

IMPLEMENTASI METODE *HYBRID AUTOREGRESSIVE INTEGRATED MOVING AVERAGE (ARIMA) – GENERALIZED AUTOREGRESSIVE CONDITIONAL HETEROSCEDASTICITY (GARCH)* DAN *LONG SHORT TERM MEMORY (LSTM)* PADA PERAMALAN HARGA SAHAM PT UNILEVER INDONESIA TBK.

Skripsi

Oleh

**YOLANDA RATULIMA BR SIMARMATA
NPM. 2217031167**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG**

2026

ABSTRACT

IMPLEMENTATION OF THE HYBRID AUTOREGRESSIVE INTEGRATED MOVING AVERAGE (ARIMA) – GENERALIZED AUTOREGRESSIVE CONDITIONAL HETEROSCEDASTICITY (GARCH) AND LONG SHORT TERM MEMORY (LSTM) METHOD IN STOCK PRICE FORECASTING OF PT UNILEVER INDONESIA TBK.

By

Yolanda Ratulima Br Simarmata

The fluctuation of stock prices is influenced by various complex economic factors, thus requiring an accurate forecasting method. This study employs a hybrid approach that combines Autoregressive Integrated Moving Average (ARIMA), Generalized Autoregressive Conditional Heteroscedasticity (GARCH), and Long Short-Term Memory (LSTM). ARIMA is used to model linear patterns, GARCH captures time-varying volatility, and LSTM learns non-linear patterns as well as long-term dependencies in time series data. The hybrid model is developed through two approaches, namely ARIMA–GARCH–E_LSTM, which uses ARIMA residuals modeled by GARCH as input to LSTM, and ARIMA–GARCH–EP_LSTM, which combines ARIMA predictions and GARCH residuals as inputs to LSTM. The data used in this study consist of stock closing prices from January 2020 to November 2025. The results show that the ARIMA–GARCH–EP_LSTM model provides the best forecasting performance compared to other models, as indicated by the smallest Kolmogorov–Smirnov (KS) statistic and its superior ability to follow data patterns. Therefore, the ARIMA–GARCH–EP_LSTM model is considered the most effective for stock price forecasting.

Keywords: stock price, prediction, forecasting, hybrid ARIMA-GARCH, hybrid ARIMA-GARCH-LSTM.

ABSTRAK

IMPLEMENTASI METODE *HYBRID AUTOREGRESSIVE INTEGRATED MOVING AVERAGE* (ARIMA) – *GENERALIZED AUTOREGRESSIVE CONDITIONAL HETEROSCEDASTICITY* (GARCH) DAN *LONG SHORT TERM MEMORY* (LSTM) PADA PERAMALAN HARGA SAHAM PT UNILEVER INDONESIA TBK.

Oleh

Yolanda Ratulima Br Simarmata

Pergerakan harga saham yang tidak stabil dipengaruhi oleh berbagai faktor ekonomi, sehingga diperlukan metode peramalan yang akurat. Penelitian ini menggunakan pendekatan *hybrid* yang menggabungkan *Autoregressive Integrated Moving Average* (ARIMA), *Generalized Autoregressive Conditional Heteroscedasticity* (GARCH), dan *Long Short-Term Memory* (LSTM). ARIMA digunakan untuk memodelkan pola linear, GARCH untuk menangkap perubahan volatilitas, dan LSTM untuk mempelajari pola non-linear serta hubungan jangka panjang dalam deret waktu. Model *hybrid* dibangun melalui dua pendekatan, yaitu ARIMA–GARCH–E.LSTM yang menggunakan residual ARIMA yang telah dimodelkan dengan GARCH sebagai *input* LSTM, serta ARIMA–GARCH–EP.LSTM yang menggabungkan prediksi ARIMA dan residual GARCH sebagai *input* LSTM. Data yang digunakan berupa harga penutupan saham periode Januari 2020 hingga November 2025. Hasil penelitian menunjukkan bahwa model ARIMA–GARCH–EP.LSTM memberikan kinerja peramalan terbaik dibandingkan model lainnya, ditunjukkan oleh nilai *Kolmogorov–Smirnov* (KS) terkecil dan kemampuan yang lebih baik dalam mengikuti pola data. Oleh karena itu, model ARIMA–GARCH–EP.LSTM dinilai paling efektif untuk peramalan harga saham.

Kata-kata kunci: harga saham, prediksi, peramalan, *hybrid* ARIMA-GARCH, *hybrid* ARIMA-GARCH-LSTM.

IMPLEMENTASI METODE *HYBRID AUTOREGRESSIVE INTEGRATED MOVING AVERAGE (ARIMA) – GENERALIZED AUTOREGRESSIVE CONDITIONAL HETEROSCEDASTICITY (GARCH) DAN LONG SHORT TERM MEMORY (LSTM)* PADA PERAMALAN HARGA SAHAM PT UNILEVER INDONESIA TBK.

YOLANDA RATULIMA BR SIMARMATA

Skripsi

Sebagai Salah Satu Syarat untuk Memperoleh Gelar
SARJANA MATEMATIKA

Pada

Jurusan Matematika

Fakultas Matematika dan Ilmu Pengetahuan Alam



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG**

2026

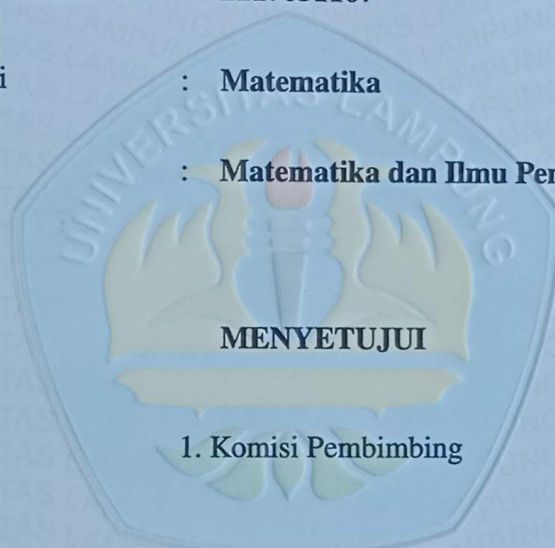
Judul Skripsi : **IMPLEMENTASI METODE *HYBRID* *AUTOREGRESSIVE INTEGRATED MOVING AVERAGE* (ARIMA)–*GENERALIZED AUTOREGRESSIVE CONDITIONAL HETEROSCEDASTICITY* (GARCH) DAN *LONG SHORT TERM MEMORY* (LSTM) PADA PERAMALAN HARGA SAHAM PT. UNILEVER INDONESIA TBK.**

Nama Mahasiswa : **Yolanda Ratulima Br Simarmata**

Nomor Pokok Mahasiswa : **2217031167**

Program Studi : **Matematika**

Fakultas : **Matematika dan Ilmu Pengetahuan Alam**



1. Komisi Pembimbing

Ir. Warsono, M.S., Ph.D.
NIP 196302161987031003

Riza Sawitri, S.Pd., M.Sc.
NIP 198905042024062001

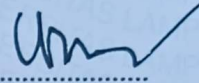
2. Ketua Jurusan Matematika

Dr. Aang Nuryaman, S.Si., M.Si.
NIP. 197403162005011001

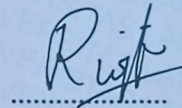
MENGESAHKAN

1. Tim Penguji

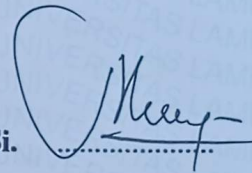
Ketua : Ir. Warsono, M.S., Ph.D.



Sekretaris : Riza Sawitri, S.Pd., M.Sc.



**Penguji
Bukan Pembimbing : Dr. Aang Nuryaman, S.Si., M.Si.**



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Dr. Eng Heri Satria, S.Si., M.Si.

NIP. 197110012005011002

Tanggal Lulus Ujian Skripsi: 17 April 2026

PERNYATAAN SKRIPSI MAHASISWA

Yang bertanda tangan di bawah ini:

Nama : **Yolanda Ratulima Br Simarmata**
Nomor Pokok Mahasiswa : **2217031167**
Jurusan : **Matematika**
Judul Skripsi : **Implementasi Metode *Hybrid Autoregressive Integrated Moving Average (ARIMA)–Generalized Autoregressive Conditional Heteroscedasticity (GARCH)* dan *Long Short Term Memory (LSTM)* pada Peramalan Harga Saham PT Unilever Indonesia Tbk.**

Dengan ini menyatakan bahwa skripsi ini adalah hasil pekerjaan saya sendiri. Apabila kemudian hari terbukti bahwa skripsi ini merupakan hasil salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan akademik yang berlaku.

Bandar Lampung, 17 April 2026

Penulis



Yolanda Ratulima Br Simarmata

RIWAYAT HIDUP

Penulis bernama lengkap Yolanda Ratulima Br Simarmata yang lahir di Balipapan, 14 Mei 2004. Penulis merupakan anak kedua dari tiga bersaudara, pasangan Bapak Eslon Simarmata dan Ibu Nelly Evelyn Napitupulu. Kedua saudara penulis bernama Rionaldo Saroha Simarmata dan Lily Aulia Evelyn Br Simarmata.

Awal pendidikan penulis dimulai dari TK Tunas Bangsa Silalahi I, Dairi, Sumatera Utara pada tahun 2009 - 2010. Selanjutnya, penulis melanjutkan pendidikan sekolah dasar di SD Negeri 1 Silahisabungan, Dairi, Sumatera Utara pada tahun 2010 - 2016. Selanjutnya, penulis menempuh pendidikan sekolah menengah pertama di SMP Negeri 1 Silahisabungan, Dairi, Sumatera Utara pada tahun 2016 - 2019. Dan dilanjutkan menempuh pendidikan sekolah menengah atas di SMA Negeri 1 Silahisabungan, Dairi, Sumatera Utara pada tahun 2019 - 2022.

Pada tahun 2022 penulis melanjutkan pendidikan ke perguruan tinggi di Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung, melalui jalur SBMPTN. Selama menjadi mahasiswa, penulis pernah menjadi bagian dari organisasi kemahasiswaan yaitu UKM-U Saintek pada tahun 2024, sebagai Staf Manajemen Sumber Daya (MSD).

Untuk menerapkan ilmu di dunia kerja, pada bulan Desember 2024 hingga Februari 2025, penulis melaksanakan Kerja Praktik (KP) di PLN Unit Induk Distribusi (UID) Lampung. Kemudian, sebagai bentuk pengabdian kepada masyarakat, pada bulan Juni hingga Agustus 2025 penulis melaksanakan Kuliah Kerja Nyata (KKN) di Kelurahan Korpri Raya, Kecamatan Sukarame, Kota Bandar Lampung.

KATA INSPIRASI

”Karena masa depan sungguh ada, dan harapanmu tidak akan hilang”
(Amsal 23 : 18)

”Jatuh 7 kali, bangkit 8 kali”
Nana Korobi Ya Oki

PERSEMBAHAN

Puji syukur kepada Tuhan Yang Maha Esa atas berkat dan karunia-Nya sehingga skripsi ini dapat terselesaikan dengan baik dan tepat pada waktunya. Dengan rasa syukur dan Bahagia, saya persembahkan rasa terimakasih saya kepada:

Ayah dan Ibuku Tercinta

Terimakasih kepada orang tuaku atas segala pengorbanan, motivasi, doa, dan dukungannya selama ini. Terimakasih telah memberikan pelajaran berharga kepada anakmu ini tentang makna perjalanan hidup yang sebenarnya sehingga kelak bisa menjadi orang yang bermanfaat bagi banyak orang.

Dosen Pembimbing dan Pembahas

Terimakasih kepada dosen pembimbing dan pembahas yang sudah sangat membantu, memberikan motivasi, memberikan arahan serta ilmu yang berharga.

Sahabat-sahabatku

Terimakasih kepada semua orang-orang baik yang telah memberikan pengalaman, semangat, motivasinya, serta doa-doanya dan senantiasa memberikan dukungan dalam hal apapun.

Almamater Tercinta

Universitas Lampung

SANWACANA

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas berkat dan karunia-Nya sehingga penulis dapat menyelesaikan skripsi ini yang berjudul "Implementasi Metode *Hybrid Autoregressive Integrated Moving Average* (ARIMA)- *Generalized Autoregressive Conditional Heteroscedasticity* (GARCH) dan *Long Short Term Memory* (LSTM) pada Peramalan Harga Saham PT Unilever Indonesia Tbk." dengan baik dan lancar serta tepat pada waktu yang telah ditentukan.

Dalam proses penyusunan skripsi ini, banyak pihak yang telah membantu memberikan bimbingan, dukungan, arahan, motivasi serta saran sehingga skripsi ini dapat terselesaikan. Oleh karena itu, dalam kesempatan ini penulis mengucapkan terimakasih kepada:

1. Bapak Ir. Warsono, M.S., Ph.D., selaku Pembimbing I yang telah banyak meluangkan waktunya untuk memberikan arahan, bimbingan, motivasi, saran serta dukungan kepada penulis sehingga dapat menyelesaikan skripsi ini.
2. Ibu Riza Sawitri, S.Pd., M.Sc., selaku Pembimbing II yang telah memberikan arahan, bimbingan dan dukungan kepada penulis sehingga dapat menyelesaikan skripsi ini.
3. Bapak Dr. Aang Nuryaman, S.Si., M.Si., selaku Ketua Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung sekaligus Penguji, atas kesediaannya memberikan kritik, saran, dan evaluasi kepada penulis sehingga penelitian ini menjadi lebih baik.
4. Bapak Dr. Subian Saidi, S.Si., M.Si., selaku dosen pembimbing akademik.
5. Seluruh dosen, staff, dan karyawan Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.

6. Bapak, Ibu, Abang dan Adik yang senantiasa memberikan doa dan motivasi kepada penulis.
7. Sahabat penulis sejak SMA diantaranya Nurlela dan Rodo yang telah banyak memberikan dukungan yang tak terlihat dan saling mendoakan satu sama lain.
8. Teman seperjuangan diperkuliahan dari awal kelas C, diantaranya Lusi dan Anisa. Terimakasih sudah memberikan canda tawa, dukungan, motivasi dan selalu ada di setiap momen suka dan duka selama berada di bangku perkuliahan.
9. Teman seperjuangan sirkel Pomp. Op Asep , diantaranya Ayu, Lusi, Winne, Gracia, Mega, Elsa, Rafael dan Agus. Terimakasih atas kebersamaan yang hangat dan menyenangkan, serta atas saling menguatkan satu sama lain sehingga kita dapat melalui masa perkuliahan ini dengan penuh semangat.
10. Teruntuk teman-temanku Elisabeth, Teli, Rupa dan Lia. Terimakasih sudah menemani perjalanan kuliah ini sejak awal di Unila. Terimakasih atas doa dan semangat yang selalu kalian berikan, bersyukur dapat melalui masa perkuliahan bersama kalian.
11. Rekan seperjuangan skripsi sepembimbingan Pak War dan Bu Riza yang selalu menemani, membantu, dan saling memberikan dukungan satu sama lain. Terimakasih sudah bersama-sama menyelesaikan ini.
12. Teman-teman seperjuangan Jurusan Matematika angkatan 2022.

Semoga skripsi ini dapat bermanfaat bagi kita semua. Penulis menyadari bahwa skripsi ini masih jauh dari sempurna, sehingga penulis mengharapkan kritik dan saran yang membangun untuk menjadikan skripsi ini lebih baik lagi.

Bandar Lampung, 17 April 2026

Yolanda Ratulima Br Simarmata

DAFTAR ISI

	Halaman
DAFTAR ISI	xiii
DAFTAR TABEL	xvi
DAFTAR GAMBAR	1
I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian	3
1.4 Manfaat Penelitian	4
II TINJAUAN PUSTAKA	5
2.1 Penelitian Terkait	5
2.2 Deret Waktu	7
2.2.1 Analisis Deret Waktu	7
2.2.2 Peramalan	7
2.3 Harga Saham	8
2.4 <i>Return</i>	9
2.5 Model Deret Waktu Univariat	9
2.5.1 Model <i>Autoregressive</i> (AR)	9
2.5.2 Model <i>Moving Average</i> (MA)	10
2.5.3 Model <i>Autoregressive Moving Average</i> (ARMA)	10
2.5.4 Model <i>Autoregressive Integrated Moving Average</i> (ARIMA)	11
2.6 Model Volatilitas (Model <i>Conditional Heteroscedastic</i>)	12
2.6.1 Volatilitas	12
2.6.2 Model ARCH (<i>Autoregressive Conditional Heteroscedasticity</i>)	13
2.6.3 Model GARCH (<i>Generalized Autoregressive Conditional Heteroskedasticity</i>)	14

2.6.4	Pengujian Efek ARCH/ GARCH	14
2.7	Stasioneritas Data	15
2.8	Residual <i>White Noise</i>	17
2.9	<i>Machine Learning</i>	18
2.9.1	<i>Scaling Data</i>	19
2.9.2	<i>Unscaling Data</i>	19
2.9.3	Fungsi Aktivasi	20
2.10	<i>Deep Learning</i>	21
2.10.1	<i>Recurrent Neural Network (RNN)</i>	22
2.10.2	<i>Long Short-Term Memory (LSTM)</i>	22
2.10.3	<i>Hyperparameter</i>	26
2.11	<i>Hybrid ARIMA-GARCH-LSTM</i>	26
2.12	Evaluasi Model	29
2.13	<i>Goodness of Fit</i>	30
III	METODE PENELITIAN	32
3.1	Waktu dan Tempat Penelitian	32
3.2	Data dan Alat Penelitian	33
3.3	Metode Penelitian	34
IV	HASIL DAN PEMBAHASAN	37
4.1	Proses <i>Input</i> dan Visualisasi Data	37
4.2	Skema Pembagian Data	38
4.3	Proses Model ARIMA-GARCH	38
4.3.1	<i>Preprocessing Data</i>	39
4.3.2	Uji Stasioner	39
4.3.3	Identifikasi Model ARIMA	40
4.3.4	Estimasi Parameter dan Uji Diagnostik Model	41
4.3.5	Pengujian Efek ARCH dalam Residual Model ARIMA	43
4.3.6	Identifikasi, Estimasi, dan Pemilihan Model GARCH	44
4.3.7	Pengujian Efek ARCH pada model GARCH	45
4.3.8	Prediksi Model ARIMA (2,1,1)	45
4.3.9	Residual Model GARCH (1,1)	47
4.3.10	Peramalan Model ARIMA(2,1,1)–GARCH (1,1)	49
4.4	Prediksi dan Peramalan Model <i>Hybrid ARIMA-GARCH-E.LSTM</i>	51
4.4.1	<i>Scaling Data</i> Residual Model GARCH	52
4.4.2	<i>Hyperparameter Tuning</i> Model E.LSTM	52

4.4.3	Membangun Model E_LSTM	53
4.4.4	Prediksi Model E_LSTM	53
4.4.5	Peramalan Model E_LSTM	55
4.4.6	Prediksi Model <i>Hybrid</i> ARIMA-GARCH-E_LSTM	57
4.4.7	Peramalan Model <i>Hybrid</i> ARIMA-GARCH-E_LSTM	58
4.5	Prediksi dan Peramalan Model <i>Hybrid</i> ARIMA-GARCH-EP_LSTM	61
4.5.1	<i>Scaling</i> Data Prediksi Model ARIMA	61
4.5.2	<i>Hyperparameter Tuning</i> Model P_LSTM	62
4.5.3	Membangun Model P_LSTM	62
4.5.4	Prediksi Model P_LSTM	63
4.5.5	Peramalan Model P_LSTM	64
4.5.6	Prediksi Model <i>Hybrid</i> ARIMA-GARCH-EP_LSTM	65
4.5.7	Peramalan Model <i>Hybrid</i> ARIMA-GARCH-EP_LSTM	67
4.6	<i>Goodness of Fit</i>	69
V	KESIMPULAN DAN SARAN	71
5.1	Kesimpulan	71
	DAFTAR PUSTAKA	73

DAFTAR TABEL

Tabel	Halaman
1. Penelitian Terdahulu	5
2. Data Harga Saham PT Unilever Indonesia Tbk.	33
3. Data <i>Input</i>	37
4. Statistika Deskriptif Variabel <i>Close</i>	37
5. Hasil Pengecekan <i>Missing Value</i>	39
6. Hasil Uji <i>Augmented Dickey-Fuller (ADF)</i>	39
7. Hasil Uji <i>Augmented Dickey-Fuller (ADF)</i> Setelah <i>Differencing</i>	40
8. <i>Diagnostic Checking</i> Dugaan Model ARIMA	42
9. Pemilihan Model ARIMA Terbaik	42
10. Uji Efek ARCH pada Model ARIMA	43
11. <i>Diagnostic Checking</i> Dugaan Model GARCH	44
12. Pemilihan Model GARCH Terbaik	45
13. Uji Efek ARCH Pada Model GARCH	45
14. Hasil Prediksi Model ARIMA	46
15. Evaluasi Model ARIMA	47
16. Residual Model GARCH	47
17. Hasil Uji <i>Ljung-Box</i>	48
18. Hasil Peramalan Model ARIMA–GARCH	49
19. Evaluasi <i>Train-Test</i>	50
20. Evaluasi <i>Forecast</i>	50
21. Evaluasi <i>Train-Test-Forecast</i>	50
22. <i>Hyperparameter Tuning</i> Model E.LSTM	53
23. Hasil <i>Hyperparameter Tuning</i> Model E.LSTM	53
24. Prediksi Model <i>E.LSTM</i>	54

25.	Hasil Prediksi Model E_LSTM	54
26.	Hasil Evaluasi Model	55
27.	Hasil Peramalan Residual Model E_LSTM	56
28.	Hasil Prediksi Model	57
29.	Evaluasi <i>Hybrid</i> ARIMA–GARCH–E_LSTM	58
30.	Hasil Peramalan Model <i>Hybrid</i> ARIMA–GARCH–E-LSTM	58
31.	Evaluasi <i>Train-Test</i>	59
32.	Evaluasi <i>Forecast</i>	59
33.	Evaluasi <i>Train-Test-Forecast</i>	60
34.	<i>Hyperparameter Tuning</i> Model P_LSTM	62
35.	Hasil <i>Hyperparameter Tuning</i> Model P_LSTM	62
36.	Prediksi Model P_LSTM	63
37.	Hasil Prediksi Model P_LSTM	63
38.	Evaluasi Model P_LSTM	64
39.	Hasil Peramalan Model P_LSTM	65
40.	Hasil Prediksi Model <i>Hybrid</i> ARIMA–GARCH–EP_LSTM	66
41.	Evaluasi Model <i>Hybrid</i> ARIMA-GARCH-EP_LSTM	67
42.	Hasil Peramalan Model <i>Hybrid</i> ARIMA-GARCH-EP LSTM	67
43.	Evaluasi <i>Train-Test</i>	68
44.	Evaluasi <i>Forecast</i>	68
45.	Evaluasi <i>Train-Test-Forecast</i>	69
46.	<i>Goodness of Fit</i> Hasil Peramalan	70

DAFTAR GAMBAR

Gambar	Halaman
1. Arsitektur LSTM (Yanti <i>et al.</i> , 2024).	23
2. <i>Flowchart</i> Model <i>Hybrid</i> ARIMA-GARCH dan E_LSTM	35
3. <i>Flowchart</i> Model <i>Hybrid</i> ARIMA-GARCH dan EP_LSTM	36
4. Plot Data Historis Penutupan Harga Saham PT Unilever Indonesia Tbk.	38
5. Plot Data Historis Penutupan Harga Saham Setelah <i>Differencing</i>	40
6. Plot ACF dan Plot PACF.	41
7. Plot Prediksi model ARIMA.	46
8. Plot Residual Model GARCH.	48
9. Plot Peramalan Model ARIMA-GARCH.	49
10. Plot Bagian <i>Forecasting</i> Model ARIMA-GARCH.	51
11. Plot Residual Prediksi Model E_LSTM.	55
12. Plot Peramalan Residual E_LSTM.	56
13. Plot Hasil Prediksi <i>Hybrid</i> ARIMA-GARCH-E_LSTM.	57
14. Plot Peramalan Model <i>Hybrid</i> ARIMA-GARCH-E_LSTM.	59
15. Plot Bagian <i>Forecasting</i> Model <i>Hybrid</i> ARIMA-GARCH-E_LSTM.	60
16. Plot Prediksi Model P_LSTM.	64
17. Plot Peramalan Model P_LSTM.	65
18. Plot Hasil Prediksi <i>Hybrid</i> ARIMA-GARCH-EP_LSTM.	66
19. Plot Peramalan Model <i>Hybrid</i> ARIMA-GARCH-EP_LSTM.	68
20. Plot Bagian <i>Forecasting Hybrid</i> ARIMA-GARCH-EP_LSTM.	69

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Pasar modal Indonesia merupakan salah satu instrumen investasi yang memiliki peranan penting dalam mendukung pertumbuhan ekonomi nasional. Pergerakan harga saham yang dinamis menyebabkan tingkat risiko investasi semakin tinggi, terutama pada saham-saham berkapitalisasi besar, seperti PT Unilever Indonesia Tbk., yang merupakan salah satu saham *blue-chip* dengan tingkat likuiditas tinggi di Bursa Efek Indonesia. Salah satu aspek penting dalam analisis risiko pasar adalah volatilitas harga saham, karena volatilitas yang tinggi menandakan ketidakstabilan harga dan berpotensi meningkatkan risiko kerugian bagi investor.

