

MODEL DEEP LEARNING UNTUK ANALISIS PREDIKSI HARGA SAHAM MENGGUNAKAN METODE LONG SHORT-TERM MEMORY (LSTM)

¹Muhammad Ruhunul Luthfi, ^{2*}Rama Dian Syah

¹Fakultas Teknologi Industri Universitas Gunadarma, ²Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Gunadarma

Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat

¹muhammad.r.luthfi@gmail.com

*Corresponding author: ²rama_ds@staff.gunadarma.ac.id

Abstrak

Investor yang berinvestasi saham selalu dituntut untuk menganalisis pasar saham agar dapat meminimalkan risiko kerugian dan meningkatkan keuntungan. Pemanfaatan teknologi Deep Learning dengan metode Long Short-Term Memory (LSTM) dapat digunakan untuk membantu analisis prediktif oleh para investor saham. Penelitian ini bertujuan untuk memprediksi harga saham PT Astra International Tbk menggunakan model Deep Learning metode Long Short Term Memory. Metode penelitian ini terdiri dari persiapan data, preprocessing data, pelatihan model, denormalisasi data, visualisasi hasil prediksi, dan evaluasi hasil prediksi. Visualisasi data menunjukkan bahwa model LSTM mampu memprediksi dengan cukup akurat dan mampu menangkap pola tren harga saham. Evaluasi model LSTM menunjukkan nilai MAE mencapai 87.69 dengan persentase MAE mencapai 1.44%, nilai RMSE mencapai 116.87 dengan persentase RMSE mencapai 1.92%, dan nilai MAPE mencapai 1.45%.

Kata Kunci: saham, prediksi, LSTM, deep learning

Abstract

Investors who invest in shares are always required to analyze the stock market in order to minimize the risk of loss and increase profits. Utilization of Deep Learning technology with the Long Short-Term Memory (LSTM) method can be used to assist predictive analysis by stock investors. This research aims to predict the share price of PT Astra International Tbk using the Long Short Term Memory method. This research method consists of data preparation, data preprocessing, model training, data denormalization, visualization of prediction results, and evaluation of prediction result. Data visualization shows that the LSTM model is able to predict quite accurately and is able to capture stock price trend patterns. Evaluation of the LSTM model shows that the MAE value reached 87.69 with an MAE percentage reaching 1.44%, the RMSE value reached 116.87 with an RMSE percentage reaching 1.92%, and the MAPE value reached 1.45%.

Keyword: share, predict, LSTM, deep learning

PENDAHULUAN

Perkembangan zaman menyebabkan investasi saham menjadi salah satu hal yang menarik di masyarakat. Investasi saham menawarkan potensi keuntungan yang menarik di masa depan karena beberapa alasan seperti adanya pertumbuhan ekonomi, pembagian dividen, menjaga nilai uang dari inflasi, dan lain-lain. Saham adalah suatu dokumen yang secara jelas menyatakan nilai nominal, nama industri atau perusahaan, serta hak dan kewajiban masing-masing pemegang saham (Ramadhan, 2020). Banyak perusahaan

besar di Indonesia menawarkan saham untuk para investor. Salah satu perusahaan besar yang menawarkan saham yaitu PT Astra International Tbk yang bergerak dalam bidang industri otomotif Indonesia. Informasi daftar dan harga terkait dengan saham PT Astra International Tbk dapat dilihat di platform penunjang bursa saham, seperti Yahoo Finance, MarketWatch, dan Bursa Efek Indonesia (BEI) yang memberikan berbagai informasi pergerakan harga saham.

Penelitian pada PT Astra International Tbk dilakukan karena perusahaan ini memiliki peran signifikan dalam industri otomotif Indonesia dan memiliki kinerja keuangan yang dapat mencerminkan kondisi ekonomi nasional. PT Astra juga memiliki diversifikasi bisnis yang luas, dari otomotif hingga jasa keuangan dan infrastruktur, menjadikannya subjek ideal untuk menganalisis berbagai aspek keuangan perusahaan, seperti profitabilitas, likuiditas, dan solvabilitas. Astra menjadi contoh penting untuk memahami determinan pertumbuhan laba dalam industri otomotif, yang berdampak pada nilai perusahaan dan pengaruhnya terhadap investor di pasar saham (Herlin, Veronica, Rosita, & Neldawaty, 2022).

Investor dapat melakukan analisis terkait dengan harga saham berdasarkan informasi yang tersedia di platform online. Analisis yang dilakukan dapat digunakan para investor untuk mengetahui tren harga saham sehingga dapat menentukan strategi untuk jual dan beli saham pada waktu yang tepat (Irahadi, Sianturi, & Kim, 2022). Analisis tren harga saham juga dilakukan untuk menentukan keputusan yang diambil investor untuk mengurangi risiko yang mungkin terjadi pada investasi saham (Pradnyawati & Sinarwati, 2022). Analisis tren harga saham juga dapat dilakukan dengan bantuan teknologi Artificial Intelligence agar analisis prediksi harga saham dapat dilakukan secara akurat. Teknologi Artificial Intelligence untuk prediksi harga saham dapat dilakukan dengan menggunakan model *deep learning*.

