

Analisis Peramalan Harga Saham PT Unilever Indonesia Menggunakan Pemodelan LSTM dengan Optimasi PSO

Dewinto Burhan¹, Isran K. Hasan², and La Ode Nashar²

¹Program Studi Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Gorontalo, Indonesia

²Program Studi Statistika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Gorontalo, Indonesia

Article History

Received 05 Januari 2026

Revised 13 Februari 2026

Accepted 26 Februari 2026

Published 28 Februari 2026



Copyright © 2026 by Authors, Published by JRMM Group. This is an open access article under the CC BY-SA License.



Abstract. Stock price movements tend to be volatile and difficult to predict due to their complex patterns and dynamic changes over time. To address this issue, this study employs a Long Short-Term Memory (LSTM) model for stock price forecasting. To achieve optimal model performance, hyperparameter optimization was conducted using the Particle Swarm Optimization (PSO) method. The optimization focused on three main LSTM parameters, namely LSTM units, dropout rate, and dense units. The optimization process resulted in six best model configurations. The testing results indicate that the fifth model achieved the best optimization performance, with an RMSE of 120,3320 and an MAPE of 3,53%, using the following hyperparameter combination: LSTM units = 137, dropout rate = 0,498, and dense units = 32. These results demonstrate that hyperparameter optimization improves forecasting accuracy compared to the LSTM model without optimization. Therefore, the combination of the LSTM model with PSO optimization produces more accurate and stable stock price forecasts. This approach can serve as an alternative for analyzing stock price movements and supporting investment decision-making.

Keywords: Stock Forecasting, LSTM, PSO, Optimization, Time Series

Abstrak. Pergerakan harga saham cenderung tidak stabil dan sulit diprediksi karena memiliki pola yang kompleks serta berubah-ubah dari waktu ke waktu. Untuk mengatasi hal tersebut, penelitian ini menggunakan model *Long Short-Term Memory* (LSTM) dalam melakukan peramalan harga saham. Agar model yang dihasilkan memiliki kinerja yang optimal, dilakukan optimasi hyperparameter menggunakan metode *Particle Swarm Optimization* (PSO). Optimasi dilakukan pada tiga parameter utama LSTM yaitu LSTM units, dropout rate, dense units. Dari proses optimasi diperoleh enam konfigurasi model terbaik. Hasil pengujian menunjukkan model ke-5 memberikan hasil optimasi paling baik dengan nilai RMSE 120,3320 dan MAPE sebesar 3,53% dengan menggunakan kombinasi hyperparameter LSTM units = 137, dropout rate = 0,498, dan dense units = 32. Hasil ini menunjukkan bahwa optimasi hyperparameter memberikan peningkatan akurasi peramalan dibandingkan LSTM tanpa optimasi. Dengan demikian, kombinasi model LSTM dengan optimasi PSO mampu menghasilkan peramalan harga saham yang lebih akurat dan stabil. Pendekatan ini dapat digunakan sebagai alternatif dalam analisis pergerakan harga saham dan mendukung pengambilan keputusan investasi.

Kata kunci: Peramalan Saham, LSTM, PSO, Optimasi, Time Series.

1. Pendahuluan

Pergerakan harga saham merupakan salah satu indikator penting dalam pasar modal yang mencerminkan kinerja serta nilai suatu perusahaan di mata investor [1]. Saham adalah lembar kertas yang menyatakan bahwa pemegang saham adalah pemilik perusahaan yang menerbitkan sekuritas tersebut [2]. Di pasar modal Indonesia, terdapat berbagai perusahaan publik yang menarik perhatian investor, salah satunya adalah PT Unilever Indonesia Tbk yang beroperasi di sektor industri barang konsumsi dan memiliki kinerja pasar yang signifikan.

PT Unilever Indonesia adalah perusahaan *fast moving consumer goods* (FMCG) terkemuka di Indonesia sejak didirikan pada 5 Desember 1933 dengan portofolio lebih dari 40 merek dan kegiatan produksi yang tersebar pada sembilan pabrik di berbagai wilayah. Perusahaan mulai menawarkan sahamnya kepada publik pada tahun 1981 dan sejak 11 Januari 1982 sahamnya resmi tercatat di Bursa Efek Indonesia (BEI) [3]. Pergerakan harga saham PT Unilever Indonesia memberikan dampak yang signifikan terhadap investor dan perekonomian secara keseluruhan. Maka dari itu penting untuk dapat meramalkan pergerakan harga saham Unilever Indonesia dengan akurasi yang tinggi untuk membantu investor mengambil keputusan investasi yang lebih baik.

Peramalan pada saham sangat sulit dan kompleks karena banyak faktor yang mempengaruhi harga saham [4]. Pada dasarnya, terdapat tiga faktor utama yang berperan dalam menganalisa pergerakan saham, yaitu analisis teknikal, analisis fundamental, dan analisis sentimen [5]. Seiring meningkatnya kebutuhan analisis data, pendekatan berbasis komputasi cerdas semakin banyak digunakan dalam peramalan harga saham. *Machine learning* memungkinkan model belajar langsung dari data historis, sedangkan *deep learning* sebagai bagiannya memanfaatkan arsitektur *neural network* yang terinspirasi oleh model biologis [6]. Dalam peramalan data time series, khususnya harga saham, Salah satu pendekatan *deep learning* yang sering dipakai adalah *Long Short Term Memory* (LSTM) [7].

Long Short-Term-Memory (LSTM) adalah metode pengembangan dari *Recurrent Neural Network* (RNN) [8]. Menurut [9] model LSTM dianggap lebih efektif dalam memprediksi harga saham dibandingkan RNN. Hal ini karena RNN memiliki keterbatasan dalam mengaitkan informasi jangka panjang, yang menyebabkan masalah hilangnya nilai gradien (*vanishing gradient*) selama proses iterasi neural network [1]. Maka dari itu LSTM dirancang khusus untuk mengatasi masalah *vanishing gradient* serta ketergantungan informasi jangka panjang (*long term dependencies*) pada RNN [10]. Dalam

pelatihan model LSTM, pengaruh hyperparameter sangat signifikan. Untuk mendapatkan hyperparameter yang optimal diperlukan algoritma optimasi. Algoritma optimasi bertujuan untuk meminimalkan kesalahan dan memaksimalkan akurasi dengan menemukan bobot optimal [11]. Salah satu algoritma optimasi yang dapat digunakan adalah *Particle Swarm Optimization* (PSO). PSO adalah algoritma evolusioner berbasis populasi yang didasarkan pada pola pergerakan kelompok ikan atau burung yang dapat mencari solusi optimal atau hampir optimal untuk berbagai jenis masalah kompleks tanpa terjebak pada minimum lokal [12].