Data keuangan seperti *return* saham diketahui memiliki karakteristik statistik yang kompleks dan sering menyimpang dari asumsi model klasik, seperti normalitas dan linearitas. Data *return* saham umumnya memperlihatkan fenomena *volatility clustering*, yaitu kecenderungan periode volatilitas tinggi diikuti oleh volatilitas tinggi lainnya, serta periode volatilitas rendah yang diikuti oleh volatilitas rendah. Selain itu, distribusi *return* saham cenderung bersifat *leptokurtic* dan menunjukkan asimetri dalam merespons guncangan positif maupun negatif. Penelitian empiris juga mengonfirmasi adanya sifat *long memory*, yakni kondisi ketika guncangan pada masa lalu tetap memberikan pengaruh jangka panjang terhadap dinamika volatilitas (Rafulta *et al.*, 2025). Kompleksitas karakteristik tersebut menuntut penggunaan pendekatan pemodelan yang mampu menangkap dinamika rata-rata, variansi bersyarat, serta pola nonlinier secara simultan.

Dalam konteks tersebut, model *Autoregressive Integrated Moving Average* (ARIMA) yang diperkenalkan oleh Box & Jenkins (1976) serta model *Generalized Autoregressive Conditional Heteroskedasticity* (GARCH) yang diperkenalkan oleh Bollerslev (1986) menjadi pendekatan statistik yang banyak digunakan

dalam analisis deret waktu keuangan. Model ARIMA efektif dalam memodelkan komponen rata-rata (*mean process*), sedangkan GARCH mampu menangkap dinamika volatilitas bersyarat dan fenomena *volatility clustering*. Namun demikian, model GARCH standar masih memiliki keterbatasan dalam merepresentasikan struktur memori panjang dan pola nonlinier yang kompleks pada volatilitas harga saham (Ding *et al.*, 1993).

Seiring dengan perkembangan teknologi dan metode analitik, pendekatan *machine learning* dan *deep learning* semakin banyak dimanfaatkan dalam peramalan deret waktu keuangan. Salah satu metode *deep learning* yang banyak digunakan adalah *Long Short-Term Memory* (LSTM) yang diperkenalkan oleh Hochreiter dan Schmidhuber (1997). LSTM dikembangkan sebagai solusi atas permasalahan *vanishing gradient* yang sering terjadi pada jaringan *Recurrent Neural Network* (RNN) konvensional, sehingga mampu mempertahankan dan mempelajari informasi dalam jangka panjang secara lebih efektif. Namun demikian, penggunaan LSTM saja masih memiliki beberapa keterbatasan, antara lain kebutuhan data dalam jumlah besar, sensitivitas terhadap proses *hyperparameter tuning*, serta potensi terjadinya *overfitting*.

Untuk mengatasi keterbatasan kedua pendekatan tersebut, sejumlah penelitian terbaru mengusulkan penggunaan metode *hybrid* yang mengombinasikan model statistik dan *deep learning*. Salah satu pendekatan yang banyak dikembangkan adalah *hybrid ARIMA–GARCH–LSTM*, yaitu model yang mengintegrasikan kemampuan ARIMA dalam memodelkan komponen rata-rata, GARCH dalam menangkap dinamika volatilitas linear, serta LSTM dalam mempelajari pola nonlinier dan dependensi jangka panjang pada deret waktu keuangan. Integrasi ketiga pendekatan ini diharapkan mampu meningkatkan akurasi prediksi volatilitas dibandingkan dengan penggunaan model ARIMA, GARCH, atau LSTM secara terpisah.

PT Unilever Indonesia Tbk. dipilih sebagai objek penelitian karena perusahaan ini memiliki kapitalisasi pasar yang besar, serta termasuk dalam indeks LQ45 dan IDX30 yang merepresentasikan saham-saham dengan tingkat likuiditas tinggi dan kapitalisasi pasar terbesar di Bursa Efek Indonesia. Selain itu, harga saham perusahaan menunjukkan pola pergerakan yang relatif stabil tetapi tetap dipengaruhi oleh dinamika pasar, kondisi makroekonomi, dan sentimen global.

Dalam penelitian ini, data harga penutupan (*closing price*) digunakan sebagai variabel utama karena harga penutupan mencerminkan harga kesepakatan akhir yang terbentuk setelah seluruh aktivitas perdagangan pada hari tersebut, sehingga dianggap dapat merepresentasikan nilai saham pada periode tersebut (Jogiyanto, 2017). Dengan mempertimbangkan karakteristik tersebut, penelitian ini bertujuan untuk menganalisis serta memodelkan harga saham PT Unilever Indonesia Tbk. menggunakan metode *hybrid* ARIMA–GARCH dan LSTM, sehingga diharapkan dapat memberikan gambaran yang lebih komprehensif dan akurat mengenai risiko pasar.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka rumusan masalah yang dibahas dalam penelitian ini, yaitu sebagai berikut:

1. Bagaimana cara membangun model *hybrid* ARIMA–GARCH, model *hybrid* ARIMA–GARCH–E_LSTM, dan model *hybrid* ARIMA–GARCH–EP_LSTM untuk peramalan harga saham PT Unilever Indonesia Tbk.?
2. Bagaimana hasil peramalan harga saham PT Unilever Indonesia Tbk. dengan model *hybrid* ARIMA–GARCH, model *hybrid* ARIMA–GARCH–E_LSTM, serta model *hybrid* ARIMA–GARCH–EP_LSTM?
3. Bagaimana perbandingan hasil peramalan harga saham PT Unilever Indonesia Tbk. dengan model *hybrid* ARIMA–GARCH, model *hybrid* ARIMA–GARCH–E_LSTM, serta model *hybrid* ARIMA–GARCH–EP_LSTM?

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini, yaitu sebagai berikut:

1. Membangun model *hybrid* ARIMA–GARCH, model *hybrid* ARIMA–GARCH–E_LSTM, dan model *hybrid* ARIMA–GARCH–EP_LSTM dengan data *input* prediksi ARIMA dan

residual GARCH untuk meramalkan harga saham PT Unilever Indonesia Tbk.

2. Mengetahui hasil peramalan harga saham PT Unilever Indonesia Tbk. dengan metode *hybrid* yang dihasilkan oleh model ARIMA–GARCH, model *hybrid* ARIMA–GARCH–E_LSTM, dan model *hybrid* ARIMA–GARCH–EP_LSTM.
3. Membandingkan kinerja model *hybrid* ARIMA–GARCH, model *hybrid* ARIMA–GARCH–E_LSTM, dan model *hybrid* ARIMA–GARCH–EP_LSTM dalam melakukan peramalan harga saham PT Unilever Indonesia Tbk.

1.4 Manfaat Penelitian

Adapun manfaat dari penelitian ini, yaitu sebagai berikut:

1. Memberikan pemahaman mengenai peramalan harga saham menggunakan metode *hybrid* ARIMA–GARCH dan LSTM.
2. Menghasilkan prediksi harga saham yang lebih akurat untuk mendukung pengambilan keputusan investasi.
3. Menjadi sumber dan referensi bagi peneliti selanjutnya dengan menggunakan metode *hybrid* ARIMA–GARCH dan LSTM.

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Penelitian terkait digunakan sebagai referensi untuk memperdalam pemahaman terhadap topik yang dikaji, yang ringkasannya disajikan pada Tabel 1 sebagai berikut:

Tabel 1. Penelitian Terdahulu

No	Penelitian	Data	Metode	MAPE
1	Peramalan Harga Saham Syariah PT Unilever Indonesia Tbk. Menggunakan GARCH (Gunawan dan Darwin, 2023)	Data harga penutupan saham harian UNVR (Januari 2018–Juni 2022)	GARCH	ARIMA: 1,69% GARCH: 1,91%
2	Penerapan Model ARIMA–GARCH untuk Peramalan Harga Emas Dunia (Beeg <i>et al.</i> , 2024)	Data harian penutupan harga emas dunia (USD) (2 Januari–18 Oktober 2023)	ARIMA–GARCH	Hybrid: 0,68%
3	<i>Application of the ARIMA–GARCH Model for Forecasting Indonesia’s Monthly Inflation Rate</i> (Anisa & Yulianto, 2025)	Data inflasi bulanan di Indonesia (Maret 2007–Oktober 2023)	ARIMA–GARCH	Hybrid: 17,78%
4	<i>A Deep LSTM Network for the Spanish Electricity Consumption Forecasting</i> (Torres <i>et al.</i> , 2022)	Kumpulan data konsumsi listrik (Januari 2007–Juni 2016) di Spanyol	LSTM	LSTM+ CVOA: 1,58% LSTM+ Random: 1,44%

Beberapa penelitian dijadikan sebagai bahan referensi dalam penelitian ini. Salah satunya adalah penelitian oleh Gunawan dan Darwin (2023), yang menganalisis peramalan saham syariah PT Unilever Indonesia Tbk. menggunakan metode GARCH dengan data harga penutupan harian dari Januari 2018 hingga Juni 2022. Pada penelitian tersebut, model ARIMA mampu memberikan tingkat akurasi yang sangat baik karena menghasilkan nilai MAPE yang relatif kecil. Namun, model ini masih mengindikasikan adanya heteroskedastisitas pada data, sehingga proses peramalan dilanjutkan dengan menggunakan model GARCH untuk mengatasi ketidakstabilan varians tersebut. Hasil peramalan dengan model GARCH memperoleh nilai MAPE sebesar 1,91%, yang menunjukkan akurasi yang sangat baik karena berada di bawah batas 10%.

Penelitian lain yang menjadi bahan referensi adalah studi yang dilakukan oleh Beeg *et al.* (2024) yang berfokus pada peramalan harga emas dunia dengan pendekatan *hybrid* ARIMA–GARCH. Dalam penelitian tersebut, digunakan data harga emas dunia dalam satuan dolar (USD) dari 2 Januari hingga 18 Oktober 2023. Hasil analisis menunjukkan bahwa model *hybrid* ARIMA (6,1,6)–GARCH (6,0) merupakan model yang paling optimal dengan nilai MAPE sebesar 0,68%. Temuan ini mengindikasikan bahwa kombinasi kedua model mampu menangkap pola tren dan volatilitas secara lebih komprehensif, sehingga menghasilkan tingkat akurasi peramalan yang sangat tinggi.

Kemudian, penelitian yang dilakukan oleh Anisa dan Yulianto (2025) membahas peramalan laju inflasi bulanan di Indonesia dengan pendekatan *hybrid* ARIMA–GARCH. Pada penelitian ini digunakan data inflasi bulanan di Indonesia dari Maret 2007 hingga Oktober 2023. Metode yang digunakan sama dengan penelitian sebelumnya, yaitu ARIMA–GARCH, dan diperoleh nilai MAPE pada metode *hybrid* sebesar 17,78%.

Penelitian oleh J. F. Torres *et al.* (2022) mengkaji peramalan konsumsi listrik menggunakan data dari Januari 2007 hingga Juni 2016. Metode yang digunakan adalah *Long Short-Term Memory* (LSTM) dengan dua pendekatan optimasi, yaitu *Random Search* dan CVOA (*Coronavirus Optimization Algorithm*). *Random Search* memilih kombinasi *hyperparameter* secara acak, sedangkan CVOA merupakan algoritma metaheuristik untuk menemukan konfigurasi model yang optimal. Hasil penelitian menunjukkan bahwa model LSTM + CVOA menghasilkan nilai MAPE sebesar 1,58%, sedangkan model LSTM + *Random Search* memperoleh nilai

MAPE sebesar 1,44%. Kedua nilai tersebut berada di bawah 10%, sehingga menunjukkan tingkat akurasi peramalan yang sangat baik.

2.2 Deret Waktu

2.2.1 Analisis Deret Waktu

Analisis deret waktu merupakan pendekatan yang mempelajari keterkaitan nilai suatu variabel dari satu periode waktu ke periode berikutnya. Tujuan utama analisis ini adalah mengidentifikasi pola atau model yang dapat menggambarkan hubungan terstruktur antarperiode, sehingga dapat digunakan untuk mengevaluasi dinamika data dan menghasilkan prediksi yang lebih akurat terhadap nilai di masa mendatang (Pankratz, 1991).

Menurut Hanke & Wichern (2005), pola dalam data deret waktu umumnya dapat dikelompokkan ke dalam empat kategori berikut:

1. Pola horizontal, yaitu kondisi ketika data berfluktuasi di sekitar nilai rata-rata yang relatif konstan tanpa adanya kecenderungan naik atau turun secara jelas.
2. Pola musiman, yakni variasi data yang berulang secara teratur dalam periode tertentu, misalnya kuartalan, bulanan, mingguan, atau harian, dan umumnya berlangsung dalam siklus satu tahun.
3. Pola siklik, yaitu gerakan naik turun data yang terjadi dalam jangka panjang dan cenderung membentuk gelombang siklus di sekitar tren utama.
4. Pola tren, yaitu kecenderungan data menunjukkan arah tertentu dalam jangka panjang, baik berupa pergerakan naik maupun penurunan secara konsisten.

2.2.2 Peramalan

Peramalan merupakan proses sistematis yang digunakan untuk memperkirakan kondisi atau nilai pada masa mendatang dengan memanfaatkan informasi historis sebagai dasar prediksi. Secara umum, peramalan dibedakan menjadi jangka pendek, jangka menengah, dan jangka panjang. Peramalan jangka pendek biasanya mencakup rentang waktu beberapa hari hingga beberapa bulan, sementara jangka

menengah umumnya mencakup periode 1–2 tahun, dan jangka panjang mencakup proyeksi dalam kurun waktu beberapa tahun (Montgomery *et al.*, 2015).

Metode peramalan dapat dikategorikan ke dalam dua kelompok utama. Pertama, metode peramalan kualitatif merupakan metode peramalan yang tidak bergantung pada data numerik historis, melainkan menggunakan penilaian, pengalaman, dan intuisi para ahli. Metode ini umumnya digunakan ketika data masa lalu tidak tersedia atau jumlahnya sangat terbatas, serta ketika faktor non-kuantitatif, seperti perubahan kebijakan, kondisi sosial, atau perkembangan teknologi, memiliki pengaruh yang signifikan terhadap objek yang diramalkan dan tujuan analisis yang ingin dicapai. Kedua, metode peramalan kuantitatif adalah metode peramalan yang menggunakan data numerik historis dan pendekatan matematis atau statistik untuk memodelkan hubungan antarvariabel atau pola dalam data. Metode ini berasumsi bahwa pola yang terjadi di masa lalu cenderung berlanjut pada periode mendatang, sehingga hasil peramalan diperoleh secara lebih objektif dan terukur (Montgomery *et al.*, 2015).

2.3 Harga Saham

Sebelum mengambil keputusan investasi, seorang investor perlu memahami terlebih dahulu bagaimana tren pergerakan harga saham perusahaan. Darmadji & Fakhruddin (2012) menjelaskan bahwa harga saham merupakan nilai yang terbentuk di bursa efek pada waktu tertentu, yang sifatnya sangat fluktuatif karena dapat berubah dengan cepat. Dinamika harga tersebut pada dasarnya dipengaruhi oleh kekuatan permintaan dan penawaran di pasar. Dengan demikian, memahami pola perubahan harga saham menjadi aspek penting bagi investor untuk menilai prospek dan risiko suatu perusahaan.

Harga penutupan (*close price*) merupakan nilai saham pada akhir sesi perdagangan dan menjadi indikator utama dalam melihat kondisi pasar pada hari tersebut. Menurut (Jogiyanto, 2017), harga penutupan menggambarkan keseimbangan akhir antara penawaran dan permintaan saham sehingga sering dijadikan acuan utama dalam analisis teknikal. Oleh karena itu, harga penutupan memiliki peran penting dan kerap digunakan sebagai dasar dalam berbagai model peramalan harga saham.

2.4 Return

Return saham merupakan tingkat keuntungan yang diperoleh investor selama periode tertentu sebagai hasil dari investasi yang dilakukan. *Return* dapat berasal dari perubahan harga saham (*capital gain*) maupun dari pendapatan berupa dividen. *Return* menjadi indikator utama yang digunakan dalam menilai kinerja suatu investasi karena mencerminkan seberapa besar keuntungan atau kerugian yang dialami investor. Menurut (Jogiyanto, 2017), *return* merupakan tingkat keuntungan yang diterima investor yang dapat dihitung berdasarkan data historis maupun ekspektasi masa depan. Dalam analisis keuangan modern, *return* historis sering digunakan untuk mengukur volatilitas, risiko, dan sebagai dasar pemodelan deret waktu, seperti ARIMA dan GARCH.

2.5 Model Deret Waktu Univariat

2.5.1 Model Autoregressive (AR)

Model *Autoregressive* (AR) merupakan pendekatan peramalan yang memanfaatkan nilai-nilai masa lalu suatu variabel untuk memprediksi nilai di masa mendatang. Konsep utamanya adalah bahwa observasi sebelumnya memiliki pengaruh terhadap nilai saat ini, sehingga informasi historis dapat digunakan untuk memodelkan perilaku data di masa depan (Box & Jenkins, 1970). Dengan demikian, model AR berasumsi bahwa terdapat hubungan yang kuat antara data pada periode sebelumnya dengan periode berikutnya. Bentuk umum model AR dituliskan pada Persamaan (2.1) sebagai berikut (Adhikari & Agrawal, 2013):

$$Y_t = \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \dots + \phi_p Y_{t-p} + \varepsilon_t \quad (2.1)$$

keterangan:

- t : indeks waktu
- Y_t : nilai pengamatan pada waktu ke- t
- Y_{t-1}, \dots, Y_{t-p} : nilai pengamatan masa lalu yang berhubungan
- ϕ_1, \dots, ϕ_p : koefisien atau parameter dari model AR
- ε_t : nilai residual pada waktu ke- t

2.5.2 Model *Moving Average* (MA)

Model *Moving Average* (MA) merupakan model deret waktu yang menyatakan bahwa nilai suatu variabel pada periode tertentu dipengaruhi oleh kombinasi linear dari komponen galat (*error terms*) pada periode saat ini dan beberapa periode sebelumnya. Dengan demikian, struktur ketergantungan dalam model MA dibentuk oleh *shock* acak masa lalu, sehingga model ini efektif dalam merepresentasikan pola fluktuasi jangka pendek yang tidak sepenuhnya dapat dijelaskan oleh model AR (Box & Jenkins, 1970). Pendekatan ini mengasumsikan bahwa *shock* atau gangguan pada waktu sebelumnya masih memiliki pengaruh pada nilai observasi saat ini. Bentuk umum model MA dituliskan pada Persamaan (2.2) sebagai berikut (Adhikari & Agrawal, 2013):

$$Y_t = \varepsilon_t - \theta_1\varepsilon_{t-1} - \theta_2\varepsilon_{t-2} - \dots - \theta_q\varepsilon_{t-q} \quad (2.2)$$

keterangan:

- Y_t : nilai pengamatan pada waktu ke- t
 $\theta_1, \dots, \theta_q$: koefisien atau parameter dari model MA
 ε_t : nilai residual pada waktu ke- t
 $\varepsilon_{t-1}, \dots, \varepsilon_{t-q}$: nilai residual pada waktu ke- $(t-1), \dots, (t-q)$

2.5.3 Model *Autoregressive Moving Average* (ARMA)

Model *Autoregressive Moving Average* (ARMA) merupakan gabungan dari dua komponen utama, yaitu model AR dan model MA. Model ini memanfaatkan hubungan antara nilai masa lalu serta pengaruh kesalahan sebelumnya untuk menggambarkan pola pada data deret waktu. Bentuk umum model ARMA dituliskan pada Persamaan (2.3) sebagai berikut (Adhikari & Agrawal, 2013):

$$Y_t = \phi_1 Y_{t-1} + \dots + \phi_p Y_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \dots - \theta_q \varepsilon_{t-q} \quad (2.3)$$

keterangan

- Y_t : nilai pengamatan pada waktu ke- t
 Y_{t-1}, \dots, Y_{t-p} : nilai pengamatan masa lalu yang berhubungan
 ϕ_1, \dots, ϕ_p : koefisien atau parameter model AR
 $\theta_1, \dots, \theta_q$: koefisien atau parameter model MA
 ε_t : nilai residual pada waktu ke- t
 $\varepsilon_{t-1}, \dots, \varepsilon_{t-q}$: nilai residual pada periode sebelumnya

2.5.4 Model *Autoregressive Integrated Moving Average* (ARIMA)

Model *Autoregressive Integrated Moving Average* (ARIMA) merupakan metode analisis deret waktu yang dikembangkan dalam kerangka metodologi Box & Jenkins (1976) untuk tujuan peramalan. Model ini terdiri atas tiga komponen utama, yaitu *Autoregressive* (AR), *Integrated* (I), dan *Moving Average* (MA). Komponen *Integrated* merepresentasikan proses pembedaan (*differencing*) yang dilakukan untuk mengatasi ketidakstasioneran data. Pada umumnya, banyak data deret waktu belum stasioner pada rata-rata, sehingga perlu dilakukan proses *differencing* agar data menjadi stasioner sebelum digunakan dalam pemodelan AR dan MA. Proses *differencing* dapat dilakukan lebih dari satu kali hingga tercapai kondisi stasioner. Tingkat *differencing* dalam model ARIMA dinyatakan dengan parameter d , sedangkan parameter p dan q masing-masing menyatakan orde komponen AR dan MA.

Model ARIMA umumnya dinotasikan sebagai $ARIMA(p, d, q)$, yang menunjukkan bahwa model tersebut menggunakan p lag variabel dependen, d tingkat *differencing*, serta q lag dari residual. Notasi ini dapat dipandang sebagai perluasan dari model-model sebelumnya. Sebagai contoh, $ARIMA(p, 0, 0)$ merupakan model $AR(p)$, $ARIMA(0, 0, q)$ merupakan model $MA(q)$, sedangkan $ARIMA(p, 0, q)$ merupakan model $ARMA(p, q)$. Hubungan tersebut dapat dijelaskan melalui persamaan model yang disajikan pada rumus (2.4) sebagai berikut (Hariadi & Sulantari, 2021):

$$\phi_p(B) (1 - B)^d Y_t = \theta_q(B) \varepsilon_t \quad (2.4)$$

keterangan

- Y_t : nilai pengamatan pada waktu ke- t
 $(1 - B)^d$: proses *differencing* orde ke- d
 $\phi_p(B)$: $(1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p)$ yaitu operator langkah mundur untuk AR
 $\theta_q(B)$: $(1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q)$, yaitu operator langkah mundur untuk MA
 ε_t : residual pada periode ke- t

2.6 Model Volatilitas (Model *Conditional Heteroscedastic*)

2.6.1 Volatilitas

Menurut Rafulta *et al.* (2025), volatilitas menggambarkan tingkat penyebaran atau variasi *return* suatu aset dan sering dijadikan indikator utama untuk menilai risiko pergerakan harga di pasar keuangan. Pada praktiknya, data keuangan umumnya menunjukkan pola volatilitas yang kompleks, tidak stasioner, serta mengandung heteroskedastisitas. Kondisi tersebut menuntut penggunaan model statistik yang mampu beradaptasi terhadap perubahan volatilitas dari waktu ke waktu secara lebih fleksibel dan akurat.

