

**IMPLEMENTASI METODE *HYBRID AUTOREGRESSIVE INTEGRATED
MOVING AVERAGE (ARIMA) – LONG SHORT TERM MEMORY (LSTM)*
DALAM PERAMALAN HARGA SAHAM BANK BRI**

(Skripsi)

Oleh

SILVI FITRIANI



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2023**

ABSTRACT

THE IMPLEMENTATION OF HYBRID AUTOREGRESSIVE INTEGRATED MOVING AVERAGE (ARIMA)– LONG SHORT TERM MEMORY (LSTM) METHOD IN BANK BRI STOCK PRICE FORECASTING

By

SILVI FITRIANI

Stock of BRI is one of the most popular investments by investors because stock of BRI is included in the Lq45 stock index. Investing in stocks is not easy and risky because of the frequent price fluctuations of stock prices. One of the things that can be done to help investors for not making wrong decisions is by forecasting the stock prices. One of the method that can be used to forecast stock prices is the Autoregressive Integrated Moving Average (ARIMA) method. However, ARIMA has a flaw because it makes linearity an assumption and the resulting residuals still contain nonlinear elements. Another method that can be used in stock price forecasting is Long Short Term Memory (LSTM). The LSTM method has advantages over the ARIMA method, which is capable of processing both linear and nonlinear data. So, in this study applies the ARIMA-LSTM hybrid method to forecast stock prices. The ARIMA-LSTM hybrid method consists of 2 main models, the first model which predicts ARIMA prediction data and the second model which predicts ARIMA residuals. In this study, the hybrid model with splitting data 90% training data 10% testing data produces the highest level of accuracy compared to the hybrid model with splitting data 80% training data 20% testing data. The best hybrid model in this study produced an RMSE value of 126,6367 and a MAPE of 0,0228%.

Keywords: Forecasting, Stocks, Hybrid ARIMA-LSTM

ABSTRAK

IMPLEMENTASI METODE *HYBRID AUTOREGRESSIVE INTEGRATED MOVING AVERAGE (ARIMA) – LONG SHORT TERM MEMORY (LSTM)* DALAM PERAMALAN HARGA SAHAM BANK BRI

Oleh

SILVI FITRIANI

Saham BRI merupakan salah satu investasi yang paling banyak diminati oleh investor karena saham BRI tergabung dalam indeks saham Lq45. Investasi pada saham bukanlah hal yang mudah dan memiliki resiko kerugian karena harga saham sering kali mengalami kenaikan maupun penurunan harga. Salah satu hal yang dapat dilakukan untuk membantu para investor agar tidak salah dalam pengambilan keputusan ketika berinvestasi adalah dengan melakukan peramalan pada harga saham. Salah satu metode yang dapat digunakan untuk meramalkan harga saham adalah metode *Autoregressive Integrated Moving Average (ARIMA)*. Namun, ARIMA memiliki kelemahan yaitu menjadikan linearitas sebagai asumsi dan residual yang dihasilkan masih terdapat unsur nonlinear. Metode lain yang dapat digunakan pada peramalan harga saham adalah *Long Short Term Memory (LSTM)*. Metode LSTM memiliki kelebihan dibandingkan metode ARIMA, yaitu mampu memproses data linear maupun nonlinear. Sehingga pada penelitian ini menerapkan metode *hybrid ARIMA-LSTM* untuk meramalkan harga saham. Metode *hybrid ARIMA-LSTM* terdiri dari 2 model utama yaitu model pertama yang memprediksi dan meramalkan data prediksi ARIMA dan model kedua yaitu model yang memprediksi dan meramalkan residual ARIMA. Model *hybrid* dengan *splitting data 90% data training 10% data testing* pada penelitian ini menghasilkan tingkat akurasi paling tinggi dibandingkan dengan model *hybrid* dengan *splitting data 80% data training 20% data testing*. Model *hybrid* terbaik pada penelitian menghasilkan nilai RMSE sebesar 126,6367 dan MAPE sebesar 0,0228%.

Kata kunci: Peramalan, Saham, *Hybrid ARIMA-LSTM*

**IMPLEMENTASI METODE *HYBRID AUTOREGRESSIVE INTEGRATED
MOVING AVERAGE (ARIMA) – LONG SHORT TERM MEMORY (LSTM)*
DALAM PERAMALAN HARGA SAHAM BANK BRI**

Oleh

Silvi Fitriani

Skripsi

**Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA MATEMATIKA**

Pada

**Jurusan Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Lampung**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2023**

Judul : **IMPLEMENTASI METODE *HYBRID*
AUTOREGRESSIVE INTEGRATED MOVING
AVERAGE (ARIMA) - LONG SHORT TERM
MEMORY (LSTM) DALAM PERAMALAN
HARGA SAHAM BANK BRI**

Nama : **Silvi Fitriani**

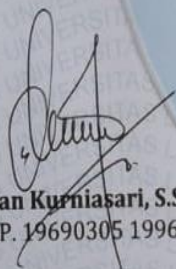
Nomor Pokok Mahasiswa : **1917031054**

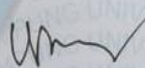
Jurusan : **Matematika**

Fakultas : **Matematika dan Ilmu Pengetahuan Alam**

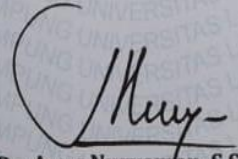


1. Komisi Pembimbing


Dian Kurniasari, S.Si., M.Sc.
NIP. 19690305 199603 2 001


Ir. Warsono, M.S., Ph.D.
NIP. 19630216 198703 1 003

2. Ketua Jurusan Matematika


Dr. Aang Nuryaman, S.Si., M.Si.
NIP. 19740316 200501 1 001

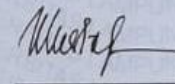
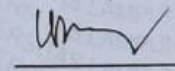
MENGESAHKAN

1. Tim Penguji

Ketua : **Dian Kurniasari, S.Si., M.Sc.**

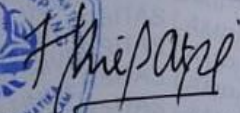
Sekretaris : **Ir. Warsono, M.S., Ph.D.**

Penguji
Bukan Pembimbing : **Prof. Drs. Mustofa Usman, MA., Ph.D.**



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam




Dr. Eng. Heri Satria, S.Si., M.Si.
NIP. 19711001 200501 1 002

Tanggal Lulus Ujian Skripsi : **15 Juni 2023**

PERNYATAAN

Yang bertanda tangan dibawah ini:

Nama : **Silvi Fitriani**
Nomor Pokok Mahasiswa : **1917031054**
Jurusan : **Matematika**
Judul Skripsi : **IMPLEMENTASI METODE *HYBRID*
AUTOREGRESSIVE INTEGRATED
MOVING AVERAGE (ARIMA) – LONG
SHORT TERM MEMORY (LSTM) DALAM
PERAMALAN HARGA SAHAM BANK
BRI**

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

Bandar Lampung, 15 Juni 2023

Penulis,



Silvi Fitriani
NPM. 1917031054

RIWAYAT HIDUP

Penulis bernama Silvi Fitriani, lahir pada tanggal 28 Desember 2001 di Negara Batin, Way Kanan. Penulis merupakan putri sulung dari Bapak Tusiman dan Ibu Sri Ningsih.

Penulis pertama kali menempuh pendidikan di Taman Kanak-Kanak Aba Purwa Agung pada tahun 2006-2007. Lalu menempuh pendidikan Sekolah Dasar di SDN 1 Purwa Negara pada tahun 2007-2013. Kemudian melanjutkan ke Sekolah Menengah Pertama (SMP) di SMPN 1 Negara Batin pada tahun 2013-2016. Selanjutnya penulis belajar pada jenjang Sekolah Menengah Atas (SMA) di SMAS Al-Kautsar Bandar Lampung pada tahun 2016-2019. Pada tahun 2019 penulis terdaftar sebagai mahasiswa Program Studi S1 Jurusan Matematika dan Ilmu Pengetahuan Alam Universitas Lampung melalui jalur SBMPTN.

Pada bulan Januari 2022 penulis melaksanakan Kuliah Kerja Nyata (KKN) selama 40 hari di Desa Pakuan Sakti, Kecamatan Pakuan Ratu, Kabupaten Way Kanan sebagai salah satu bentuk pengabdian kepada masyarakat. Lalu pada bulan Juli 2022 penulis melaksanakan Kuliah Kerja Praktik (KP) di Badan Pengelola Pajak dan Retribusi Daerah (BPPRD) selama 40 hari.

KATA INSPIRASI

“Apa yang melewatkanmu tidak ada pernah menjadi takdirmu dan apa yang ditakdirkan untukmu tidak ada pernah melewatkanmu”

(Umar bin Khattab)

PERSEMBAHAN

Alhamdulillah, puji dan syukur kepada Allah SWT atas nikmat serta hidayahnya sehingga skripsi ini dapat terselesaikan dengan baik dan tepat pada waktunya.

Oleh karena itu, dengan rasa syukur dan bahagia saya persembahkan rasa terima-kasih saya kepada

Ibu dan Bapak

Terimakasih untuk kedua orangtuaku yang selalu mendukung semua hal yang aku inginkan. Terimakasih atas dukungan yang selalu diberikan baik dalam bentuk doa, kasih sayang maupun secara finansial. Terimalah bukti kecil ini sebagai kado keseriusanku untuk membalas semua pengorbanan, keikhlasan, dan jerih payah yang selama ini kalian lakukan.

Dosen Pembimbing dan Pembahas

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

Almamater Tercinta Universitas Lampung

SANWACANA

Segala puji dan syukur penulis ucapkan kepada Allah SWT atas segala nikmat dan karunia-Nya yang tak terhingga sehingga penulis dapat menyelesaikan skripsi yang berjudul ***“Implementasi Metode Hybrid Autoregressive Integrated Moving Average (ARIMA) – Long Short Term Memory (LSTM) Dalam Peramalan Harga Saham Bank BRI”***. Dalam penulisan skripsi ini tidak dapat terselesaikan tanpa adanya bimbingan, bantuan dan dukungan dari berbagai pihak. Sehingga, dalam kesempatan ini penulis mengucapkan terimakasih kepada:

1. Ibu Dian Kurniasari S.Si., M.Sc. selaku dosen pembimbing I yang senantiasa membimbing dengan sabar, memberi masukan serta saran serta mendukung penulis dalam menyelesaikan skripsi ini.
2. Bapak Ir. Warsono, M.S., Ph.D. selaku dosen pembimbing II yang senantiasa memberikan bimbingan, arahan, serta saran sehingga penulis dapat menyelesaikan skripsi ini.
3. Bapak Prof. Drs. Mustofa Usman, MA., Ph.D. selaku dosen penguji yang telah memberikan kritik dan saran yang membangun sehingga skripsi ini dapat terselesaikan.
4. Bapak Dr.Aang Nuryaman, S.Si., M.Si. selaku Ketua Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
5. Bapak Dr. Eng. Suropto Dwi Yuwono, S.Si., M.T. selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
6. Seluruh dosen, staff, karyawan Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
7. Teruntuk kedua orang tuaku tercinta dan semua keluargaku, terimakasih atas doa, dukungan, pengorbanan, cinta kasih, perhatian, demi kesuksesan penulis

semoga dikemudian hari dapat membahagiakan dan menjadi kebanggaan kalian.