LSTM banyak digunakan dalam meramalkan harga saham. [13] dalam penelitiannya menerapkan LSTM dan *artificial rabbits optimization* (ARO) yang menunjukkan bahwa LSTM-ARO lebih unggul dari model LSTM dan ANN lainnya. Penelitian lain oleh [14] menggunakan LSTM dengan menggunakan parameter yang bervariasi terhadap waktu dalam memprediksi harga mata uang kripto. [15] melakukan peramalan harga mata uang kripto menggunakan LSTM yang mendapatkan prediksi terbaik pada koin DODGE dengan Epoch 20 dan nilai RMSE 0,0630. [16] membahas metode LSTM dalam prediksi harga saham syariah menggunakan beberapa parameter pada *layers*, *epoch* dan *time step* serta menggunakan arsitektur *multiple layer* LSTM.

Berdasarkan penelitian-penelitian sebelumnya, masih terdapat kesenjangan dalam penentuan kombinasi hyperparameter LSTM yang optimal secara sistematis untuk peramalan harga saham, khususnya pada saham PT Unilever Indonesia. Banyak penelitian menggunakan penentuan hyperparameter secara manual atau terbatas pada parameter tertentu, sehingga hasil peramalan belum sepenuhnya optimal dan konsisten. Oleh karena itu, penelitian ini bertujuan untuk membangun model peramalan harga saham PT Unilever Indonesia menggunakan metode *Long Short-Term Memory* (LSTM) dengan optimasi hyperparameter berbasis *Particle Swarm Optimization* (PSO). Penelitian ini bersifat univariate, dengan hanya menggunakan satu variabel input berupa harga penutupan (*Close*) dalam proses peramalan.

Kebaruan dari penelitian ini terletak pada penerapan optimasi *Particle Swarm Optimization* (PSO) secara terfokus pada tiga hyperparameter utama LSTM (LSTM units, dropout rate, dan dense units) dalam peramalan harga saham PT Unilever Indonesia, serta evaluasi beberapa konfigurasi model hasil optimasi untuk menentukan model terbaik secara kuantitatif. Penelitian ini menyajikan pendekatan optimasi yang lebih sistematis dan terukur. Kontribusi penelitian ini diharapkan dapat memperkaya kajian ilmiah terkait integrasi metode deep learning dan algoritma optimasi dalam peramalan harga saham, serta memberikan referensi praktis bagi investor dan peneliti dalam membangun model peramalan harga saham yang lebih akurat, stabil, dan adaptif terhadap pola data yang dinamis.

2. Metode

Penelitian ini menggunakan studi komputasional berbasis data historis untuk meramalkan harga saham PT Unilever Indonesia. Pendekatan yang digunakan bersifat prediktif dan analitis, dengan fokus pada pengembangan model deret waktu menggunakan *Long Short-Term Memory* (LSTM) yang

dioptimalkan melalui algoritma *Particle Swarm Optimization* (PSO).

2.1. Sumber Data

Dataset yang digunakan pada penelitian ini adalah data sekunder dari harga saham PT. Unilever Indonesia. Data ini bersumber dari situs web resmi data pergerakan harga saham Yahoo Finance (finance.yahoo.com).

2.2. Variabel Penelitian

Data yang digunakan dalam penelitian ini adalah data historis harga saham PT Unilever Indonesia. Variabel yang digunakan terdiri dari tanggal (*date*) dan harga penutupan (*close*), di mana variabel *close* yang digunakan sebagai variabel input dalam pemodelan, sedangkan variabel *date* berfungsi sebagai indeks waktu.

Table 1: Variabel Penelitian

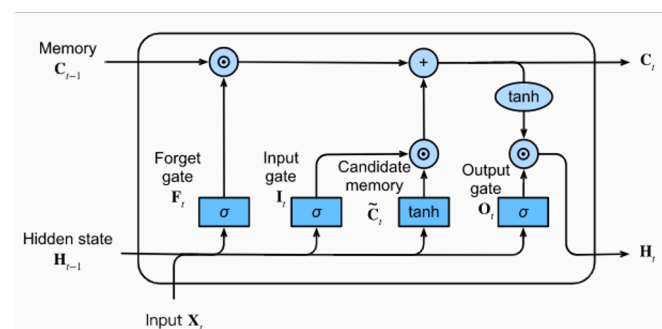
Variabel	Keterangan	Skala
Date	Tanggal harian saham	Interval
Close	Harga penutupan saham	Rasio

2.3. Software Aplikasi

Software yang digunakan adalah *Google Colab* dengan menggunakan bahasa pemrograman *Python*. *Software* ini akan digunakan dalam pengolahan data, pemodelan, hingga visualisasi peramalan harga saham PT. Unilever Indonesia menggunakan metode *Long Short-Term Memory* dengan algoritma *Particle Swarm Optimization*

2.4. Long Short-Term Memory (LSTM)

Long Short Term Memory pertama kali diperkenalkan oleh Hochreiter dan Schmidhuber tahun 1997. LSTM adalah turunan dari metode *Recurrent Neural Network*. Arsitektur LSTM tersusun atas *input layer*, *hidden layer*, dan *output layer*, sehingga model ini mampu menyimpan serta mengelola informasi jangka panjang secara lebih efektif [10], sebagaimana ditunjukkan pada Gambar 1.



Gambar 1: Arsitektur Long Short-Term Memory (LSTM)

Di dalam lapisan tersembunyi (*hidden layer*) LSTM terdapat *memory cell* dan tiga *gate units*, yaitu *forget gate*, *input gate*, dan *output gate*, yang mengatur aliran informasi pada setiap waktu ke- t [17].

1. *Forget Gate* (f_t)

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

f_t menyatakan nilai *forget gate*, σ fungsi sigmoid, W_f bobot, b_f bias, h_{t-1} *hidden state* sebelumnya, dan x_t input pada waktu ke- t .