Dalam instrumen seperti harga saham, volatilitas *return* menjadi aspek yang sangat penting dalam merumuskan strategi investasi, karena fluktuasi harga yang besar berpotensi memengaruhi keputusan manajemen risiko portofolio. Oleh sebab itu, pemahaman yang komprehensif mengenai pola dan dinamika volatilitas tidak hanya memperkaya literatur keuangan, tetapi juga memberikan manfaat praktis bagi investor maupun pengelola portofolio dalam menghadapi ketidakpastian pasar (Rafulta *et al.*, 2025).

Salah satu karakteristik penting dalam analisis volatilitas adalah *volatility clustering*, yaitu kondisi ketika variabilitas data saling berkaitan antar-periode. Periode dengan volatilitas tinggi cenderung diikuti oleh periode tinggi, dan sebaliknya. Hal ini ditunjukkan oleh adanya autokorelasi pada kuadrat residual yang mengindikasikan persistensi volatilitas.

Secara umum, volatilitas menunjukkan besar perubahan data akibat ketidakpastian. Pada data acak, volatilitas dibedakan menjadi dua, yaitu konstan dan berubah-ubah. Salah satu ukuran volatilitas adalah varians, yaitu ukuran penyimpangan data dari rata-rata. Perbedaan ini membagi model deret waktu menjadi dua kelompok. Model dengan varians konstan meliputi AR, MA, dan ARMA, sedangkan varians yang berubah diwakili oleh ARCH dan GARCH untuk menangkap perubahan volatilitas.

Dalam bidang keuangan, pemodelan volatilitas memegang peranan penting. Gospodinov *et al.* (2006) mengemukakan beberapa alasan utama perlunya pemodelan volatilitas, yakni sebagai berikut:

1. Prediksi volatilitas membantu investor menilai risiko yang timbul ketika memegang suatu aset keuangan.
2. Estimasi selang kepercayaan dapat menjadi lebih akurat apabila ragam residual yang berubah-ubah berhasil dimodelkan dengan tepat.
3. Proses pendugaan parameter model juga menjadi lebih efisien ketika ketidakhomogenan ragam residual dapat ditangani secara memadai. Dengan demikian, pemodelan volatilitas menjadi komponen penting dalam analisis pasar keuangan modern.

2.6.2 Model ARCH (*Autoregressive Conditional Heteroscedasticity*)

Model ARCH diperkenalkan oleh Engle (1982). Model ini digunakan untuk mengatasi varians galat yang tidak konstan dalam data deret waktu. Pada model ARCH, varians galat (σ_t^2) dipengaruhi oleh galat pada periode sebelumnya (ε_{t-1}^2). Bentuk umum model ARCH dapat dilihat pada Persamaan (2.5) berikut (Beeg *et al.*, 2024):

$$\sigma_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + \dots + \alpha_p \varepsilon_{t-p}^2 \quad (2.5)$$

keterangan:

- σ_t^2 : varians kondisional (volatilitas) pada waktu ke- t
 α_0 : konstanta
 $\alpha_1, \dots, \alpha_p$: parameter ARCH yang menunjukkan pengaruh *shock* masa lalu
 $\varepsilon_{t-1}^2, \dots, \varepsilon_{t-p}^2$: kuadrat residual pada periode sebelumnya (*shock*)

2.6.3 Model GARCH (*Generalized Autoregressive Conditional Heteroskedasticity*)

Bollerslev (1986) memperkenalkan model GARCH yang merupakan pengembangan dari model ARCH. Pada model ini, varians tidak hanya dipengaruhi oleh residual masa lalu, tetapi juga oleh nilai varians kondisional pada periode sebelumnya. Dengan demikian, GARCH memberikan pemodelan volatilitas yang lebih komprehensif. Bentuk umum model GARCH dapat dilihat pada Persamaan (2.6) berikut (Beeg *et al.*, 2024):

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^p \beta_i \sigma_{t-i}^2 + \sum_{j=1}^q \alpha_j \varepsilon_{t-j}^2 \quad (2.6)$$

keterangan:

- σ_t^2 : varians kondisional (volatilitas) pada waktu ke- t
- α_0 : konstanta
- β_1, \dots, β_p : koefisien GARCH yang menunjukkan pengaruh varians masa lalu
- $\sigma_{t-1}^2, \dots, \sigma_{t-p}^2$: varians kondisional pada periode sebelumnya
- $\alpha_1, \dots, \alpha_q$: koefisien ARCH yang menunjukkan pengaruh residual kuadrat
- $\varepsilon_{t-1}^2, \dots, \varepsilon_{t-q}^2$: residual kuadrat pada periode sebelumnya (*shock*)

2.6.4 Pengujian Efek ARCH/ GARCH

Heteroskedastisitas adalah kondisi ketika varians residual tidak konstan antar pengamatan. Pengujian dilakukan untuk mendeteksi adanya perbedaan varians tersebut. Jika nilai signifikansi lebih dari 0,05, maka tidak terdapat heteroskedastisitas, sedangkan nilai kurang dari atau sama dengan 0,05 menunjukkan adanya heteroskedastisitas. Untuk mendeteksi pola ARCH–GARCH, Engle (2001) mengembangkan uji *Lagrange Multiplier* (ARCH–LM) yang bertujuan menguji hubungan pada kuadrat residual deret waktu. Tahapan uji ARCH–LM adalah sebagai berikut:

1. Hipotesis

$H_0 : \alpha_1 = \dots = \alpha_q = 0$ (Tidak terdapat unsur ARCH–GARCH)

$H_1 : \text{minimal terdapat satu } \alpha_i \neq 0$ (Terdapat unsur ARCH–GARCH)

2. Taraf Signifikansi

$$\alpha = 5\% = 0,05$$

3. Statistik Uji Uji ARCH–LM menggunakan statistik uji yang dituliskan pada Persamaan (2.7) sebagai berikut:

$$LM = TR^2 \quad (2.7)$$

keterangan:

T : jumlah observasi

R^2 : koefisien determinasi

4. Kriteria uji:

- Tolak H_0 jika nilai statistik uji $LM > \chi^2(\alpha, k)$ atau $p\text{-value} < \alpha$
- Tidak tolak H_0 jika nilai statistik uji $LM \leq \chi^2(\alpha, k)$ atau $p\text{-value} \geq \alpha$

5. Keputusan dan Kesimpulan

Apabila masih terdapat indikasi heteroskedastisitas dalam varians residual maka pemodelan akan dilanjutkan pada model GARCH.

2.7 Stasioneritas Data

Dalam analisis deret waktu, stasioneritas menunjukkan bahwa data memiliki rata-rata dan varians yang konstan sepanjang waktu. Salah satu penyebab ketidakstasioneran adalah adanya akar unit (*unit root*) yang membuat data mengikuti pola *random walk*. Dickey & Fuller (1979) mengembangkan uji untuk mendeteksi keberadaan akar unit dalam model autoregresif dengan menguji apakah koefisien lag sama dengan satu. Jika hipotesis nol tidak dapat ditolak, maka deret waktu dinyatakan tidak stasioner. Untuk mengatasi autokorelasi residual, pengujian ini dikembangkan menjadi bentuk *Augmented Dickey-Fuller* (ADF) dengan menambahkan lag dari variabel terdiferensiasi, dengan langkah sebagai berikut:

1. Hipotesis

H_0 : terdapat akar unit (data tidak stasioner)

H_1 : tidak terdapat akar unit (data stasioner)

2. Taraf Signifikansi

$$\alpha = 5\% = 0,05$$

3. Statistika uji dengan Persamaan (2.8) sebagai berikut:

$$ADF_{hitung} = \frac{\hat{\gamma}}{Se(\hat{\gamma})} \quad (2.8)$$

keterangan:

$\hat{\gamma}$: nilai dugaan parameter *autoregressive* (AR) dengan orde p

$Se(\hat{\gamma})$: *standard error*

4. Kriteria uji:

- Jika $ADF_{hitung} > ADF_{tabel}$ atau $p\text{-value} < \alpha$ maka H_0 ditolak
- Jika $ADF_{hitung} < ADF_{tabel}$ atau $p\text{-value} > \alpha$ maka H_0 tidak ditolak

5. Keputusan dan Kesimpulan

Jika deret waktu awal terbukti tidak stasioner, maka dilakukan *differencing* untuk menghilangkan tren atau pola yang menyebabkan ketidakstasioneran. Setelah itu, dilakukan kembali uji stasioneritas menggunakan *Augmented Dickey-Fuller* (ADF) untuk memastikan bahwa data hasil *differencing* sudah stasioner dan layak digunakan dalam analisis lanjutan yang mensyaratkan kestasioneran.

Dalam proses pemisahan data deret waktu, sering muncul perbedaan waktu data dengan t data pada waktu data $t - 1$. Langkah ini dikenal sebagai *differencing* orde pertama. Jika setelah itu data masih belum stasioner, proses *differencing* dapat dilanjutkan ke orde kedua, ketiga, dan seterusnya. Secara matematis, operasi *differencing* dihitung menggunakan Persamaan (2.9) sebagai berikut:

$$\Delta Y_t = Y_t - Y_{t-1} \quad (2.9)$$

keterangan:

ΔY_t : data *differencing*

Y_t : data periode saat ini

Y_{t-1} : data periode sebelumnya

Proses membuat data menjadi stasioner, baik melalui transformasi maupun *differencing*, merupakan langkah penting agar komponen *Autoregressive* (AR) dan

Moving Average (MA) dalam model ARIMA dapat diidentifikasi dengan tepat. Setelah data mencapai kestasioneran, baik dari sisi rata-rata maupun varians, dilakukan analisis melalui plot ACF dan PACF. Kedua plot tersebut berfungsi untuk menentukan orde AR dan MA yang sesuai dalam pemodelan ARIMA (Wei, 2006).

2.8 Residual *White Noise*

Uji asumsi terhadap residual dilakukan untuk memastikan bahwa model yang dibangun memenuhi kriteria kelayakan secara statistik. Salah satu pengujian penting adalah uji *white noise*, yang berfungsi untuk menilai apakah residual bersifat independen serta tidak menunjukkan pola sistematis dari waktu ke waktu. Residual yang memenuhi karakteristik *white noise* dianggap tidak memiliki autokorelasi dan memiliki distribusi yang seragam. Untuk menguji kondisi tersebut, digunakan uji *Ljung-Box*, yaitu prosedur yang dikembangkan oleh Ljung & Box (1978) untuk mendeteksi keberadaan autokorelasi pada berbagai lag. Apabila hasil uji menunjukkan tidak adanya autokorelasi signifikan, maka residual dapat dinyatakan memenuhi asumsi *white noise*, sehingga model dianggap valid untuk digunakan pada tahap analisis lebih lanjut. Proses pengujian dilakukan sebagai berikut:

1. Hipotesis

H_0 : residual memenuhi asumsi *white noise*

H_1 : residual tidak memenuhi asumsi *white noise*

2. Taraf Signifikansi

$\alpha = 5\% = 0,05$

3. Statistik uji Uji asumsi residual *white noise* menggunakan statistik uji yang ditulis pada Persamaan (2.10), sebagai berikut:

$$Q = n(n + 2) \sum_{i=1}^m \frac{\rho_i^2}{(n - i)} \quad (2.10)$$

keterangan

n : banyaknya data pengamatan

ρ_i : autokorelasi lag ke- i

i : indeks lag

m : jumlah lag

4. Kriteria uji

- Tolak H_0 jika hasil statistik uji $Q > X_{1-\alpha, m}^2$ atau $p\text{-value} > \alpha$
- Tidak tolak H_0 jika hasil statistik uji $Q < X_{1-\alpha, m}^2$ atau $p\text{-value} < \alpha$

5. Keputusan dan Kesimpulan.

2.9 Machine Learning

Machine learning merupakan cabang dari kecerdasan buatan (*artificial intelligence*) yang berfokus pada pengembangan algoritma dan model yang memungkinkan komputer untuk belajar dari data. *Machine learning* memungkinkan sistem komputer untuk mempelajari pola dari data historis dan menggunakan pola dan tren data untuk membuat prediksi dan keputusan. Algoritma *machine learning* dapat dibedakan berdasarkan jenis pengalaman atau informasi yang digunakan dalam proses pembelajarannya yaitu (Alpaydin, 2004):

1. *Supervised Learning*

Supervised learning adalah metode pembelajaran yang menggunakan data yang sudah memiliki label atau target yang telah ditentukan. *Supervised learning* merupakan salah satu komputasi dalam *machine learning* yang digunakan dalam melakukan proses klasifikasi dan regresi. Model dilatih dengan data tersebut agar dapat mempelajari pola dan mampu memprediksi atau mengklasifikasikan data baru dengan baik (Alpaydin, 2004). Hasil dari regresi berupa nilai numerik atau nilai nyata, seperti harga, berat produk, atau berat badan.

2. *Unsupervised Learning*

Unsupervised Learning adalah metode pembelajaran yang menggunakan data tanpa label atau target yang telah ditentukan. Pada pendekatan ini, algoritma belajar sendiri untuk menemukan pola atau struktur tersembunyi dalam data, sehingga dapat mengelompokkan data ke dalam beberapa kategori berdasarkan kemiripan tertentu tanpa memerlukan target *output* (Alpaydin, 2004). Pembelajaran tidak terarah sangat efektif digunakan dalam proses pengelompokan atau *clustering* data.

3. *Reinforcement Learning*

Reinforcement Learning adalah metode pembelajaran di mana agen belajar melalui interaksi dengan lingkungan dengan mencoba berbagai tindakan. Agen

menerima umpan balik berupa *reward* atau *punishment* dan menyesuaikan strateginya untuk memperoleh hasil terbaik, sehingga banyak digunakan pada sistem pengambilan keputusan otomatis seperti robotika dan permainan (Alpaydin, 2004).

2.9.1 *Scaling Data*

Scaling data merupakan teknik untuk menyesuaikan nilai numerik dalam suatu dataset ke dalam skala umum. Metode ini bertujuan mengurangi kesalahan serta meningkatkan efisiensi pengolahan data dengan memperkecil ukuran data, tanpa mengubah nilai atau informasi utama yang sebenarnya. *Scaling* data dilakukan untuk mempercepat proses pembelajaran pada *machine learning*. Salah satu teknik yang umum digunakan adalah *min-max scaler*, terutama ketika data memiliki rentang nilai yang besar. Metode ini mentransformasikan data ke dalam interval [0, 1] sehingga setiap fitur memiliki kontribusi yang seimbang dalam proses pelatihan (Zhang *et al.*, 2021). Persamaan untuk *min-max scaler* ditulis pada Persamaan (2.11) berikut:

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (2.11)$$

keterangan

x' : nilai hasil *scaling*

x : nilai data aktual

x_{min} : nilai minimum data aktual

x_{max} : nilai maksimum data aktual

2.9.2 *Unscaling Data*

Unscaling data merupakan proses untuk mengembalikan nilai hasil prediksi ke dalam skala aslinya sehingga diperoleh nilai yang sesuai dengan kondisi sebenarnya (Lattifia *et al.*, 2022). Proses ini diperlukan karena pada tahap *preprocessing*, data terlebih dahulu dilakukan *scaling* untuk meningkatkan kinerja model. Akibatnya, hasil prediksi yang dihasilkan masih berada dalam skala yang telah diubah, sehingga perlu dikonversi kembali ke skala awal agar dapat dibandingkan dengan data aktual dan dievaluasi secara tepat. Persamaan *unscaling data* menggunakan

metode *min-max scaler* dinyatakan pada Persamaan (2.12) sebagai berikut:

$$x_i = y_i (\max(x) - \min(x)) + \min(x) \quad (2.12)$$

keterangan

- x_i : nilai data aktual
- y_i : nilai hasil *output*
- $\min(x)$: data dengan nilai minimum
- $\max(x)$: data dengan nilai maksimum

2.9.3 Fungsi Aktivasi

Fungsi aktivasi merupakan komponen fundamental dalam jaringan saraf tiruan yang berperan dalam memodelkan hubungan nonlinier antara data masukan dan keluaran. Fungsi ini menentukan bagaimana sebuah neuron memproses sinyal yang diterima sebelum meneruskannya ke neuron lain di dalam jaringan. Penerapan fungsi aktivasi dilakukan pada keluaran setiap neuron di lapisan tertentu untuk mengatur respons neuron terhadap stimulus yang diberikan. Dengan kata lain, fungsi aktivasi bertindak sebagai mekanisme penyaring yang mengubah nilai input menjadi output dalam rentang tertentu, sehingga memungkinkan jaringan saraf untuk mempelajari pola yang kompleks dan tidak linear (Haykin, 2009). Beberapa fungsi aktivasi yang dapat digunakan sebagai berikut:

1. Fungsi sigmoid (fungsi logistik)

Fungsi sigmoid atau fungsi logistik merupakan jenis fungsi aktivasi yang sering digunakan dalam jaringan saraf dan model statistik. Pada fungsi ini, jika *input* semakin kecil maka *output* mendekati 0, sedangkan jika *input* semakin besar maka *output* mendekati 1. Menurut Haykin (2009), fungsi ini bersifat nonlinier dan terdiferensialkan, sehingga mendukung proses pembelajaran menggunakan algoritma berbasis gradien seperti *backpropagation*. Fungsi sigmoid sering digunakan untuk merepresentasikan *output* dalam bentuk probabilitas, tetapi memiliki keterbatasan berupa kejenuhan (*saturation*) pada nilai *input* ekstrem yang dapat menyebabkan gradien menjadi sangat kecil dan memperlambat proses pelatihan jaringan saraf. Bentuk persamaan fungsi sigmoid ditulis pada Persamaan (2.13) sebagai berikut:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (2.13)$$

keterangan

e : bilangan *euler*

x : data

2. Fungsi tangen hiperbolik (fungsi tanh)

Fungsi tangen hiperbolik (tanh) merupakan fungsi aktivasi nonlinier yang memetakan nilai input ke dalam rentang -1 hingga 1 dan digunakan untuk menghasilkan vektor keluaran baru dalam jaringan saraf. Menurut Haykin (2009), fungsi tanh bersifat kontinu dan dapat diturunkan, sehingga mendukung proses pembelajaran berbasis metode gradien. Selain itu, keluaran fungsi tanh berpusat di sekitar nol (*zero-centered*), yang memungkinkan proses konvergensi pelatihan jaringan saraf berlangsung lebih cepat dibandingkan dengan fungsi sigmoid. Namun, pada nilai *input* yang sangat besar atau sangat kecil, fungsi ini dapat mengalami kejenuhan yang menyebabkan nilai gradien menjadi sangat kecil (*vanishing gradient*). Bentuk persamaan fungsi tanh ditulis pada Persamaan (2.14) sebagai berikut:

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.14)$$

keterangan

e : bilangan *euler*

x : data

2.10 Deep Learning

Deep learning merupakan pengembangan dari *machine learning* yang memanfaatkan jaringan saraf tiruan berlapis (*deep neural networks*) untuk mempelajari pola data yang kompleks. Model *deep learning* umumnya tersusun atas lapisan masukan (*input layer*), beberapa lapisan tersembunyi (*hidden layers*), dan lapisan keluaran (*output layer*). Menurut LeCun *et al.* (2015), keberadaan banyak lapisan tersembunyi memungkinkan model mengekstraksi representasi fitur secara bertahap dan hierarkis dari data mentah. Proses transformasi nonlinier pada setiap lapisan tersembunyi membuat *deep learning* mampu mengubah hubungan

data yang kompleks menjadi representasi yang lebih sederhana dan informatif, sehingga meningkatkan kinerja pemodelan dan klasifikasi (Schmidhuber, 2015).

2.10.1 Recurrent Neural Network (RNN)

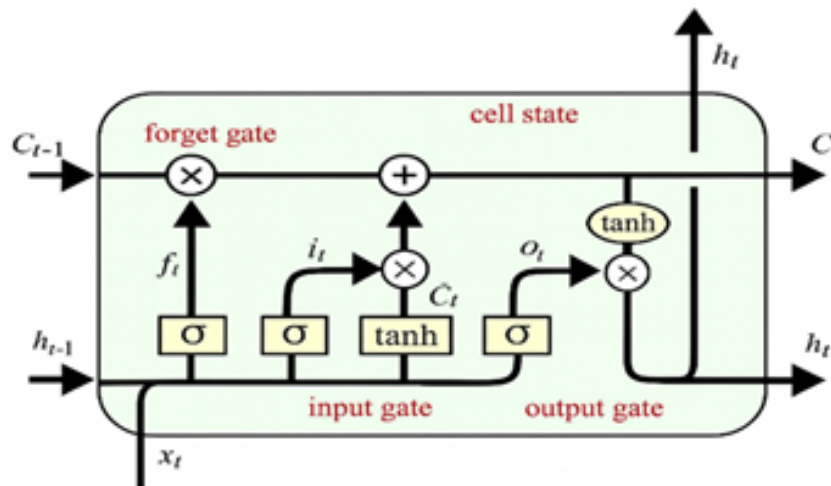
Recurrent Neural Network (RNN) merupakan arsitektur jaringan saraf tiruan yang dikembangkan untuk menangani data berurutan (*sequential data*). RNN memiliki kemampuan menyimpan dan memanfaatkan informasi dari langkah waktu sebelumnya, sehingga efektif digunakan pada permasalahan yang melibatkan urutan data, seperti teks, audio, dan deret waktu. Keunggulan utama RNN terletak pada kemampuannya memproses data dengan panjang urutan yang bervariasi, yang membedakannya dari jaringan saraf *multi-layer* konvensional yang umumnya memerlukan ukuran *input* tetap.

RNN dirancang untuk mengolah data berurutan. RNN bekerja dengan memanfaatkan informasi dari keadaan tersembunyi sebelumnya untuk membentuk keadaan saat ini, sehingga mampu menangkap pola yang bergantung pada urutan waktu. Namun, RNN memiliki keterbatasan dalam menyimpan informasi jangka panjang karena masalah *vanishing gradient*, yang menyebabkan kinerja model menurun pada data dengan interval waktu yang panjang. Untuk mengatasi kelemahan tersebut, dikembangkan model turunan RNN seperti *Long Short-Term Memory* (LSTM) yang dirancang dengan mekanisme gerbang untuk mengatur aliran informasi secara lebih efektif dan mampu menangkap ketergantungan jangka panjang dalam data berurutan (Hochreiter & Schmidhuber, 1997).

2.10.2 Long Short-Term Memory (LSTM)

Long Short-Term Memory (LSTM) pertama kali diperkenalkan oleh Hochreiter dan Schmidhuber (1997). LSTM merupakan pengembangan dari arsitektur RNN yang dirancang untuk mengatasi permasalahan *vanishing gradient* dalam pemrosesan data jangka panjang melalui penambahan *memory cell* yang berfungsi menyimpan informasi penting dalam waktu yang lebih lama. Model LSTM terdiri dari sekumpulan *memory cell* yang menggantikan neuron pada lapisan tersembunyi RNN. Arsitektur LSTM menggunakan mekanisme gerbang untuk mengatur aliran informasi dalam memperbarui dan mempertahankan isi *memory cell*. Terdapat tiga gerbang utama dalam LSTM, yaitu *forget gate*, *input gate*, dan *output gate*,

yang masing-masing berfungsi menyaring informasi yang dibuang, disimpan, dan dikeluarkan dari *memory cell*. Mekanisme ini membuat LSTM lebih efektif dalam menangani ketergantungan jangka panjang pada data berurutan (Sherstinsky, 2020). Ilustrasi struktur jaringan LSTM ditunjukkan pada Gambar 1 sebagai berikut:



Gambar 1. Arsitektur LSTM (Yanti *et al.*, 2024).