Penelitian terkait prediksi harga saham menggunakan model Deep Learning, dengan metode LSTM telah dilakukan. Penelitian oleh Budiprasetyo, Hani'ah, dan Aflah, (2023) menggunakan informasi harga penutupan saham dari lima perusahaan yang terdaftar sebagai emiten syariah. Data yang digunakan dari website Yahoo Finance dengan periode pengamatan selama lima tahun, mulai dari tahun 2016 hingga 2021. Penelitian ini bertujuan untuk memprediksi harga saham syariah dengan menggunakan metode LSTM untuk membantu investor untuk menemukan momen yang ideal dalam membeli dan menjual saham. Hasil penelitian menunjukkan bahwa model LSTM yang dihasilkan memiliki performa terbaik dengan nilai Mean Absolute Percentage Error (MAPE) mencapai 1.51%. Penelitian oleh Julian dan Pribadi (2021) menggunakan data harga saham dari emiten pertambangan yang terdaftar di Bursa Efek Indonesia (BEI), yaitu PT Aneka Tambang Tbk (ANTM), PT Timah Tbk (TINS), dan PT Vale Indonesia Tbk (INCO). Data historis yang digunakan mencakup periode dari tanggal 1 Januari 2016 hingga 31 Desember 2020, dan diperoleh dari dataset publik yang tersedia di website Yahoo Finance. Hasil penelitian yang telah dilakukan menunjukkan bahwa nilai *Root Mean Square Error* (RMSE) dalam proses prediksi harga saham mengalami penurunan seiring dengan peningkatan jumlah *epoch* yang diterapkan. Penurunan RMSE ini menunjukkan bahwa model semakin mampu memprediksi harga saham secara akurat seiring bertambahnya *epoch*, namun di sisi lain peningkatan epoch juga berdampak pada waktu komputasi yang dibutuhkan. Pada emiten PT Timah Tbk (TINS), nilai RMSE terkecil tercatat sebesar 31.76 pada 200 *epoch*, menandakan akurasi prediksi yang lebih tinggi dibandingkan emiten lainnya. Sementara itu, pada emiten PT Vale Indonesia Tbk (INCO), nilai RMSE terbesar tercatat sebesar 139,67 pada 50 *epoch*, yang menunjukkan

bahwa prediksi untuk emiten ini kurang akurat pada variasi *epoch* yang lebih rendah. Penelitian oleh Rosyd, Purnamasari dan Ali (2024) menggunakan data harga saham PT Bank Central Asia (BBCA.JK). Data historis yang digunakan mencakup periode dari tanggal 1 Januari 2020 dan tanggal akhir 30 Oktober 2023, dan diperoleh dari dataset publik yang tersedia di website Yahoo Finance. Penelitian berfokus pada variabel *low*, yaitu harga terendah harian saham. Hasil penelitian membuktikan bahwa metode LSTM terbukti efektif dalam menganalisis pergerakan harga saham BBKA. Model prediksi menunjukkan akurasi yang tinggi, dengan nilai RMSE sebesar 40.85, MAPE sebesar 0.71%, dan MSE sebesar 6662.76.

Model *deep learning* mampu mengidentifikasi pola kompleks dalam data dan memiliki kemampuan luar biasa untuk mengekstrak informasi relevan dari deret waktu keuangan. Hal ini menjadikannya alat yang sangat efektif dalam analisis dan prediksi pasar harga saham. Dalam konteks model *deep learning*, model *Long Short-Term Memory* (LSTM) telah menunjukkan kesuksesan signifikan berkat kemampuannya untuk mempertahankan memori dari peristiwa terkini dan membangun hubungan antara data dalam urutan waktu. LSTM dilengkapi dengan tiga gerbang yang dirancang untuk mengatasi masalah yang muncul dalam metode tradisional, serta dapat memproses baik titik data tunggal maupun keseluruhan urutan data, sehingga dapat meningkatkan akurasi dalam prediksi harga pasar saham (Nabipour et al., 2020).

Metode LSTM diperkenalkan oleh Hochreiter dan Schmidhuber pada tahun 1997 sebagai bentuk khusus RNN (Recurrent neural network). LSTM dirancang untuk memecahkan masalah ketergantungan jangka panjang. Memori jangka panjang dan jangka pendek dapat dengan mudah menyimpan informasi dalam jangka waktu yang lama karena memiliki struktur jaringan berulang yang unik. LSTM memiliki tiga jenis gerbang utama yaitu *forget gate*, *input gate*, dan *output gate* yang masing-masing memiliki fungsinya tersendiri dalam mengelola informasi status sel (Akbar, Santoso, & Warsito, 2023).