2. *Input Gate* (i_t)

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad (2)$$

$$\tilde{C}_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \quad (3)$$

i_t adalah *input gate*, $\tilde{C}_t$ kandidat *cell state*, $\tanh$ fungsi aktivasi hiperbolik, sedangkan W_i , W_c dan b_i , b_c masing-masing bobot dan bias.

3. *Cell State* (c_t)

$$c_t = f_t c_{t-1} + i_t \tilde{C}_t \quad (4)$$

c_t adalah *cell state* saat ini dan c_{t-1} *cell state* sebelumnya.

4. *Output Gate* (o_t)

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (5)$$

$$h_t = o_t \tanh(c_t) \quad (6)$$

o_t adalah *output gate*, W_o bobot, b_o bias, dan h_t *hidden state* pada waktu ke- t [18].

2.5. Particle Swarm Optimization (PSO)

Particle Swarm Optimization dikembangkan pada tahun 1995 oleh James Kennedy dan Russ Eberhart sebagai metode optimasi berbasis populasi, yang konsep dasarnya diadaptasi dari perilaku sosial kawanan burung dan gerombolan ikan dalam bergerak secara berkelompok dan terkoordinasi tanpa saling bertabrakan, meskipun berada pada jarak yang relatif dekat. Dalam PSO, setiap individu diasumsikan sebagai sebuah partikel yang merepresentasikan solusi terhadap suatu permasalahan, yang masing-masing partikel memiliki posisi (x) dan kecepatan (v) [19].

Proses algoritma PSO dimulai dengan pembentukan sekumpulan partikel secara acak, di mana masing-masing partikel merepresentasikan satu kandidat solusi. Selanjutnya, posisi dan kecepatan masing-masing partikel diperbarui secara berulang untuk memperoleh solusi terbaik. Proses optimasi akan dihentikan apabila solusi terbaik telah tercapai atau kriteria penghentian yang telah ditentukan terpenuhi. [20]. Algoritma PSO memiliki tahapan sebagai berikut:

1. Melakukan inisialisasi posisi dan kecepatan awal partikel.
2. Menghitung nilai *fitness* pada partikel ke- i . Nilai *fitness* berfungsi sebagai indikator tingkat kualitas suatu partikel dalam merepresentasikan solusi. Nilai *fitness* setiap partikel dihitung dengan persamaan berikut:

$$fitness = RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (7)$$

3. Menentukan *Personal Best* ($Pbest$)

Pada iterasi awal, nilai $Pbest$ ditetapkan berdasarkan posisi awal masing-masing partikel.

4. Menentukan *Global Best* ($Gbest$)

Nilai $Gbest$ diperoleh dari partikel yang memiliki nilai $Pbest$ dengan nilai *fitness* paling optimal (terkecil).

5. Pembaruan Kecepatan dan Posisi Partikel

Pembaruan kecepatan partikel dinyatakan sebagai berikut:

$$v_{i,j}(t+1) = wv_{i,j}(t) + c_1r_1(Pbest_{i,j}(t) - x_{i,j}(t)) + c_2r_2(Gbest_j(t) - x_{i,j}(t)) \quad (8)$$

Pada persamaan tersebut, $v_{i,j}(t+1)$ adalah kecepatan partikel ke- i pada dimensi ke- j di iterasi $t+1$, sedangkan $v_{i,j}(t)$ adalah kecepatan sebelumnya. Parameter w menyatakan bobot inersia. Koefisien c_1 dan c_2 masing-masing adalah faktor kognitif dan sosial. Variabel acak r_1 dan r_2 berada pada rentang $[0, 1]$. Simbol $Pbest_{i,j}(t)$ adalah posisi terbaik partikel ke- i , $Gbest_j(t)$ adalah posisi terbaik global pada dimensi ke- j , dan $x_{i,j}(t)$ adalah posisi partikel saat ini. Pembaruan posisi partikel diberikan oleh:

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1) \quad (9)$$

Pada persamaan ini, $x_{i,j}(t+1)$ adalah posisi baru partikel, $x_{i,j}(t)$ adalah posisi sebelumnya, dan $v_{i,j}(t+1)$ adalah kecepatan terbaru yang telah diperbarui.

6. Pembaruan nilai $Pbest$ untuk partikel ke- i dengan melakukan perbandingan nilai *fitness* antara iterasi saat ini dan iterasi sebelumnya.
7. Pembaruan nilai $gbest$ dilakukan berdasarkan nilai *fitness* terbaik yang dihasilkan oleh seluruh partikel.
8. Ketika jumlah iterasi belum mencapai batas maksimum, maka langkah 2 hingga 7 diulangi. [20]

2.6. Penyetelan Hyperparameter Model

Keberhasilan proses pelatihan LSTM sangat dipengaruhi oleh pemilihan hyperparameter. Hyperparameter adalah variabel yang nilainya ditetapkan secara manual ke model untuk membantu dalam proses pelatihan [11]. Hyperparameter utama yang digunakan dalam penelitian ini yaitu *LSTM Units* yang merupakan jumlah neuron LSTM yang mempelajari pola sekuensial, *Dropout Rate* yang berfungsi sebagai teknik regulasi untuk mencegah overfitting, serta *Dense Units* yang berfungsi sebagai lapisan output yang memetakan fitur output LSTM ke prediksi akhir [21]. Dalam penelitian ini, model LSTM standar menggunakan pendekatan ruang hyperparameter diskrit melalui pemilihan kombinasi secara acak dari nilai-nilai tertentu, sedangkan model LSTM dengan optimasi PSO menggunakan ruang hyperparameter kontinu, di mana nilai LSTM units, dropout rate, dan dense units dieksplorasi dalam rentang yang telah ditentukan. Pendekatan kontinu memungkinkan PSO untuk menemukan konfigurasi hyperparameter yang lebih optimal meskipun dengan jumlah evaluasi yang terbatas.

Untuk menjaga keadilan dalam perbandingan kinerja model, jumlah kombinasi hyperparameter yang dievaluasi pada model LSTM dengan optimasi PSO dan model LSTM standar dibuat setara, yaitu masing-masing sebanyak enam kombinasi. Kedua model menggunakan ruang hyperparameter yang sama, meliputi jumlah LSTM units pada rentang 50–200, dropout rate pada rentang 0,1–0,5, dan dense units pada rentang 8–32.