1. Forget Gate

Langkah awal pada model LSTM dimulai melalui *forget gate* (f_t). Pada tahap ini, informasi yang dianggap tidak relevan atau kurang penting untuk proses pembelajaran akan disaring dan dihilangkan dengan menggunakan fungsi sigmoid. Variabel x_t merepresentasikan data masukan pada waktu ke- t , sedangkan h_{t-1} merupakan vektor keadaan tersembunyi pada waktu sebelumnya ($t - 1$). Proses pada *forget gate* dirumuskan dalam Persamaan (2.15) sebagai berikut (Rowan *et al.*, 2022):

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (2.15)$$

keterangan

f_t : nilai *forget gate* pada waktu ke- t

σ : fungsi aktivasi sigmoid

W_f : bobot (*weight*) pada *forget gate*

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

x_t : masukan (*input*) pada waktu ke- t

b_f : bias pada *forget gate*

2. *Input Gate*

Pada langkah selanjutnya, informasi diproses melalui komponen *input gate* (i_t). Komponen ini berfungsi untuk menyeleksi serta menentukan informasi baru yang akan diperbarui dan disimpan ke dalam *cell state*. Proses pemilihan informasi tersebut dilakukan dengan menggunakan fungsi aktivasi sigmoid. Fungsi aktivasi ini menghasilkan nilai antara 0 dan 1 yang menunjukkan tingkat kepentingan informasi baru untuk disimpan dalam *cell state*. Proses pada *input gate* dirumuskan dalam Persamaan (2.16) sebagai berikut sebagai berikut (Rowan *et al.*, 2022):

$$i_t = \sigma (W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2.16)$$

keterangan

- i_t : *input gate* pada waktu ke- t
- σ : fungsi aktivasi sigmoid
- W_i : bobot pada *input gate*
- h_{t-1} : keluaran (*hidden state*) pada waktu ke- $(t - 1)$
- x_t : masukan (*input*) pada waktu ke- t
- b_i : bias pada *input gate*

Selanjutnya, dilakukan proses pembaruan informasi pada *cell state*. *Cell state* yang telah diperbarui (\hat{C}_t) dibentuk dengan menggunakan fungsi aktivasi tanh untuk mengendalikan besarnya informasi baru yang ditambahkan. Mekanisme pembaruan ini dirumuskan dalam Persamaan (2.17) sebagai berikut (Rowan *et al.*, 2022):

$$\hat{C}_t = \tanh (W_c \cdot [h_{t-1}, x_t] + b_c) \quad (2.17)$$

keterangan

- \hat{C}_t : nilai baru yang ditambahkan pada *cell state*
- \tanh : fungsi aktivasi tanh
- W_c : bobot pada pembaruan *cell state*
- h_{t-1} : keluaran pada waktu ke- $(t - 1)$
- x_t : masukan pada waktu ke- t ,
- b_c : bias pada pembaruan *cell state*

Langkah selanjutnya adalah memperbarui nilai *cell state* lama (C_{t-1}) menjadi

cell state baru (C_t) menggunakan Persamaan (2.18) berikut:

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \hat{C}_t \quad (2.18)$$

keterangan

C_t : nilai *cell state* baru

f_t : *forget gate* pada waktu ke- t

C_{t-1} : *cell state* pada waktu ke- $(t - 1)$

i_t : *input gate* pada waktu ke- t

\hat{C}_t : nilai baru yang ditambahkan pada *cell state*

3. *Output Gate*

Langkah terakhir terjadi pada proses *output gate*, di mana fungsi aktivasi sigmoid digunakan untuk menghasilkan keluaran pada *hidden state*. Pada tahap ini, *cell state* terlebih dahulu diproses dengan fungsi aktivasi tanh sebelum dikombinasikan dengan *output gate* untuk membentuk keluaran akhir. Proses ini dirumuskan dalam Persamaan (2.19) sebagai berikut (Rowan *et al.*, 2022):

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (2.19)$$

keterangan

o_t : *output gate* pada waktu ke- t

σ : fungsi aktivasi sigmoid

W_o : bobot pada *output gate*

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

x_t : masukan (*input*) pada waktu ke- t

b_o : bias pada *output gate*

Setelah diperoleh nilai aktivasi sigmoid dan nilai aktivasi tanh, kedua hasil tersebut dikalikan untuk menghasilkan *hidden state* akhir sebagai berikut (Rowan *et al.*, 2022):

$$h_t = o_t \cdot \tanh(C_t) \quad (2.20)$$

keterangan

h_t : *hidden state* pada waktu ke- t

o_t : nilai *output gate*

C_t : *cell state* pada waktu ke- t

2.10.3 *Hyperparameter*

Algoritma *machine learning* dan *deep learning* memiliki seperangkat pengaturan awal yang berfungsi sebagai mekanisme pengendali dalam proses pembelajaran model, yang secara umum disebut sebagai *hyperparameter*. Berbeda dengan parameter model yang diperoleh melalui proses pembelajaran berbasis data, *hyperparameter* tidak diperbarui secara otomatis selama pelatihan, melainkan harus ditentukan sebelum proses pelatihan dimulai (Yang *et al.*, 2023). Penentuan konfigurasi *hyperparameter* yang tepat dilakukan melalui proses *hyperparameter tuning*, yaitu suatu pendekatan optimasi yang bertujuan untuk mencari kombinasi nilai terbaik sehingga model mampu mencapai kinerja yang optimal. Sejumlah penelitian terkini menunjukkan bahwa proses *hyperparameter tuning* memiliki peran krusial dalam meningkatkan kemampuan generalisasi model sekaligus meminimalkan risiko terjadinya *underfitting* maupun *overfitting* (Zhang *et al.*, 2024).

Dalam kerangka *deep learning*, pemilihan *hyperparameter* yang sesuai menjadi faktor penentu keberhasilan model karena secara langsung memengaruhi dinamika proses pelatihan, termasuk laju konvergensi, stabilitas pembelajaran, serta kualitas hasil prediksi yang dihasilkan. *Hyperparameter* yang umum digunakan pada model *deep learning* mencakup jumlah *hidden layer*, jumlah neuron pada setiap lapisan, *batch size*, jumlah *epoch*, tingkat *dropout*, jenis *optimizer*, serta nilai *learning rate*, yang masing-masing berperan dalam mengatur kompleksitas model dan efisiensi proses pembelajaran jaringan saraf tiruan (Kumar *et al.*, 2025).

2.11 *Hybrid ARIMA-GARCH-LSTM*

Model *hybrid* merupakan pendekatan yang menggabungkan dua atau lebih model ke dalam satu sistem untuk meningkatkan kinerja peramalan. Dalam penelitian ini, pendekatan *hybrid* diterapkan dengan mengombinasikan model ARIMA, GARCH,

dan LSTM untuk menangkap karakteristik data yang berbeda. Pendekatan ini memungkinkan pemodelan yang lebih komprehensif karena setiap model memiliki keunggulan dalam menangkap pola tertentu pada data. Menurut Zhang (2003), pengembangan model *hybrid* didasarkan pada beberapa pertimbangan sebagai berikut:

1. Dalam peramalan, sering kali sulit untuk menentukan apakah suatu data deret waktu mengikuti pola linear atau nonlinear. Kondisi ini mendorong peneliti untuk menggunakan lebih dari satu model dan membandingkan hasilnya agar diperoleh prediksi yang paling akurat. Oleh karena itu, penggabungan beberapa model dapat membantu meminimalkan kesalahan dalam pemilihan model.
2. Jarang sekali ditemukan data deret waktu yang sepenuhnya bersifat linear maupun nonlinear. Pada umumnya, data merupakan kombinasi dari kedua pola tersebut. Akibatnya, penggunaan satu model saja menjadi kurang efektif karena tidak mampu menangkap karakteristik pola campuran secara menyeluruh.
3. Dalam semua situasi, tidak ada satu model peramalan yang dapat digunakan secara universal. Pada kenyataannya, data yang kompleks sering kali tidak dapat dimodelkan dengan baik hanya dengan satu model, sehingga penggunaan kombinasi beberapa model menjadi alternatif yang lebih efektif.

Pendekatan *hybrid* digunakan untuk meningkatkan akurasi peramalan dengan menggabungkan beberapa model yang mampu menangkap karakteristik data yang berbeda. Data deret waktu dapat dipisahkan menjadi komponen linear dan nonlinier seperti disajikan dalam Persamaan (2.21) sebagai berikut (Zhang, 2003):

$$Y_t = L_t + NL_t + a_t \quad (2.21)$$

keterangan:

Y_t : data aktual pada periode ke- t

L_t : komponen linear

NL_t : komponen nonlinier

a_t : residual

Komponen linear dalam penelitian ini dimodelkan menggunakan ARIMA yang dikembangkan oleh Box dan Jenkins (1976). Model ini digunakan untuk

menangkap hubungan linear dalam data deret waktu, sehingga diperoleh bentuk Persamaan (2.22) sebagai berikut:

$$Y_t = \hat{Y}_t^{\text{ARIMA}} + e_t \quad (2.22)$$

keterangan:

\hat{Y}_t^{ARIMA} : L_t (hasil peramalan komponen linear)

e_t : nilai residual pada waktu ke- t

Residual yang dihasilkan dari model ARIMA selanjutnya dimodelkan menggunakan GARCH untuk menangkap volatilitas. Model GARCH dikembangkan oleh Bollerslev (1986) sebagai pengembangan dari model ARCH oleh Engle (1982). Persamaan model GARCH dituliskan pada Persamaan (2.23) dan Persamaan (2.24) sebagai berikut:

$$e_t = \sigma_t z_t \quad (2.23)$$

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^p \beta_i \sigma_{t-i}^2 + \sum_{j=1}^q \alpha_j \varepsilon_{t-j}^2 \quad (2.24)$$

keterangan:

e_t : nilai residual pada waktu ke- t

σ_t^2 : variansi kondisional

z_t : residual yang telah distandardisasi

Komponen nonlinier dimodelkan menggunakan *Long Short-Term Memory* (LSTM) yang diperkenalkan oleh Hochreiter dan Schmidhuber (1997). Model ini mampu menangkap pola nonlinier kompleks dalam data deret waktu. Secara umum, model LSTM dituliskan pada Persamaan (2.25) sebagai berikut:

$$\hat{z}_t = f_{\text{LSTM}}(z_{t-1}, z_{t-2}, \dots, z_{t-n}) \quad (2.25)$$

keterangan:

\hat{z}_t : nilai prediksi pada waktu ke- t

$z_{t-1}, z_{t-2}, \dots, z_{t-n}$: deret nilai observasi sebelumnya (lag) hingga lag ke- n

Hasil prediksi kemudian dikembalikan ke skala semula:

$$\hat{e}_t^{\text{LSTM}} = \hat{z}_t \cdot \sigma_t \quad (2.26)$$

keterangan:

\hat{e}_t^{LSTM} : NL_t (komponen nonlinier)

Berdasarkan ketiga komponen tersebut, model *hybrid* ARIMA–GARCH–LSTM dibangun secara bertahap (*sequential*), di mana setiap model menangkap karakteristik data yang berbeda. Hasil peramalan akhir diperoleh dengan menggabungkan hasil prediksi ARIMA dan hasil prediksi residual dari model LSTM, sehingga diperoleh Persamaan (2.27) sebagai berikut:

$$\hat{Y}_t = \hat{Y}_t^{\text{ARIMA}} + \hat{e}_t^{\text{LSTM}} \quad (2.27)$$

keterangan:

\hat{Y}_t : hasil peramalan model *hybrid*

\hat{Y}_t^{ARIMA} : L_t (komponen linear)

\hat{e}_t^{LSTM} : NL_t (komponen nonlinier)

2.12 Evaluasi Model

Evaluasi model dilakukan untuk mengukur tingkat ketepatan hasil prediksi dengan cara membandingkan nilai yang dihasilkan oleh model terhadap nilai aktual atau nilai observasi. Dalam penelitian ini, evaluasi kinerja model dilakukan menggunakan dua ukuran *error*, yaitu:

1. Mean Absolute Percentage Error (MAPE)

Mean Absolute Percentage Error (MAPE) merupakan ukuran evaluasi yang digunakan untuk menilai rata-rata kesalahan prediksi dalam bentuk persentase absolut dengan membandingkan nilai hasil peramalan terhadap data aktual. Nilai MAPE yang lebih kecil menunjukkan tingkat akurasi model yang lebih baik. Secara umum, nilai MAPE di bawah 10% dikategorikan sangat baik, 10%–20% baik, 20%–50% cukup, dan di atas 50% menunjukkan akurasi yang rendah (Beeg *et al.*, 2024). Bentuk umum persamaan MAPE ditulis pada Persamaan (2.28) berikut:

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| \frac{\hat{Y}_t - Y_t}{Y_t} \right| \times 100\% \quad (2.28)$$

keterangan

n : jumlah periode yang diprediksi

Y_t : nilai aktual pada periode ke- t

\hat{Y}_t : nilai prediksi pada periode ke- t

2. Root Mean Square Error (RMSE)

Root Mean Square Error (RMSE) merupakan ukuran evaluasi yang digunakan untuk menilai besarnya kesalahan prediksi model terhadap data aktual. RMSE dihitung dari selisih antara nilai observasi dan nilai prediksi yang kemudian dikuadratkan, dirata-ratakan, dan diakarkan. Nilai RMSE yang semakin kecil menunjukkan bahwa hasil prediksi semakin mendekati nilai sebenarnya, sehingga kinerja model peramalan semakin baik (Lattifia *et al.*, 2022). Persamaan umum RMSE dituliskan pada Persamaan (2.29).

$$RMSE = \sqrt{\frac{\sum_{t=1}^n (A_t - F_t)^2}{n}} \quad (2.29)$$

keterangan

n : jumlah periode yang diprediksi

A_t : nilai aktual pada periode ke- t

F_t : nilai prediksi pada periode ke- t

2.13 Goodness of Fit

Goodness of fit merupakan metode statistik yang digunakan untuk menilai kesesuaian antara model atau distribusi teoretis dengan data empiris yang diamati. Pendekatan ini bertujuan untuk mengetahui sejauh mana distribusi yang diasumsikan dapat merepresentasikan karakteristik data aktual. Salah satu metode *goodness of fit* yang sering digunakan adalah uji *Kolmogorov–Smirnov* (KS). Uji ini dilakukan dengan membandingkan fungsi distribusi kumulatif empiris dari data pengamatan dengan fungsi distribusi kumulatif dari distribusi teoretis yang diasumsikan. Selisih maksimum antara kedua fungsi distribusi tersebut digunakan

sebagai dasar dalam menilai kesesuaian distribusi. Statistik uji KS dinyatakan pada Persamaan (2.30) (Kolmogorov, 1933).

$$KS = \sup_x |F_n(x) - F(x)| \quad (2.30)$$

keterangan

KS : statistik uji Kolmogorov–Smirnov

$F_n(x)$: fungsi distribusi kumulatif empiris

$F(x)$: fungsi distribusi kumulatif teoretis

\sup : nilai supremum (nilai maksimum) dari selisih absolut antara kedua fungsi distribusi

Nilai KS yang kecil menunjukkan distribusi data observasi semakin mendekati distribusi teoretis.

BAB III

METODE PENELITIAN

3.1 Waktu dan Tempat Penelitian

1. Tempat Penelitian

Penelitian ini dilakukan pada Semester Ganjil Tahun Akademik 2025/2026 di Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung yang beralamatkan di Jalan Prof. Dr. Ir. Soemantri Brojonegoro, Gedong Meneng, Kecamatan Rajabasa, Kota Bandar Lampung, Provinsi Lampung.

2. Waktu Penelitian

Penelitian ini dilaksanakan pada Semester Ganjil Tahun Akademik 2025/2026 dan dimulai pada bulan Agustus 2025. Pelaksanaan penelitian dilakukan melalui beberapa tahapan utama. Tahap pertama meliputi studi literatur dengan mengumpulkan referensi dari jurnal ilmiah dan buku yang relevan dengan topik penelitian. Kegiatan ini bertujuan untuk membangun landasan teoritis yang kuat sebagai dasar penyusunan proposal penelitian hingga tahap akhir. Pada tahap ini juga dilakukan pengumpulan data yang akan digunakan sebagai bahan analisis dalam penelitian. Tahap kedua merupakan tahap pengolahan dan pemodelan data, yang mencakup proses *preprocessing* data, membangun model ARIMA dan model GARCH berdasarkan residual ARIMA, pelaksanaan *hyperparameter tuning*, pengembangan model *hybrid* ARIMA-GARCH-LSTM, serta evaluasi kinerja model. Tahap ketiga adalah penyusunan laporan penelitian berdasarkan hasil analisis dan pemodelan yang telah dilakukan. Hasil penelitian tersebut kemudian dipresentasikan dan dipertanggungjawabkan melalui seminar hasil dan sidang komprehensif.

3.2 Data dan Alat Penelitian

1. Data Penelitian

Penelitian ini menggunakan data sekunder berupa harga penutupan (*closing price*) saham PT Unilever Indonesia Tbk. yang dicatat secara harian. Data diperoleh dari situs *Yahoo Finance* dan diakses melalui halaman riwayat perdagangan saham UNVR.JK. Periode data yang digunakan dari Januari 2020 hingga November 2025, dengan total 1.427 observasi. Ringkasan data tersebut disajikan pada Tabel 2 sebagai berikut:

Tabel 2. Data Harga Saham PT Unilever Indonesia Tbk.

Tanggal	Close
02/01/2020	8.550
03/01/2020	8.575
06/01/2020	8.475
07/01/2020	8.450
08/01/2020	8.325
⋮	⋮
24/11/2025	2.670
25/11/2025	2.630
26/11/2025	2.600
27/11/2025	2.600
28/11/2025	2.600

2. Alat Penelitian

Penelitian ini dilaksanakan dengan memanfaatkan berbagai perangkat dan alat bantu yang berfungsi untuk mendukung serta menunjang seluruh proses penelitian.

(a) Perangkat Keras

Penelitian ini menggunakan satu unit laptop sebagai perangkat keras dengan spesifikasi sebagai berikut:

- Processor : Gen 8th Intel Core i5 @ 3.00GHz 2.90 GHz
- RAM : 8,00 GB

(b) Perangkat Lunak yang digunakan

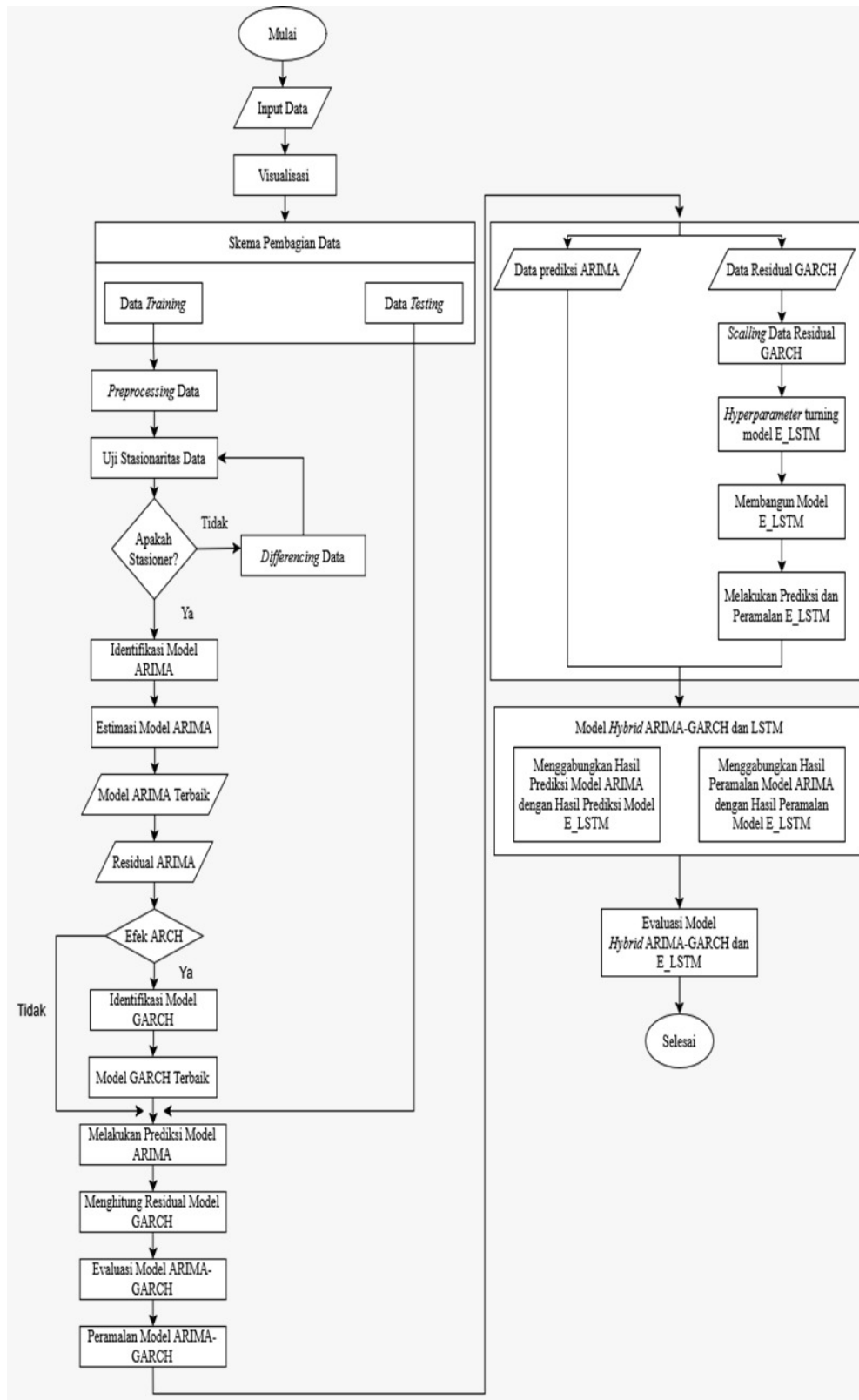
- Google Colaboratory (Python 3.10.12)
- Sistem Operasi Windows 10 Pro 64-bit