Penggunaan metode Long Short-Term Memory (LSTM) dapat dimanfaatkan oleh investor untuk meningkatkan peluang meraih keuntungan serta mengurangi risiko kerugian dalam berinvestasi saham. Melalui analisis prediktif, LSTM mampu menyelami pola-pola rumit yang terdapat dalam kumpulan data besar dan menyimpan informasi sepanjang periode waktu yang panjang. Selain itu, LSTM dapat membantu para investor dengan menyediakan rekomendasi berbasis data yang dapat mendukung pengambilan keputusan yang lebih baik dalam investasi saham. Berdasarkan uraian tersebut, tujuan dari penelitian ini adalah untuk mengembangkan model prediktif berbasis LSTM untuk memprediksi harga saham PT Astra International Tbk. Penelitian ini diharapkan dapat memberikan manfaat yang signifikan bagi investor dan praktik investasi yang berbasis data, serta berkontribusi terhadap perkembangan ilmu pengetahuan di bidang keuangan dan analitik data, dengan menawarkan pendekatan yang terintegrasi antara teknologi *deep learning* dan investasi saham. Hasil dari penelitian ini dapat dijadikan referensi bagi penelitian selanjutnya yang mengeksplorasi penggunaan teknologi serupa dengan konteks yang berbeda, serta memberikan wawasan bagi praktisi di industri keuangan untuk mengadopsi metode analisis efektif dalam strategi investasi.

KERANGKA TEORI

Saham merupakan suatu bukti bahwa seseorang atau entitas memiliki sebagian dari suatu perusahaan yang artinya jika seseorang tersebut memiliki beberapa saham

terhadap sebuah perusahaan maka orang tersebut memiliki sebagian atas kepemilikan perusahaan tersebut (Permana, Hartoko, & Dewi, 2023). Persentase saham yang dimiliki seseorang terhadap sebuah perusahaan menentukan besar kecilnya kepemilikan perusahaan tersebut. Harga saham yang ditawarkan ditentukan oleh perusahaan tersebut untuk para investor yang ingin memiliki hak kepemilikan saham (Merida & Sari, 2022). Investor dapat melakukan analisis prediktif menggunakan bantuan teknologi *deep learning* menggunakan metode LSTM.

Menurut Raup, Ridwan, Khoeriyah, Supiana, dan Zaqiah (2022), *deep learning* merupakan bagian dari kecerdasan buatan (*Artificial Intelligence*) dan *machine learning* yang mengembangkan jaringan syaraf tiruan berlapis-lapis untuk meningkatkan ketepatan dalam tugas-tugas seperti deteksi objek, pengenalan suara, dan terjemahan bahasa. Metode *deep learning* memanfaatkan jaringan saraf buatan yang terdiri dari berbagai lapisan, dirancang mirip dengan otak manusia dengan neuron-neuron saling terhubung untuk membentuk jaringan yang sangat rumit. Metode ini, yang sering disebut sebagai pembelajaran terstruktur mendalam, pembelajaran hierarkis, atau jaringan saraf dalam mengaplikasikan transformasi non-linier secara berlapis dan dapat dianggap sebagai perpaduan antara *machine learning* dan kecerdasan buatan dengan menggunakan jaringan saraf buatan.

Menurut Nurashila, Hamami, dan Kusumasari (2023), Long Short-Term Memory (LSTM) adalah pengembangan dari arsitektur Recurrent Neural Network (RNN) yang diperkenalkan oleh Hochreiter dan Schmidhuber pada tahun 1997. LSTM menawarkan sejumlah keunggulan dibandingkan RNN tradisional, berkat arsitekturnya yang mampu mengingat dan melupakan informasi dengan baik, memungkinkan output untuk diproses ulang menjadi input yang lebih efektif. Selain itu, LSTM memiliki kemampuan khusus untuk menyimpan kesalahan yang muncul selama proses pelatihan, sehingga dapat menghindari peningkatan kesalahan yang berlebihan..