Pada model LSTM+PSO, pemilihan kombinasi hyperparameter dilakukan menggunakan algoritma PSO, sedangkan pada model LSTM standar kombinasi dipilih melalui random sampling dalam rentang yang sama. Seluruh kombinasi yang diuji dipastikan unik dan tidak mengalami duplikasi.

2.7. Normalisasi Data

Normalisasi data merupakan teknik untuk menskalakan ulang nilai sehingga menjadi lebih kecil. Proses ini penting ketika terdapat perbedaan skala antar fitur dalam suatu dataset, dengan demikian normalisasi diperlukan untuk menyeimbangkan skala tiap fitur ke rasio yang lebih kecil [22]. Salah satu metode normalisasi yang sering dipakai adalah *Min-Max*. *Min-Max Normalization* adalah metode yang menerapkan penskalaan data secara linear ke batas minimum serta maksimum yang telah ditentukan, umumnya rentang $[0,1]$ [23]. Rumus normalisasi *Min-Max* ditunjukkan pada persamaan berikut.

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (10)$$

Pada persamaan tersebut, x menyatakan nilai asli, x' adalah nilai hasil normalisasi, $\min(x)$ merupakan nilai minimum pada data, dan $\max(x)$ adalah nilai maksimum pada data. Transformasi ini memetakan data ke dalam rentang $[0, 1]$. Prosedur implementasi *Min-Max Normalization* dalam penelitian ini dilakukan dengan:

1. Menghitung parameter min dan max dari data latih saja
2. Melakukan fit scaler pada data latih
3. Mentransformasi data uji menggunakan parameter scaler yang sama

2.8. Pembentukan Dataset Time Series

Teknik *windowing* atau *sliding window* digunakan untuk mengubah data time series menjadi format *supervised learning*. *Sliding window* adalah teknik yang membentuk variabel input X dari beberapa langkah waktu sebelumnya untuk memprediksi variabel output Y pada langkah waktu berikutnya [24]. *Sliding window* memiliki parameter *lookback* yang menentukan jumlah time steps sebelumnya yang digunakan sebagai input dan Horizon prediksi menunjukkan berapa banyak langkah waktu ke depan yang ingin diprediksi [25].

Penentuan jumlah *lookback* bersifat eksperimental, maka peneliti menetapkan ukuran *lookback* sebesar 7, yang mengindikasikan bahwa setiap prediksi didasarkan pada 7 langkah waktu sebelumnya. Horizon prediksi menggunakan strategi *one-step prediction* dengan horizon=1 untuk memprediksi satu nilai pada periode berikutnya. Proses *shifting* menggunakan faktor pergeseran sebesar 1, sehingga *window* bergeser satu langkah maju pada setiap iterasi. Selain itu, data tidak mengalami proses shuffling untuk mempertahankan urutan kronologis yang diperlukan model dalam pembelajaran pola *time series*.

2.9. Metrik Evaluasi Parameter

Pada penelitian ini, evaluasi parameter yang digunakan yaitu *Mean Absolute Percentage Error* (MAPE) dan *Root Mean Squared Error* (RMSE). MAPE dan RMSE adalah dua metrik

yang umum diterapkan dalam mengevaluasi kinerja model prediksi [9]. Model prediksi menjadi lebih baik jika nilai MAPE semakin rendah. RMSE digunakan untuk mengukur perbedaan antara target estimasi dan target aktual dengan menghitung nilai akar kuadrat dari MSE. Semakin besar nilai yang dihasilkan oleh RMSE, semakin kecil tingkat akurasi, dan sebaliknya, jika nilai RMSE yang dihasilkan lebih rendah, tingkat akurasi lebih tinggi [11]. Seluruh perhitungan nilai MAPE dan RMSE pada penelitian ini dilakukan menggunakan data pada skala asli, dengan terlebih dahulu mengembalikannya hasil prediksi dari skala normalisasi melalui proses *inverse transform*. Rumus MAPE dan RMSE ditunjukkan pada persamaan berikut:

$$\text{MAPE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100\% \quad (11)$$

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (12)$$

Pada persamaan tersebut, MAPE (*Mean Absolute Percentage Error*) mengukur rata-rata kesalahan persentase absolut, sedangkan RMSE (*Root Mean Squared Error*) mengukur akar dari rata-rata kuadrat kesalahan. Simbol y_i menyatakan nilai aktual, $\hat{y}_i$ nilai prediksi, dan n jumlah data pengamatan.

3. Hasil dan Pembahasan

3.1. Deskripsi Data

Data yang digunakan dalam penelitian ini adalah pergerakan harga saham harian dari perusahaan PT. Unilever Indonesia Tbk dengan kode saham "UNVR.JK". Data ini diambil melalui situs resmi keuangan Yahoo Finance dengan rincian historis dari tanggal 2 Januari 2020 hingga 30 Desember 2024. Berikut visualisasi time series dari variable *close*



Gambar 2: Grafik Harga Penutupan PT. Unilever Indonesia

Pada **Gambar 2** ditampilkan grafik harga penutupan saham PT. Unilever Indonesia selama periode pengamatan. Grafik tersebut memperlihatkan fluktuasi harga penutupan dari tahun ke tahun, dengan titik tertinggi sebesar 8.575 pada tanggal 3 Januari 2020 dan titik terendah sebesar 1.735 pada tanggal 7 November 2024. Dari grafik dapat diamati bahwa harga saham mengalami tren penurunan dalam beberapa tahun terakhir, yang ditandai dengan perbedaan signifikan antara harga tertinggi dan terendah.

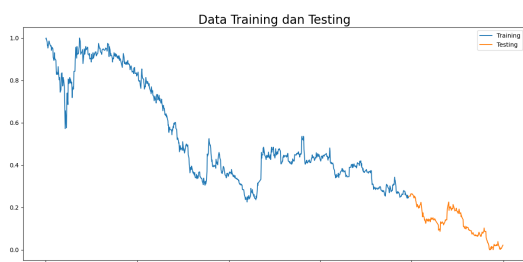
3.2. Data Preparation

Tahap *data preparation* meliputi pengecekan *missing value* untuk memastikan kelengkapan data. Pengecekan nilai hilang

dilakukan menggunakan fungsi *isna()* dari pustaka *pandas*. Hasil pengecekan menunjukkan bahwa seluruh variabel pada dataset tidak memiliki *missing value*.