(c) Package yang digunakan

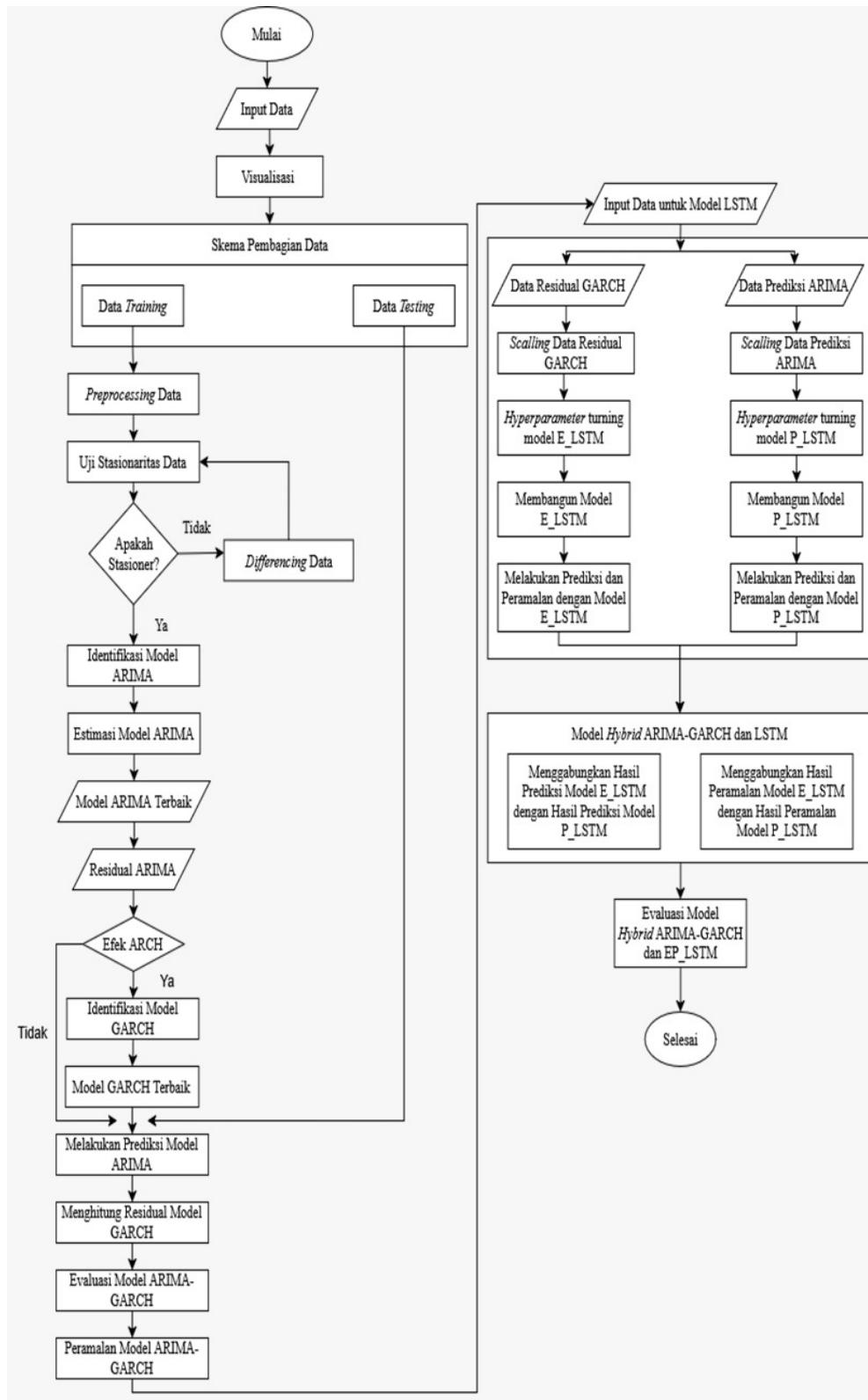
- NumPy (1.23.5)
- Pandas (1.5.3)
- Matplotlib (3.7.1)
- Scipy (1.16.3)
- Arch (8.0.0)
- Sklearn (1.5.2)
- Statsmodels (0.14.4)
- Tensorflow (2.12.0)

3.3 Metode Penelitian

Tahap pertama dalam penelitian ini adalah melakukan studi literatur dengan menelusuri berbagai sumber seperti buku, jurnal ilmiah, artikel, serta sumber daring lainnya untuk memperoleh informasi yang relevan dengan topik penelitian. Selanjutnya, seluruh tahapan penelitian yang dilakukan disajikan dalam bentuk diagram alur pada Gambar 2 dan Gambar 3, sebagai berikut:



Gambar 2. Flowchart Model Hybrid ARIMA-GARCH dan E_LSTM



Gambar 3. Flowchart Model Hybrid ARIMA-GARCH dan EP-LSTM

BAB IV

HASIL DAN PEMBAHASAN

4.1 Proses *Input* dan Visualisasi Data

Langkah pertama yang dilakukan dalam penelitian ini adalah melakukan *input* data ke dalam perangkat lunak *Phyton*. Data yang di-*input* merupakan data penutupan harian harga saham PT Unilever Indonesia Tbk. selama periode Januari 2020 hingga Desember 2025. Data tersebut selanjutnya disajikan pada Tabel 3 berikut:

Tabel 3. Data *Input*

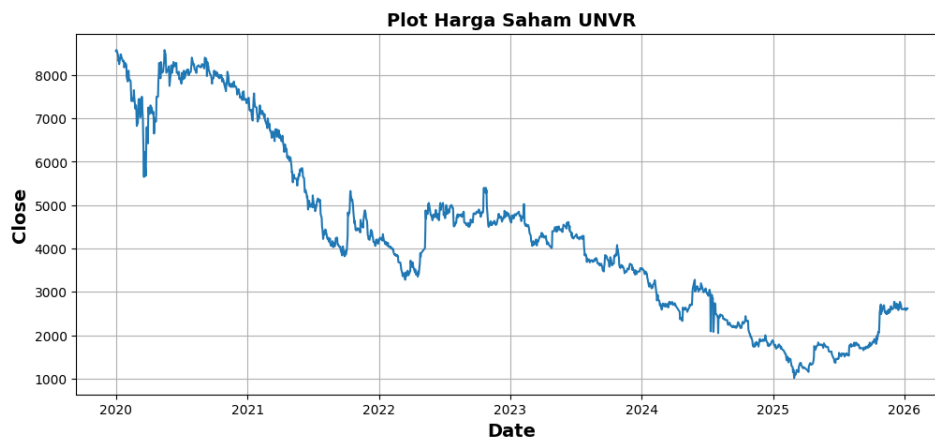
<i>Date</i>	<i>Open</i>	<i>High</i>	<i>Low</i>	<i>Close</i>	<i>Adj Close</i>	<i>Volume</i>
01/01/2020	8.500	8.700	8.500	8.550	6.994.39	11.059.800
02/01/2020	8.675	8.675	8.550	8.575	7.017.84	8.071.700
03/01/2020	8.575	8.600	8.350	8.475	6.933.04	7.913.300
06/01/2020	8.475	8.500	8.400	8.450	6.933.04	5.793.000
07/01/2020	8.450	8.450	8.275	8.325	6.810.33	8.261.400
⋮	⋮	⋮	⋮	⋮	⋮	⋮
28/11/2025	2.620	2.640	2.590	2.600	2.513.99	18.013.700

Dalam penelitian ini digunakan metode ARIMA dengan variabel utama *close*. Selanjutnya dilakukan statistik deskriptif yang disajikan pada Tabel 4 berikut:

Tabel 4. Statistika Deskriptif Variabel *Close*

Statistika Deskriptif	<i>Close</i>
Jumlah Data	1427
Mean	4.384
Standar Deviasi	2,055
Minimum	1.015
Q1	2.695
Median	4.230
Q3	5.262
Maksimum	8.575

Setelah data di-*input*, langkah selanjutnya adalah melakukan visualisasi data untuk melihat pola dalam data yang dipresentasikan pada Gambar 4 sebagai berikut:



Gambar 4. Plot Data Historis Penutupan Harga Saham PT Unilever Indonesia Tbk.

Berdasarkan plot data historis saham harian, harga saham PT Unilever Indonesia Tbk. menunjukkan tren penurunan signifikan selama periode 2020–2025, dari kisaran 8.000 pada awal periode menjadi di bawah 2.000 pada akhir periode. Hal ini menunjukkan adanya kecenderungan penurunan kinerja harga saham secara signifikan dalam jangka panjang.

4.2 Skema Pembagian Data

Skema pembagian data dilakukan sebelum data tersebut diproses pada model ARIMA. Pada penelitian ini, pembagian data dilakukan dengan satu skema, yaitu 90% data *training* dan 10% data *testing*. Berdasarkan skema tersebut, diperoleh data *training* sebanyak 1284 dan data *testing* sebanyak 143.

4.3 Proses Model ARIMA-GARCH

Langkah awal dalam peramalan harga saham menggunakan model *hybrid* ARIMA-GARCH-LSTM adalah melakukan pemodelan ARIMA untuk menstasionerkan *mean* data. Selanjutnya, model GARCH digunakan untuk memodelkan varians residual yang dihasilkan dari model ARIMA. Tahapan untuk memperoleh nilai prediksi model ARIMA dan residual model GARCH adalah sebagai berikut:

4.3.1 Preprocessing Data

Sebelum proses pemodelan menggunakan metode ARIMA dilakukan, terlebih dahulu dilakukan tahap *preprocessing* data. Tahapan ini meliputi pemeriksaan data untuk mendeteksi adanya nilai yang hilang (*missing values*). Hasil pengecekan data hilang disajikan pada Tabel 5 berikut:

Tabel 5. Hasil Pengecekan *Missing Value*

Tanggal	Variabel	<i>Missing Value</i>
-	<i>Close</i>	0

Berdasarkan Tabel 5 tidak terdapat nilai data variabel yang hilang, sehingga data dapat dilakukan proses selanjutnya.

4.3.2 Uji Stasioner

Setelah tahap *preprocessing* data selesai dilakukan, langkah selanjutnya adalah melakukan uji stasioneritas menggunakan uji *Augmented Dickey-Fuller* (ADF) terhadap variabel *close*. Uji ADF bertujuan untuk mengetahui apakah data *close* telah memenuhi asumsi stasioneritas. Jika data sudah stasioner, maka data tersebut memenuhi asumsi dasar metode ARIMA. Namun, apabila data belum stasioner, perlu dilakukan proses *differencing*. Hasil uji ADF disajikan pada Tabel 6 berikut:

Hipotesis:

H_0 : data *close* tidak stasioner

H_1 : data *close* stasioner

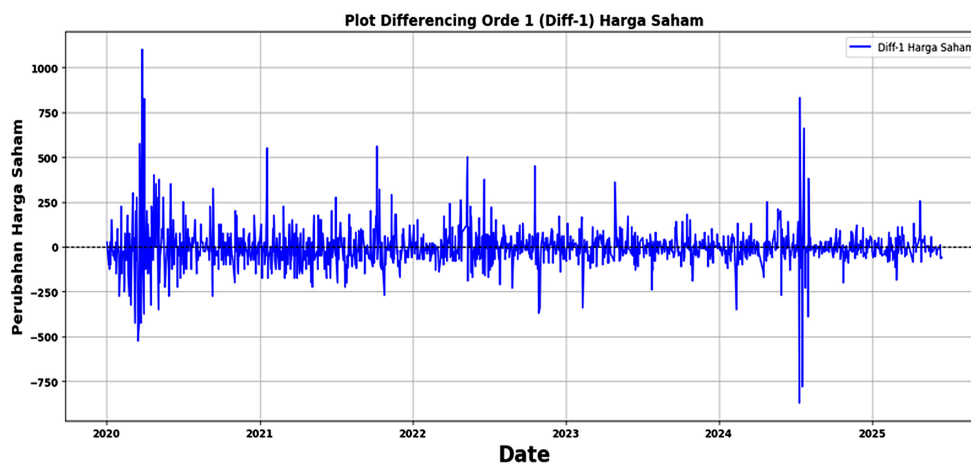
Jika nilai $p\text{-value} < \alpha = 0,05$ maka H_0 ditolak.

Tabel 6. Hasil Uji *Augmented Dickey-Fuller* (ADF)

Statistik Uji ADF	$p\text{-value}$
-1.8078	0.6289

Berdasarkan Tabel 6, diperoleh nilai $p\text{-value}$ sebesar 0.6289. Karena tingkat signifikansi (α) yang digunakan dalam penelitian ini adalah 5%, maka perbandingan antara $p\text{-value}$ dan α menunjukkan bahwa $p\text{-value} > \alpha$. Dengan demikian, dapat disimpulkan bahwa data yang digunakan tidak stasioner, sehingga

diperlukan proses *differencing*. Visualisasi data yang diperoleh setelah dilakukan *differencing* pertama disajikan pada Gambar 5 sebagai berikut:



Gambar 5. Plot Data Historis Penutupan Harga Saham Setelah *Differencing*.

Berdasarkan Gambar 5, dapat dilihat bahwa data sudah stasioner dalam rata-rata. Hal ini merupakan indikasi awal bahwa data tersebut sudah stasioner. Kestasioneran data selanjutnya diuji dengan menggunakan uji ADF. Hasil uji ADF setelah proses *differencing* ditampilkan pada Tabel 7 berikut:

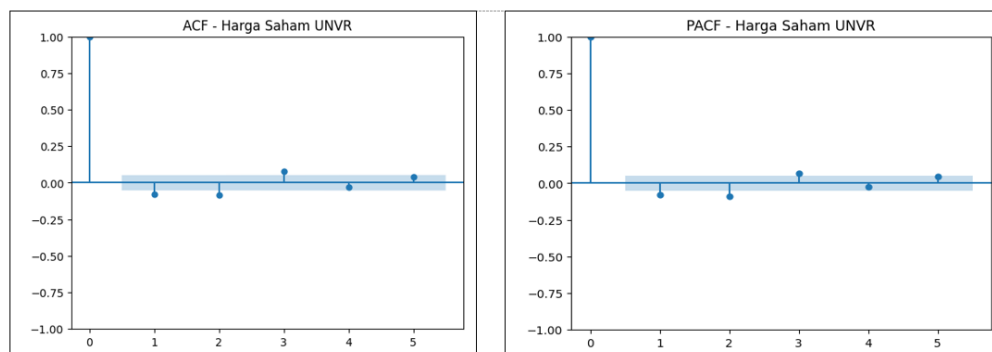
Tabel 7. Hasil Uji *Augmented Dickey-Fuller* (ADF) Setelah *Differencing*

Statistik Uji ADF	<i>p-value</i>
-14.6108	0.00000

Berdasarkan Tabel 7, diperoleh *p-value* sebesar 0.00000. Karena *p-value* tersebut lebih kecil dari nilai α , maka dapat disimpulkan bahwa data sudah bersifat stasioner. Dengan demikian, parameter *differencing* pada model ARIMA (p, d, q) ditetapkan sebesar $d = 1$.

4.3.3 Identifikasi Model ARIMA

Setelah data memenuhi asumsi stasioner, tahap selanjutnya yaitu melakukan identifikasi model. Identifikasi model ARIMA (p, d, q) diperoleh dengan cara membuat plot data harga saham harian PT Unilever Indonesia Tbk. ke dalam bentuk plot ACF dan PACF. Plot ACF digunakan untuk menentukan nilai q pada model MA, sedangkan plot PACF digunakan untuk menentukan nilai p pada model AR. Visualisasi plot ACF dan PACF dapat dilihat pada Gambar 6 berikut:



Gambar 6. Plot ACF dan Plot PACF.

Berdasarkan Gambar 6, hasil visualisasi plot ACF setelah dilakukan *differencing* satu kali menunjukkan bahwa lag ke-1, ke-2, dan ke-3 pada *project line* ACF melewati batas signifikansi. Oleh karena itu, kandidat model untuk parameter q adalah MA(0), MA(1), MA(2), dan MA(3).

Selanjutnya, berdasarkan plot PACF pada Gambar 6 setelah dilakukan *differencing* satu kali, terlihat bahwa lag ke-1, ke-2, dan ke-3 pada *project line* PACF juga melewati batas signifikansi. Dengan demikian, kandidat model untuk parameter p adalah AR(0), AR(1), AR(2), dan AR(3).

4.3.4 Estimasi Parameter dan Uji Diagnostik Model

Tahap selanjutnya yang dilakukan adalah menentukan model ARIMA terbaik dari nilai p , d , dan q yang diperoleh dari plot ACF dan PACF. Kombinasi model ARIMA (p, d, q) yang terbentuk sebanyak 15 model, yaitu ARIMA (0, 1, 1), ARIMA (0, 1, 2), ARIMA (0, 1, 3), ARIMA (1, 1, 0), ARIMA (1, 1, 1), ARIMA (1, 1, 2), ARIMA (1, 1, 3), ARIMA (2, 1, 0), ARIMA (2, 1, 1), ARIMA (2, 1, 2), ARIMA (2, 1, 3), ARIMA (3, 1, 0), ARIMA (3, 1, 1), ARIMA (3, 1, 2), ARIMA (3, 1, 3).

Tahap selanjutnya adalah melakukan analisis untuk menentukan model ARIMA terbaik. Proses ini dilakukan melalui *diagnostic checking* terhadap model-model dugaan, yang mencakup evaluasi signifikansi parameter dan pengujian apakah residual bersifat *white noise*. Parameter model dikatakan signifikan apabila nilai p -value yang dihasilkan lebih kecil dari tingkat signifikansi yang digunakan, yaitu $\alpha = 5\%$. Model dikatakan memenuhi syarat pengujian residual *white noise* apabila hasil p -value dari uji *Ljung-Box* memiliki nilai p -value $> \alpha$, dengan nilai $\alpha = 5\%$. Hasil dari proses *diagnostic checking* tersebut disajikan pada Tabel 8 berikut:

Tabel 8. *Diagnostic Checking* Dugaan Model ARIMA

Model	Parameter Signifikan	Residual <i>White Noise</i>
ARIMA (0,1,1)	Signifikan	Signifikan
ARIMA (0,1,2)	Signifikan	Signifikan
ARIMA (0,1,3)	Signifikan	Signifikan
ARIMA (1,1,0)	Signifikan	Signifikan
ARIMA (1,1,1)	Signifikan	Signifikan
ARIMA (1,1,2)	Signifikan	Signifikan
ARIMA (1,1,3)	Signifikan	Signifikan
ARIMA (2,1,0)	Signifikan	Signifikan
ARIMA (2,1,1)	Signifikan	Signifikan
ARIMA (2,1,2)	Tidak Signifikan	Signifikan
ARIMA (2,1,3)	Tidak Signifikan	Signifikan
ARIMA (3,1,0)	Signifikan	Signifikan
ARIMA (3,1,1)	Tidak Signifikan	Signifikan
ARIMA (3,1,2)	Tidak Signifikan	Signifikan
ARIMA (3,1,3)	Tidak Signifikan	Signifikan

Berdasarkan hasil *diagnostic checking*, diperoleh beberapa model yang memenuhi asumsi, yaitu ARIMA (0,1,1), ARIMA (0,1,2), ARIMA (0,1,3), ARIMA (1,1,0), ARIMA (1,1,1), ARIMA (1,1,2), ARIMA (1,1,3), ARIMA (2,1,0), ARIMA (2,1,1), dan ARIMA (3,1,0). Selanjutnya, model terbaik ditentukan berdasarkan nilai AIC terkecil. Hasil perbandingan ditunjukkan pada Tabel 9 sebagai berikut:

Tabel 9. Pemilihan Model ARIMA Terbaik

ARIMA (p, d, q)	AIC
ARIMA (0,1,1)	17552.557
ARIMA (0,1,2)	17547.096
ARIMA (0,1,3)	17532.771
ARIMA (1,1,0)	17534.094
ARIMA (1,1,1)	17542.015
ARIMA (1,1,2)	17529.518
ARIMA (1,1,3)	17529.863
ARIMA (2,1,0)	17534.863
ARIMA (2,1,1)	17527.790
ARIMA (3,1,0)	17530.428

Berdasarkan Tabel 9 dapat diketahui bahwa model ARIMA (2,1,1) merupakan model yang telah memenuhi syarat signifikansi parameter dan residual *white noise* dengan nilai AIC terkecil. Langkah selanjutnya adalah menganalisis deteksi heteroskedastisitas pada model ARIMA terbaik. Pengujian ini dilakukan untuk mengetahui apakah terdapat variansi residual yang tidak konstan. Jika ditemukan adanya heteroskedastisitas, maka model ARIMA saja belum cukup untuk menangkap karakteristik volatilitas data, sehingga diperlukan pendekatan lanjutan seperti model GARCH untuk memodelkan varians.

4.3.5 Pengujian Efek ARCH dalam Residual Model ARIMA

Selanjutnya dilakukan uji heteroskedastisitas untuk mengetahui apakah residual dari model ARIMA (2,1,1) memiliki varians yang konstan (homoskedastisitas). Pengujian ini dilakukan menggunakan uji ARCH (*Autoregressive Conditional Heteroskedasticity*). Hasil uji ARCH dapat dilihat pada Tabel 10 sebagai berikut:

Hipotesis:

$$H_0 : \alpha_1 = \alpha_2 = \dots = \alpha_m = 0$$

$$H_1 : \alpha_1 = \alpha_2 = \dots = \alpha_m \neq 0$$

Jika nilai *p-value* $< \alpha = 0,05$ maka H_0 ditolak.

Tabel 10. Uji Efek ARCH pada Model ARIMA

Uji ARCH-LM	<i>p-value</i>
235.258	5.10×10^{-45}

Berdasarkan Tabel 10 diperoleh nilai *p-value* $< \alpha$, sehingga H_0 ditolak. Hal ini mengindikasikan adanya heteroskedastisitas dalam bentuk efek ARCH pada residual model. Kondisi tersebut menunjukkan bahwa varians residual tidak konstan sepanjang waktu (*time-varying*) dan dipengaruhi oleh nilai residual pada periode sebelumnya.

Dengan demikian, model ARIMA (2,1,1) belum sepenuhnya memenuhi asumsi homoskedastisitas. Meskipun ARIMA baik dalam memodelkan rata-rata, uji ARCH-LM menunjukkan residual masih memiliki variansi yang tidak konstan. Oleh karena itu, diperlukan model lanjutan yang mampu menangkap dinamika variansi tersebut, yaitu model GARCH.

4.3.6 Identifikasi, Estimasi, dan Pemilihan Model GARCH

Setelah teridentifikasi adanya efek ARCH berdasarkan uji ARCH-LM, selanjutnya dilakukan pemodelan varians residual menggunakan model GARCH. Pemodelan GARCH dilakukan terhadap residual model ARIMA untuk menangkap dinamika variansi yang tidak konstan, sehingga dipertimbangkan beberapa model dengan orde rendah, yaitu GARCH (1,1), GARCH (1,2), GARCH (2,1), dan GARCH (2,2). Selanjutnya, masing-masing model diestimasi dan dievaluasi untuk menentukan model terbaik berdasarkan kriteria signifikansi parameter, uji diagnostik residual untuk memastikan tidak adanya efek ARCH yang tersisa, serta nilai AIC terkecil. Hasil *diagnostic checking* dari model-model GARCH yang diestimasi disajikan pada Tabel 11 sebagai berikut:

Tabel 11. *Diagnostic Checking* Dugaan Model GARCH

Model	Parameter	Signifikansi Parameter	<i>White Noise</i>
ARCH (1)	α_0	Signifikan	Signifikan
	α_1	Signifikan	Signifikan
ARCH (2)	α_0	Signifikan	Signifikan
	α_1	Signifikan	Signifikan
	α_2	Tidak Signifikan	Signifikan
GARCH (1,1)	α_0	Signifikan	Signifikan
	α_1	Signifikan	Signifikan
	β_1	Signifikan	Signifikan
GARCH (1,2)	α_0	Signifikan	Signifikan
	α_1	Signifikan	Signifikan
	β_1	Tidak Signifikan	Signifikan
	β_2	Signifikan	Signifikan
GARCH (2,1)	α_0	Signifikan	Signifikan
	α_1	Tidak Signifikan	Signifikan
	α_2	Signifikan	Signifikan
	β_1	Signifikan	Signifikan
GARCH (2,2)	α_0	Signifikan	Signifikan
	α_1	Signifikan	Signifikan
	α_2	Tidak Signifikan	Signifikan
	β_1	Tidak Signifikan	Signifikan
	β_2	Signifikan	Signifikan

Berdasarkan hasil *diagnostic checking* pada Tabel 12, didapatkan model ARCH (1) dan GARCH (1,1) yang memenuhi uji signifikansi parameter dan *white noise*. Selanjutnya, dari model-model yang memenuhi kriteria tersebut, dipilih model terbaik berdasarkan nilai AIC terkecil. Hasil perbandingan ditunjukkan pada Tabel 12 sebagai berikut:

Tabel 12. Pemilihan Model GARCH Terbaik

GARCH (m, s)	AIC
ARCH (1)	17180.018800
GARCH (1,1)	16956.208173

Dari hasil Tabel 12 didapatkan model GARCH (1,1) sebagai model terbaik dalam memodelkan volatilitas harga saham karena memenuhi uji signifikansi parameter dan *white noise*, serta memiliki nilai AIC paling rendah dibandingkan model lainnya yaitu sebesar 16956.208.