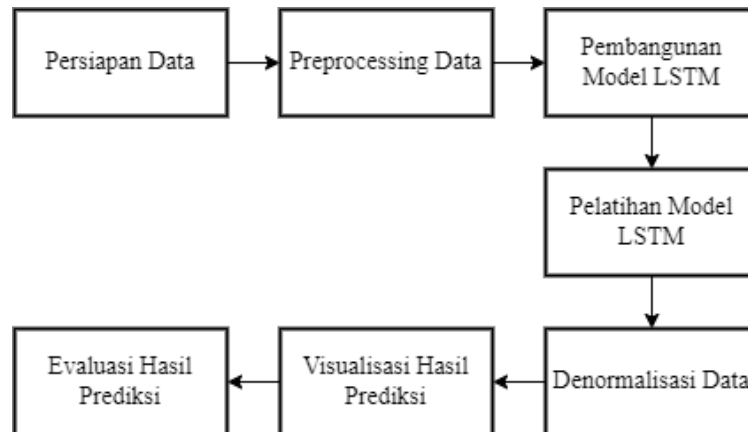
Penelitian terkait prediksi harga saham dengan teknologi *deep learning* telah dilakukan oleh peneliti terdahulu. Penelitian dengan melakukan prediksi harga saham syariah menggunakan metode LSTM dengan tujuan untuk membantu investor menentukan waktu jual dan beli. Hasil penelitian tersebut yaitu performa terbaik model LSTM yang dihasilkan dengan nilai MAPE mencapai 1,51 (Budiprasetyo et al., 2023). Adi dan Sudianto (2022) melakukan prediksi harga komoditas pangan menggunakan metode LSTM, hasil penelitian tersebut yaitu performa model menunjukkan nilai RMSE terendah mencapai 79.19%. Penelitian oleh Rizkilloh dan Widiyanesti (2022) dilakukan untuk prediksi harga cryptocurrency menggunakan metode LSTM, hasil dari penelitian tersebut yaitu performa model memiliki nilai RMSE mencapai 0.0630.

Riyantoko, Fahrudin, Hindrayani, dan Safitri (2020) melakukan analisis prediksi harga saham sektor perbankan menggunakan metode LSTM, hasil penelitian tersebut yaitu nilai RMSE terhadap prediksi harga saham bank BRI mencapai 57.31 dengan akurasi 95%. Puteri (2023) melakukan implementasi LST dan BiLSTM dalam prediksi harga saham syariah, hasil penelitian tersebut yaitu performa model terdapat pada PT Indofood Sukses Makmur Tbk dengan nilai MAPE mencapai 1.05%.

Variabel pada penelitian terkait dengan model *deep learning* untuk prediksi harga saham PT Astra International yaitu harga penutupan harga saham sesuai dengan tanggal, harga pembukaan ketika perdagangan saham dimulai, harga tertinggi dan terendah, serta volume jumlah saham yang diperdagangkan. Perumusan hipotesis pada penelitian ini yaitu model *deep learning* mampu memprediksi harga saham PT Astra International dengan akurasi lebih baik dibandingkan metode konvensional.

METODE PENELITIAN

Penelitian ini dilakukan menggunakan Google Collabolatory dengan dataset data historis sahama PT Astra International Tbk dengan ticker 'ASII.JK'. Metode penelitian terdiri dari beberapa tahap yaitu persiapan data, *preprocessing* data, pembagian data, pembangunan model, Pelatihan model, visualisasi hasil prediksi. Metode penelitian terdapat pada Gambar 1.



Gambar 1. Metode Penelitian

Persiapan Data

Data untuk penelitian ini merupakan data historis saham PT Astra International Tbk dengan kode saham "ASII.JK", yang diperoleh langsung dari website yahoo finance dengan cara memakai salah satu library Python 'yfinance'. Data ini mencakup harga saham harian dari tanggal 1 Januari 2010 hingga tanggal saat penelitian ini dibuat yaitu 22 Mei 2024. Data tersebut mencakup kolom-kolom: Date, Open, High, Low, Close, Adj Close, dan Volume, dengan total jumlah recordnya sebanyak 3549 data. Gambar 2 merupakan sampel lima data awal dan lima data akhir data perusahaan saham yang diperoleh dari yahoo finance menggunakan pustaka Python 'yfinance'.

data.head()							data.tail()						
	Open	High	Low	Close	Adj Close	Volume		Open	High	Low	Close	Adj Close	Volume
Date							Date						
2010-01-04	3530.0	3550.0	3465.0	3530.0	2001.564819	40	2024-05-15	4570.0	4620.0	4530.0	4530.0	4530.0	109962400
2010-01-05	3550.0	3570.0	3485.0	3550.0	2012.905029	40	2024-05-16	4590.0	4720.0	4540.0	4710.0	4710.0	149039700
2010-01-06	3530.0	3580.0	3515.0	3530.0	2001.564819	40	2024-05-17	4710.0	4810.0	4690.0	4800.0	4800.0	101492400
2010-01-07	3420.0	3560.0	3415.0	3420.0	1939.193481	40	2024-05-20	4800.0	4830.0	4730.0	4740.0	4740.0	58021800
2010-01-08	3440.0	3450.0	3380.0	3440.0	1950.533325	40	2024-05-21	4750.0	4770.0	4700.0	4700.0	4700.0	57558000