Selanjutnya, dilakukan analisis terhadap skala data pada variabel *Close*. Berdasarkan visualisasi pada [Gambar 2](#), terlihat bahwa nilai harga saham memiliki rentang yang cukup besar, sehingga diperlukan proses normalisasi untuk meningkatkan stabilitas dan kinerja model.

Sebelum proses normalisasi, dataset dibagi menjadi dua bagian, yaitu 968 data (80%) sebagai data latih (*training data*) dan 243 data (20%) sebagai data uji (*testing data*). Data latih mencakup periode 2 Januari 2020 hingga 19 Desember 2023, sedangkan data uji mencakup periode 20 Desember 2023 hingga 30 Desember 2024. Pembagian ini dilakukan secara kronologis tanpa melakukan *shuffling* atau pengacakan untuk mempertahankan karakteristik data deret waktu.



Gambar 3: Grafik Harga Penutupan PT. Unilever Indonesia

Setelah pembagian data, normalisasi dilakukan menggunakan metode *MinMax* seperti pada Persamaan (10). Parameter nilai minimum dan maksimum dihitung berdasarkan data latih, kemudian parameter tersebut digunakan untuk mentransformasikan data uji ke dalam rentang 0 hingga 1. Pendekatan ini bertujuan untuk menghindari terjadinya *data leakage*.

3.3. Long Short-Term Memory

Tahap pemodelan LSTM dilakukan dengan menguji tiga hyperparameter utama, yaitu LSTM units, dropout rate dan dense units. Ketiga hyperparameter tersebut diuji dengan beberapa nilai untuk menemukan kombinasi terbaik. Berikut adalah tabel hyperparameter yang digunakan pada penelitian ini.

Table 2: Hyperparameter LSTM

Hyper-parameter	Nilai	Status	Keterangan
LSTM units	50, 100, 200	Divariasikan	Jumlah unit/neuron pada layer LSTM
dropout rate	0.1, 0.2, 0.3	Divariasikan	Nilai dropout pada setiap layer
dense units	8, 16, 32	Divariasikan	Jumlah unit/neuron pada <i>dense layer</i>
epochs	100	Tetap	Jumlah <i>epoch</i> pelatihan
batch size	16	Tetap	Ukuran <i>batch</i> saat training
optimizer	<i>adam</i>	Tetap	Algoritma optimasi
activation (Dense)	<i>relu</i>	Tetap	Fungsi aktivasi pada <i>dense layer</i>
loss function	<i>mse</i>	Tetap	Fungsi <i>loss</i> untuk training

Berdasarkan parameter pada [Tabel 2](#), berikut kombinasi hyperparameter utama yang akan digunakan dalam model LSTM.

Table 3: Kombinasi Hyperparameter LSTM

LSTM units	dropout rate	dense units
50	0.1	8
100	0.2	16
200	0.3	32
50	0.2	16
100	0.3	32
200	0.1	8

Berdasarkan [Tabel 3](#) peneliti akan menggunakan 6 kombinasi hyperparameter utama beserta hyperparameter tetap lainnya yang dapat dilihat pada [Tabel 2](#).

3.3.1. Hasil Evaluasi Model LSTM

Setelah arsitektur model dan nilai hyperparameter LSTM ditentukan, Selanjutnya Peneliti melakukan proses training dan evaluasi model LSTM. Pada tahap ini, data yang telah melalui proses normalisasi dan pembagian data digunakan sebagai masukan untuk melatih model LSTM. Proses pelatihan dilakukan menggunakan data latih (*training data*), sedangkan proses evaluasi dilakukan menggunakan data uji (*testing data*). Berikut adalah hasil evaluasi dari enam kombinasi hyperparameter LSTM yang telah ditentukan.

Table 4: Hasil Evaluasi 6 Kombinasi Hyperparameter Model LSTM

LSTM units	dropout rate	dense units	RMSE	MAPE (%)
50	0.1	8	452.4475	17.85
100	0.2	16	484.9711	19.11
200	0.3	32	479.0121	18.62
50	0.2	16	521.7426	20.03
100	0.3	32	263.0356	9.91
200	0.1	8	687.7263	27.11

[Tabel 4](#) menunjukkan hasil evaluasi model LSTM menggunakan enam kombinasi hyperparameter dengan menggunakan data skala asli. Setiap kombinasi diuji dan performa model diukur menggunakan metrik RMSE dan MAPE. Kombinasi hyperparameter terbaik diperoleh setelah training yaitu LSTM units = 100, dropout rate = 0.3, dan dense units = 32 dengan nilai *error* RMSE sebesar 263.0356 dan MAPE sebesar 9.91%.

3.4. Particle Swarm Optimization

Pada penelitian ini algoritma PSO digunakan dalam tuning hyperparameter pada *Long Short Term Memory* dengan hyperparameter utama yang digunakan yaitu LSTM Units, dropout rate dan dense units. Dalam algoritma PSO terdapat sejumlah hyperparameter yang berperan penting untuk mendukung dan mengoptimalkan proses tuning. Hyperparameter PSO dapat dilihat pada tabel berikut.

Table 5: Hyperparameter PSO

Hyper-parameter	Nilai	Keterangan
<i>no_particle</i>	10	Jumlah partikel (solusi) dalam swarm/populasi
<i>no_dim</i>	3	Jumlah dimensi/parameter yang dioptimasi
<i>x_range</i>	[(50,200), (0.1,0.5), (8,32)]	Batas bawah dan atas setiap parameter (ruang pencarian)
<i>v_range</i>	[(1,10), (0.01,0.1), (1,10)]	Batas bawah dan atas kecepatan partikel
<i>iw_range</i>	(0.5, 1.0)	Rentang bobot inersia (<i>inertia weight</i>)
<i>c</i>	(1.4962, 1.4962)	Koefisien kognitif (c_1) dan sosial (c_2)
<i>iter</i>	6	Jumlah iterasi/maksimum pengulangan PSO

Tabel 5 menampilkan hyperparameter PSO yang digunakan dalam penelitian ini. Dapat dilihat pada tabel, *x_range* menyimpan nilai parameter LSTM yang akan dioptimasi, yaitu LSTM units berada pada *range* (50, 200), dropout rate = (0.1, 0.5) dan dense units = (8, 32). Setelah menentukan hyperparameter yang akan digunakan, selanjutnya melakukan optimasi pada hyperparameter LSTM untuk mendapatkan hasil yang optimal. Berikut adalah hasil dari pencarian 6 hyperparameter terbaik selama proses PSO.