8. Teruntuk sahabatku Ghina terimakasih selalu ada dan selalu memberikan semangat dan menemani penulis dalam keadaan apapun
9. Teruntuk sahabat kuliahku, Alfira, Melisa, Niken, Widya, Nur Azizah, Nurul, Dhifa, Yusril, Rahma Adji, terimakasih atas dukungan, bantuan, dan waktu yang diberikan terhadap penulis sejak awal perkuliahan hingga selesai.
10. Teruntuk sahabat, teman seperbimbingan, dan seperjuanganku Azzahra, terimakasih telah menjadi orang yang selalu mendengarkan keluh kesah penulis selama mengerjakan skripsi ini dan telah mengajarkan banyak hal semenjak masa awal perkuliahan.
11. Orang-orang baik yang namanya tidak bisa saya sebutkan satu persatu yang telah menjadi teman terbaik penulis yang selalu memberikan semangat dan menemani penulis dalam keadaan apapun serta telah memberikan pengalaman dan banyak cerita selama masa perkuliahan.
12. Teman-teman seperbimbinganku Fiqih, Dea, Mia, Tiara, Putri, Lisna, Zida, Anin, terimakasih atas doa, motivasi, dukungan, semangat, yang selalu di berikan kepada penulis. Semoga kalian menjadi orang yang sukses dan selalu bermanfaat bagi orang disekeliling kalian.

Penulis menyadari masih banyak kekurangan dalam penulisan skripsi ini. Oleh karena itu, penulis mengharapkan masukan serta saran untuk dijadikan pelajaran kedepannya.

Bandar Lampung, Juni 2023

Penulis,

Silvi Fitriani

DAFTAR ISI

	Halaman
DAFTAR TABEL	xv
DAFTAR GAMBAR	xvii
I. PENDAHULUAN	1
1.1 Latar Belakang dan Masalah.....	1
1.2 Tujuan Penelitian	4
1.3 Manfaat Penelitian	5
II. TINJAUAN PUSTAKA	6
2.1 Peramalan	6
2.2 <i>Time Series</i>	7
2.3 Stasioneritas	11
2.4 Data Mining	12
2.5 <i>Machine Learning</i>	14
2.6 <i>Deep Learning</i>	15
2.6.1 <i>Recurrent Neural Network (RNN)</i>	16
2.6.2 <i>Long Short Term Memory (LSTM)</i>	17
2.7 Parameter <i>Long Short Term Memory (LSTM)</i>	21
2.8 <i>Hybrid ARIMA-LSTM</i>	22
2.9 Akurasi	24
III. METODOLOGI PENELITIAN	27
3.1 Waktu dan Tempat Penelitian.....	27
3.2 Spesifikasi Perangkat	27
3.3 Data Penelitian	27
3.4 Metode Penelitian.....	28
IV. HASIL DAN PEMBAHASAN	32
4.1 <i>Input Data</i>	32
4.2 Visualisasi Data.....	33
4.3 Proses Prediksi Data dengan ARIMA	34
4.3.1 <i>Preprocessing Data I</i>	35
4.3.2 Uji Stasioner	35

4.3.3	Identifikasi model	38
4.3.4	Estimasi Parameter Model dan Uji Diagnostik Model.....	39
4.3.5	Prediksi ARIMA	40
4.3.6	Menghitung Residual	42
4.4	Proses <i>Hybrid</i> ARIMA-LSTM	43
4.4.1	Membangun Model LSTM Pertama	46
4.4.2	<i>Hyperparameter Tuning</i> Model LSTM Pertama	46
4.4.3	<i>Hyperparameter Tuning</i> Model LSTM Kedua.....	49
4.4.4	Prediksi Model LSTM Pertama	51
4.4.5	Prediksi Model LSTM Kedua.....	54
4.4.6	Kombinasi Prediksi Model Pertama dan Model Kedua LSTM (<i>Hybrid</i>).....	57
4.5	Membangun Model LSTM.....	60
4.5.1	<i>Hyperparameter Tuning</i> Model LSTM.....	61
4.5.2	Prediksi Model LSTM.....	63
4.6	Hasil Peramalan	66
4.6.1	Hasil Peramalan LSTM	66
4.6.2	Hasil Peramalan <i>Hybrid</i> ARIMA-LSTM	69
4.6.2.1	Hasil Peramalan Model LSTM Pertama	70
4.6.2.2	Hasil Peramalan Model LSTM Kedua.....	71
4.6.2.3	Hasil Peramalan Model <i>Hybrid</i>	73
V.	KESIMPULAN	77
	DAFTAR PUSTAKA	79
	LAMPIRAN	

DAFTAR TABEL

Tabel	Halaman
1. Data Harga Saham BRI	28
2. Data <i>Input</i>	32
3. Statistika Deskriptif Variabel <i>Close</i>	33
4. Hasil <i>Differencing</i> Pertama	36
5. Pemilihan Model ARIMA Terbaik	39
6. Hasil Prediksi ARIMA	41
7. Perbandingan Data Aktual dan Prediksi ARIMA.....	41
8. Nilai Residual	42
9. Perbandingan Data Prediksi ARIMA dan Data Prediksi Model LSTM Pertama dengan <i>Splitting</i> 80% Data <i>Training</i> 20% Data <i>Testing</i>	51
10. Perbandingan Data Prediksi ARIMA dan Data Prediksi Model LSTM Pertama dengan <i>Splitting</i> 90% Data <i>Training</i> 10% Data <i>Testing</i>	52
11. Evaluasi Model LSTM Pertama	54
12. Perbandingan Data Residual ARIMA dan Data Prediksi Residual Model LSTM Kedua dengan <i>Splitting</i> 80% Data <i>Training</i> 20% Data <i>Testing</i>	55
13. Perbandingan Data Residual ARIMA dan Data Prediksi Residual Model LSTM Kedua dengan <i>Splitting</i> 90% Data <i>Training</i> 10% Data <i>Testing</i>	56

14. Evaluasi Model LSTM Kedua	57
15. Perbandingan Data Aktual dan Data Prediksi Model <i>Hybrid</i> dengan <i>Splitting</i> 80% Data <i>Training</i> 20% Data <i>Testing</i>	58
16. Perbandingan Data Aktual dan Data Prediksi Model <i>Hybrid</i> dengan <i>Splitting</i> 90% Data <i>Training</i> 10% Data <i>Testing</i>	59
17. Evaluasi Model <i>Hybrid</i>	60
18. Perbandingan Data Aktual dan Data Prediksi Model LSTM dengan <i>Splitting</i> 80% data <i>Training</i> 20% data <i>Testing</i>	63
19. Perbandingan Data Aktual dan Data Prediksi Model LSTM dengan <i>Splitting</i> 90% data <i>Training</i> 10% data <i>Testing</i>	64
20. Evaluasi Model LSTM	66
21. Perbandingan Hasil Peramalan Model <i>Hybrid</i> dengan <i>Splitting</i> 80% Data <i>Training</i> 20% Data <i>Testing</i> dengan Data <i>Ter-update</i>	74
22. Perbandingan Hasil Peramalan Model <i>Hybrid</i> dengan <i>Splitting</i> 90% Data <i>Training</i> 10% Data <i>Testing</i> dengan Data <i>Ter-update</i>	75

DAFTAR GAMBAR

Gambar	Halaman
1. Arsitektur <i>Recurrent Neural Network</i>	17
2. Struktur <i>Long Short Term Memory</i>	18
3. Diagram Alir Metode Penelitian.....	31
4. Plot Data Historis Harga Penutupan Saham BRI.....	34
5. Syntax untuk Mengecek <i>Missing Value</i>	35
6. Syntax untuk Mengisi <i>Missing Value</i>	35
7. Syntax untuk Uji Stasioner Data.....	36
8. Syntax untuk <i>Differencing</i> Pertama	36
9. Plot Data <i>Differencing</i> Pertama	37
10. Syntax untuk Uji Stasioner Data <i>Differencing</i>	37
11. Syntax untuk Menampilkan Plot ACF dan PACF	38
12. Plot ACF dan PACF.....	38
13. Syntax untuk Menampilkan <i>Summary</i> Model ARIMA	40
14. <i>Summary</i> Model ARIMA(2, 1, 2).....	40
15. Syntax untuk Prediksi ARIMA.....	40

16. Syntax untuk <i>Undifferencing</i> Prediksi ARIMA	41
17. Plot Perbandingan Data Aktual dengan Prediksi ARIMA	42
18. Plot Residual	43
19. Syntax <i>Splitting</i> Data Prediksi dan Residual 90% <i>Training</i> 10% <i>Testing</i>	44
20. Syntax <i>Splitting</i> Data Prediksi dan Residual 80% <i>Training</i> 20% <i>Testing</i>	44
21. Syntax Normalisasi Data Prediksi ARIMA	45
22. Syntax Normalisasi Data Residual ARIMA	45
23. <i>Hyperparameter Tuning</i> Model LSTM Pertama dengan <i>Splitting Data</i> 80% Data <i>Training</i> 20% Data <i>Testing</i>	47
24. <i>Hyperparameter Tuning</i> Model Pertama LSTM dengan <i>Splitting Data</i> 90% Data <i>Training</i> 10% Data <i>Testing</i>	48
25. <i>Hyperparameter Tuning</i> Model LSTM Kedua dengan <i>Splitting Data</i> 80% Data <i>Training</i> 20% Data <i>Testing</i>	49
26. <i>Hyperparameter Tuning</i> Model LSTM Kedua dengan <i>Splitting Data</i> 90% Data <i>Training</i> 10% Data <i>Testing</i>	50
27. Plot Prediksi Model LSTM Pertama dengan <i>Splitting Data</i> 80% Data <i>Training</i> 20% Data <i>Testing</i>	52
28. Plot Prediksi Model Pertama LSTM dengan <i>Splitting Data</i> 90% data <i>Training</i> 10% data <i>Testing</i>	53
29. Plot Prediksi Model LSTM Kedua dengan <i>Splitting Data</i> 80% Data <i>Training</i> 20% Data <i>Testing</i>	55
30. Plot Prediksi Model Kedua LSTM dengan <i>Splitting Data</i> 90% data <i>Training</i> 10% data <i>Testing</i>	56
31. Plot Prediksi Model <i>Hybrid</i> dengan <i>Splitting Data</i> 80% Data <i>Training</i> 20% Data <i>Testing</i>	58