4.3.7 Pengujian Efek ARCH pada model GARCH

Setelah pemodelan GARCH dilakukan pengujian kembali terhadap efek ARCH pada residual. Hasil perbandingan ditunjukkan pada Tabel 13 sebagai berikut:

Tabel 13. Uji Efek ARCH Pada Model GARCH

Uji ARCH-LM	p -value
125.158	0.2481

Berdasarkan Tabel 13, diperoleh nilai p -value $> \alpha$ maka H_0 tidak ditolak, yang menunjukkan bahwa tidak terdapat efek ARCH yang signifikan, sehingga dapat disimpulkan bahwa model GARCH (1,1) telah mampu mengatasi heteroskedastisitas pada data. Oleh karena itu, model dapat dilakukan proses prediksi serta perhitungan residual untuk mengevaluasi ukuran kebaikan model.

4.3.8 Prediksi Model ARIMA (2,1,1)

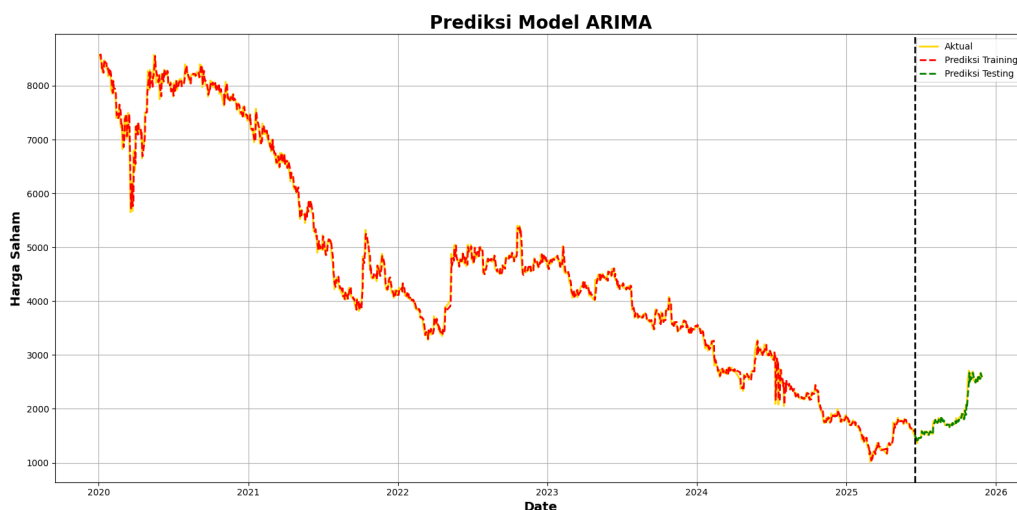
Setelah melakukan estimasi parameter model dengan menggunakan data *training*, diperoleh model ARIMA (2,1,1)-GARCH (1,1) sebagai model yang sesuai untuk memodelkan data. Model ARIMA (2,1,1) digunakan untuk menghitung nilai

prediksi berdasarkan pola rata-rata. Hasil prediksi model ARIMA (2,1,1), disajikan pada Tabel 14 sebagai berikut:

Tabel 14. Hasil Prediksi Model ARIMA

Tanggal	Aktual	Prediksi
02/01/2020	8.575	8.565.96
03/01/2020	8.475	8.564.28
06/01/2020	8.450	8.490.79
07/01/2020	8.325	8.457.87
08/01/2020	8.350	8.420.40
⋮	⋮	⋮
28/11/2025	2.600	2.608.51

Berdasarkan Tabel 14, hasil prediksi model ARIMA mendekati nilai aktual meskipun terdapat perbedaan kecil. Hal ini menunjukkan bahwa model ARIMA mampu menangkap pola fluktuasi data deret waktu yang dianalisis. Visualisasi pola fluktuasi prediksi disajikan pada Gambar 7 sebagai berikut:



Gambar 7. Plot Prediksi model ARIMA.

Berdasarkan Gambar 7, pola prediksi model ARIMA mengikuti pola data aktual dengan baik pada data training maupun testing, yang menunjukkan keakuratan model dalam melakukan prediksi. Keakuratan ini juga dapat dievaluasi menggunakan RMSE, MAPE, dan akurasi. Hasil evaluasi model ARIMA disajikan pada Tabel 15 berikut:

Tabel 15. Evaluasi Model ARIMA

Evaluasi Model	Nilai
RMSE	2.5286
MAPE	1.8423%
Akurasi	98.157%

Evaluasi model ARIMA pada Tabel 15 menunjukkan nilai RMSE dan MAPE yang sangat rendah, serta nilai akurasi sebesar 98.1577%. Hal ini menunjukkan bahwa model ARIMA memiliki performa yang sangat baik dalam melakukan prediksi.

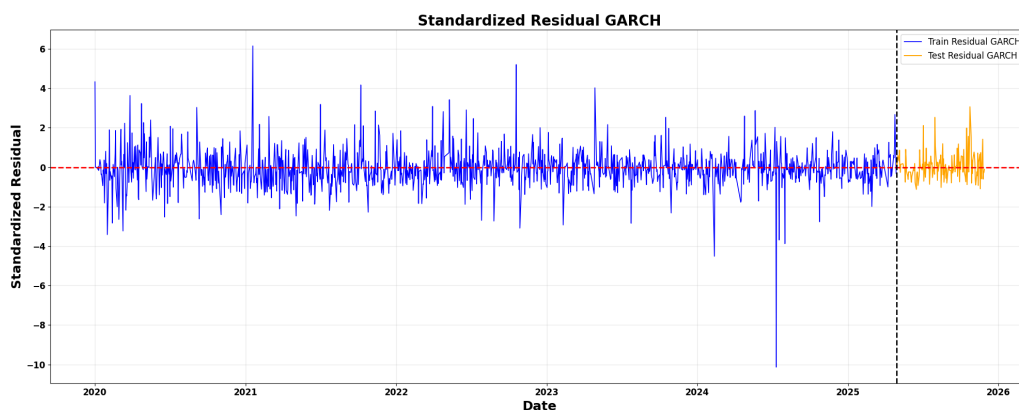
4.3.9 Residual Model GARCH (1,1)

Setelah diperoleh nilai prediksi dari model ARIMA, langkah selanjutnya adalah menghitung nilai residual, yaitu selisih antara nilai aktual dan nilai prediksi. Nilai residual tersebut kemudian digunakan sebagai dasar dalam pemodelan GARCH untuk memodelkan volatilitas data. Nilai residual yang dihasilkan disajikan pada Tabel 16 sebagai berikut:

Tabel 16. Residual Model GARCH

Tanggal	Residual
02/01/2020	4.321482
03/01/2020	-0.017435
06/01/2020	-0.004115
07/01/2020	-0.047587
08/01/2020	0.013016
⋮	⋮
28/11/2025	-0.909968

Visualisasi pola dan fluktuasi residual model GARCH dapat dilihat pada Gambar 8 sebagai berikut:



Gambar 8. Plot Residual Model GARCH.

Plot residual model GARCH pada Gambar 8 menunjukkan fluktuasi residual di sekitar nilai nol tanpa adanya tren, yang merupakan indikasi awal bahwa model telah menangkap pola data. Namun, untuk memastikan bahwa residual bersifat acak dan tidak memiliki pola tertentu, diperlukan uji residual *white noise*. Uji residual *white noise* dilakukan untuk mengetahui apakah residual yang dihasilkan bersifat independen dan memiliki karakteristik yang identik. Pengujian dilakukan dengan menggunakan uji *Ljung-Box* dengan hipotesis sebagai berikut:

Hipotesis:

H_0 : residual memenuhi asumsi *white noise*

H_1 : residual tidak memenuhi asumsi *white noise*

Jika nilai $p\text{-value} < \alpha = 0,05$ maka H_0 ditolak.

Tabel 17. Hasil Uji *Ljung-Box*

Uji <i>Ljung-Box</i>	$p\text{-value}$
2.936519	0.9828

Hasil uji *Ljung-Box* pada Tabel 17 menunjukkan nilai $p\text{-value}$ adalah 0.9828. Taraf signifikansi yang digunakan yaitu 5%, sehingga nilai $p\text{-value} > 5\%$, maka keputusan yang diambil tidak tolak H_0 . Hal ini menunjukkan bahwa residual dari kedua variabel tersebut telah memenuhi asumsi *white noise*. Oleh karena itu, model yang dibangun dapat digunakan untuk peramalan dan analisis lebih lanjut.

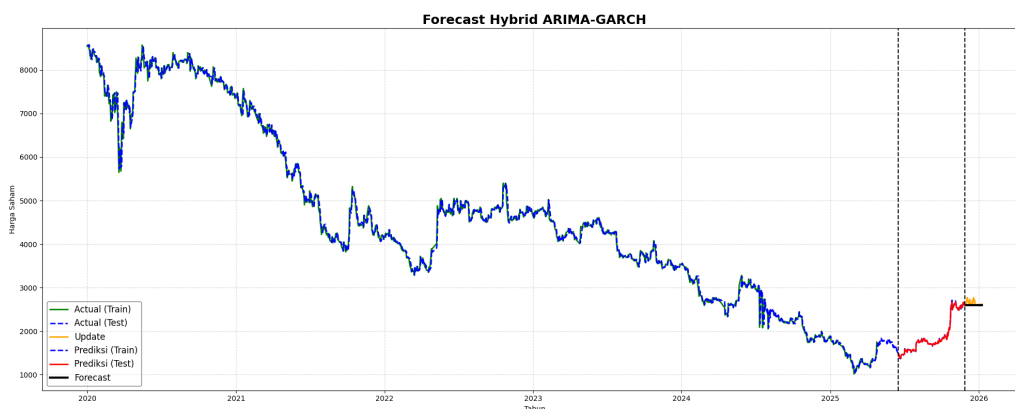
4.3.10 Peramalan Model ARIMA(2,1,1)–GARCH (1,1)

Setelah model ARIMA (2,1,1)–GARCH (1,1) dievaluasi dan residualnya telah memenuhi asumsi *white noise*, maka model tersebut dapat digunakan untuk melakukan peramalan. Peramalan dilakukan untuk periode 30 hari ke depan, dari 1 Desember 2025 hingga 9 Januari 2026. Hasil peramalan dari model ARIMA(2,1,1)–GARCH (1,1) disajikan pada Tabel 18 sebagai berikut:

Tabel 18. Hasil Peramalan Model ARIMA–GARCH

Tanggal	<i>Update</i>	Peramalan
01/12/2025	2.620	2.597.020
02/12/2025	2.650	2.589.796
03/12/2025	2.770	2.577.612
04/12/2025	2.770	2.697.334
05/12/2025	2.710	2.678.172
⋮	⋮	⋮
09/01/2026	2.620	2.597.531

Berdasarkan Tabel 18, hasil peramalan harga saham menggunakan model ARIMA–GARCH menunjukkan pola pergerakan yang relatif stabil selama periode peramalan. Nilai prediksi mengalami fluktuasi berupa kenaikan dan penurunan pada beberapa titik waktu, tetapi perubahan tersebut terjadi dalam rentang yang sempit sehingga tidak mencerminkan volatilitas yang tinggi. Visualisasi pola peramalan model ARIMA–GARCH disajikan pada Gambar 9 sebagai berikut:



Gambar 9. Plot Peramalan Model ARIMA-GARCH.

Berdasarkan Gambar 9, model ARIMA–GARCH mampu mengikuti pola data historis dengan cukup baik, terutama pada bagian prediksi data *training* dan *testing*. Hal ini ditunjukkan oleh kedekatan antara nilai prediksi dan nilai aktual pada kedua periode tersebut. Namun, pada bagian *forecasting*, hasil prediksi cenderung lebih halus sehingga belum mampu menggambarkan fluktuasi harga saham secara optimal. Untuk mengevaluasi kinerja model dapat dilihat pada Tabel 19, Tabel 20, dan Tabel 21 sebagai berikut:

Tabel 19. Evaluasi *Train-Test*

Evaluasi Model	Nilai
RMSE	2.5286
MAPE	0.8423%
Akurasi	99.157%

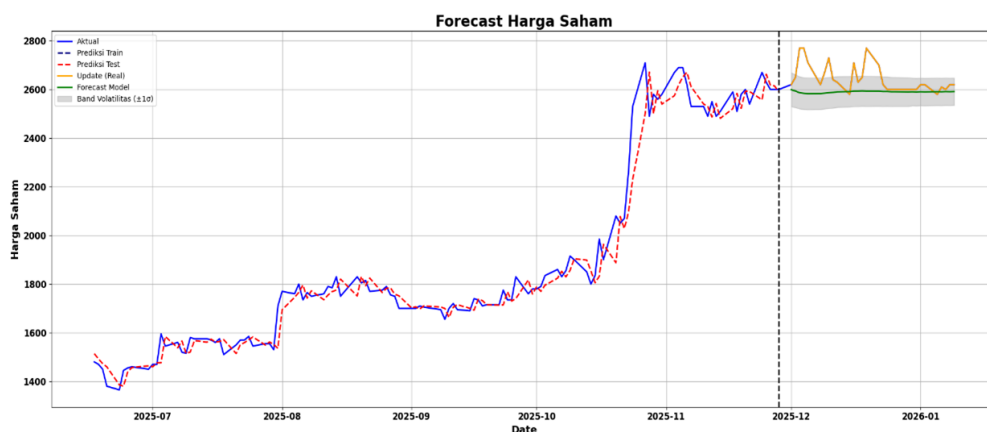
Tabel 20. Evaluasi *Forecast*

Evaluasi Model	Nilai
RMSE	150.6030
MAPE	5.6499%
Akurasi	94.350%

Tabel 21. Evaluasi *Train-Test-Forecast*

Evaluasi Model	Nilai
RMSE	185.7977
MAPE	1.7005%
Akurasi	98.299%

Berdasarkan hasil evaluasi tersebut, model ARIMA–GARCH menunjukkan nilai MAPE yang relatif kecil dan tingkat akurasi yang tinggi pada seluruh skenario pengujian. Hal ini mengindikasikan bahwa model memiliki kemampuan yang baik dalam memprediksi data secara relatif. Tetapi, nilai RMSE yang cukup besar menunjukkan bahwa kesalahan absolut masih tinggi, sehingga model belum mampu menangkap fluktuasi dengan baik. Untuk melihat hasil peramalan yang lebih jelas dapat dilihat pada Gambar 10 berikut:



Gambar 10. Plot Bagian *Forecasting Model* ARIMA-GARCH.

Berdasarkan Gambar 10, hasil peramalan menunjukkan pola yang relatif datar dibandingkan data aktual. Meskipun residual telah memenuhi asumsi *white noise*, model ARIMA–GARCH masih belum optimal dalam menangkap fluktuasi harga saham secara menyeluruh. Ini mengindikasikan adanya kemungkinan komponen nonlinier yang belum tertangkap oleh model.

4.4 Prediksi dan Peramalan Model *Hybrid* ARIMA-GARCH-E.LSTM

Hasil peramalan dengan model ARIMA–GARCH menunjukkan bahwa model tersebut masih belum optimal dalam merepresentasikan dinamika fluktuasi pada data *update*. Keterbatasan ini mengindikasikan bahwa pola nonlinier dalam data belum sepenuhnya tertangkap. Oleh karena itu, digunakan model *hybrid* ARIMA–GARCH–E.LSTM, yang menggabungkan *output* prediksi model ARIMA dengan model E.LSTM yang dilatih berdasarkan residual GARCH, sehingga diharapkan mampu meningkatkan akurasi peramalan.

Model E.LSTM dilatih untuk menangkap struktur nonlinier yang masih tersisa pada residual GARCH. Selanjutnya, *output* prediksi dari model E.LSTM dikombinasikan dengan hasil peramalan ARIMA–GARCH untuk memperoleh nilai prediksi dan peramalan akhir. Pendekatan penggabungan ini dilakukan untuk memperbaiki keterbatasan ARIMA–GARCH dalam memodelkan pola nonlinier, sehingga diharapkan dapat meningkatkan kinerja peramalan secara keseluruhan. Berikut disajikan tahapan prediksi dan peramalan menggunakan model *hybrid* ARIMA–GARCH–E.LSTM.

4.4.1 *Scaling Data Residual Model GARCH*

Pada tahap ini, residual yang diperoleh dari model ARIMA dianalisis untuk mengidentifikasi adanya pola heteroskedastisitas. Hasil pengujian menunjukkan bahwa residual model ARIMA belum sepenuhnya memenuhi asumsi *white noise*. Oleh karena itu, dilakukan pemodelan lanjutan menggunakan model GARCH untuk menangkap perilaku volatilitas yang bersifat berubah-ubah (*volatility clustering*) pada data harga saham. Nilai residual yang dihasilkan dari model GARCH kemudian dilakukan proses *scaling* menggunakan metode *Min-Max Scaler* dengan mengubah nilai residual ke dalam rentang [0, 1] tanpa mengubah pola distribusi data asli. Pemilihan *Min-Max Scaler* didasarkan pada karakteristik residual yang masih memiliki rentang nilai cukup besar akibat dinamika volatilitas harga saham. Dengan menstandarkan nilai residual, model LSTM dapat mempelajari pola nonlinear secara lebih stabil dan efisien.

Proses *scaling* ini juga bertujuan mengurangi risiko *exploding gradient* selama pelatihan model LSTM. Data residual yang telah melalui proses *scaling* kemudian digunakan sebagai *input* dalam pelatihan model LSTM. Hasil prediksi residual dari model LSTM selanjutnya digabungkan dengan hasil prediksi ARIMA untuk membentuk model *hybrid* ARIMA–GARCH–E_LSTM. Melalui pendekatan ini, komponen linear ditangkap oleh ARIMA, komponen volatilitas oleh GARCH, dan komponen nonlinear ditangkap oleh LSTM.

4.4.2 *Hyperparameter Tuning Model E_LSTM*

Sebelum membangun model E_LSTM, dilakukan *hyperparameter tuning* untuk memperoleh kombinasi parameter optimal. Karena tidak ada parameter tetap, pencarian dilakukan secara sistematis menggunakan metode *grid search*. Hasilnya disajikan pada Tabel 22 sebagai berikut:

Tabel 22 menunjukkan *hyperparameter tuning* yang digunakan pada model E_LSTM. *Hyperparameter* yang sudah ditetapkan yaitu *epoch* sebesar 100, *loss function* menggunakan *Mean Squared Error*, *learning rate* sebesar 0,001, *activation function* menggunakan tanh dan sigmoid. Sementara itu, *hidden neuron*, *batch size*, dan *dropout* dilakukan tuning dengan menggunakan metode *grid search*. *Hidden neuron* dan *batch size* menggunakan variasi nilai 32, 64, dan 128, sedangkan *dropout* menggunakan variasi nilai 0,1 dan 0,2.

Tabel 22. *Hyperparameter Tuning Model E.LSTM*

Kelompok	Hyperparameter	Nilai
Tidak perlu tuning	<i>Epoch</i>	100
	<i>Loss function</i>	<i>Mean Squared Error</i>
	<i>Learning rate</i>	0,001
	<i>Activation</i>	Tanh, sigmoid
Perlu tuning	<i>Hidden neuron (LSTM unit dan dense unit)</i>	32, 64, 128
	<i>Batch size</i>	32, 64, 128
	<i>Dropout</i>	0,1, 0,2

4.4.3 Membangun Model E.LSTM

Setelah proses *hyperparameter tuning* pada model E.LSTM, diperoleh parameter terbaik yang digunakan untuk membangun model optimal. Hasilnya ditampilkan pada Tabel 23 sebagai berikut:

Tabel 23. Hasil *Hyperparameter Tuning Model E.LSTM*

Hyperparameter	Skema Pembagian E.LSTM
<i>Activation</i>	3 <i>hidden layer</i> : tanh <i>Output layer</i> : sigmoid
<i>Hidden neuron</i>	LSTM unit: 64 Dense unit: 32
<i>Batch size</i>	128
<i>Dropout</i>	0,2

Dari Tabel 23 menunjukkan hasil *hyperparameter tuning* yang digunakan untuk membangun model E.LSTM. Model E.LSTM dibangun menggunakan 3 *hidden layer* dengan fungsi aktivasi tanh, lapisan *output* dengan fungsi aktivasi sigmoid, 64 *neuron* pada LSTM unit, 32 *dense unit*, *batch size* 128, dan *dropout* sebesar 0,2.

4.4.4 Prediksi Model E.LSTM

Model E.LSTM yang dibangun dengan menggunakan kombinasi parameter terbaik yang telah diperoleh selanjutnya digunakan untuk melakukan prediksi. Hasil prediksi nilai residual dari model E.LSTM disajikan pada Tabel 24 sebagai berikut:

Tabel 24. Prediksi Model *E-LSTM*

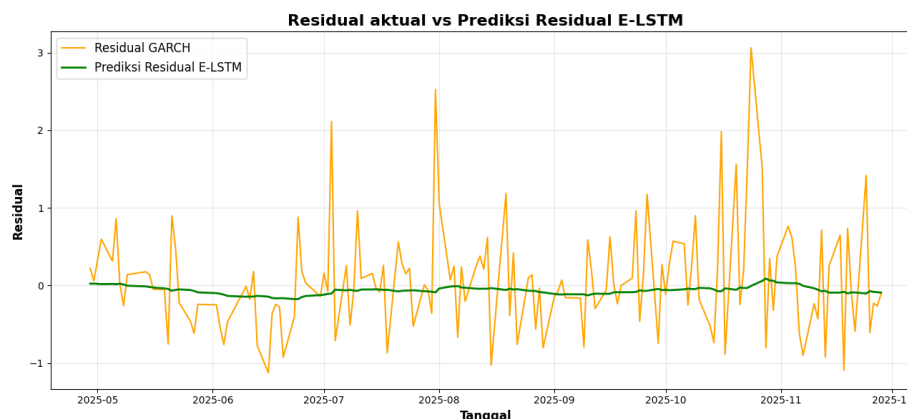
Tanggal	Prediksi Residual
29/04/2025	0.628205
30/04/2025	0.627536
02/05/2025	0.626393
05/05/2025	0.626482
06/05/2025	0.625862
⋮	⋮
28/11/2025	0.616473

Hasil prediksi nilai residual Model *E-LSTM* pada Tabel 24 diperoleh melalui residual yang telah melalui proses *scaling*, sehingga nilai prediksi yang dihasilkan juga berada dalam skala tersebut. Oleh karena itu, *unscaling* perlu dilakukan untuk mengembalikan hasil prediksi ke dalam skala residual GARCH agar dapat diinterpretasikan dengan baik. Hasil prediksi model *E-LSTM* setelah dilakukan *unscaling* disajikan pada Tabel 25 sebagai berikut:

Tabel 25. Hasil Prediksi Model *E-LSTM*

Tanggal	Residual GARCH	Residual Prediksi
29/04/2025	0.221318	0.092485
30/04/2025	0.061252	0.081586
02/05/2025	0.598289	0.062979
05/05/2025	0.314381	0.064422
06/05/2025	0.861927	0.054330
⋮	⋮	⋮
28/11/2025	-0.909968	-0.098529

Hasil prediksi model *E-LSTM* pada Tabel 25 menunjukkan bahwa setelah dilakukan proses *unscaling*, masih terdapat perbedaan yang cukup signifikan antara nilai residual GARCH dengan prediksi residual *E-LSTM* yang dihasilkan. Gambaran yang lebih jelas mengenai pola dan tren prediksi nilai residual model *E-LSTM* disajikan pada Gambar 11 sebagai berikut:



Gambar 11. Plot Residual Prediksi Model E.LSTM.