Table 6: Hasil Optimasi hyperparameter LSTM+PSO

LSTM units	dropout rate	dense nits	RMSE
87	0.181	32	0.0376
95	0.201	32	0.0472
69	0.100	32	0.0474
95	0.288	32	0.0501
137	0.498	32	0.0520
63	0.100	28	0.0551

Tabel 6 di atas menunjukkan hasil optimasi hyperparameter LSTM menggunakan algoritma PSO. Kombinasi hyperparameter dengan nilai RMSE terendah selama proses optimasi dicatat sebagai hasil optimasi, sebagaimana ditampilkan pada Tabel 6.

3.4.1. Hasil Training dan Evaluasi LSTM+PSO

Pada tahap ini peneliti menggunakan 6 model dari 6 kombinasi hyperparameter terbaik PSO yang dilatih dari kombinasi 3 hyperparameter utama, yaitu LSTM units, dropout rate, dan dense units. Berikut adalah hasil evaluasi 6 model LSTM+PSO.

Table 7: Hasil Evaluasi 6 Model LSTM+PSO

LSTM units	dropout rate	dense units	RMSE	MAPE (%)
87	0.181	32	364.2773	14.20
95	0.201	32	121.4365	3.54
69	0.100	32	503.7554	19.81
95	0.288	32	367.1762	14.34
137	0.498	32	120.3320	3.52
63	0.100	28	256.5222	9.92

Tabel 7 menunjukkan hasil evaluasi ulang model LSTM+PSO menggunakan 6 hyperparamter terbaik hasil optimasi dengan menggunakan data pada skala asli. Setiap kombinasi diuji kembali dan performa model diukur menggunakan metrik RMSE dan MAPE. Kombinasi hyperparameter terbaik diperoleh setelah *retraining* yaitu pada Model 5 dengan LSTM units = 137, dropout rate = 0.498, dan dense units = 32 dengan nilai RMSE sebesar 120.3320 dan MAPE sebesar 3.52%. Berikut adalah hasil prediksi dan *error* pada model terbaik LSTM+PSO.

Table 8: Hasil Prediksi Model 5 LSTM+PSO

Index	Aktual (y_i)	Prediksi ($\hat{y}_i$)
0	3530.0	3424.587158
1	3520.0	3435.488525
2	3520.0	3444.555664
3	3480.0	3452.480713
4	3430.0	3457.660156
⋮	⋮	⋮
231	1750.0	1900.402588
232	1785.0	1873.888794
233	1780.0	1854.100098
234	1840.0	1840.462646
235	1885.0	1837.430664

Tabel 8 di atas memperlihatkan perbandingan antara nilai aktual dan prediksi hasil model LSTM+PSO pada data uji. Dapat diamati bahwa prediksi model hampir sama dengan nilai aktual, baik pada data awal maupun data akhir. Perlu diperhatikan bahwa jumlah data uji yang diprediksi sebesar 236 sampel, berkurang dari 243 sampel data asli karena penerapan *lookback* window sebesar 7 langkah waktu. Pada proses windowing, 7 data pertama dari data uji tidak dapat digunakan untuk membentuk sequence input model, sehingga hanya 236 data (243 - 7) yang tersedia untuk prediksi. Berikut hasil perhitungan error pada data uji sebagai evaluasi performa model.

Table 9: Error Hasil Evaluasi Model 5 LSTM+SPO

Indeks	$\frac{y_i - \hat{y}_i}{y_i}$	$(y_i - \hat{y}_i)^2$
0	0.029862	11111.87
1	0.024009	7142.19
2	0.021433	5691.85
3	0.007908	757.31
4	0.008064	765.08
⋮	⋮	⋮
231	0.085944	22620.94
232	0.049798	7901.22
233	0.041629	5490.82
234	0.000251	0.21
235	0.025236	2262.84
Total	8.299234	3417233.04

Tabel 9 di atas menunjukkan hasil perhitungan *error* dari prediksi yang telah dilakukan. Tabel ini kemudian dievaluasi menggunakan MAPE dan RMSE, dengan tujuan untuk mengetahui tingkat akurasi prediksi terhadap data aktual. Hasil perhitungan MAPE dan RMSE dapat dilihat pada penjelasan berikut.

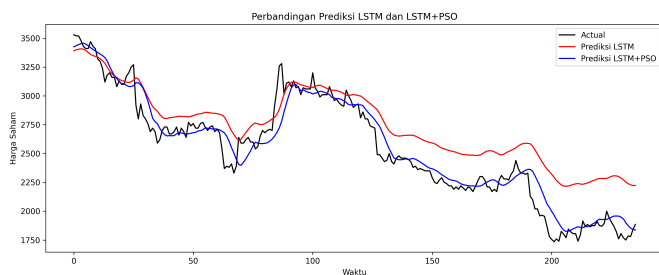
$$\text{MAPE} = \frac{1}{240} \times 8.299234 \times 100\% \approx 3.52\%$$

$$\text{RMSE} = \sqrt{\frac{1}{240} \times 3417233.04} \approx 120.3320$$

Berdasarkan hasil perhitungan MAPE dan RMSE tersebut, tingkat *error* dari data hasil prediksi terhadap data aktual adalah sebesar 3.52% dan 120.3320.

3.5. Perbandingan Hasil Prediksi

Setelah melakukan prediksi harga penutupan saham PT. Unilever Indonesia TBK menggunakan dua pendekatan yaitu model LSTM standar dan LSTM yang dioptimasi menggunakan *Particle Swarm Optimization* (PSO), langkah selanjutnya melakukan perbandingan performa dari kedua model tersebut. Perbandingan hasil prediksi dengan data aktual dari kombinasi terbaik antara LSTM standar dan LSTM+PSO dapat dilihat pada gambar berikut.



