024-08-22	6650	80,55556
2024-08-23	6700	86,11111
2024-08-26	6850	87,5
2024-08-27	6800	82,92683
2024-08-28	6825	81,57895
2024-08-29	6800	78,37838
2024-08-30	6850	81,57895
2024-09-02	6725	70
2024-09-03	6775	65,71429
2024-09-04	6700	62,16216
2024-09-05	6975	69,56522
2024-09-06	7025	75,55556
2024-09-09	6925	67,3913
2024-09-10	7050	68,08511
2024-09-11	7000	64,58333
2024-09-12	7150	67,30769
2024-09-13	7100	60,41667
2024-09-17	7100	63,04348
2024-09-18	7075	60,86957
2024-09-19	7075	62,22222
2024-09-20	7150	63,04348
2024-09-23	7100	67,44186

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2024-09-24	7175	68,18182
2024-09-25	7175	73,17073
2024-09-26	7225	65,625
2024-09-27	7175	59,375
2024-09-30	7050	57,57576
2024-10-01	7175	57,57576
2024-10-02	7075	54,28571
2024-10-03	7100	46,66667
2024-10-04	7050	46,66667
2024-10-07	6925	40
2024-10-08	7000	45,94595
2024-10-09	6925	42,5
2024-10-10	7025	43,90244
2024-10-11	7000	45
2024-10-14	7050	43,58974
2024-10-15	7200	51,11111
2024-10-16	7175	47,72727
2024-10-17	7175	50
2024-10-18	7425	65,95745
2024-10-21	7400	60,46512
2024-10-22	7425	67,5
2024-10-23	7500	69,04762
2024-10-24	7525	73,17073
2024-10-25	7500	81,08108
2024-10-28	7525	80
2024-10-29	7575	88,23529
2024-10-30	7475	76,47059
2024-11-01	7300	65
2024-11-04	7450	68,18182
2024-11-05	7875	74,54545
2024-11-06	7800	71,92982
2024-11-07	7900	73,77049
2024-11-08	7700	59,32203
2024-11-11	7800	62,90323
2024-11-12	7850	63,49206
2024-11-13	7725	56,92308
2024-11-14	7700	55,38462
2024-11-15	7725	56,92308
2024-11-18	7725	56,25
2024-11-19	7800	56,92308
2024-11-20	7800	60,65574
2024-11-21	7625	60,65574

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2024-11-22	7475	50,81967
2024-11-25	7600	38,77551
2024-11-26	7650	43,75
2024-11-27	7295	29,20962
2024-11-28	7650	48,4472
2024-11-29	7550	42,23602
2024-12-02	7550	40,38462
2024-12-03	7775	51,50602
2024-12-04	7850	54,38596
2024-12-05	7875	54,38596
2024-12-06	7925	55,68182
2024-12-09	8200	60,20408
2024-12-10	8150	58,70647
2024-12-11	8200	65,25199
2024-12-12	8200	70,89337
2024-12-13	8175	67,5841
2024-12-16	8125	64,52599
2024-12-17	8050	77,85978
2024-12-18	7825	57,14286
2024-12-19	7650	53,84615
2024-12-20	7625	52,83019
2024-12-23	7725	47,91667

<i>Date</i>	<i>Close</i>	<i>RSI</i>
2024-12-24	7625	40,81633
2024-12-27	7700	43,13725
2024-12-30	7700	40,81633
2025-01-02	7475	19,14894
2025-01-03	7550	25
2025-01-06	7375	18,86792
2025-01-07	7450	23,21429
2025-01-08	7475	25
2025-01-09	7500	27,27273
2025-01-10	7725	39,34426
2025-01-13	7725	46,15385
2025-01-14	7525	45,28302
2025-01-15	7700	52,54237
2025-01-16	7575	45
2025-01-17	7600	49,12281
2025-01-20	7375	39,68254
2025-01-21	7450	42,42424
2025-01-22	7625	54,6875
2025-01-23	7500	48,48485
2025-01-24	7550	55,7377
2025-01-30	7625	55,7377
2025-01-31	7850	60,86957