32. Plot Prediksi Model <i>Hybrid</i> dengan <i>Splitting Data</i> 90% Data <i>Training</i> 10% Data <i>Testing</i>	59
33. <i>Hyperparameter Tuning</i> Model LSTM dengan <i>Splitting Data</i> 80% Data <i>Training</i> 20% Data <i>Testing</i>	61
34. <i>Hyperparameter Tuning</i> Model LSTM dengan <i>Splitting Data</i> 90% Data <i>Training</i> 10% Data <i>Testing</i>	62
35. Plot Prediksi Model LSTM dengan <i>Splitting Data</i> 80% data <i>Training</i> 20% data <i>Testing</i>	64
36. Plot Prediksi Model LSTM dengan <i>Splitting Data</i> 90% data <i>Training</i> 10% data <i>Testing</i>	65
37. Proses Peramalan Model LSTM	67
38. Plot Peramalan Model LSTM dengan <i>Splitting Data</i> 80% Data <i>Training</i> 20% Data <i>Testing</i>	68
39. Plot Peramalan Model LSTM dengan <i>Splitting Data</i> 90% data <i>Training</i> 10% data <i>Testing</i>	69
40. Plot Peramalan Model Pertama LSTM dengan <i>Splitting Data</i> 80% data <i>Training</i> 20% data <i>Testing</i>	70
41. Plot Peramalan Model Pertama LSTM dengan <i>Splitting Data</i> 90% data <i>Training</i> 10% data <i>Testing</i>	71
42. Plot Peramalan Model Kedua LSTM dengan <i>Splitting Data</i> 80% data <i>Training</i> 20% data <i>Testing</i>	72
43. Plot Peramalan Model Kedua LSTM dengan <i>Splitting Data</i> 90% data <i>Training</i> 10% data <i>Testing</i>	72
44. Plot Peramalan Model <i>Hybrid</i> dengan <i>Splitting Data</i> 80% data <i>Training</i> 20% data <i>Testing</i>	73
45. Plot Peramalan Model <i>Hybrid</i> dengan <i>Splitting Data</i> 90% data <i>Training</i> 10% data <i>Testing</i>	76

I. PENDAHULUAN

1.1 Latar Belakang dan Masalah

Saham saat ini menjadi salah satu bentuk investasi yang paling banyak diminati oleh investor. Saham merupakan surat berharga tanda penyertaan modal seseorang dalam suatu perusahaan atau Perseroan Terbatas (PT). Saham perbankan merupakan sektor saham yang banyak diminati investor. Terdapat banyak perusahaan perbankan, baik milik negara maupun swasta. Salah satu bank terbesar milik negara adalah Bank Rakyat Indonesia (BRI). Bank Rakyat Indonesia (BRI) merupakan bank yang didirikan oleh Raden Aria Wirjaatmadja pada tanggal 16 Desember 1895 di Purwokerto, Jawa Tengah. Saham BRI merupakan saham unggulan di Bursa Efek Indonesia karena saham BRI tergabung dalam indeks saham Lq45. Selain itu, BRI memiliki kantor cabang serta mesin ATM yang tersebar di seluruh wilayah Indonesia, tidak terbatas di kota atau desa besar saja. Berdasarkan hal tersebut, banyak investor yang tertarik untuk berinvestasi pada BRI. Namun, investasi pada saham bukanlah hal yang mudah dan memiliki resiko kerugian karena harga saham sering kali mengalami kenaikan maupun penurunan

Salah satu hal yang dapat dilakukan untuk membantu para investor agar tidak salah dalam pengambilan keputusan ketika berinvestasi adalah dengan melakukan peramalan pada pergerakan harga saham. Harga saham termasuk dalam kategori data *time series*. Peramalan *time series* merupakan pengamatan data masa lalu dari suatu variabel, dikumpulkan dan dianalisis untuk mengembangkan model yang menggambarkan hubungan yang mendasarinya. Model tersebut lalu digunakan untuk memprediksi nilai suatu data *time series* dimasa depan (Zhang, 2001).

Langkah dasar dalam menyelesaikan masalah data *time series* yaitu dengan mengumpulkan serta memilih variabel yang tepat, lalu memilih model terbaik sehingga menghasilkan akurasi terbaik (Berradi dan Lazaar, 2019). Salah satu metode *time series* yang sering digunakan dalam peramalan adalah metode *Autoregressive Integrated Moving Average* (ARIMA). *Autoregressive Integrated Moving Average* (ARIMA) merupakan gabungan dari model *Autoregressive* (AR) dan model *Moving Average* (MA) yang *didifference*. Model ARIMA terdiri dari model *non seasonal* (non-musiman) dan *seasonal* (musiman). Model *non seasonal* yang merupakan model stasioner terdiri dari AR(p), MA(q), dan ARMA(p,q), sedangkan model ARIMA (p,d,q) merupakan bentuk model non stasioner (Zhang, 2001). Keterbatasan utama dalam model ARIMA adalah menjadikan linearitas sebagai asumsi dan residual yang dihasilkan masih terdapat unsur nonlinear. Metode ARIMA memiliki akurasi tinggi dalam peramalan jangka pendek, namun akurasi akan menurun ketika digunakan dalam peramalan jangka panjang. Untuk mengatasi permasalahan yang terdapat pada metode ARIMA, pada penelitian ini metode ARIMA akan dikombinasikan dengan metode *Long Short-Term Memory* (LSTM). *Long Short-Term Memory* (LSTM) mampu memproses data linear maupun nonlinear, namun dibutuhkan waktu yang lama dalam memproses data serta parameter yang banyak untuk mencapai hasil yang maksimal. *Long Short-Term Memory* (LSTM) dapat juga digunakan dalam peramalan jangka panjang karena LSTM memiliki memori sel yang membuat metode ini dapat menyimpan informasi dalam jangka waktu yang panjang (Wiranda dan Sadikin, 2019). Keterbatasan yang dimiliki oleh metode ARIMA dapat diatasi dengan mengkombinasikan ARIMA dengan model LSTM agar pola data non-linear dapat didefinisikan dan membuat akurasi dalam melakukan peramalan jangka panjang menjadi lebih baik.

Penelitian tentang metode *hybrid* sudah pernah dilakukan sebelumnya. Penelitian yang dilakukan oleh Zhang melakukan peramalan terhadap tiga jenis saham dengan menggunakan metode *hybrid* ARIMA-ANN dan membandingkan hasil akurasi dari metode ARIMA, ANN, dan *hybrid* ARIMA-ANN. Pada penelitian ini diperoleh

bahwa metode *hybrid* ARIMA-ANN memiliki nilai MSE yang lebih kecil dibandingkan dengan ARIMA maupun ANN saja (Zhang, 2001).

Penelitian tentang prediksi terhadap pasar saham menggunakan metode *hybrid* ARIMA-LSTM dilakukan oleh Kulshreshtha dan A. Pada penelitian ini membandingkan akurasi dari hasil peramalan menggunakan metode ARIMA dan *hybrid* ARIMA-LSTM. Hasil yang diperoleh yaitu performa dari metode *hybrid* ARIMA-LSTM lebih baik dibandingkan metode ARIMA saja. Metode ARIMA menghasilkan nilai MAPE sebesar 0,077%, sedangkan metode *hybrid* ARIMA-LSTM menghasilkan nilai MAPE sebesar 0,009% (Kulshreshtha dan A, 2020).

Temur dan kawan-kawan melakukan peramalan penjualan rumah di Turki dengan menggunakan metode ARIMA, LSTM, dan *hybrid* ARIMA-LSTM. Hasil yang diperoleh pada penelitian ini yaitu performa dari metode *hybrid* ARIMA-LSTM lebih baik dibandingkan dengan metode ARIMA ataupun LSTM saja. Metode ARIMA menghasilkan nilai MAPE sebesar 0,121%, LSTM menghasilkan MAPE sebesar 0,150%, dan *hybrid* ARIMA-LSTM menghasilkan MAPE sebesar 0,072% (Temur, dkk., 2019)

Penelitian tentang peramalan kasus positif COVID-19 di Jawa Timur menggunakan metode *hybrid* ARIMA-LSTM dilakukan oleh Rowan dan kawan-kawan. Penelitian ini bertujuan mengambil tindakan antisipasi agar sumber daya yang dibutuhkan dalam penanganan COVID-19 seperti tenaga kesehatan dan obat-obatan dapat tersedia sebelum kasus COVID-19 melonjak. Penelitian ini menghasilkan akurasi yang baik yaitu dengan MAPE sebesar 1,8%. Berdasarkan hasil peramalan pada penelitian ini diharapkan dapat membantu dalam mengambil keputusan bagi pihak-pihak terkait (Rowan, dkk., 2022).

Temur dan Yilzid melakukan suatu penelitian yang bertujuan untuk meramalkan anggaran kuantitas penjualan bulanan suatu perusahaan manufaktur. Data yang digunakan dalam penelitian ini adalah data penjualan produk dari bisnis yang

beroperasi di industri konstruksi dari Januari 2010 sampai dengan Desember 2017. Hasil yang diperoleh pada penelitian ini yaitu performa dari metode *hybrid* ARIMA-LSTM lebih baik dibandingkan dengan metode ARIMA ataupun LSTM saja. Metode ARIMA menghasilkan nilai MAPE sebesar 0,234%, LSTM menghasilkan MAPE sebesar 0,309%, dan *hybrid* ARIMA-LSTM menghasilkan MAPE sebesar 0,133% (Temur dan Yilzid , 2021).

Namini dan kawan-kawan melakukan penelitian dengan membandingkan metode ARIMA dan LSTM yang bertujuan untuk mengetahui metode mana yang menghasilkan akurasi lebih baik dalam meramalkan harga saham. Kedua metode tersebut diterapkan pada sekumpulan data harga saham dan hasilnya menunjukkan bahwa bahwa LSTM lebih unggul dibandingkan ARIMA. Hal tersebut dapat dibuktikan dengan nilai RMSE yang dihasilkan saat penelitian. Nilai RMSE yang dihasilkan oleh metode ARIMA sebesar 511,481 dan nilai RMSE yang dihasilkan oleh metode LSTM sebesar 64,213 (Namini, dkk., 2018).

Berdasarkan penjelasan diatas, penulis tertarik untuk melakukan peramalan terhadap harga penutupan saham BRI dengan menggunakan metode *hybrid* ARIMA-LSTM. Hasil peramalan dari penelitian ini dapat digunakan sebagai acuan calon investor dalam mengambil keputusan mengenai waktu yang tepat untuk bisa berinvestasi pada saham Bank BRI sehingga dapat meminimalisir resiko kerugian dan memperoleh keuntungan sebesar-besarnya.

1.2 Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai berikut:

1. Membangun model *hybrid* ARIMA-LSTM dengan data input prediksi ARIMA dan residual ARIMA.
2. Mengetahui performa model yang telah dibangun dari metode *hybrid* ARIMA-LSTM dengan melihat nilai RMSE dan MAPE.

3. Untuk meramalkan harga penutupan saham Bank BRI selama 30 periode kedepan.

1.3 Manfaat Penelitian

Manfaat dari penelitian ini adalah sebagai berikut:

1. Menambah wawasan penulis terhadap metode *hybrid* ARIMA-LSTM serta pengaplikasiannya.
2. Dapat dijadikan referensi bagi investor dalam melakukan investasi saham jangka panjang.
3. Dapat menjadi referensi bagi peneliti selanjutnya dalam melakukan peramalan menggunakan metode *hybrid* ARIMA-LSTM dengan jenis data yang berbeda.
4. Dapat dijadikan referensi bagi BRI dalam mengambil suatu keputusan.