Gambar 4: Visualisasi Perbandingan Hasil Prediksi

Berdasarkan visualisasi pada [Gambar 4](#) dapat disimpulkan bahwa baik model LSTM standar maupun model LSTM yang dioptimalkan dengan PSO sama-sama mampu mengikuti pola pergerakan harga saham aktual dengan cukup baik. Namun, prediksi yang dihasilkan oleh LSTM+PSO tampak lebih mendekati nilai aktual di bandingkan LSTM standar. Hal ini menunjukkan bahwa penerapan optimasi PSO pada model LSTM berhasil meningkatkan akurasi prediksi saham, sehingga model LSTM+PSO lebih adaptif dalam menangkap dinamika data historis dibandingkan LSTM tanpa optimasi

4. Kesimpulan

Berdasarkan hasil analisis dan pembahasan mengenai peramalan harga saham PT. Unilever Indonesia menggunakan metode *Long Short Term Memory* (LSTM) yang dioptimasi dengan *Particle Swarm Optimization* (PSO), dapat disimpulkan beberapa hal sebagai berikut:

1. Kinerja model *Long Short-Term Memory* (LSTM) dalam peramalan harga saham PT. Unilever Indonesia menunjukkan bahwa enam model dengan kombinasi hyperparameter yang diuji mampu memprediksi harga penutupan saham dengan baik. Model terbaik (LSTM units = 100, dropout rate = 0,3, dense units = 32) menghasilkan nilai RMSE sebesar 263,0356 dan MAPE sebesar 9,91% pada data uji, yang menandakan tingkat error prediksi yang relatif rendah.
2. Optimasi hyperparameter menggunakan algoritma *Particle Swarm Optimization* (PSO) terbukti mampu meningkatkan performa model LSTM. Hasil retraining dengan kombinasi hyperparameter terbaik dari PSO

(LSTM units = 137, dropout rate = 0,498, dense units = 32) menghasilkan RMSE sebesar 120,3320 dan MAPE sebesar 3,52%, yang lebih baik dibandingkan semua kombinasi pada model LSTM standar.

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan, berikut beberapa saran yang dapat diberikan:

1. Penelitian selanjutnya dapat mempertimbangkan untuk menggunakan metode peramalan *Deep Learning* lainnya seperti *Gated Recurrent Unit* (GRU) untuk dibandingkan dengan metode *Long Short-Term Memory*.
2. Penelitian selanjutnya dapat menggunakan algoritma optimasi lain, seperti *Genetic Algorithm* (GA), *Bayesian Optimization*, atau algoritma metaheuristik lainnya, guna di bandingkan efektivitasnya dengan algoritma *Particle Swarm Optimization*.

Pernyataan Kontribusi Penulis (CRediT)

Dewinto Burhan: Konseptualisasi, Metodologi, Perangkat Lunak, Analisis Formal, Investigasi, Kurasi Data, Penulisan–Draf Awal, Penulisan–Telaah dan Penyuntingan, Visualisasi. **Isran K. Hasan:** Konseptualisasi, Metodologi, Perangkat Lunak, Penulisan–Draf Awal, Penulisan–Telaah dan Penyuntingan, Supervisi. **La Ode Nashar:** Konseptualisasi, Metodologi, Validasi, Penulisan–Draf Awal, Penulisan–Telaah dan Penyuntingan, Supervisi.

Deklarasi Penggunaan AI atau Teknologi Berbasis AI

Model ChatGPT versi 5 digunakan secara terbatas untuk membantu penyusunan draf awal, penyesuaian tata bahasa akademik, dan konsistensi gaya penulisan sesuai standar jurnal. Semua analisis matematis, formulasi, dan interpretasi hasil dilakukan sepenuhnya oleh penulis.

Deklarasi Konflik Kepentingan

Penulis menyatakan tidak ada konflik kepentingan.

Pendanaan dan Ucapan Terima Kasih

Penelitian ini tidak menerima pendanaan eksternal.

Ketersediaan Data

Data yang digunakan dalam penelitian ini merupakan data sekunder dan tersedia secara publik. Dataset harga saham PT Unilever Indonesia Tbk (kode saham: UNVR.JK) diperoleh dari situs web Yahoo Finance².

Daftar Pustaka

- [1] W. Hastomo, A. Satyo, B. Karno, N. Kalbuana, E. Nisfiani, and L. E. -, "Optimasi deep learning untuk prediksi saham di masa pandemi covid-19," *JEPIN (Jurnal Edukasi Penelitian Informatika)*, vol. 7, pp. 133–140, 2 2021.