Gambar 2. Dataset

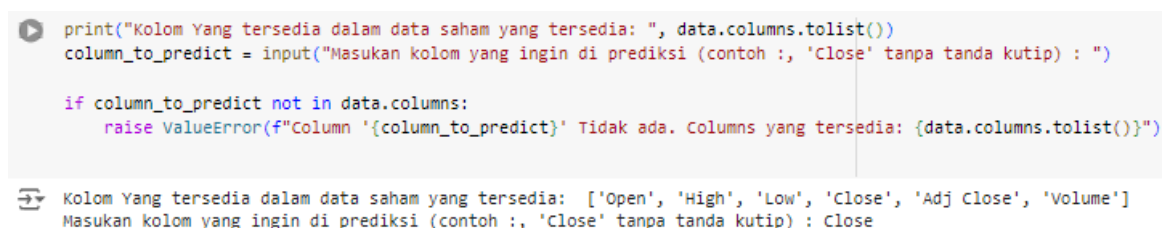
Gambar 2 terdapat kolom 'Date' yang merupakan kolom untuk menunjukan tanggal data harga saham dicatat. Kolom ini menunjukan setiap hari penjualan saham di jual. Kolom 'Open' menunjukkan harga saham saat pasar dibuka pada hari perdagangan tertentu. Kolom 'High' mencatat harga tertinggi yang ditawarkan pada hari tersebut. Kolom 'Low' mencerminkan harga terendah yang ditawarkan pada hari yang sama. Kolom 'Close' mencantumkan harga saham saat pasar ditutup pada hari tertentu. Harga adj close adalah harga penutupan yang telah disesuaikan untuk faktor-faktor tertentu,

seperti dividen, pemecahan saham, dan tindakan korporat lainnya. Volume menggambarkan total jumlah saham yang diperdagangkan pada hari tersebut.

Preprocessing Data

Data historis saham yang telah didapatkan akan melalui tahap *preprocessing* terlebih dahulu sebelum diterapkan dalam model LSTM, preprocessing data ini bertujuan agar data bisa terbaca dan diproses dengan baik pada model LSTM sehingga bisa menghasilkan prediksi yang baik. Tahapan *preprocessing* terdiri dari input kolom, normalisasi, pembagian data, dan pembuatan data sekuensial.

Langkah pertama dalam *preprocessing* data adalah menentukan kolom yang ingin diprediksi. Dalam program ini pengguna bisa memilih kolom yang ingin diprediksi dengan beberapa pilihan kolom yaitu Date, Open, High, Low, Close, Adj Close, dan Volume. Gambar 3 menunjukkan tahapan preprocessing untuk input kolom.



```
print("Kolom Yang tersedia dalam data saham yang tersedia: ", data.columns.tolist())
column_to_predict = input("Masukan kolom yang ingin di prediksi (contoh :, 'Close' tanpa tanda kutip) : ")

if column_to_predict not in data.columns:
    raise ValueError(f"Column '{column_to_predict}' Tidak ada. Columns yang tersedia: {data.columns.tolist()}")
```

Kolom Yang tersedia dalam data saham yang tersedia: ['Open', 'High', 'Low', 'Close', 'Adj Close', 'Volume']
Masukan kolom yang ingin di prediksi (contoh :, 'Close' tanpa tanda kutip) : Close

Gambar 3. Input Kolom

Langkah kedua tahapan dalam *preprocessing* yaitu normalisasi data yang digunakan untuk mengubah skala asli dari data menjadi rentang tertentu. Setelah memilih kolom 'Close' kemudian dilanjutkan ke tahap normalisasi data menggunakan teknik Min-Max Scaling dengan menggunakan fungsi 'MinMaxScaler' dari pustaka Python yang bernama 'sklearn.preprocessing'. Teknik Min-Max Scaling dapat mengubah skala data menjadi rentang antara 0 dan 1. Teknik MinMaxScaler juga berfungsi untuk memastikan bahwa semua nilai data saham berada dalam rentang yang sama, sehingga model LSTM dapat lebih mudah melakukan perhitungan dan mengurangi kemungkinan adanya bias akibat perbedaan antara skala dan fitur. Gambar 4 menunjukkan tahapan *preprocessing* untuk normalisasi.

Data Asli:	
Date	Close
2010-01-04	3530.0
2010-01-05	3550.0
2010-01-06	3530.0
2010-01-07	3420.0
2010-01-08	3440.0
Data Dinormalisasi:	
Date	Close_Scaled
2010-01-04	0.042589
2010-01-05	0.045997
2010-01-06	0.042589
2010-01-07	0.023850
2010-01-08	0.027257

Gambar 4. Normalisasi

Langkah ketiga tahapan dalam *preprocessing* adalah pembagian dataset menjadi data latih (train data) dan data uji (test data). Data latih berfungsi untuk pelatihan algoritma model LSTM, sedangkan data uji berfungsi untuk pengujian dan mengukur

performa model yang didapatkan setelah melakukan pengujian (testing). Data yang digunakan dalam penelitian ini adalah kolom 'Close'. Penelitian ini membagi data latih dan data uji dengan proposi perbandingan 80:20. Data latih sebanyak 2839 dan data uji sebanyak 710 data. Gambar 5 merupakan tahapan *preprocessing* data untuk pembagian dataset.

```
[ ] training_size = int(len(data_scaled)*0.80)
    test_size=len(data_scaled)-training_size
    train_data,test_data=data_scaled[0:training_size,:],data_scaled[training_size:len(data_scaled),:]
    training_size,test_size
```