II. TINJAUAN PUSTAKA

2.1 Peramalan

Peramalan adalah prediksi beberapa peristiwa di masa depan. Peramalan dapat juga diartikan sebagai prediksi kejadian atau peristiwa di masa yang akan datang dengan menggunakan data-data masa lalu dari suatu variabel yang nantinya akan digunakan dalam pengambilan keputusan (Makridakis, dkk., 1997). Peramalan merupakan masalah penting yang mencakup banyak bidang termasuk pemerintahan, ekonomi, kesehatan, ilmu sosial, politik, keuangan, bisnis dan industri (Montgomery, dkk., 2015).

Dalam peramalan dibutuhkan perhitungan yang tepat agar mendapatkan hasil peramalan yang akurat. Menurut Cahyono, dkk. (2018), terdapat dua langkah dasar yang harus dilakukan agar menghasilkan peramalan yang akurat dan berguna. Langkah pertama adalah pengumpulan data yang relevan dengan tujuan peramalan yang dimaksud. Langkah kedua adalah memilih metode peramalan yang sesuai dengan data yang telah dikumpulkan, sehingga hasil peramalan akan memiliki akurasi yang baik.

Menurut Montgomery, dkk. (2015), terdapat dua jenis teknik peramalan, yaitu:

1. Teknik peramalan kuantitatif adalah teknik peramalan yang menggunakan data-data kuantitatif masa lalu yang diperoleh dari nilai-nilai pengamatan sebelumnya.
2. Teknik peramalan kualitatif adalah teknik peramalan yang didasarkan pada data kualitatif masa lalu. Teknik peramalan ini digunakan dalam situasi dimana

hanya ada sedikit atau tidak ada data historis yang menjadi dasar peramalan. Dasar penilaian dalam teknik peramalan ini bersifat subjektif karena didasarkan pada opini, pengalaman maupun pengetahuan pendapat pihak yang bersangkutan.

2.2 *Time Series*

Time series merupakan serangkaian pengamatan atau observasi terhadap suatu variabel yang diamati secara berurutan dalam interval waktu yang tetap (Montgomery, dkk., 2015). Sedangkan data *time series* merupakan data yang diperoleh dari pengamatan suatu variabel atau objek yang terdiri dari beberapa periode waktu yang konstan seperti dalam jam, hari, minggu, bulan atau tahun. Pada data *time series* nilai pengamatan suatu periode waktu diasumsikan dipengaruhi oleh nilai pengamatan pada periode waktu sebelumnya. Berdasarkan asumsi tersebut, analisis data *time series* dapat digunakan untuk melakukan prediksi beberapa peristiwa di masa yang akan datang.

Menurut Makridakis, dkk. (1997), pola data *time series* dapat dibedakan menjadi empat kategori, yaitu:

1. Pola acak atau pola horizontal, pola ini terjadi bila data mengalami kenaikan atau penurunan disekitar rata-ratanya.
2. Pola tren, pola ini terjadi bila data mengalami kenaikan atau penurunan jangka panjang dalam data.
3. Pola musiman, pola ini terjadi apabila suatu data dipengaruhi oleh faktor musiman (misalnya kuartal, bulanan, atau harian) dan biasanya berulang setiap tahun.
4. Pola siklis, pola ini ditandai dengan adanya fluktuasi jangka panjang yang membentuk pola gelombang atau siklus disekitar garis tren. Pola ini terjadi jika data dipengaruhi oleh fluktuasi ekonomi jangka panjang, biasanya terkait dengan siklus bisnis.

Berdasarkan banyaknya variabel yang diamati, model *time series* dapat dikelompokkan menjadi dua bagian, yaitu model *time series* univariat dan model *time series* multivariat. Model *time series* univariat merupakan model yang hanya menggunakan satu variabel pengamatan saja, sedangkan model *time series* multivariat merupakan model yang menggunakan lebih dari satu variabel pengamatan. *Autoregressive (AR)*, *Moving Average (MA)*, *Autoregressive Moving Average (ARMA)*, dan *Autoregressive Integrated Moving Average (ARIMA)* termasuk dalam model *time series* univariat. Model ARIMA (p,d,q) termasuk dalam model *seasonal* atau non stasioner, sedangkan AR(p), MA(q), dan ARMA(p,q) termasuk dalam model *non seasonal* atau stasioner (Zhang, 2001).

1. *Autoregressive (AR)*

Model AR memiliki asumsi bahwa nilai pengamatan periode saat ini dipengaruhi oleh nilai pengamatan periode sebelumnya. Menurut Box, dkk. (2016), bentuk umum model AR dengan orde p atau AR(p) dapat dinyatakan sebagai berikut:

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

dengan:

- Y_t : variabel dependent
- Y_{t-1}, \dots, Y_{t-p} : variabel dependent atau data masa lalu
- ϕ_1, \dots, ϕ_p : koefisien parameter *autoregressive*
- ε_t : residual periode ke-t

Persamaan (2.1) dapat dituliskan menggunakan operator B atau operator *backshift*. Operator *backshift* mempunyai pengaruh menggeser data satu waktu ke belakang. Operator *backshift* dapat didefinisikan sebagai berikut:

$$B^i Y_t = Y_{t-i} \quad (2.2)$$

Dengan menggunakan persamaan operator *backshift* maka persamaan (2.1) dapat dituliskan sebagai berikut:

$$Y_t = \phi_1 B Y_t + \phi_2 B^2 Y_t + \dots + \phi_p B^p Y_t + \varepsilon_t$$

$$\varepsilon_t = (1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p) Y_t$$

atau

$$\varepsilon_t = \phi_p(B) Y_t \quad (2.3)$$

2. *Moving Average* (MA)

Model MA memiliki asumsi bahwa nilai pengamatan periode saat ini dipengaruhi oleh nilai-nilai kesalahan atau residual periode saat ini dan periode sebelumnya. Menurut Box, dkk. (2016), bentuk umum model MA dengan orde q atau MA(q) dapat dinyatakan sebagai berikut:

$$Y_t = \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \dots - \theta_q \varepsilon_{t-q} \quad (2.4)$$

dengan:

- Y_t : variabel dependent periode ke-t
- $\theta_1, \dots, \theta_q$: koefisien parameter *moving average*
- ε_t : residual periode ke-t
- $\varepsilon_{t-1}, \dots, \varepsilon_{t-q}$: residual periode periode ke-t-1, ..., t-p

Persamaan (2.4) dapat dituliskan menggunakan operator B atau operator *backshift* sehingga menjadi persamaan berikut:

$$Y_t = \varepsilon_t - \theta_1 B \varepsilon_t - \theta_2 B^2 \varepsilon_t - \dots - \theta_q B^q \varepsilon_t$$

$$Y_t = (1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q) \varepsilon_t$$

atau

$$Y_t = \theta_q(B) \varepsilon_t \quad (2.5)$$

3. *Autoregressive Moving Average* (ARMA)

Model ARMA merupakan gabungan dari model AR dan model MA. Model ARMA memiliki asumsi bahwa nilai pengamatan periode saat ini dipengaruhi oleh nilai pengamatan dan residual periode sebelumnya. Menurut Box, dkk. (2016), bentuk umum model ARMA dengan orde p dan q atau ARMA(p,q) dapat dinyatakan sebagai berikut:

$$Y_t = \phi_1 Y_{t-1} + \dots + \phi_p Y_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \dots - \theta_q \varepsilon_{t-q} \quad (2.6)$$

dengan:

- Y_t : variabel dependent periode ke-t
- Y_{t-1}, \dots, Y_{t-p} : variabel dependent periode ke-t-1, ..., t-p
- ϕ_1, \dots, ϕ_p : koefisien parameter *autoregressive*
- $\theta_1, \dots, \theta_q$: koefisien parameter *moving average*
- ε_t : residual periode ke-t
- $\varepsilon_{t-1}, \dots, \varepsilon_{t-q}$: residual periode sebelumnya

Persamaan (2.6) dapat dituliskan menggunakan operator B atau operator *backshift* sehingga menjadi persamaan berikut:

$$\begin{aligned} Y_t &= \phi_1 B Y_t + \dots + \phi_p B^p Y_t + \varepsilon_t - \theta_1 B \varepsilon_t - \dots - \theta_q B^q \varepsilon_t \\ (1 - \phi_1 B - \phi_2 B^2 - \dots - \phi_p B^p) Y_t &= (1 - \theta_1 B - \theta_2 B^2 - \dots - \theta_q B^q) \varepsilon_t \\ &\text{atau} \\ \phi_p(B) Y_t &= \theta_q(B) \varepsilon_t \end{aligned} \quad (2.7)$$

4. *Autoregressive Integrated Moving Average* (ARIMA)

Autoregressive Integrated Moving Average (ARIMA) atau yang juga dikenal sebagai metode Box-Jenkins dikembangkan pada tahun 1970 oleh George Box dan Gwilym Jenkins. Model ARIMA merupakan kombinasi antara model AR(p) dan model MA(q) nonstasioner. Pada model ARMA (p,q) nonstasioner perlu dilakukan proses *differencing* sebanyak d kali agar data stasioner, sehingga model ARMA (p,q) menjadi model ARIMA (p,d,q). *Differencing* merupakan proses mencari selisih antara data pada suatu periode dengan periode sebelumnya secara berurutan. Bentuk umum model ARIMA dapat dituliskan dalam persamaan berikut:

$$\phi_p(B)(1 - B)^d Y_t = \theta_q(B) \varepsilon_t \quad (2.8)$$

dengan:

- Y_t : variabel dependent
- B : operator *backshift*
- d : banyaknya *differencing*

- $\phi_p(B)$: operator AR(p)
 $\theta_q(B)$: operator MA(q)
 ε_t : residual periode ke-t

2.3 Stasioneritas

Stasioner diasumsikan sebagai proses yang tetap dalam kestabilan statistik dengan probabilitas yang tidak berubah dari waktu ke waktu atau dengan kata lain nilai tengah dan ragamnya konstan (Box, dkk., 2016). Metode yang dapat digunakan untuk menguji kestasioneran data adalah dengan menggunakan uji *Augmented Dickey Fuller* (ADF).