Lampiran 2: *Script Phyton Prediksi Saham Menggunakan Metode Long Short Term Memory*

```

import pandas as pd
import numpy as np
import seaborn as sns
import mplfinance as mpf
from sklearn.preprocessing import MinMaxScaler
from math import sqrt
from sklearn.metrics import mean_squared_error,
mean_absolute_percentage_error
from sklearn.metrics import mean_squared_error, mean_absolute_error
from sklearn.model_selection import GridSearchCV
from scikeras.wrappers import KerasRegressor
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import LSTM, Dense, Dropout, Input
from tensorflow.keras.callbacks import EarlyStopping
import matplotlib.pyplot as plt
import datetime

# Import data saham
file_path = 'D:/INDFJK(3).xlsx'
data = pd.read_excel(file_path)
data.head()

data['Date'] = pd.to_datetime(data['Date']) # Convert 'Date' column to datetime
format
data.set_index('Date', inplace=True) # Set 'Date' as the DataFrame index

plt.figure(figsize=(16,6))
plt.title('Close Price History')
plt.plot(data['Close'])
plt.xlabel('Date', fontsize=18)
plt.ylabel('Close Price IDR', fontsize=18)
plt.show()

# Convert 'Date' column to datetime format
data['Date'] = pd.to_datetime(data['Date'], errors='coerce')

# Filter hanya kolom 'Date' dan 'Close'
data = data[['Date', 'Close']].dropna()
data

# Mengecek deskriptif dataset
deskriptif = data.describe()
print(deskriptif)

```

```

# Ascending
data1 = data.sort_values(by='Date', ascending=True).reset_index(drop=True)
data1

# Menghitung RSI (Relative Strength Index)
window_length = 14
delta = data1['Close'].diff(1)
gain = (delta.where(delta > 0, 0)).rolling(window=window_length).mean()
loss = (-delta.where(delta < 0, 0)).rolling(window=window_length).mean()
rs = gain / loss
data1['RSI'] = 100 - (100 / (1 + rs))
data2 = data1.dropna(subset=['RSI'])
data2

# Mengecek deskriptif dataset
deskriptif = data2.describe()
print(deskriptif)

# Pisahkan data testing (1 Nov 2024 - 31 Jan 2025)
test_mask = (data2["Date"] >= "2024-11-01") & (data2["Date"] <= "2025-01-31")
data_test = data2[test_mask]
# Sisa data untuk training & validation
data_remaining = data2[~test_mask]

# Normalisasi Data
scaler = MinMaxScaler(feature_range=(0, 1))
dataset_train = scaler.fit_transform(data_remaining[['Close', 'RSI']])
data_test = scaler.transform(data_test[['Close', 'RSI']])

plt.subplots(figsize = (16,6))
plt.plot(dataset_train)
plt.xlabel("Days as 1st, 2nd, 3rd..")
plt.ylabel("Close Price and RSI")
plt.show()

# Fungsi untuk membuat dataset dengan time step
def create_sequences(data2, time_step):
    X, y = [], []
    for i in range(time_step, len(data2)):
        X.append(data2[i-time_step:i])
        y.append(data2[i, :])
    return np.array(X), np.array(y)

time_step = 30 # Gunakan 30 hari terakhir sebagai input
X_train, y_train = create_sequences(dataset_train, time_step)
X_test, y_test = create_sequences(data_test, time_step)