(2839, 710)

Gambar 5. Pembagian Data Latih dan Data Uji

Langkah keempat tahapan dalam *preprocessing* yaitu pembuatan data sekuensial untuk proses dimana data diatur dalam urutan waktu yang digunakan dalam model prediksi. Pada tahapan pembuatan data sekuensial dilakukan pengolahan data pelatihan (*train_data*) dan data pengujian (*test_data*) yang sudah di normalisasi data untuk menghasilkan input yang akan dipakai dalam model LSTM. Gambar 6 merupakan tahapan *preprocessing* data untuk pembuatan data sekuensial.

```
[ ] time_step = 90
    X_train, y_train = [], []
    for i in range(len(train_data) - time_step - 1):
        a = train_data[i:(i + time_step), 0]
        X_train.append(a)
        y_train.append(train_data[i + time_step, 0])

    X_test, y_test = [], []
    for i in range(len(test_data) - time_step - 1):
        a = test_data[i:(i + time_step), 0]
        X_test.append(a)
        y_test.append(test_data[i + time_step, 0])

    X_train = np.array(X_train).reshape(-1, time_step, 1)
    X_test = np.array(X_test).reshape(-1, time_step, 1)
    y_train = np.array(y_train)
    y_test = np.array(y_test)
```

Gambar 6. Pembuatan Data Sekuensial

Pembangunan Model

Lapisan LSTM dalam penelitian ini terdiri dari empat lapisan. Lapisan pertama LSTM memiliki 128 unit sel memori (sel neuron) dan mengembalikan urutan keluaran (*return_sequence = True*) untuk digunakan oleh lapisan LSTM berikutnya. Lapisan pertama ini menerima inputan dengan bentuk '*X_train.shape[1],1*' sebagai data latih dengan jumlah timestep dan '1' merupakan jumlah fitur per timestep. Aktivasi yang digunakan adalah 'tanh'. Terdapat lapisan dropout sebesar 40% (0.4) untuk mencegah overfitting. Lapisan kedua LSTM juga memiliki 128 unit sel memori (sel neuron) dan mengembalikan urutan keluaran. Aktivasi yang digunakan ialah 'tanh'. Terdapat lapisan dropout dengan rasio 40% (0.4). Lapisan ketiga LSTM memiliki 64 unit sel memori (sel neuron) dan mengembalikan urutan keluaran. Aktivasi yang digunakan adalah 'tanh'. Terdapat lapisan dropout dengan rasio 40% (0.4). Lapisan keempat LSTM memiliki 64 unit sel memori (sel neuron) dan tidak mengembalikann urutan keluaran (*'return_sequence = False'*). Aktivitas yang digunakan 'tanh'. Kemudian terdapat lapisan dense satu unit. Lapisan ini akan menghasilkan nilai prediksi akhir dari model. Gambar 7 merupakan model LSTM yang dibangun.

Model: "sequential"

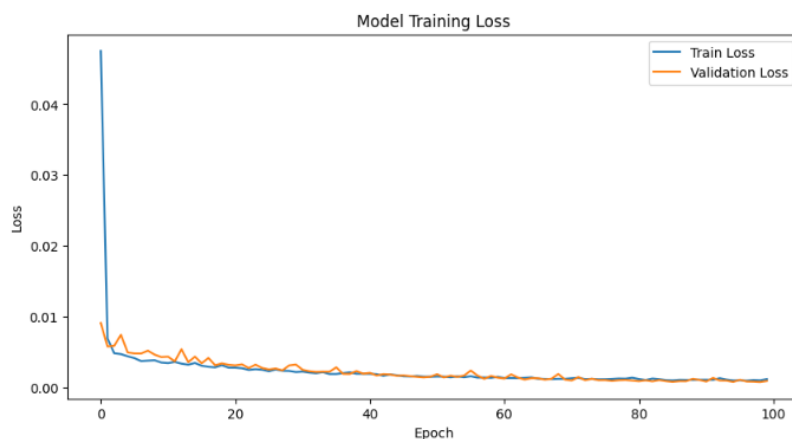
Layer (type)	Output Shape	Param #
lstm (LSTM)	(None, 180, 128)	66560
dropout (Dropout)	(None, 180, 128)	0
lstm_1 (LSTM)	(None, 180, 128)	131584
dropout_1 (Dropout)	(None, 180, 128)	0
lstm_2 (LSTM)	(None, 180, 64)	49408
dropout_2 (Dropout)	(None, 180, 64)	0
lstm_3 (LSTM)	(None, 64)	33024
dense (Dense)	(None, 1)	65

=====
Total params: 280641 (1.07 MB)
Trainable params: 280641 (1.07 MB)
Non-trainable params: 0 (0.00 Byte)

Gambar 7. Arsitektur Model LSTM

Pelatihan Model

Pelatihan model ini menggunakan data pelatihan ('X_train') dan ('y_train'). Model akan dilatih selama 100 *epoch* dengan ukuran batch sebesar 64. Data pelatihan dibagi menjadi dua bagian yaitu 80% untuk pelatihan dan 20% validasi. Data validasi digunakan untuk mengevaluasi kinerja model pada data yang tidak dapat dilihat sehingga membantu mendeteksi *overfitting*. Gambar 8 merupakan visualisasi hasil dari pelatihan model.