Data *time series* dikatakan stasioner dalam rata-rata apabila data tersebut mengalami kenaikan maupun penurunan disekitar nilai rata-rata secara konstan dari waktu ke waktu. Data yang tidak stasioner dalam rata-rata dapat distasionerkan dengan melakukan *differencing*. *Differencing* merupakan proses mencari selisih antara data satu periode dengan periode sebelumnya. *Differencing* pertama atau d=1 didefinisikan sebagai berikut:

$$\begin{aligned}
 \nabla Y_t &= Y_t - Y_{t-1} \\
 \nabla Y_t &= Y_t - BY_t \\
 \nabla Y_t &= (1 - B)Y_t
 \end{aligned}
 \tag{2.9}$$

dengan:

- Y_t : data aktual periode ke-t
 B : operator *backshift*

Apabila setelah melakukan *differencing* pertama data masih belum stasioner dalam rata-rata, maka dilakukan *differencing* lagi pada periode selanjutnya dari hasil *differencing* pertama untuk semua t. *Differencing* kedua atau d=2 didefinisikan sebagai berikut:

$$\nabla^2 Y_t = \nabla Y_t - \nabla Y_{t-1} \tag{2.10}$$

$$\nabla^2 Y_t = (Y_t - Y_{t-1}) - (Y_{t-1} - Y_{t-2})$$

$$\nabla^2 Y_t = Y_t - 2Y_{t-1} + Y_{t-2}$$

$$\nabla^2 Y_t = Y_t - 2BY_t + B^2 Y_t$$

$$\nabla^2 Y_t = Y_t(1 - B)^2$$

Apabila terdapat *differencing* ke-d untuk mencapai stasioneritas, maka dapat dinyatakan sebagai berikut:

$$\nabla^d Y_t = Y_t(1 - B)^d \quad (2.11)$$

2.4 Data Mining

Data mining merupakan proses penambangan atau penemuan informasi baru dengan mencari aturan atau pola tertentu dari sejumlah data yang sangat besar (Davies dan Beynon, 2004). Menurut Hartama, dkk. (2019), data mining adalah proses yang menggunakan teknik matematika, statistika, kecerdasan buatan, dan pembelajaran mesin untuk mengekstraksi dan mengidentifikasi informasi dan pengetahuan yang bermanfaat yang terkait dengan data yang besar. Berdasarkan pemaparan diatas dapat diambil kesimpulan bahwa data mining merupakan suatu teknik menggali informasi tersembunyi pada data yang sangat besar sehingga ditemukan pola tertentu yang sebelumnya tidak diketahui (Han, dkk., 2012).

Data mining dalam pengaplikasiannya merupakan bagian dari proses *Knowledge Discover in Database*. Tahapan dalam KDD dapat dinyatakan sebagai berikut:

1. *Data cleaning*, berfungsi untuk memeriksa dan mengisi data yang hilang, mengidentifikasi dan menghapus outlier, serta memperbaiki data yang tidak konsisten.
2. *Data integration*, menggabungkan data dari beberapa sumber ke dalam penyimpanan data, seperti *data warehouse*.
3. *Data selection*, data yang terdapat pada database sering kali tidak digunakan semuanya. Oleh karena itu, hanya data yang sesuai untuk dianalisis yang akan diambil dari database. Data yang tidak sesuai akan dihilangkan.

4. Transformasi data

Data diubah ke dalam format yang sesuai untuk diproses dalam data mining. Salah satu teknik dalam transformasi data adalah normalisasi. Normalisasi data merupakan proses transformasi nilai suatu data dalam rentang nilai tertentu. Beberapa metode untuk normalisasi data adalah *min-max normalization* dan *standardscaler*.

a. *Min-max Normalization*

Min-max normalization merubah nilai data aktual menjadi nilai dengan *range* interval [0,1] dengan persamaan sebagai berikut:

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \quad (2.12)$$

dengan:

x' : nilai hasil *min-max normalization*

x : nilai data aktual

x_{\max} : nilai maksimum data aktual

x_{\min} : nilai minimum data aktual

b. *Standardscaler*

Standardscaler merubah nilai data aktual menjadi data yang memiliki $\mu = 0$ dan $\sigma = 1$, dengan persamaan sebagai berikut:

$$x' = \frac{x - \mu}{\sigma} \quad (2.13)$$

dengan:

x' : nilai hasil *standardscaler*

x : nilai data aktual

μ : nilai rata-rata

σ : standar deviasi

5. Data mining

Data mining adalah proses dalam mencari pola yang sesuai dari data yang ingin ditampilkan dengan menggunakan metode tertentu.

6. *Pattern evaluation*, untuk mengidentifikasi atau menerjemahkan pola yang dihasilkan dari proses data mining.
7. *Knowledge presentation*, pola data atau informasi yang telah dihasilkan dari proses data mining perlu ditampilkan dalam bentuk yang mudah dipahami oleh

pihak yang berkepentingan. Visualisasi data ini dapat berupa tabel, laporan, dan berbagai format lainnya.

2.5 *Machine Learning*

Machine learning merupakan salah satu cabang dari ilmu kecerdasan buatan (*artificial intelligence*) yang bertujuan untuk memprogram komputer agar dapat belajar dan bertindak seperti manusia, serta meningkatkan kemampuan belajarnya dari waktu ke waktu dengan cara menyuplai data dan informasi sebagai bentuk pengalaman, sehingga komputer dapat belajar secara otomatis mengenali pola yang kompleks dan membuat keputusan cerdas berdasarkan data yang diberikan. Menurut Rowan, dkk. (2022), pembelajaran mesin dikembangkan pada berbagai bidang seperti matematika, statistik, dan data mining sehingga mesin dapat belajar dengan cara menganalisis data tanpa perlu instruksi maupun pemograman ulang. Ciri dari *machine learning* adalah adanya proses pembelajaran (*training*) dan pelatihan (*testing*). *Training* merupakan proses konstruksi model sedangkan *testing* merupakan proses yang digunakan dalam menguji kinerja model pembelajaran yang telah dibangun.

Machine learning memiliki dua teknik pembelajaran, yaitu:

1. *Supervised learning* (pembelajaran terarah) merupakan algoritma *machine learning* yang menerima informasi dari data-data yang telah memiliki label. *Output* pada pembelajaran terarah telah ditentukan sebelumnya dan proses pembelajaran akan berhenti ketika algoritma telah mencapai hasil yang diinginkan. *Supervised learning* biasanya digunakan dalam membuat prediksi dan klasifikasi.
2. *Unsupervised learning* (pembelajaran tidak terarah) merupakan proses pembelajaran yang menerima informasi dari data yang tidak memiliki label. Tujuan dari teknik pembelajaran tidak terarah ini adalah untuk mengelompokkan data yang memiliki karakteristik hampir sama dalam suatu

area tertentu dan tidak memerlukan target output. Teknik pembelajaran tidak terarah cocok digunakan untuk *clustering*.

2.6 Deep Learning

Deep learning merupakan bagian dari *machine learning* yang pada dasarnya merupakan jaringan saraf dengan tiga atau lebih lapisan jaringan saraf tiruan yang memungkinkannya untuk belajar dan beradaptasi dengan data yang jumlahnya besar dan mampu untuk memecahkan berbagai masalah yang sulit dipecahkan oleh algoritma pembelajaran mesin lainnya (Rowan, dkk., 2022). Salah satu algoritma *deep learning* adalah *Recurrent Neural Network* (RNN).

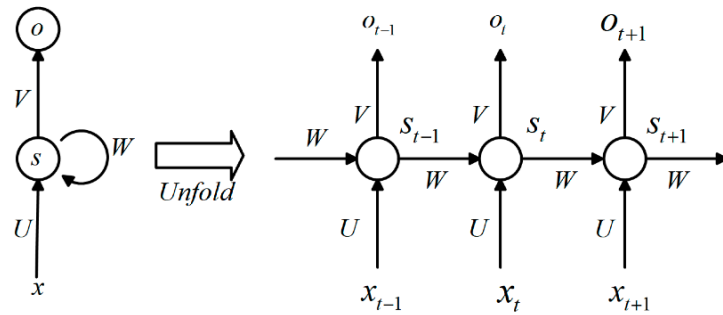
Berawal dari *neural network* sederhana yang merupakan bagian dari *artificial intelligent* yang dibentuk meniru cara kerja otak manusia. Model awal ini terdiri dari tiga *layer*, yaitu *input layer*, *hidden layer*, dan *output layer*. Model ini juga terdiri dari prosesor yang sangat sederhana dan saling berhubungan yang disebut *neuron* (Kurniawansyah, 2018). Model awal ini memiliki kelemahan yaitu tidak mampu menghubungkan atau mengaitkan informasi baru dengan informasi sebelumnya. Berdasarkan kelemahan tersebut, model awal ini tidak dapat digunakan untuk mengolah atau memproses data *time series* karena data *time series* memiliki keterkaitan dengan data periode sebelumnya. Agar model awal tersebut dapat digunakan untuk mengolah data *time series*, maka hasil output pada proses sebelumnya digunakan untuk memperbarui nilai bobot yang baru. Bobot baru yang dihasilkan digunakan sebagai bobot untuk perhitungan data yang lebih baru dan begitu seterusnya, sehingga ada keterkaitan antara data periode saat ini dengan data periode sebelumnya. Model pengembangan ini disebut dengan *Recurrent Neural Network* (RNN). Dalam arsitektur model RNN terdapat *multilayer neural network* dengan melibatkan perulangan dalam pemrosesan data yang biasanya adalah data *time series*. Keterlibatan *multilayer* tersebut yang menyebabkan RNN termasuk dalam algoritma *deep learning* (Karno, 2020).

Namun RNN memiliki kelemahan yaitu tidak dapat menangani ketergantungan jangka panjang dikarenakan RNN tidak dapat menyimpan informasi sebelumnya dengan baik yang disebabkan oleh masalah *vanishing gradient* atau *exploding gradient*. Menurut Salehinejad, dkk. (2018), *vanishing gradien* terjadi saat *neural network* memiliki banyak *layer* dengan *input* yang panjang sehingga nilai bobot saat proses *backpropagation* menjadi kecil sekali atau mendekati nol (menghilang) sehingga sinyal gradien menjadi lambat atau bahkan berhenti bekerja, oleh karena itu terjadilah proses *vanishing gradient*. Namun sebaliknya, saat nilai bobot dalam saat proses *backpropagation* besar, dapat menyebabkan sinyal gradien akan sangat besar dan menimbulkan masalah *exploding gradient*. Untuk mengatasi masalah *vanishing gradient* atau *exploding gradient* maka dikembangkan sel *Long Short Term Memory* (LSTM) dengan cara memodifikasi arsitektur model dengan mengkombinasikan beberapa unit gerbang (*gate*) yang dirancang khusus sehingga memiliki jaringan blok memori yang mampu mengolah data dan menyimpan informasi dengan periode waktu yang lama (Karno, 2020).

2.6.1 Recurrent Neural Network (RNN)

Recurrent Neural Network (RNN) adalah jaringan saraf berulang yang merupakan salah satu jenis dari jaringan saraf tiruan yang dirancang khusus agar dapat memproses data sekuensial sehingga dapat digunakan untuk memproses data *time series*. *Recurrent Neural Network* (RNN) mempunyai kemampuan untuk melihat korelasi tersembunyi yang terjadi pada data dalam aplikasi untuk pemrosesan bahasa alami, pengenalan suara, dan peramalan deret waktu (Tian, dkk., 2018). Jaringan saraf tiruan dengan koneksi berulang disebut dengan RNN. RNN dikatakan sebagai jaringan saraf berulang dikarenakan nilai pada *hidden layer* sebelumnya akan digunakan kembali sebagai data input bagi pemrosesan selanjutnya.