```

```

print("Shape of X_train before reshape :",X_train.shape)
print("Shape of y_train before reshape :",y_train.shape)
print("Shape of X_test before reshape :",X_test.shape)
print("Shape of y_test before reshape :",y_test.shape)

# Fungsi untuk membuat model
def create_lstm_model(input_shape):
    model = Sequential([
        Input(shape=input_shape),
        LSTM(units=50, activation='tanh', recurrent_activation='sigmoid',
return_sequences=True),
        LSTM(units=50, activation='tanh', recurrent_activation='sigmoid'),
        Dense(units=25),
        Dense(units=2)
    ])
    model.compile(optimizer='adam', loss='mean_squared_error')
    return model

# Fungsi evaluasi model (RMSE dan MAPE)
def evaluate_model(y_true, y_pred):
    rmse = root_mean_squared_error(y_true, y_pred)
    mape = mean_absolute_percentage_error(y_true, y_pred) * 100
    return rmse, mape

# Grid search parameter
epochs_list = [50, 100]
batch_sizes = [4, 16, 32, 64]
tscv = TimeSeriesSplit(n_splits=5)

best_rmse = float('inf')
best_mape = float('inf')
best_params = None
best_history = None

# Tambahan variabel untuk menyimpan data dari fold terbaik
best_X_tr, best_X_val = None, None
best_y_tr, best_y_val = None, None

# Loop grid search
for epochs in epochs_list:
    for batch_size in batch_sizes:
        fold_rmse = []
        fold_mape = []
        print(f"\nEvaluating: epochs={epochs}, batch_size={batch_size}")

        history_list = []

        for fold, (train_idx, val_idx) in enumerate(tscv.split(X_train)):

```

```

X_tr, X_val = X_train[train_idx], X_train[val_idx]
y_tr, y_val = y_train[train_idx], y_train[val_idx]

model = create_lstm_model((X_train.shape[1], X_train.shape[2]))
history = model.fit(X_tr, y_tr, validation_data=(X_val, y_val),
epochs=epochs, batch_size=batch_size, verbose=0)

y_val_pred = model.predict(X_val)
y_val_inv = scaler.inverse_transform(y_val.reshape(-1, 1))
y_val_pred_inv = scaler.inverse_transform(y_val_pred.reshape(-1, 1))

rmse, mape = evaluate_model(y_val_inv, y_val_pred_inv)

print(f'Fold {fold+1} - RMSE: {rmse:.4f}, MAPE: {mape:.2f}%')
fold_rmse.append(rmse)
fold_mape.append(mape)

avg_rmse = np.mean(fold_rmse)
avg_mape = np.mean(fold_mape)
print(f'Average RMSE: {avg_rmse:.4f}, Average MAPE:
{avg_mape:.2f}%')

# Simpan model terbaik
if avg_rmse < best_rmse:
    best_rmse = avg_rmse
    best_mape = avg_mape
    best_params = (epochs, batch_size)
    best_history = history

# Output hasil terbaik
print(f'\nModel terbaik → Epochs: {best_params[0]}, Batch size:
{best_params[1]}')
print(f'Best CV RMSE: {best_rmse:.4f}')
print(f'Best CV MAPE: {best_mape:.2f}%')

# Nilai weight dan bias
first_layer = final_model.layers[0] # Ambil layer LSTM pertama
weights, recurrent_weights, biases = first_layer.get_weights()

# Print nilai weight & bias
print("Weight (Input -> Hidden):")
print(weights)

print("\nWeight (Hidden -> Hidden / Recurrent):")
print(recurrent_weights)

print("\nBias:")
print(biases)

```

```

# Simpan data dari fold terbaik
best_X_tr, best_y_tr = X_tr, y_tr
best_X_val, best_y_val = X_val, y_val

# Simpan model terbaik
if avg_rmse < best_rmse:
    best_rmse = avg_rmse
    best_mape = avg_mape
    best_params = (epochs, batch_size)
    best_history = history
    best_X_tr, best_y_tr = X_tr, y_tr
    best_X_val, best_y_val = X_val, y_val

#Train ulang model terbaik pada seluruh data train
final_model = create_lstm_model((X_train.shape[1], X_train.shape[2]))
final_history = final_model.fit(
    best_X_tr, best_y_tr,
    validation_data=(best_X_val, best_y_val),
    epochs=best_params[0],
    batch_size=best_params[1],
    verbose=1
)

plt.figure(figsize=(10, 5))
plt.plot(final_history.history['loss'], label='Train Loss') # default: garis penuh
plt.plot(final_history.history['val_loss'], label='Validation Loss', linestyle='--') #
garis putus-putus
plt.title('Training vs Validation Loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()

# Prediksi data testing
y_test_pred = final_model.predict(X_test)

# Inverse transform (kembalikan ke skala IDR)
y_test_reshaped = y_test.reshape(-1, 1)
y_test_pred_reshaped = y_test_pred.reshape(-1, 1)

y_test_inv = scaler.inverse_transform(y_test_reshaped)
y_test_pred_inv = scaler.inverse_transform(y_test_pred_reshaped)

# Evaluasi hasil prediksi di data testing (dalam satuan IDR)
rmse_test, mape_test = evaluate_model(y_test_inv, y_test_pred_inv)