Gambar 8. Visualisasi Hasil Pelatihan Model

Gambar 8 menunjukkan bahwa pada awal pelatihan menunjukkan nilai *loss* untuk data latih cukup tinggi, tetapi seiring berjalannya pelatihan dan *epoch* terlihat nilai *loss* untuk data latih semakin berkurang secara stabil. Sementara untuk nilai data validasi terlihat bahwa lebih stabil yang menandakan bahwa model LSTM yang dibangun memiliki kemampuan mempelajari yang baik. Visualisasi hasil pelatihan model menunjukkan nilai *loss* yang berkurang seiring dengan bertambahnya jumlah *epoch*.

Denormalisasi Data

Denormalisasi data merupakan tahap mengembalikan data yang sudah di normalisasikan ke skala aslinya. Denormalisasi data bertujuan agar hasil prediksi data yang telah dilakukan oleh model LSTM dapat dipahami. Dalam penelitian ini, denormalisasi data menggunakan fungsi 'inverse_transform' yang berasal dari teknik

‘MinMaxScaler’ dari pustaka Python yang bernama ‘sklearn.preprocessing’. Fungsi ‘inverse_transform’ dapat mengubah skala data normalisasi yang telah diprediksi menjadi skala asli data. Gambar 9 menunjukkan hasil dari denormalisasi data.

```
print(y_test_denorm[:5])
```

```
[[6050.]
 [6150.]
 [6000.]
 [6000.]
 [6025.]]
```

```
[ ] print(y_pred_denorm[:5])
print()
```

```
[[6129.795 ]
 [6062.288 ]
 [6092.0396]
 [6031.1416]
 [5996.4595]]
```

Gambar 9. Denormalisasi Data

HASIL DAN PEMBAHASAN

Pengujian model dan visualisasi data dilakukan pada model yang telah dilatih. Pengujian model dilakukan untuk melihat hasil pada kolom ‘Close’. Prediksi dilakukan pada data close prediksi terhadap data close asli. Berikut pengujian model terdapat pada Gambar 10.

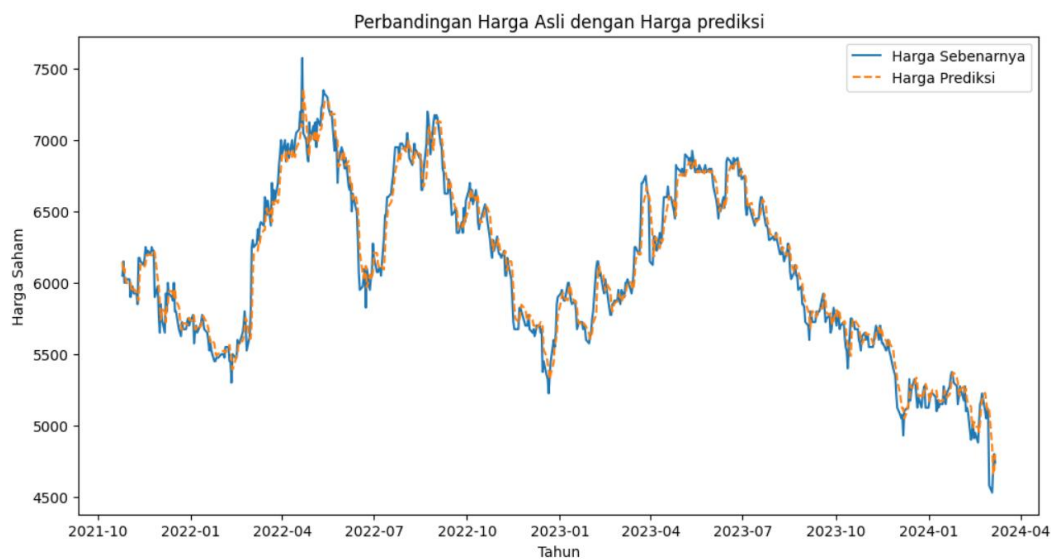
	Tahun	Data_Close_Asli	Data_Close_Prediksi
0	2021-10-25	6050.0	6129.794922
1	2021-10-26	6150.0	6062.288086
2	2021-10-27	6000.0	6092.039551
3	2021-10-28	6000.0	6031.141602
4	2021-10-29	6025.0	5996.459473

Gambar 10. Hasil Pengujian Model

Gambar 10 menunjukkan bahwa model LSTM telah mampu memprediksi harga kolom ‘Close’ dengan cukup baik. Terdapat perbedaan nilai yang tidak terlalu signifikan antara harga asli dengan harga prediksi, ini menunjukkan bahwa model telah berhasil menangkap pola dalam data kolom ‘Close’ dan mampu melakukan prediksi. Penelitian ini sejalan dengan penelitian oleh Budiprasetyo, et al. (2023) dan Patriya, Latif, dan Handayani. (2023), yang membuktikan bahwa LSTM mampu melakukan prediksi harga saham dengan cukup baik.