Adapun arsitektur dari RNN adalah sebagai berikut:



Gambar 1. Arsitektur *Recurrent Neural Network*.
(Sumber: Zhu, dkk., 2019)

Berdasarkan Gambar 1, S_t dan O_t dapat direpresentasikan sebagai berikut:

$$S_t = f((U * X_t) + (W * S_{t-1})) \quad (2.14)$$

$$O_t = g(V * S_t) \quad (2.15)$$

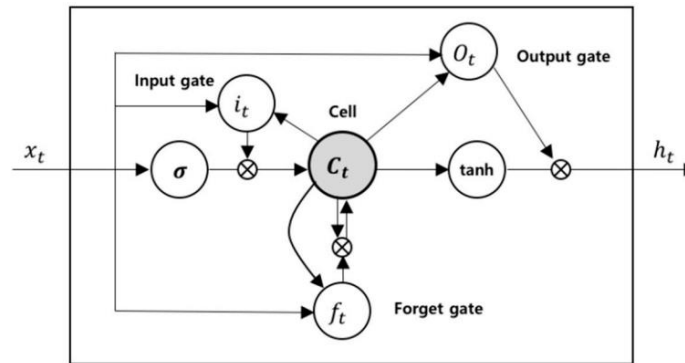
dengan:

- S_t : memori jaringan pada waktu ke-t
- X_t dan O_t : input dan output pada waktu ke-t
- U, V, dan W : bobot disetiap *layer*
- $f(\dots)$ dan $g(\dots)$: fungsi nonlinear

2.6.2 Long Short Term Memory (LSTM)

Long Short Term Memory (LSTM) merupakan salah satu pengembangan dari RNN, arsitektur ini pertama kali diperkenalkan oleh Hochreiter dan Schmidhuber pada tahun 1997. LSTM dibangun untuk mengatasi masalah pada RNN, yaitu pada saat pemrosesan data sekuensial jangka panjang RNN tidak dapat menghubungkan informasi baru dengan informasi lama karena memori lama yang tersimpan akan semakin tidak berguna dan tertimpa dengan memori yang baru, sedangkan LSTM mampu mengatasi masalah tersebut karena dapat mengatur memori pada setiap masukannya dengan menggunakan *memory cell* dan *gate units* (Arfan dan Lusiana, 2019). LSTM dapat menghubungkan interval waktu dalam jangka yang panjang tanpa kehilangan informasi (Hochreiter dan Schmidhuber, 1997). LSTM mengganti

lapisan RNN dengan *memory cell* menggunakan mekanisme gerbang yang terdiri dari *forget gate*, *input gate*, dan *output gate* (Felix, dkk., 2000). LSTM tersusun dari *neuron* yang diproses secara berulang, sama halnya dengan RNN.



Gambar 2. Struktur *Long Short Term Memory*.
(Sumber: Chung & Shin, 2018)

Gambar 2 menampilkan isi dari *hidden layer* LSTM yaitu *memory cell*. *Memory cell* di LSTM menyimpan sebuah nilai baik untuk periode waktu yang panjang ataupun singkat. Berikut ini merupakan penjelasan untuk gerbang-gerbang yang ada pada satu sel memori LSTM.

1. *Forget gate* (f_t) adalah gerbang pertama pada LSTM yang digunakan dalam menentukan informasi mana yang akan dipertahankan dari *cell state*. *Forget gate* menerima input h_{t-1} dan x_t untuk menghasilkan nilai 0 atau 1 pada C_{t-1} . Saat *forget gate* bernilai 1 maka informasi akan disimpan oleh *cell state*, namun jika bernilai 0 maka informasi akan dibuang dari *cell state* (Nugroho, dkk., 2021).

Persamaan *forget gate* dapat diuraikan sebagai berikut:

$$f_t = \sigma(W_f * [h_{t-1}, x_t] + b_f) \quad (2.16)$$

dengan:

- f_t : *forget gate*
- σ : fungsi aktivasi sigmoid
- W_f : nilai *weight* untuk *forget gate*
- h_{t-1} : nilai *output* periode ke t-1
- x_t : nilai input periode ke t

b_f : nilai bias pada *forget gate*

2. *Input gate* (i_t) adalah gerbang kedua pada LSTM dan memiliki dua fungsi aktivasi yaitu *sigmoid* dan *tanh*. Fungsi aktivasi *sigmoid* berfungsi untuk memutuskan nilai mana yang akan diperbarui, sedangkan fungsi aktivasi *tanh* membentuk kandidat vektor baru \tilde{C}_t untuk ditambahkan ke *cell state*

Persamaan *input gate* dapat diuraikan sebagai berikut:

$$i_t = \sigma(W_i * [h_{t-1}, x_t] + b_i) \quad (2.17)$$

dengan:

i_t : *input gate*
 σ : fungsi aktivasi *sigmoid*
 W_i : nilai *weight* untuk *input gate*
 h_{t-1} : nilai *output* periode ke t-1
 x_t : nilai input periode ke t
 b_i : nilai bias pada *input gate*

Persamaan kandidat vektor baru dapat diuraikan sebagai berikut:

$$\tilde{C}_t = \tanh(W_c * [h_{t-1}, x_t] + b_c) \quad (2.18)$$

dengan:

\tilde{C}_t : kandidat vektor baru
 \tanh : fungsi *tangen hiperbolik*
 W_c : nilai *weight* untuk *cell state*
 h_{t-1} : nilai *output* periode ke t-1
 x_t : nilai input periode ke t
 b_c : nilai bias pada *cell state*

3. Pada langkah ini nilai state lama c_{t-1} diperbarui menjadi nilai state baru c_t dengan mengalikan *cell state* lama dengan f_t untuk menghapus nilai pada *forget gate* sebelumnya lalu dijumlahkan dengan *input gate* yang dikalikan dengan \tilde{C}_t

sebagai nilai baru yang akan digunakan untuk memperbarui nilai *cell state* (Nugroho, dkk., 2021).

Persamaan *cell state* dapat diuraikan sebagai berikut:

$$c_t = f_t * c_{t-1} + i_t * \tilde{c}_t \quad (2.19)$$

dengan:

- c_t : nilai memori sel baru
- f_t : *forget gate*
- c_{t-1} : nilai memori sel sebelumnya
- i_t : *input gate*
- \tilde{c}_t : kandidat vektor baru

4. *Output gate* (o_t) adalah gerbang terakhir pada LSTM yang berfungsi untuk menentukan *output* dari *cell state*. Pertama, fungsi aktivasi *sigmoid* akan menentukan bagian dari *cell state* mana yang akan menjadi *output*. Kemudian, *output* tersebut akan dimasukkan ke dalam lapisan *tanh* lalu dikalikan dengan fungsi aktivasi *sigmoid* agar *output* yang dihasilkan sesuai dengan yang telah ditentukan sebelumnya (Nugroho, dkk., 2021).

Persamaan *output gate* dapat diuraikan sebagai berikut:

$$o_t = \sigma(W_o * [h_{t-1}, x_t] + b_o) \quad (2.20)$$

$$h_t = \tanh(c_t) * o_t \quad (2.21)$$

dengan:

- o_t : *output gate*
- h_t : nilai output periode ke t
- σ : fungsi aktivasi sigmoid
- \tanh : fungsi *tangen hiperbolik*
- W_o : nilai *weight* untuk *output gate*
- h_{t-1} : nilai *output* periode ke t-1
- x_t : nilai input periode ke t
- b_o : nilai bias pada *output gate*

Fungsi aktivasi berfungsi menentukan hubungan antara *input* dan *output* dari suatu *neuron* dan suatu jaringan saraf tiruan. Fungsi aktivasi pada dasarnya mentransformasi nilai *input* untuk disalurkan menjadi *output* (Setiawan, dkk., 2022). Fungsi aktivasi yang digunakan dalam LSTM adalah sebagai berikut:

1. Fungsi *sigmoid* merupakan fungsi aktivasi nonlinear yang menghasilkan nilai dengan *range* 0 sampai 1. Fungsi *sigmoid* didefinisikan sebagai berikut:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (2.22)$$

dengan:

e : bilangan euler

x : data

2. Fungsi *tanh* atau *tangen hiperbolik* merupakan fungsi aktivasi nonlinear yang *outputnya* memiliki *range* -1 sampai dengan 1. Fungsi *tanh* didefinisikan sebagai berikut:

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.23)$$

dengan:

e : bilangan euler

x : data

2.7 Parameter *Long Short Term Memory* (LSTM)

Pada proses pembuatan model LSTM diperlukan beberapa parameter yang ditentukan diawal dan tetap konstan selama proses pelatihan (*training*). Parameter yang digunakan antara lain jumlah *neuron*, *epoch*, dan *batch size*. *Epoch* merupakan parameter yang menentukan berapa kali algoritma *deep learning* bekerja melewati dataset baik secara *backward* maupun *forward*. *Epoch* juga dapat disebut sebagai jumlah iterasi selama proses pelatihan dan juga memperbarui bobot pada jaringan (Vijayalakshmi dan Venkatachalapathy, 2019). Satu *epoch* tercapai jika algoritma

deep learning telah melewati seluruh dataset baik secara *backward* maupun *forward* sebanyak satu kali. Satu *epoch* saat proses pelatihan dapat berlangsung cukup lama, sehingga perlu pembagian data pada setiap *batch* yang disebut dengan *batch size*. *Batch size* merupakan jumlah data dalam satu iterasi. Hal tersebut dilakukan untuk mempersingkat waktu saat proses pelatihan berlangsung. *Epoch* dan *batch size* dibutuhkan saat dataset yang digunakan saat penelitian terlalu besar untuk dilatih sekaligus oleh komputer. Hal tersebut disebabkan memori komputer juga memiliki keterbatasan. Oleh karena itu, data dibagi menjadi ukuran-ukuran yang lebih kecil.

Parameter-parameter tersebut harus dikombinasikan dengan cara yang ideal untuk mendapatkan model LSTM yang optimal. Dalam mencari kombinasi parameter terbaik membutuhkan waktu yang cukup lama karena pelatihan harus dilakukan pada setiap kombinasi. Berdasarkan hal tersebut, diperlukan suatu metode untuk mengidentifikasi kombinasi terbaik dengan waktu yang lebih singkat atau dapat disebut juga dengan *hyperparameter tuning* atau *hypertuning*. *Hyperparameter tuning* menggunakan *early stopping* sehingga ketika kondisi terpenuhi atau telah ditemukan kombinasi parameter terbaik untuk membangun sebuah model, maka proses pembelajaran model berakhir (Li, dkk., 2018).

2.8 Hybrid ARIMA-LSTM

Model *hybrid* merupakan suatu metode yang menggabungkan dua atau lebih model dalam suatu sistem. Menurut Zhang (2001), berikut ini beberapa alasan yang mendasari dikembangkannya metode *hybrid*:

1. Sulit untuk menentukan apakah *time series* yang diteliti mengandung pola linear atau nonlinear sehingga membuat peneliti menggunakan lebih dari satu metode lalu dibandingkan untuk mendapatkan hasil yang paling akurat. Untuk mengatasi hal tersebut, maka dilakukan penggabungan metode yang berbeda sehingga permasalahan seleksi model dapat diminimalisir.

2. Permasalahan *time series* yang murni linear maupun nonlinear sangat jarang terjadi. Pola data lebih sering mengandung keduanya. Jika hal tersebut terjadi, maka penggunaan metode tunggal dinilai kurang efektif karena tidak dapat menangkap pola campuran.
3. Dalam peramalan, tidak ada metode tunggal yang terbaik dalam setiap situasi. Hal ini didukung oleh fakta bahwa pola data yang kompleks dalam kasus nyata tidak dapat ditangani dengan baik menggunakan metode tunggal.