Berdasarkan Gambar 11, plot menunjukkan bahwa residual GARCH pada data aktual berfluktuasi secara acak di sekitar nol, sedangkan prediksi residual E.LSTM cenderung lebih halus dan stabil. Hal ini mengindikasikan bahwa setelah pemodelan GARCH, tidak terdapat pola nonlinier yang kuat pada residual sehingga E.LSTM tidak mampu menangkap struktur tambahan yang signifikan. Kondisi ini menunjukkan bahwa model GARCH telah bekerja secara optimal dalam memodelkan data. Hasil evaluasi model E.LSTM disajikan pada Tabel 26 sebagai berikut:

Tabel 26. Hasil Evaluasi Model

Evaluasi Model	Nilai
RMSE	6.9800
MAPE	60.581%
Akurasi	39.419%

Evaluasi model E.LSTM pada Tabel 26, menunjukkan nilai RMSE yang masih sangat rendah, tetapi pada nilai MAPE cukup tinggi, sehingga nilai akurasi hanya sebesar 39.419%. Hal ini menunjukkan bahwa model E.LSTM masih belum cukup memiliki performa yang sangat baik dalam melakukan prediksi nilai residual.

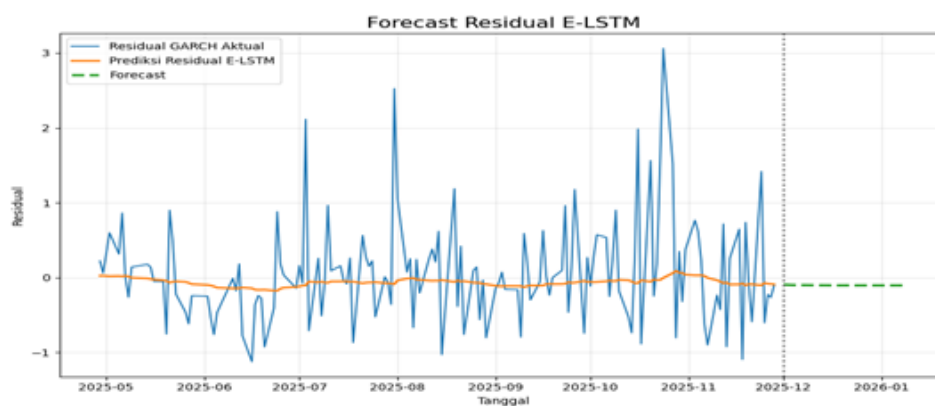
4.4.5 Peramalan Model E.LSTM

Tahapan selanjutnya adalah melakukan peramalan menggunakan model E.LSTM. Peramalan dilakukan untuk periode 30 hari ke depan. Dengan hasil peramalan nilai residual menggunakan model E.LSTM disajikan pada Tabel 27 sebagai berikut:

Tabel 27. Hasil Peramalan Residual Model E.LSTM

Tanggal	Peramalan Residual
01/12/2025	-0.108623
02/12/2025	-0.111724
03/12/2025	-0.114219
04/12/2025	-0.116401
05/12/2025	-0.118628
⋮	⋮
09/01/2026	-0.125784

Dari Tabel 27, model E.LSTM menghasilkan peramalan nilai residual yang berfluktuasi kecil di sekitar nol, yang menunjukkan bahwa sebagian besar struktur pola telah berhasil ditangkap oleh model GARCH. Hal ini mengindikasikan bahwa residual yang tersisa bersifat mendekati *white noise* sehingga kontribusi koreksi E.LSTM pada periode peramalan relatif kecil. Visualisasi yang lebih jelas mengenai pola dan tren prediksi nilai residual model E.LSTM disajikan pada Gambar 12 sebagai berikut:



Gambar 12. Plot Peramalan Residual E.LSTM.

Hasil peramalan residual menggunakan E.LSTM menunjukkan pola yang relatif konstan di sekitar nol. Hal ini mengindikasikan bahwa setelah pemodelan ARIMA-GARCH, sisa komponen residual tidak lagi mengandung struktur nonlinier yang signifikan untuk dimodelkan, sehingga E.LSTM menghasilkan proyeksi residual yang mendekati nilai rata-rata historis.

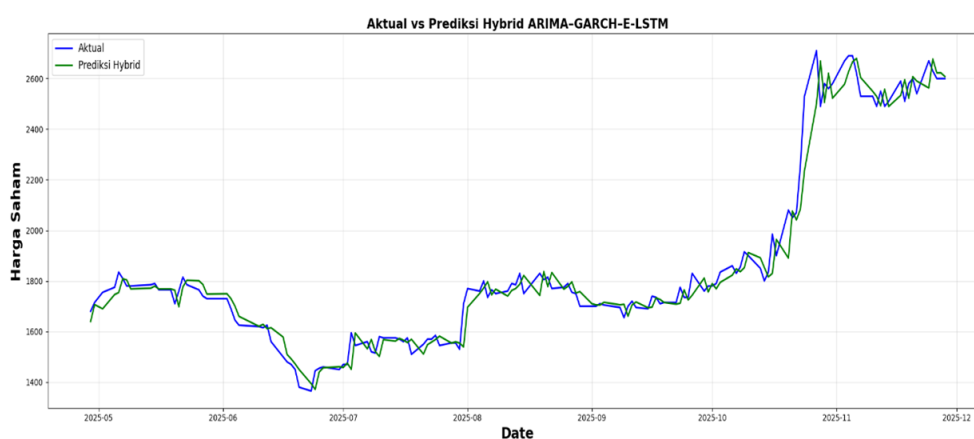
4.4.6 Prediksi Model *Hybrid* ARIMA-GARCH-E_LSTM

Model *hybrid* ARIMA–GARCH–E_LSTM dibangun dari penggabungan hasil prediksi model ARIMA dengan prediksi model E_LSTM menggunakan residual GARCH. Hasil prediksi model *hybrid* ARIMA–GARCH–E_LSTM disajikan pada Tabel 28 sebagai berikut:

Tabel 28. Hasil Prediksi Model

Tanggal	Aktual	Prediksi
29/04/2025	1.680	1.639.7077
30/04/2025	1.715	1.706.4726
02/05/2025	1.755	1.689.8351
05/05/2025	1.775	1.746.4727
06/05/2025	1.835	1.754.0280
⋮	⋮	⋮
28/11/2025	2,600	2.608.4071

Berdasarkan Tabel 28, hasil prediksi model *hybrid* ARIMA–GARCH–E LSTM mendekati nilai aktual dengan perbedaan yang relatif kecil. Hal ini menunjukkan bahwa model memiliki kemampuan yang cukup baik dalam menangkap pola pergerakan data. Visualisasi pola dan tren prediksi disajikan pada Gambar 13 sebagai berikut:



Gambar 13. Plot Hasil Prediksi *Hybrid* ARIMA–GARCH–E_LSTM.

Pada Gambar 13 pola prediksi model *hybrid* ARIMA–GARCH–E_LSTM dapat mengikuti pola dengan sangat baik. Hal ini mengindikasikan bahwa

model cukup akurat dalam melakukan prediksi. Hasil evaluasi model *hybrid* ARIMA–GARCH–E.LSTM disajikan pada Tabel 29 berikut:

Tabel 29. Evaluasi *Hybrid* ARIMA–GARCH–E.LSTM

Evaluasi Model	Nilai
RMSE	6.0371
MAPE	2.1686%
Akurasi	97.831%

Berdasarkan Tabel 29, hasil evaluasi model *hybrid* ARIMA–GARCH–E.LSTM pada periode pengujian menghasilkan nilai RMSE sebesar 6,04 dan MAPE sebesar 2,1686% dengan akurasi sebesar 97,831%. Hasil ini menunjukkan bahwa model memiliki kinerja yang baik dalam memprediksi data dengan tingkat kesalahan yang relatif kecil.

4.4.7 Peramalan Model *Hybrid* ARIMA–GARCH–E.LSTM

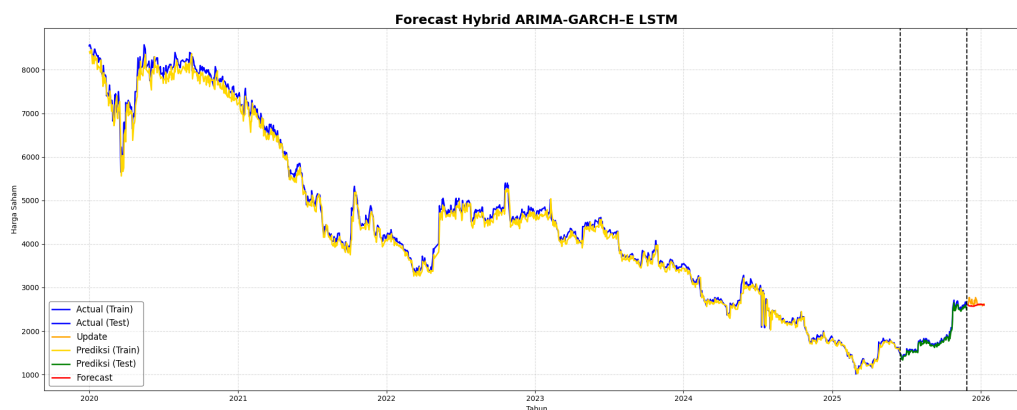
Peramalan menggunakan model *hybrid* ARIMA–GARCH–E.LSTM diperoleh dengan menjumlahkan hasil peramalan yang diperoleh dari model ARIMA–GARCH dengan hasil peramalan yang diperoleh dari model E.LSTM. Hasil peramalan selanjutnya dibandingkan dengan data *update* untuk mengevaluasi kinerja model dalam melakukan peramalan. Hasil peramalan selama 30 hari ke depan dengan menggunakan model *hybrid* ARIMA–GARCH–E.LSTM ditampilkan pada Tabel 30 sebagai berikut:

Tabel 30. Hasil Peramalan Model *Hybrid* ARIMA–GARCH–E-LSTM

Tanggal	<i>Update</i>	Peramalan
01/12/2025	2.620	2.639.0312
02/12/2025	2.650	2.633.6102
03/12/2025	2.770	2.686.5018
04/12/2025	2.770	2.721.1672
05/12/2025	2.710	2.694.0323
⋮	⋮	⋮
09/01/2026	2.620	2.651.2995

Tabel 30 menunjukkan bahwa masih terdapat perbedaan yang cukup signifikan

antara hasil peramalan model *hybrid* ARIMA–GARCH–E.LSTM dengan data *update* yang sebenarnya. Hasil peramalan cenderung stabil dan belum mampu mengikuti pola fluktuasi dari data *update* sepanjang periode peramalan. Visualisasi yang lebih jelas mengenai pola dan tren peramalan model *hybrid* ARIMA–GARCH–E.LSTM disajikan pada Gambar 14 sebagai berikut:



Gambar 14. Plot Peramalan Model *Hybrid* ARIMA–GARCH–E.LSTM.

Berdasarkan Gambar 14, model *hybrid* ARIMA–GARCH–E.LSTM mampu mengikuti pola dan fluktuasi data dengan lebih baik pada bagian prediksi *training* dan *testing*. Pada bagian *forecasting*, hasil peramalan juga terlihat sedikit fluktuasi, sehingga lebih mampu merepresentasikan pergerakan harga saham dibandingkan model sebelumnya. Untuk mengevaluasi kinerja model dapat dilihat pada Tabel 31, Tabel 31, dan Tabel 32 sebagai berikut:

Tabel 31. Evaluasi *Train-Test*

Evaluasi Model	Nilai
RMSE	6.0371
MAPE	2.1686%
Akurasi	97.831%

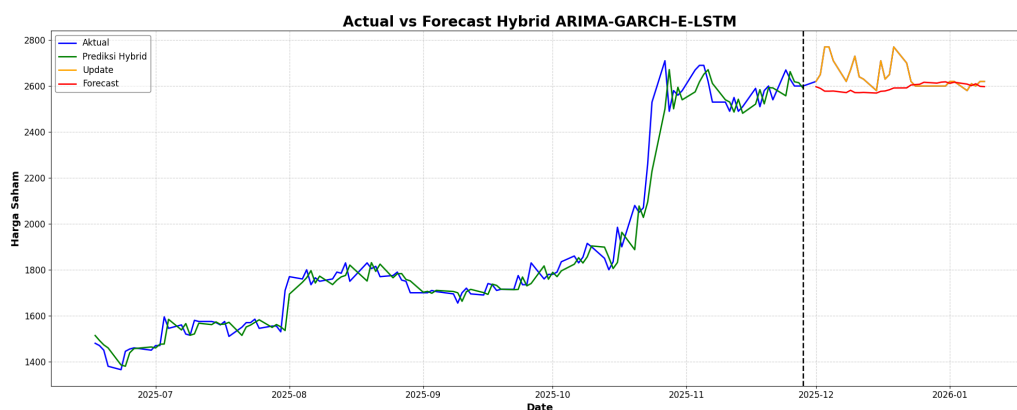
Tabel 32. Evaluasi *Forecast*

Evaluasi Model	Nilai
RMSE	4.6030
MAPE	3.7509%
Akurasi	96.249%

Tabel 33. Evaluasi *Train-Test-Forecast*

Evaluasi Model	Nilai
RMSE	15.2487
MAPE	2.0049%
Akurasi	97.995%

Berdasarkan hasil evaluasi pada Tabel di atas, model *hybrid* ARIMA–GARCH–E_LSTM menunjukkan nilai MAPE yang relatif kecil dan tingkat akurasi yang tinggi pada seluruh skenario pengujian. Hal ini mengindikasikan bahwa model memiliki kemampuan yang cukup baik dalam memprediksi data secara relatif. Tetapi nilai RMSE yang masih muncul menunjukkan bahwa kesalahan absolut tetap ada, sehingga model belum sepenuhnya mampu menangkap fluktuasi data dengan baik. Untuk melihat hasil peramalan yang lebih jelas dapat dilihat pada Gambar 15 berikut:

Gambar 15. Plot Bagian *Forecasting* Model *Hybrid* ARIMA–GARCH–E_LSTM.

Plot hasil peramalan model *hybrid* ARIMA–GARCH–E_LSTM pada Gambar 15, menunjukkan bahwa hasil peramalan model *hybrid* ARIMA–GARCH–E_LSTM sudah menunjukkan adanya sedikit fluktuasi, tetapi belum mampu mengikuti pola fluktuasi dari data *update* dengan baik. Hal ini mengindikasikan bahwa kontribusi model E-LSTM dalam meningkatkan hasil peramalan masih terbatas, yang sejalan dengan karakteristik residual yang telah mendekati *white noise*. Oleh karena itu, model *hybrid* ARIMA–GARCH–E_LSTM masih belum cukup optimal dalam melakukan peramalan.

4.5 Prediksi dan Peramalan Model *Hybrid* ARIMA–GARCH–EP_LSTM

Model *hybrid* ARIMA–GARCH–E_LSTM menunjukkan kinerja peramalan yang lebih baik dibandingkan model ARIMA–GARCH. Namun, model tersebut masih belum sepenuhnya mampu menangkap seluruh pola yang terdapat pada data *update*. Oleh karena itu, dikembangkan pendekatan lanjutan berupa model *hybrid* ARIMA–GARCH–EP_LSTM yang diharapkan dapat menghasilkan peramalan yang lebih optimal. Model *hybrid* ARIMA–GARCH–EP_LSTM memanfaatkan dua jenis informasi dari nilai residual model GARCH dan hasil prediksi ARIMA.

Model *hybrid* ARIMA–GARCH–EP_LSTM dibangun dengan menggabungkan dua model LSTM yang dilatih secara terpisah, yaitu E_LSTM menggunakan residual GARCH dan P_LSTM menggunakan prediksi ARIMA. *Output* dari kedua model kemudian dijumlahkan untuk menghasilkan prediksi dan peramalan akhir. Penggabungan ini bertujuan mengatasi keterbatasan ARIMA–GARCH dalam menangkap pola nonlinier sehingga meningkatkan akurasi. Tahapan prediksi dan peramalan menggunakan model *hybrid* ARIMA–GARCH–EP_LSTM dijelaskan sebagai berikut.

4.5.1 *Scaling* Data Prediksi Model ARIMA

Pada tahap ini, nilai prediksi yang diperoleh dari model ARIMA terlebih dahulu melalui proses *scaling* menggunakan metode *min-max scaler*. Proses ini dilakukan untuk mengubah rentang nilai data ke dalam interval $[0, 1]$, sehingga data menjadi lebih terstandarisasi dan setiap variabel dapat memberikan kontribusi yang seimbang selama proses pelatihan model.

Setelah proses *scaling* pada nilai prediksi model ARIMA selesai dilakukan, data tersebut kemudian digunakan sebagai *input* dalam pelatihan model P_LSTM. Model P_LSTM akan menghasilkan nilai prediksi dan peramalan yang selanjutnya digabungkan dengan hasil prediksi dan peramalan dari model E_LSTM melalui proses penjumlahan. Hasil penggabungan tersebut membentuk model *hybrid* ARIMA–GARCH–EP_LSTM.

4.5.2 Hyperparameter Tuning Model P_LSTM

Sebelum membangun model P_LSTM, dilakukan proses *hyperparameter tuning* menggunakan metode *grid search*. Proses ini bertujuan untuk memperoleh kombinasi parameter yang optimal. Parameter yang diuji pada tahap tersebut disajikan pada Tabel 34 sebagai berikut:

Tabel 34. *Hyperparameter Tuning Model P_LSTM*

Kelompok	Hyperparameter	Nilai
Tidak perlu tuning	<i>Epoch</i>	100
	<i>Loss function</i>	<i>Mean Squared Error</i>
	<i>Learning rate</i>	0,001
	<i>Activation</i>	tanh, sigmoid
Perlu tuning	<i>Hidden neuron (LSTM unit dan dense unit)</i>	32, 64, 128
	<i>Batch size</i>	32, 64, 128

Tabel 34 menunjukkan *hyperparameter tuning* yang digunakan pada model P_LSTM. *Hyperparameter* yang telah ditetapkan yaitu *epoch* sebesar 100, *loss function* menggunakan *Mean Squared Error*, *learning rate* sebesar 0,001, dan *activation function* menggunakan *tanh* dan *sigmoid*. Sementara itu, *hidden neuron* dan *batch size* dilakukan proses *tuning* menggunakan metode *grid search* dengan variasi nilai 32, 64, dan 128.

4.5.3 Membangun Model P_LSTM

Setelah *hyperparameter tuning* pada model P_LSTM, diperoleh kombinasi parameter terbaik yang digunakan untuk membangun model optimal. Hasilnya disajikan pada Tabel 35 sebagai berikut:

Tabel 35. Hasil *Hyperparameter Tuning Model P_LSTM*

<i>Hyperparameter</i>	P_LSTM
<i>Activation</i>	3 hidden layer: tanh Output layer: <i>sigmoid</i>
<i>Hidden neuron</i>	LSTM unit: 32 Dense unit: 32
<i>Batch size</i>	64

Tabel 35 menunjukkan hasil proses *hyperparameter tuning* yang digunakan dalam pembangunan model P_LSTM. Berdasarkan hasil tersebut, model P_LSTM dibangun dengan tiga *hidden layer* yang menggunakan fungsi aktivasi *tanh*, serta lapisan *output* dengan fungsi aktivasi *sigmoid*. Selain itu, model ini menggunakan 32 *neuron* pada unit LSTM, 32 *neuron* pada *dense unit*, dan *batch size* sebesar 64.

4.5.4 Prediksi Model P_LSTM

Setelah dibangun dengan parameter terbaik, model P_LSTM digunakan untuk menghasilkan prediksi harga saham. Hasilnya disajikan pada Tabel 36 sebagai berikut:

Tabel 36. Prediksi Model P_LSTM

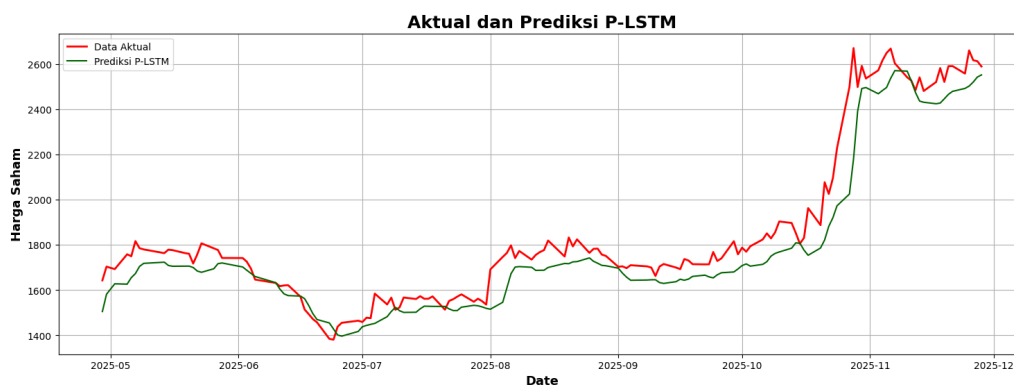
Tanggal	Prediksi
29/04/2025	0.6200
30/04/2025	0.6500
02/05/2025	0.7880
05/05/2025	0.7703
06/05/2025	0.7603
⋮	⋮
28/11/2025	0.8120

Nilai prediksi pada Tabel 36 masih berada pada skala normalisasi karena melalui proses *scaling*. Oleh karena itu, dilakukan *unscaling* untuk mengembalikannya ke skala asli agar mudah diinterpretasikan. Perbandingan data aktual dan hasil prediksi setelah *unscaling* disajikan pada Tabel 37 sebagai berikut:

Tabel 37. Hasil Prediksi Model P_LSTM

Tanggal	Aktual	Prediksi
29/04/2025	1.643	1.505.1345
30/04/2025	1.704	1.581.8604
02/05/2025	1.693	1.628.1234
05/05/2025	1.758	1.626.5517
06/05/2025	1.750	1.655.3437
⋮	⋮	⋮
28/11/2025	2.600	2576.5476

Tabel 37 menunjukkan bahwa perbedaan antara prediksi model P_LSTM dan data aktual tidak signifikan, sehingga model memiliki akurasi yang cukup baik. Visualisasi pola hasil prediksi ditampilkan pada Gambar 16 berikut:



Gambar 16. Plot Prediksi Model P_LSTM.

Plot pada Gambar 16 menunjukkan bahwa pola prediksi yang dihasilkan oleh model P_LSTM mampu mengikuti pola data aktual dengan cukup baik, meskipun masih terdapat sedikit perbedaan yang tidak terlalu signifikan. Ini menunjukkan model mampu melakukan prediksi dengan baik. Selain itu, tingkat keakuratan model juga dapat dinilai menggunakan ukuran evaluasi seperti RMSE, MAPE, dan akurasi. Hasil evaluasi model P_LSTM tersebut disajikan pada Tabel 38 sebagai berikut:

Tabel 38. Evaluasi Model P_LSTM

Evaluasi Model	Nilai
RMSE	2.0441
MAPE	1.040%
Akurasi	98.96%

Hasil evaluasi model P_LSTM pada Tabel 38, menunjukkan nilai RMSE dan MAPE yang sangat rendah, serta nilai akurasi sebesar 98,96% yang menunjukkan bahwa model P_LSTM memiliki performa yang sangat baik dalam melakukan prediksi.

4.5.5 Peramalan Model P_LSTM

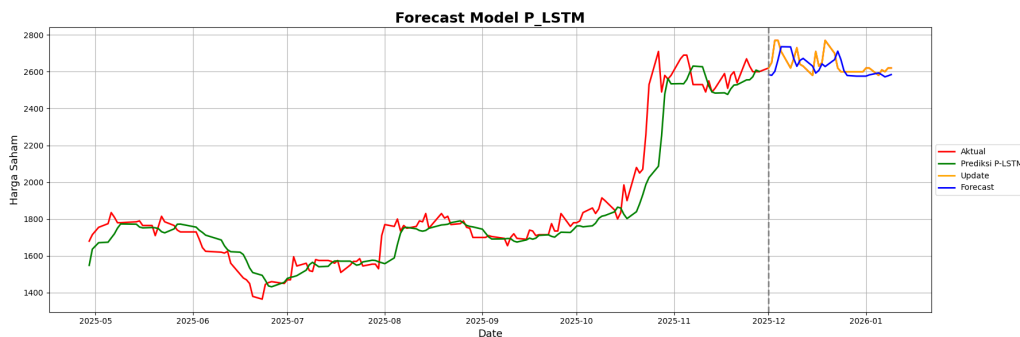
Selanjutnya dilakukan peramalan selama 30 hari ke depan menggunakan model P_LSTM. Hasil peramalan tersebut kemudian dibandingkan dengan data *update*

untuk mengevaluasi kinerja model dalam memprediksi data. Hasil peramalan tersebut disajikan pada Tabel 39 sebagai berikut.