²<https://finance.yahoo.com/quote/UNVR.JK/>

- [2] R. E. Arini, Y. Iskandar, and H. F. Ningrum, “Faktor – faktor yang mempengaruhi harga saham pada perusahaan sektor keuangan,” *Jurnal Proaksi*, vol. 7, pp. 39–53, 1 Apr. 2020. DOI: [10.32534/jpk.v7i1.1008](https://doi.org/10.32534/jpk.v7i1.1008).
- [3] Unilever, *Unilever indonesia homepage / unilever*, Akses Pada: 11-05-2024, 2024. Available online.
- [4] A. Picasso, S. Merello, Y. Ma, L. Oneto, and E. Cambria, “Technical analysis and sentiment embeddings for market trend prediction,” *Expert Systems with Applications*, vol. 135, pp. 60–70, Nov. 2019. DOI: [10.1016/j.eswa.2019.06.014](https://doi.org/10.1016/j.eswa.2019.06.014).
- [5] R. Julian and M. R. Pribadi, “Peramalan harga saham pertambangan pada bursa efek indonesia (bei) menggunakan long short term memory (lstm),” *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 8, 3 2021. Available online.
- [6] D. Kobiela, D. Krefta, W. Król, and P. Weichbroth, “Arima vs lstm on nasdaq stock exchange data,” in *Procedia Computer Science*, vol. 207, Elsevier B.V., 2022, pp. 3830–3839. DOI: [10.1016/j.procs.2022.09.445](https://doi.org/10.1016/j.procs.2022.09.445).
- [7] R. Yotenka, F. Fikri, and E. Huda, “Implementasi long short-term memory pada harga saham perusahaan perkebunan di indonesia,” *Jurnal UJMC*, vol. 6, pp. 9–18, 1 2020.
- [8] D. I. Puteri, “Implementasi long short term memory (lstm) dan bidirectional long short term memory (bilstm) dalam prediksi harga saham syariah,” *Euler : Jurnal Ilmiah Matematika, Sains dan Teknologi*, vol. 11, pp. 35–43, 1 May 2023. DOI: [10.34312/euler.v11i1.19791](https://doi.org/10.34312/euler.v11i1.19791).
- [9] D. M. Gunarto, S. Sa’adah, and D. Q. Utama, “Predicting cryptocurrency price using rnn and lstm method,” *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 12, pp. 1–8, 1 Mar. 2023. DOI: [10.32736/sisfokom.v12i1.1554](https://doi.org/10.32736/sisfokom.v12i1.1554).
- [10] L. Wiranda and M. Sadikin, “Penerapan long short term memory pada data time series untuk memprediksi penjualan produk pt. metiska farma,” *Jurnal Nasional Pendidikan Teknik Informatika : JANAPATI*, vol. 8, pp. 184–196, 2019. DOI: <https://doi.org/10.23887/janapati.v8i3.19139>.
- [11] Y. L. Sukestiyarno, D. T. Wiyanti, L. Azizah, W. Widada, and K. U. Z. Nugroho, “Algorithm optimizer in ga-lstm for stock price forecasting,” *Contemporary Mathematics (Singapore)*, vol. 5, 1 2024. DOI: [10.37256/cm.5120243367](https://doi.org/10.37256/cm.5120243367).
- [12] N. Kumar and S. Susan, “Particle swarm optimization of partitions and fuzzy order for fuzzy time series forecasting of covid-19,” *Applied Soft Computing*, vol. 110, Oct. 2021. DOI: [10.1016/j.asoc.2021.107611](https://doi.org/10.1016/j.asoc.2021.107611).
- [13] B. Gulmez, “Stock price prediction with optimized deep lstm network with artificial rabbits optimization algorithm,” *Expert Systems with Applications*, vol. 227, Oct. 2023. DOI: [10.1016/j.eswa.2023.120346](https://doi.org/10.1016/j.eswa.2023.120346).
- [14] L. Boongasame and P. Songram, “Cryptocurrency price forecasting method using long short-term memory with time-varying parameters,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 30, pp. 435–443, 1 Apr. 2023. DOI: [10.11591/ijeecs.v30.i1.pp435-443](https://doi.org/10.11591/ijeecs.v30.i1.pp435-443).
- [15] M. F. Rizkilloh and S. Widiyanesti, “Prediksi harga cryptocurrency menggunakan algoritma long short term memory (lstm),” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 6, pp. 25–31, 1 Feb. 2022. DOI: [10.29207/resti.v6i1.3630](https://doi.org/10.29207/resti.v6i1.3630).
- [16] G. Budiprasetyo, M. Hani’ah, and D. Z. Aflah, “Prediksi harga saham syariah menggunakan algoritma long short-term memory (lstm),” *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 8, pp. 164–172, 3 Jan. 2023. DOI: [10.25077/teknosi.v8i3.2022.164-172](https://doi.org/10.25077/teknosi.v8i3.2022.164-172).
- [17] N. Selle, N. Yudistira, and C. Dewi, “Perbandingan prediksi penggunaan listrik dengan menggunakan metode long short term memory (lstm) dan recurrent neural network (rnn),” *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 9, pp. 155–162, 1 Feb. 2022. DOI: [10.25126/jtiik.2022915585](https://doi.org/10.25126/jtiik.2022915585).
- [18] L. N. I. Afida, F. A. Bachtiar, and I. Cholissodin, “Klasifikasi aktivitas manusia menggunakan metode long short-term memory,” *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 11, pp. 357–368, 2 Apr. 2024. DOI: [10.25126/jtiik.20241127060](https://doi.org/10.25126/jtiik.20241127060).
- [19] Y. P. Sugandhi, B. Warsito, and A. R. Hakim, “Prediksi harga saham harian menggunakan cascade forward neural network (cfn) dengan particle swarm optimization (pso),” *STATISTIKA Journal of Theoretical Statistics and Its Applications*, vol. 19, pp. 71–82, 2 Dec. 2019. DOI: [10.29313/jstat.v19i2.4878](https://doi.org/10.29313/jstat.v19i2.4878).
- [20] A. D. Ihsanuddin, D. Ispriyanti, and T. Tarno, “Penerapan metode fuzzy time series menggunakan particle swarm optimization algorithm untuk peramalan indeks saham lq45,” *Jurnal Gaussian*, vol. 12, pp. 10–19, 1 May 2023. DOI: [10.14710/j.gauss.12.1.10-19](https://doi.org/10.14710/j.gauss.12.1.10-19).
- [21] GeeksforGeeks, *Lstm networks*, 2026. Available online.
- [22] P. P. Allorerung, A. Erna, M. Bagussahrir, and S. Alam, “Analisis performa normalisasi data untuk klasifikasi k-nearest neighbor pada dataset penyakit,” *JISKA (Jurnal Informatika Sunan Kalijaga)*, vol. 9, pp. 178–191, 3 Sep. 2024. DOI: [10.14421/jiska.2024.9.3.178-191](https://doi.org/10.14421/jiska.2024.9.3.178-191).
- [23] D. Singh and B. Singh, “Investigating the impact of data normalization on classification performance,” *Applied Soft Computing*, vol. 97, p. 105524, Dec. 2020. DOI: [10.1016/j.asoc.2019.105524](https://doi.org/10.1016/j.asoc.2019.105524).
- [24] A. Essien and C. Giannetti, “A deep learning framework for univariate time series prediction using convolutional lstm stacked autoencoders,” in *2019 IEEE International Symposium on INnovations in Intelligent SysTems and Applications (INISTA)*, IEEE, Jul. 2019, pp. 1–6. DOI: [10.1109/INISTA.2019.8778417](https://doi.org/10.1109/INISTA.2019.8778417).
- [25] R. E. Wahyuni, “Optimasi prediksi inflasi dengan neural network pada tahap windowing adakah pengaruh perbedaan window size,” *Technologia: Jurnal Ilmiah*, vol. 12, p. 176, 3 Jul. 2021. DOI: [10.31602/tji.v12i3.5181](https://doi.org/10.31602/tji.v12i3.5181).