ARIMA dan LSTM memiliki kelebihan dan kekurangan masing-masing. ARIMA baik dalam memproses data *time series* yang bersifat linear dan memiliki akurasi peramalan yang tinggi saat digunakan untuk peramalan jangka pendek, namun ARIMA kesulitan saat data bersifat nonlinear dan akurasi peramalannya akan menurun saat melakukan peramalan jangka panjang. Sedangkan jaringan saraf tiruan mampu bekerja dengan baik dengan data linear maupun nonlinear, namun dibutuhkan waktu yang lama dalam memproses data serta parameter yang banyak untuk mencapai hasil yang maksimal. LSTM dapat juga digunakan dalam peramalan jangka panjang karena LSTM memiliki memori sel yang membuat metode ini dapat menyimpan informasi dalam jangka waktu yang panjang. Berdasarkan faktor-faktor yang telah dijelaskan, sebuah model gabungan dirancang untuk saling menutupi kekurangan tiap model dan memanfaatkan masing-masing kelebihan model sehingga dapat meningkatkan akurasi dari hasil ramalan (Rowan, dkk., 2022). Menurut Zhang (2001), kombinasi dari model *time series* yang memiliki struktur autokorelasi linear dan nonlinear dapat dituliskan sebagai berikut:

$$y_t = L_t + N_t \quad (2.24)$$

dengan:

- y_t : data periode ke-t
- L_t : komponen linear periode ke-t
- N_t : komponen nonlinear periode ke-t
- t : waktu

2.9 Akurasi

Akurasi merupakan parameter yang dapat digunakan untuk mengukur seberapa baik model yang telah dibangun. Menurut Temur, dkk. (2019), dalam melihat akurasi model dapat digunakan ukuran sebagai berikut:

1. *Mean Absolute Percentage Error* (MAPE)

MAPE merupakan nilai *absolute* dari persentase *error* data terhadap rata-rata. Sehingga diperoleh persamaan sebagai berikut:

$$MAPE = \frac{\sum_{t=1}^n \frac{|Y_t - \hat{Y}_t|}{Y_t}}{n} \times 100\% \quad (2.25)$$

dengan:

- \hat{Y}_t : nilai data peramalan pada periode ke-t
- Y_t : nilai data aktual pada periode ke-t
- t : waktu/periode waktu
- n : banyaknya data pengamatan

Berdasarkan nilai evaluasi yang dihasilkan MAPE memiliki kriteria sebagai berikut (Warmansyah dan Hilpiah, 2019):

- a. $MAPE \leq 10\%$: akurasi peramalan tinggi
- b. $10\% < MAPE \leq 20\%$: akurasi peramalan baik
- c. $20\% < MAPE \leq 50\%$: akurasi peramalan cukup
- d. $MAPE \geq 50\%$: akurasi peramalan rendah

2. *Root Mean Square Error* (RMSE)

RMSE merupakan akar kuadrat dari penjumlahan kuadrat selisih antara nilai aktual dengan nilai peramalan, kemudian membagi jumlah tersebut dengan banyaknya jumlah data. Sehingga diperoleh persamaan untuk RMSE adalah sebagai berikut:

$$RMSE = \sqrt{\frac{\sum_{t=1}^n (Y_t - \hat{Y}_t)^2}{n}} \quad (2.26)$$

dengan:

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

Y_t : nilai data aktual pada periode ke-t

t : waktu/periode

n : banyaknya data pengamatan

III. METODOLOGI PENELITIAN

3.1 Waktu dan Tempat Penelitian

Penelitian ini dilakukan pada semester ganjil tahun akademik 2022/2023, bertempat di Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung.

3.2 Spesifikasi Perangkat

Perangkat yang digunakan dalam penelitian ini adalah laptop dengan merek Acer Aspire A514-53 dengan tipe 64-bit *operating system*, *x64-based processor*. Spesifikasi *hardware* perangkat tersebut adalah sebagai berikut:

Processor : Intel(R) Core(TM) i3-1005G1 CPU @ 1.20GHz 1.19 GHz
Memori : SSD 475 GB
RAM : 8 GB

3.3 Data Penelitian

Data yang digunakan dalam penelitian ini merupakan data sekunder yang diperoleh secara online dari <https://finance.yahoo.com/quote/BBRI.JK?p=BBRI.JK> yang merupakan data *time series* mengenai harga saham harian Bank BRI terhitung sejak Januari 2017 sampai dengan Oktober 2022. Data memiliki enam variabel yaitu *open*,

high, *low*, *close*, *adj close*, dan *volume*. Namun variabel yang digunakan dalam penelitian ini hanya variabel *close* karena disesuaikan dengan metode yang digunakan yaitu ARIMA yang merupakan metode untuk data univariat sehingga hanya variabel *close* saja yang digunakan.

Tabel 1. Data Harga Saham BRI

<i>Date</i>	<i>Open</i>	<i>High</i>	<i>Low</i>	<i>Close</i>	<i>Adj Close</i>	<i>Volume</i>
02/01/2017	2335	2335	2335	2335	1942,108	0
03/01/2017	2350	2380	2315	2380	1979,536	67580500
04/01/2017	2380	2440	2350	2440	2029,441	79891000
05/01/2017	2460	2465	2405	2420	2012,806	145357000
06/01/2017	2420	2430	2395	2420	2012,806	73241000
⋮	⋮	⋮	⋮	⋮	⋮	⋮
24/10/2022	4430	4540	4430	4500	4500	186997700
25/10/2022	4560	4610	4550	4590	4590	233948800
26/10/2022	4600	4610	4500	4530	4530	178298500
27/10/2022	4550	4600	4530	4600	4600	131385300
28/10/2022	4600	4630	4560	4630	4630	227256400

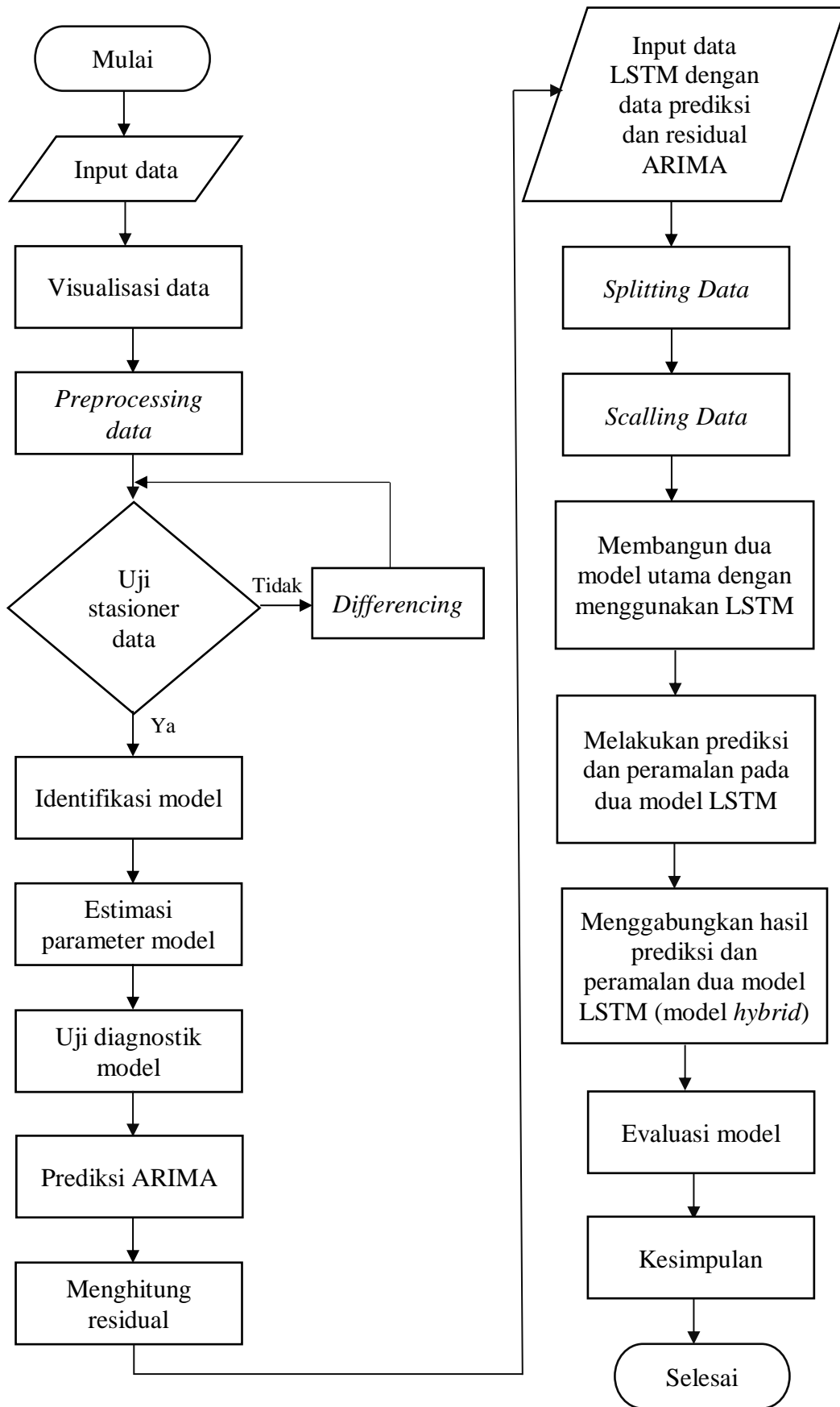
3.4 Metode Penelitian

Langkah-langkah yang dilakukan dalam melakukan peramalan menggunakan metode *hybrid* ARIMA-LSTM adalah sebagai berikut:

1. Melakukan studi literatur mengenai metode *hybrid* ARIMA-LSTM yang diambil dari jurnal, buku serta dibantu oleh narasumber yang memahami tentang metode *hybrid* ARIMA-LSTM.
2. Mengumpulkan data harga saham Bank BRI yang digunakan untuk proses peramalan dengan metode *hybrid* ARIMA-LSTM.
3. Menginput data harga saham Bank BRI.
4. Melakukan visualisasi data dengan *plotting* untuk melihat tren dan pola dalam data.
5. Melakukan *preprocessing data*, yaitu dengan memeriksa data yang hilang pada data.