Tabel 39. Hasil Peramalan Model P_LSTM

Tanggal	Update	Peramalan
01/12/2025	2.620	2.583.9500
02/12/2025	2.650	2.580.3706
03/12/2025	2.770	2.601.9226
04/12/2025	2.770	2.663.9990
05/12/2025	2.710	2.735.6516
⋮	⋮	⋮
09/01/2026	2.620	2.584.3490

Berdasarkan Tabel 39, masih terdapat perbedaan antara hasil peramalan model P_LSTM dengan data *update*. Visualisasi yang lebih jelas mengenai pola dan fluktuasi hasil peramalan model P_LSTM disajikan pada Gambar 17 sebagai berikut:



Gambar 17. Plot Peramalan Model P_LSTM.

Gambar 17 menampilkan plot hasil peramalan model P_LSTM yang menunjukkan bahwa pola hasil peramalan telah mengikuti pola data *update* dengan cukup baik. Hal ini mengindikasikan bahwa model P_LSTM memiliki kemampuan yang cukup baik dalam melakukan peramalan.

4.5.6 Prediksi Model *Hybrid* ARIMA–GARCH–EP_LSTM

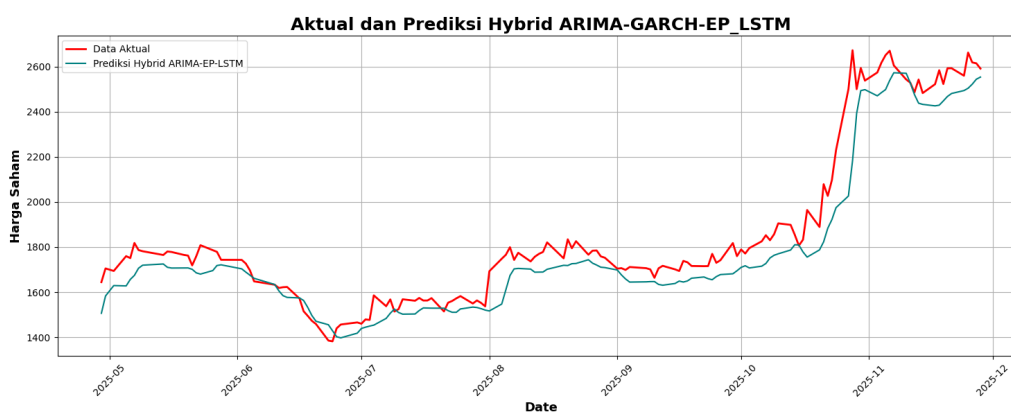
Model *hybrid* ARIMA–GARCH–EP_LSTM dibangun dari penggabungan model E_LSTM dengan model P_LSTM. Hasil prediksi model *hybrid*

ARIMA–GARCH–EP_LSTM disajikan pada Tabel 40 sebagai berikut:

Tabel 40. Hasil Prediksi Model *Hybrid* ARIMA–GARCH–EP_LSTM

Tanggal	Aktual	Prediksi
29/04/2025	1.643	1.505.1696
30/04/2025	1.704	1.581.8988
02/05/2025	1.693	1.628.1575
05/05/2025	1.758	1.626.5906
06/05/2025	1.750	1.655.3795
⋮	⋮	⋮
28/11/2025	2.600	2576.6162

Berdasarkan Tabel 40, terdapat perbedaan yang relatif kecil antara prediksi model *hybrid* ARIMA–GARCH–EP_LSTM dengan nilai data aktual. Visualisasi yang lebih jelas mengenai pola prediksi model *hybrid* ARIMA–GARCH–EP_LSTM disajikan pada Gambar 18 sebagai berikut:



Gambar 18. Plot Hasil Prediksi *Hybrid* ARIMA-GARCH-EP_LSTM.

Berdasarkan plot pada Gambar 18, hasil prediksi model *hybrid* ARIMA–GARCH–EP_LSTM menunjukkan pola yang sangat mendekati data aktual. Meskipun masih terdapat sedikit perbedaan antara nilai prediksi dan nilai aktual, selisih tersebut relatif kecil sehingga tidak memberikan pengaruh yang signifikan. Hal ini menunjukkan bahwa model *hybrid* ARIMA–GARCH–EP_LSTM memiliki kemampuan yang baik dalam menghasilkan prediksi yang akurat. Selain itu, tingkat keakuratan model juga dievaluasi menggunakan beberapa ukuran, yaitu RMSE, MAPE, dan akurasi.

Hasil perhitungan evaluasi model ARIMA–GARCH–EP_LSTM ditampilkan pada Tabel 41 sebagai berikut:

Tabel 41. Evaluasi Model *Hybrid* ARIMA-GARCH-EP_LSTM

Evaluasi Model	Nilai
RMSE	2.0187
MAPE	0.0279%
Akurasi	99.972%

Berdasarkan evaluasi model *hybrid* ARIMA–GARCH–EP_LSTM pada Tabel 41, menunjukkan nilai RMSE dan MAPE yang sangat rendah, serta nilai akurasi sebesar 99,9721%. Hal ini menegaskan bahwa model *hybrid* ARIMA–GARCH–EP_LSTM memiliki performa yang sangat baik dalam melakukan prediksi.

4.5.7 Peramalan Model *Hybrid* ARIMA–GARCH–EP_LSTM

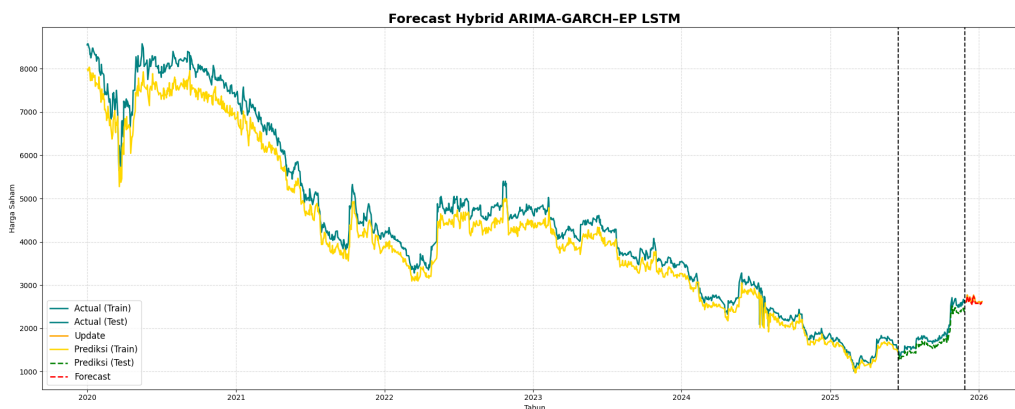
Hasil peramalan model *hybrid* ARIMA–GARCH–EP_LSTM diperoleh dengan menjumlahkan hasil peramalan dari model E.LSTM dan model P.LSTM. Selanjutnya, hasil peramalan tersebut dibandingkan dengan data *update* untuk mengevaluasi kinerja model dalam melakukan peramalan. Hasil peramalan untuk 30 hari ke depan menggunakan model *hybrid* ARIMA–GARCH–EP_LSTM disajikan pada Tabel 42 sebagai berikut:

Tabel 42. Hasil Peramalan Model *Hybrid* ARIMA-GARCH-EP_LSTM

Tanggal	<i>Update</i>	Peramalan
01/12/2025	2.620	2.576.1721
02/12/2025	2.650	2.584.9641
03/12/2025	2.770	2.603.7441
04/12/2025	2.770	2.663.9990
05/12/2025	2.710	2.735.6516
⋮	⋮	⋮
09/01/2026	2.620	2.584.3491

Pada Tabel 42 terlihat bahwa hasil peramalan model *hybrid* ARIMA–GARCH–EP_LSTM memiliki selisih yang relatif kecil dibandingkan

dengan data *update*. Visualisasi yang lebih jelas mengenai pola dan tren hasil peramalan model *hybrid* ARIMA–GARCH–EP LSTM disajikan pada Gambar 19 berikut:



Gambar 19. Plot Peramalan Model *Hybrid* ARIMA–GARCH–EP LSTM.

Berdasarkan Gambar 19, model *hybrid* ARIMA–GARCH–EP LSTM mampu mengikuti pola dan fluktuasi data dengan baik pada bagian prediksi data *training* dan *testing*. Pada bagian *forecasting*, hasil peramalan menunjukkan bahwa hasil peramalan sudah dapat mengikuti pola dan mampu merepresentasikan fluktuasi data dengan lebih baik dibandingkan model sebelumnya, meskipun masih terdapat sedikit perbedaan dengan data aktual yang tidak terlalu signifikan. Untuk mengevaluasi kinerja model dapat dilihat pada Tabel 43, Tabel 44, dan Tabel 45 sebagai berikut:

Tabel 43. Evaluasi *Train-Test*

Evaluasi Model	Nilai
RMSE	2.0187
MAPE	0.0279%
Akurasi	99.972%

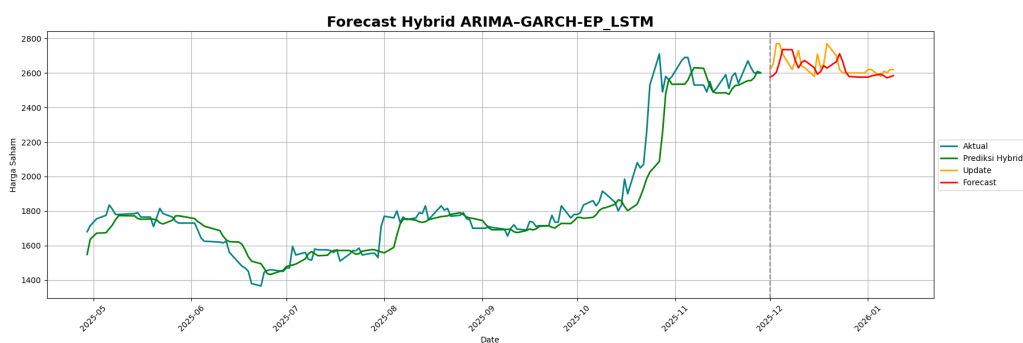
Tabel 44. Evaluasi *Forecast*

Evaluasi Model	Nilai
RMSE	3.2570
MAPE	0.7533%
Akurasi	99.246%

Tabel 45. Evaluasi *Train-Test-Forecast*

Evaluasi Model	Nilai
RMSE	1.8349
MAPE	1.0582%
Akurasi	98.941%

Berdasarkan hasil evaluasi pada Tabel di atas, model *hybrid* ARIMA–GARCH–EP_LSTM menunjukkan nilai MAPE dan RMSE yang relatif kecil dan tingkat akurasi yang tinggi pada seluruh skenario pengujian. Hal ini mengindikasikan bahwa model memiliki kemampuan yang baik dalam memprediksi data harga saham, sehingga model sudah mampu menangkap fluktuasi data dengan baik dibandingkan model sebelumnya. Untuk melihat hasil peramalan yang lebih jelas dapat dilihat pada Gambar 20 berikut:

Gambar 20. Plot Bagian *Forecasting Hybrid* ARIMA–GARCH–EP_LSTM.

Plot pada Gambar 20 menunjukkan bahwa hasil peramalan model *hybrid* ARIMA–GARCH–EP_LSTM cenderung mengikuti pola pergerakan data *update*. Meskipun nilai peramalannya belum sepenuhnya mendekati data *update*, dibandingkan dengan model lainnya, model ini menghasilkan peramalan yang paling mendekati data tersebut. Oleh karena itu, model *hybrid* ARIMA–GARCH–EP_LSTM dinilai cukup baik untuk digunakan dalam peramalan.

4.6 Goodness of Fit

Setelah diperoleh hasil peramalan dari model ARIMA–GARCH, model *hybrid* ARIMA–GARCH–E_LSTM, dan model *hybrid* ARIMA–GARCH–EP_LSTM,

selanjutnya dilakukan uji *goodness of fit* menggunakan uji *Kolmogorov-Smirnov* (KS) antara data *update* dan hasil peramalan dari ketiga model tersebut. Uji ini bertujuan untuk menilai tingkat kesesuaian distribusi antara data *update* dan data hasil peramalan melalui nilai selisih maksimum fungsi distribusi kumulatif keduanya. Semakin kecil nilai statistik KS yang diperoleh, maka semakin baik kesesuaian antara data *update* dan hasil peramalan. Hasil uji *goodness of fit* disajikan pada Tabel 46 sebagai berikut:

Tabel 46. *Goodness of Fit* Hasil Peramalan

Tanggal	<i>Update</i>	ARIMA–GARCH	<i>Hybrid</i> ARIMA– GARCH– E.LSTM	<i>Hybrid</i> ARIMA– GARCH– EP.LSTM
01/12/2025	2.620	2.597.020	2.639.031	2.576.172
02/12/2025	2.650	2.589.797	2.633.610	2.584.964
03/12/2025	2.770	2.577.612	2.686.501	2.603.744
04/12/2025	2.770	2.697.334	2.721.167	2.663.999
05/12/2025	2.710	2.678.172	2.694.032	2.735.652
⋮	⋮	⋮	⋮	⋮
09/01/2026	2.620	2.597.531	2.651.299	2.584.349
Statistik Uji KS		0.666667	0.900000	0.400000

Berdasarkan Tabel 46, hasil peramalan model *hybrid* ARIMA–GARCH–EP.LSTM untuk variabel harga saham menunjukkan nilai statistik uji KS yang paling kecil (0,40000) dibandingkan model ARIMA–GARCH (0.6667) dan model *hybrid* ARIMA –GARCH–E.LSTM (0.9000). Hal ini menunjukkan bahwa model *hybrid* ARIMA– GARCH– EP.LSTM merupakan model terbaik dalam meramalkan harga penutupan saham harian PT Unilever Indonesia Tbk.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Model *hybrid* ARIMA–GARCH–LSTM yang digunakan untuk melakukan peramalan dikembangkan melalui dua pendekatan, yaitu ARIMA–GARCH–E.LSTM dan ARIMA–GARCH–EP.LSTM. Model ARIMA–GARCH–E.LSTM diperoleh dengan menggabungkan hasil peramalan dari model ARIMA–GARCH dan model E.LSTM. Sementara itu, model ARIMA–GARCH–EP.LSTM dibentuk dari penggabungan *output* model E.LSTM dan model P.LSTM. Model E.LSTM merupakan model LSTM yang dibangun menggunakan residual dari model ARIMA yang selanjutnya dimodelkan dengan GARCH. Sebaliknya, model P.LSTM dibangun berdasarkan nilai prediksi yang dihasilkan oleh model ARIMA. Pada kedua pendekatan tersebut, model LSTM dikembangkan menggunakan kombinasi parameter terbaik yang diperoleh melalui proses *hyperparameter tuning*, dengan kombinasi parameter sebagai berikut:

1. Model E.LSTM menggunakan kombinasi parameter optimal berupa 64 unit LSTM, 32 unit *dense*, *batch size* sebesar 128, serta *dropout* sebesar 0,2.
2. Model P.LSTM menggunakan kombinasi parameter terbaik yang terdiri dari 32 unit LSTM, 32 unit *dense*, dan *batch size* sebesar 64.

Peramalan untuk periode 30 hari ke depan, yaitu dari 1 Desember 2025 hingga 9 Januari 2026, menggunakan model *hybrid* ARIMA–GARCH–EP.LSTM menunjukkan perbedaan yang cukup jelas dibandingkan dengan model ARIMA–GARCH maupun model *hybrid* ARIMA–GARCH–E.LSTM. Model ARIMA–GARCH menghasilkan pola peramalan yang cenderung linear selama periode peramalan sehingga belum mampu mengikuti fluktuasi pada data *update*. Sementara itu, model *hybrid* ARIMA–GARCH–E.LSTM mulai

memperlihatkan adanya variasi dalam pola pergerakan, tetapi kemampuan model tersebut dalam merepresentasikan fluktuasi data *update* masih belum optimal. Sebaliknya, model *hybrid* ARIMA–GARCH–EP_LSTM mampu mengikuti pola fluktuasi data *update* dengan lebih baik. Oleh karena itu, model *hybrid* ARIMA–GARCH–EP_LSTM dapat dinilai sebagai model yang paling baik dalam menghasilkan peramalan dibandingkan dengan model ARIMA–GARCH maupun model *hybrid* ARIMA–GARCH–E_LSTM.

Berdasarkan hasil uji *goodness of fit* menggunakan uji *Kolmogorov-Smirnov*, model *hybrid* ARIMA–GARCH–EP_LSTM menghasilkan nilai statistik uji KS yang paling kecil dibandingkan dengan model ARIMA–GARCH maupun model *hybrid* ARIMA–GARCH–E_LSTM pada variabel harga saham. Hal ini menunjukkan bahwa model *hybrid* ARIMA–GARCH–EP_LSTM memiliki kemampuan yang lebih baik dalam menyesuaikan distribusi hasil peramalan dengan distribusi data *update*. Oleh karena itu, dapat disimpulkan bahwa model *hybrid* ARIMA–GARCH–EP_LSTM merupakan model dengan kinerja terbaik dalam melakukan peramalan dibandingkan model ARIMA–GARCH maupun model *hybrid* ARIMA–GARCH–E_LSTM.

DAFTAR PUSTAKA

- Adhikari, R., & Agrawal, R. K. (2013). *An Introductory Study on Time Series Modeling and Forecasting*. LAP Lambert Academic Publishing.
- Alpaydin, E. (2004). *Introduction to Machine Learning*. MIT Press.
- Anisa, I. Y., & Yulianto, T. (2025). Application of the ARIMA–GARCH Model for Forecasting Indonesia’s Monthly Inflation Rate. *Contemporary Mathematics and Applications*, **7**(1), 1–12.
- Beeg, F. A. F., Paendong, M. S., & Mananohas, M. L. (2024). Penerapan Model ARIMA–GARCH untuk Peramalan Harga Emas Dunia. *d’Cartesian: Jurnal Matematika dan Aplikasi*, **13**(2), 73–79.
- Bollerslev, T. (1986). Generalized Autoregressive Conditional Heteroskedasticity. *Journal of Econometrics*, **31**(3), 307–327.
- Box, G. E. P., & Jenkins, G. M. (1976). *Time Series Analysis: Forecasting and Control*. San Francisco, CA: Holden-Day.
- Darmadji, T., & Fakhruddin, H. M. (2012). *Pasar Modal di Indonesia: Pendekatan Tanya Jawab*. Salemba Empat.
- Dickey, D. A., & Fuller, W. A. (1979). Distribution of the Estimators for Autoregressive Time Series with a Unit Root. *Journal of the American Statistical Association*, **74**(366), 427–431.
- Ding, Z., Granger, C. W. J., & Engle, R. F. (1993). A Long Memory Property of Stock Market Returns and A New Model. *Journal of Empirical Finance*, **1**(1), 83–106.

- Engle, R. F. (1982). Autoregressive Conditional Heteroscedasticity With Estimates of the Variance of United Kingdom Inflation. *Econometrica: Journal of the econometric society*, **50**(4), 987–1007.
- Engle, R. F. (2001). GARCH 101: The Use of ARCH/GARCH Models in Applied Econometrics. *Journal of Economic Perspectives*, **15**(4), 157–168.
- Gospodinov, N., Gavala, A., & Jiang, D. (2006). Forecasting Volatility. *Journal of Forecasting*, **25**(6), 381–400.
- Gunawan, D. (2023). Peramalan Harga Saham Syariah PT Unilever Indonesia Tbk. Menggunakan GARCH. *Krigan: Journal of Management and Sharia Business*, **1**(2), 88–101.
- Hanke, J. E., & Wichern, D. W. (2005). *Business Forecasting* (8th ed.). Pearson Prentice Hall.
- Hariadi, W., & Sulantari, S. (2021). Application of ARIMA Model for Forecasting Additional Positive Cases of COVID-19 in Jember Regency. *Enthusiastic: International Journal of Applied Statistics and Data Science*, **1**(1), 20–27.
- Haykin, S. (2009). *Neural Networks and Learning Machines* (3rd ed.). Pearson Education.
- Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, **9**(8), 1735–1780.
- Jogiyanto. (2017). *Teori Portofolio dan Analisis Investasi* (Edisi ke-11). BPFE Yogyakarta.
- Kolmogorov, A. (1933). Sulla Determinazione Empirica di Una Legge di Distribuzione. *Giornale dell'Istituto Italiano degli Attuari*, **4**, 89–91.
- Kumar, J., Dalal, N., & Sethi, M. (2025). Hyperparameter Selection and Optimization in Deep Learning Models. *Journal of Artificial Intelligence Research*, **72**, 215–240.
- Lattifa, T., Buana, P. W., & Rusjyanthi, N. K. D. (2022). Model Prediksi Cuaca Menggunakan Metode LSTM. *Jurnal Ilmu Teknologi dan Komputer (JITTER)*, **3**(1), 994–1000.

- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep Learning. *Nature*, **521**(7553), 436–444.
- Ljung, G. M., & Box, G. E. P. (1978). On A Measure of Lack of Fit in Time Series Models. *Biometrika*, **65**(2), 297–303.
- Montgomery, D. C., Jennings, C. L., & Kulahci, M. (2015). *Introduction to Time Series Analysis and Forecasting* (2nd ed.). Wiley.
- Pankratz, A. (1991). *Forecasting with Dynamic Regression Models*. Wiley–Interscience.
- Rafulta, E., Yanuar, F., Devianto, D., & Maiyastri. (2025). Pemodelan dan Peramalan Volatilitas Memori Panjang pada Return Saham ANTM: Studi Komparatif Model GARCH dan FIGARCH. *Journal of Mathematics Education and Applied*, **5**(1), 75–76.
- Rowan, R., Muflikhah, L., & Cholissodin, I. (2022). Peramalan Kasus Positif COVID–19 di Jawa Timur Menggunakan Metode Hybrid ARIMA–LSTM. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, **6**(9), 4146–4153.
- Schmidhuber, J. (2015). Deep Learning in Neural Networks: An Overview. *Neural Networks*, **61**, 85–117.
- Sherstinsky, A. (2020). Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network. *Physica D: Nonlinear Phenomena*, **404**, 132–306.
- Torres, J. F., Martínez-Álvarez, F., & Troncoso, A. (2022). A Deep LSTM Network for Spanish Electricity Consumption Forecasting. *Neural Computing and Applications*, **34**(13), 10533–10545.
- Wei, W. W. (2006). *Time Series Analysis: Univariate and Multivariate Methods* (2nd ed.). Pearson Addison–Wesley.
- Yang, L., Shami, A., & El Saddik, A. (2023). On Hyperparameter Optimization of Machine Learning Algorithms: A Recent Review. *IEEE Access*, **11**, 123456–123470.

- Yanti, F., Sari, B. N., & Defiyanti, S. (2024). Implementasi Algoritma Lstm Pada Peramalan Stok Obat. *JATI (Jurnal Mahasiswa Teknik Informatika)*, **8**(4), 6082-6089.
- Zhang, G. P. (2003). Time Series Forecasting Using a Hybrid ARIMA and Neural Network Model. *Neurocomputing*, **50**, 159–175.
- Zhang, Y., Liu, X., & Chen, W. (2024). Hyperparameter Tuning Strategies for Deep Learning Models: A Systematic Review. *Applied Soft Computing*, **146**, 110–782.
- Zhang, Z., Dong, Z., Lin, H., He, Z., Wang, M., He, Y., & Gao, M. (2021). An Improved Bidirectional Gated Recurrent Unit Method for Accurate State-of-Charge Estimation. *IEEE Access*, **9**, 11252–11263.