```

```

rmse_test/= 1000

print(f"\nEvaluasi pada Data Testing:")
print(f"RMSE: {rmse_test:.4f}")
print(f"MAPE: {mape_test:.2f}%")

# Prediksi data testing
y_test_pred = final_model.predict(X_test)

# Evaluasi hasil prediksi di data testing
rmse_test, mape_test = evaluate_model(y_test, y_test_pred)

print(f"\nEvaluasi pada Data Testing:")
print(f"RMSE: {rmse_test:.4f}")
print(f"MAPE: {mape_test:.2f}%")

# Visualisasi hasil prediksi vs aktual Close Price
n = len(y_test_inv)
tanggal = pd.date_range(start='2024-12-01', end='2025-01-31', periods=n)

# Plot
plt.figure(figsize=(12, 6))
plt.plot(tanggal, y_test_inv, label='Actual Price')
plt.plot(tanggal, y_test_pred_inv, label='Predicted Price', linestyle='--')

plt.title('Actual vs Predicted Close Price (Testing Set)', fontsize=16)
plt.xlabel('Date', fontsize=14)
plt.ylabel('Close Price (IDR)', fontsize=14)
plt.ylim(5000, 10000)
plt.legend(fontsize=12)
plt.grid(True)
plt.tight_layout()
plt.xticks(rotation=45, fontsize=14)
plt.show()

# Visualisasi hasil prediksi vs aktual RSI
n = len(y_test_inv)
tanggal = pd.date_range(start='2024-12-01', end='2025-01-31', periods=n)

# Plot
plt.figure(figsize=(12, 6))
plt.plot(tanggal, y_test_inv, label='Actual Price')
plt.plot(tanggal, y_test_pred_inv, label='Predicted Price', linestyle='--')

plt.title('Actual vs Predicted RSI (Testing Set)', fontsize=16)
plt.xlabel('Date', fontsize=14)
plt.ylabel('Close Price (IDR)', fontsize=14)

```

```
plt.ylim(0, 300)
plt.legend(fontsize=14)
plt.grid(True)
plt.tight_layout()
plt.xticks(rotation=45, fontsize=14)
plt.show()
```

RIWAYAT HIDUP



Khoiro Amalia lahir di Kota Singkawang pada tanggal 29 Oktober 2002. Penulis merupakan anak bungsu dari tiga bersaudara, putri dari pasangan almarhum Bapak Abdul Mukid dan Ibu Khalifah. Penulis menghabiskan masa kecil di Kota Singkawang selama 15 tahun. Pendidikan formal dimulai di MI Al-Hidayah Singkawang dan lulus pada tahun 2014, kemudian melanjutkan jenjang pendidikan menengah pertama di SMP Negeri 3 Singkawang hingga lulus pada tahun 2017. Selanjutnya, penulis menempuh pendidikan menengah atas di Kabupaten Jombang, tepatnya di Pondok Pesantren Darul Ulum, Peterongan. Pendidikan SMA diselesaikan di SMA Darul Ulum 1 Unggulan BPP-T Jombang dan lulus pada tahun 2020. Setelah itu, penulis melanjutkan studi Strata 1 pada Program Studi Matematika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Selama masa perkuliahan, penulis aktif dalam berbagai kegiatan akademik dan organisasi kemahasiswaan, antara lain menjadi Musyrifah di Ma'had Al-Jami'ah UIN Maulana Malik Ibrahim Malang, asisten praktikum pada beberapa mata kuliah, serta menjadi anggota Himpunan Mahasiswa Program Studi (HMPS) Integral Matematika. Selain itu, penulis juga mengikuti sejumlah organisasi lain, baik intra maupun ekstra kampus yang dapat mendukung pengembangan diri penulis. Penulis juga pernah melaksanakan kegiatan magang di Bank Rakyat Indonesia (BRI) sebagai bagian dari pemenuhan program Praktik Kerja Lapangan (PKL) universitas. Seluruh pengalaman tersebut menjadi bekal berharga dalam membentuk karakter, memperluas wawasan, serta mempersiapkan diri untuk berkontribusi secara profesional maupun sosial.