6. Melakukan uji stasioner data dengan menggunakan uji *Augmented Dickey-Fuller* (ADF). Jika data tidak stasioner perlu dilakukan proses *differencing*.
7. Setelah data stasioner, maka identifikasi orde p , d , dan q untuk model ARIMA berdasarkan plot PACF, banyaknya *differencing* yang dilakukan hingga data stasioner, dan plot ACF.
8. Melakukan estimasi parameter model dan uji diagnostik model untuk mengetahui apakah model sudah cukup baik untuk digunakan dalam prediksi dan peramalan. Model yang baik residualnya bersifat *white noise*.
9. Menentukan model ARIMA terbaik berdasarkan nilai AIC terkecil.
10. Berdasarkan model ARIMA terbaik yang telah dipilih, kemudian dilakukan prediksi dan peramalan untuk harga penutupan saham Bank BRI.
11. Menghitung nilai residual dengan cara mengurangi data aktual dari data harga penutupan saham Bank BRI dengan data hasil prediksi yang diperoleh dari metode ARIMA yang telah di- *undifferencing*.
12. Data prediksi dan residual dari ARIMA digunakan sebagai input yang akan diolah dengan menggunakan metode LSTM.
13. Melakukan *splitting* untuk kedua data dengan skema 80% *training* 20% *testing* dan 90% *training* 10% *testing*.
14. Melakukan *preprocessing* data, yaitu dengan melakukan normalisasi pada data prediksi dan residual ARIMA menggunakan *minmaxscaler*.
15. Membangun model LSTM pertama dengan data prediksi ARIMA dan model LSTM kedua dengan data residual ARIMA menggunakan parameter-parameter terbaik yang telah diperoleh dari proses *hyperparameter tuning*. Parameter-parameter yang dibutuhkan untuk membangun model, yaitu jumlah neuron, *epochs*, dan *batch size*.
16. Melakukan prediksi dan peramalan model LSTM pertama dengan menggunakan data prediksi ARIMA.
17. Melakukan prediksi dan peramalan model LSTM kedua dengan menggunakan data residual ARIMA.
18. Menggabungkan hasil prediksi dan peramalan kedua model LSTM dengan proses penjumlahan.

19. Menghitung akurasi model *hybrid* ARIMA-LSTM berdasarkan nilai MAPE dan RMSE.



Gambar 3. Diagram Alir Metode Penelitian.

V. KESIMPULAN

Berdasarkan hasil dari pembahasan mengenai peramalan harga penutupan saham BRI dengan menggunakan metode *hybrid* ARIMA-LSTM dapat diperoleh kesimpulan sebagai berikut.

1. Model *hybrid* ARIMA-LSTM dibangun dengan menggunakan kombinasi jumlah *neuron*, *batch size*, dan *epoch* terbaik yang diperoleh pada proses *hyperparameter tuning* dengan rincian sebagai berikut:
 - a. Untuk model LSTM dengan data input prediksi ARIMA kombinasi jumlah *neuron*, *batch size*, dan *epoch* terbaik yang diperoleh untuk membangun model dengan *splitting data* 80% data *training* 20% data *testing* yaitu 128, 4, 100, dan untuk *splitting data* 90% data *training* 10% data *testing* yaitu 128, 4, 100.
 - b. Sedangkan untuk model LSTM dengan data input residual ARIMA, kombinasi jumlah *neuron*, *batch size*, dan *epoch* terbaik yang diperoleh untuk membangun model dengan *splitting data* 80% data *training* 20% data *testing* yaitu 128, 16, 50, dan untuk *splitting data* 90% data *training* 10% data *testing* yaitu 64, 4, 100.
2. Metode *hybrid* ARIMA-LSTM merupakan metode yang baik untuk digunakan melakukan prediksi maupun peramalan untuk data harga penutupan saham BRI karena menghasilkan nilai RMSE dan MAPE yang rendah. Model *hybrid* dengan *splitting data* 80% data *training* 20% data *testing* menghasilkan nilai RMSE sebesar 140,3567 dan MAPE sebesar 0,0233%. Sedangkan model *hybrid* dengan *splitting data* 90% data *training* 10% data *testing* menghasilkan nilai RMSE sebesar 126,6367 dan MAPE sebesar 0,0228%. Berdasarkan hal tersebut dapat disimpulkan bahwa model *hybrid* dengan *splitting data* 90% data

training 10% data *testing* memiliki tingkat akurasi lebih tinggi dibandingkan dengan model *hybrid* dengan *splitting data* 80% data *training* 20% data *testing*.

3. Hasil peramalan yang diperoleh selama 30 periode kedepan (31 Oktober 2022 sampai dengan 9 Desember 2022) dengan menggunakan metode *hybrid* ARIMA-LSTM dengan *splitting data* 90% data *training* 10% data *testing* menghasilkan nilai peramalan yang paling baik dibandingkan dengan model *hybrid* ARIMA-LSTM dengan *splitting data* 80% data *training* 20% maupun model LSTM saja. Hasil peramalan dengan menggunakan metode *hybrid* ARIMA-LSTM dengan *splitting data* 90% data *training* 10% data *testing* menghasilkan nilai peramalan yang naik dari waktu ke waktu, sedangkan hasil peramalan dengan model *hybrid* ARIMA-LSTM dengan *splitting data* 80% data *training* 20% dan model LSTM saja menghasilkan nilai peramalan yang mengalami penurunan dari waktu ke waktu.

DAFTAR PUSTAKA

- Arfan, A. dan Lusiana, E.T.P. 2019. Prediksi Harga Saham di Indonesia Menggunakan Algoritma Long Short Term Memory. *Prosiding Seminar Nasional Teknologi Informasi dan Komunikasi*. **3**(1): 225-230.
- Berradi, Z. dan Lazaar, M. 2019. Integration of Principal Component Analysis and Recurrent Neural Network to Forecast the Stock Price of Casablanca Stock Exchange. *Procedia Computer Science*. **148**(2019): 55-61.
- Box, G.E.P., Jenkins, G.M., Reinsel, G.C., dan Ljung, G.M. 2016. *Time Series Analysis Forecasting and Control*. 5th Edition. John Wiley & Sons, New Jersey.
- Cahyono, F.R., Irwan, dan Nurfadilah. 2018. Peramalan Tingkat Suku Bunga Pasar Uang Antar Bank (PUAB) dengan Vector Autoregressive Exogeneous (VARX). *Jurnal Matematika dan Statistika serta Aplikasinya*. **6**(1): 51-60.
- Chung, H., and Shin, K. S. 2019. Genetic Algorithm-Optimized Long Short – Term Memory Network for Stock Market Prediction. *Sustainability Journal*. **10**(10): 1-18.
- Davies dan Benon, P. 2004. *Database Systems*. 3th Edition. Hampshire: Palgrave Macmillan, New York.
- Felix, A.G., Schmidhuber, J., dan Cummins, F. 2000. Learning to Forget: Continual Prediction with LSTM. *Neural Computation*. **12**(10): 2451-2471.
- Han, J., Kamber, M., dan Pei, J. 2012. *Data Mining: Concepts and Techniques*. 3th Edition. Morgan Kaufmann, San Francisco.

- Hartama, D., Windarto, A.P., dan Wanto, A. 2019. The Application of Data Mining in Determining Patterns of Interest of High School Graduates. *Journal of Physics: Conference Series*. **1339**(2019): 2-6.
- Hochreiter, S. dan Schmidhuber, J. 1997. Long Short-Term Memory. *Neural Computation*. **9**(8): 1735-1780.
- Karno, A.S.B. 2020. Prediksi Data Time Series Saham Bank BRI Dengan Mesin Belajar LSTM (Long Short Term Memory). *JIFORTY*. **1**
- Kulshreshtha, S. dan A, V. 2020. An ARIMA-LSTM Hybrid Model for Stock Market Prediction Using Live Data. *Journal of Engineering Science and Technology Review*. **13**(4): 117-123.
- Kurniawansyah, A.S. 2018. Implementasi Metode Artificial Neural Network dalam Memprediksi Hasil Ujian Kebidanan. *Jurnal Pseudocode*. **5**(1): 37-44.
- Li, L., Jamieson, K., Desalvo, G., Rostamizadeh, A., dan Talwalkar, A. 2018. Hyperband: A Novel Bandit-Based Approach to Hyperparameter Optimization. *Journal of Machine Learning Research*. **18**(2018): 1-52.
- Makridakis, S., Wheelwright, S.C., dan Hyndman, R.J. 1997. *Forecasting: Methods and Applications*. 3th Edition. John Wiley & Sons, New York.
- Montgomery, D.C., Jennings, C.L., dan Kulahci, M. 2015. *Introduction to time series analysis and forecasting*. 2th Edition. John Wiley & Sons, New Jersey.
- Namini, S.S., Tavakoli, N., dan Namin, A.S. 2018. A Comparison of ARIMA and LSTM in Forecasting Time Series. *IEEE International Conference on Machine Learning and Applications*. 1394-1401.
- Nugroho, K.S., Akbar, I., Suksmawati, A.N., dan Istiadi. 2021. Deteksi Depresi dan Kecemasan Pengguna Twitter Menggunakan Bidirectional LSTM. *The 4th Conference on Innovation and Application of Science and Technology (CIASTECH 2021)*. 287-296.

- Rowan, Muflikhah, L., dan 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.
- Salehinejad, H., Sankar, S., Barfett, J., Colak, E., dan Valaee, S. 2018. Recent Advance in Recurrent Neural Network. 1-21.
- Selle, N., Yudistira, N., dan Dewi, C. Perbandingan Prediksi Penggunaan Listrik dengan Menggunakan Metode Long Short Term Memory (LSTM) dan Recurrent Neural Network. *Journal Teknologi Informasi dan Ilmu Komputer*. **9**(1): 155-162.
- Setiawan, Y., Tarno, dan Kartikasari, P. 2022. Prediksi Harga Jual Kakao dengan Metode Long Short Term Memory Menggunakan Metode Optimasi Root Mean Square Propagation dan Adaptive Moment Estimation Dilengkapi Gui Rshiny. *Jurnal Gaussian*. **11**(1): 99-107.
- Temur, A.S., Akgun, M., dan Temur, G. 2019. Predicting Housing Sales in Turkey Using ARIMA, LSTM, and Hybrid Models. *Journal of Business Economics and Management*. **20**(5): 920-938.
- Temur, A.S dan Yilzid, S. 2021. Comparison of Forecasting Performance of ARIMA LSTM and Hybrid Model for The Sales Volume Budget of a Manufacturing Enterprise. *Istanbul Business Research*. **50**(1): 15-46.
- Tian, C., Ma, J., Zhang, C., dan Zhan, P. 2018. A Deep Neural Network Model for Short-Term Load Forecast Based on Long Short-Term Memory Network and Convolutional Neural Network. *Energies*. 11
- Vijayalakshmi, V. dan Venkatachalapathy, K. Deep Neural Network for Multi-Class Prediction of Student Performance in Educational Data. *International Journal of Recent Technology and Engineering (IJRTE)*. **8**(2): 5073-5081.
- Warmansyah, J. dan Hilpiah, D. 2019. Penerapan metode fuzzy sugeno untuk prediksi persediaan bahan baku. *Jurnal Ilmiah Teknologi Informasi dan Sains*. **9**(2): 12-20.

Wiranda, L. dan Sadikin, M. 2019. Penerapan Long Short Term Memory pada Data Time Series untuk Memprediksi Penjualan Produk PT. Metiska Farma. *Jurnal Nasional Pendidikan Teknik Informatika*. 8(3): 184-196.

Zhang, G.P. 2001. Time Series Forecasting Using A Hybrid ARIMA and Neural Network Model. *Elsevier Neuro Computing*. 50(2003): 159-175.

Zhu, J., Yang, Z., Guo, Y., Zhang, J., dan Yang, H. 2019. Short-Term Load Forecasting for Electric Vehicle Charging Stations Based on Deep Learning Approaches. *Applied Sciences*. 9(9).