KEMENTERIAN AGAMA RI
UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM MALANG
FAKULTAS SAINS DAN TEKNOLOGI
Jl. Gajayana No.50 Dinoyo Malang Telp. / Fax. (0341)558933

BUKTI KONSULTASI SKRIPSI

Nama : Khoiro Amalia
NIM : 210601110083
Fakultas / Jurusan : Sains dan Teknologi / Matematika
Judul Skripsi : Implementasi *Long Short Term Memory* pada Prediksi Harga Saham Perusahaan di Bidang Pangan
Pembimbing I : Hisyam Fahmi, M.Kom
Pembimbing II : Dr. Fachrur Rozi, M.Si.

No	Tanggal	Hal	Tanda Tangan
1.	23 September 2024	Konsultasi Topik dan Data	1.
2.	08 Oktober 2024	Konsultasi Bab I, II, dan III	2.
3.	09 Oktober 2024	Konsultasi Kajian Agama Bab I dan II	3.
4.	10 Oktober 2024	ACC Bab I, II, dan III	4.
5.	14 Oktober 2024	ACC Kajian Agama Bab I dan II	5.
6.	04 November 2024	Konsultasi Revisi Seminar Proposal	6.
7.	07 November 2024	Konsultasi Bab IV dan V	7.
8.	08 November 2024	Konsultasi Bab IV dan V	8.
9.	11 November 2024	Konsultasi Revisi Seminar Proposal	9.
10.	15 November 2024	Konsultasi Bab IV dan V	10.
11.	22 November 2024	ACC Bab IV dan V	11.
12.	25 November 2024	Konsultasi Revisi Seminar Proposal	12.
13.	26 November 2024	Konsultasi Revisi Seminar Proposal	13.
14.	27 November 2024	Konsultasi Revisi Seminar Proposal	14.
15.	28 November 2024	Konsultasi Revisi Seminar Proposal	15.



KEMENTERIAN AGAMA RI
UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM MALANG
FAKULTAS SAINS DAN TEKNOLOGI

Jl. Gajayana No.50 Dinoyo Malang Telp. / Fax. (0341)558933

16.	28 November 2024	Konsultasi Revisi Seminar Proposal	16.
17.	09 Desember 2024	Revisi Seminar Proposal	17.
18.	10 Januari 2025	Konsultasi Bab IV dan V	18.
19.	12 Februari 2025	Konsultasi Kajian Agama Bab IV	19.
20.	14 Februari 2025	ACC Kajian Agama Bab IV	20.
21.	17 februari 2025	ACC Seminar Hasil	21.
22.	14 Maret 2025	Konsultasi Revisi Seminar Hasil	22.
23.	18 Maret 2025	Konsultasi Revisi Seminar Hasil	23.
24.	7 Mei 2025	ACC Sidang Skripsi	24.
25.	23 Mei 2025	ACC Keseluruhan	25.

Malang, 23 Mei 2025

Mengetahui,

Ketua Program Studi Matematika



Dwi Susanti, M.Sc.

19741129 200012 2 005