Visualisasi Data

Visualisasi data dilakukan untuk melihat hasil dari prediksi yang dilakukan. Visualisasi ini bertujuan untuk melihat bagaimana perbandingan harga asli dengan harga prediksi dengan menggunakan model LSTM yang sudah dilatih. Visualisasi ini menggunakan pustaka Python yaitu ‘Matplotlib.pyplot’ untuk membuat plot perbandingan dari harga asli dengan harga prediksi. Gambar 11 menunjukkan visualisasi perbandingan harga asli dan harga prediksi.



Gambar 11. Visualisasi Perbandingan antara Harga Asli dengan Prediksi

Pada Gambar 11 terlihat hasil akhir dari perbandingan harga asli dengan harga prediksi periode bulan Oktober 2021 hingga bulan April 2024 yang dilakukan dengan model LSTM menunjukkan bahwa model mampu memprediksi dengan cukup akurat dan mampu menangkap pola tren.

Evaluasi Hasil Prediksi

Tahap evaluasi hasil prediksi menggunakan metode *Mean Absolute Error (MAE)*, *Root Mean Squared Error (RMSE)*, dan *Mean Absolute Percentage Error (MAPE)*. MAE (*Mean Absolute Error*) merupakan perhitungan untuk mengukur rata-rata kesalahan antara nilai asli dari data dengan nilai prediksi yang telah dilakukan. Gambar 12 menunjukkan rata-rata kesalahan absolut antara nilai prediksi dengan nilai sebenarnya sebesar 87.69. Untuk nilai persentase *Mean Absolute Error* sebesar 1.44% yang menunjukkan bahwa model prediksi LSTM memiliki tingkat akurasi yang cukup baik.

```

mae = np.mean(np.abs(y_test_denorm - y_pred_denorm))
print("Mean Absolute Error (MAE): {:.2f}".format(mae))
# Persentase MAE
mae_percentage = (mae / mean_y_test) * 100
print("Persentase MAE: {:.2f}%".format(mae_percentage))

```

Mean Absolute Error (MAE): 87.69
 Persentase MAE: 1.44%

Gambar 12. Evaluasi Nilai MAE

Root Mean Squared Error (RMSE) merupakan perhitungan untuk mengukur seberapa besar kesalahan prediksi model dengan nilai aslinya. *Root Mean Squared Error* memberikan penilaian yang lebih besar dibandingkan dengan Mean Absolute Error, karena kesalahan yang dihitung dengan kuadrat dari selisih antara nilai prediksi dan nilai asli sebelum diambil rata-ratanya. Gambar 12 menunjukkan bahwa nilai RMSE sebesar 116.87 menunjukkan rata-rata kesalahan prediksi model dengan nilai aslinya. Nilai RMSE ini menunjukkan prediksi yang dihasilkan oleh model LSTM yang dibuat memiliki kesalahan rata-rata sebesar 116.87 dari nilai sebenarnya, sedangkan persentase untuk

RMSE sebesar 1.92 % yang menunjukkan kesalahan prediksi model sangat kecil dibandingkan dengan nilai aslinya. Nilai RMSE ini menunjukkan bahwa model LSTM memiliki kinerja yang cukup baik dalam memprediksi data yang diuji.

```
[ ] # Root Mean Squared Error (RMSE)
rmse = np.sqrt(np.mean((y_test_denorm - y_pred_denorm) ** 2))
print("Root Mean Squared Error (RMSE): {:.2f}".format(rmse))
rmse_percentage = (rmse / mean_y_test) * 100
print("Persentase RMSE: {:.2f}%".format(rmse_percentage))
```

➤ Root Mean Squared Error (RMSE): 116.87
Persentase RMSE: 1.92%

Gambar 13. Evaluasi Nilai RMSE

Mean Absolute Percentage Error (MAPE) merupakan perhitungan untuk mengukur akurasi model prediksi dalam bentuk persentase. MAPE memberikan rata-rata kesalahan absolut antara nilai prediksi dan nilai sebenarnya yang dinyatakan sebagai persentase dari nilai sebenarnya. Gambar 14 menunjukkan nilai Mean Absolute Percentage Error (MAPE) sebesar 1.45% menunjukkan rata-rata kesalahan absolut prediksi model LSTM adalah 1.45% dari nilai sebenarnya. Nilai MAPE ini menjelaskan bahwa model LSTM sangat akurat dalam memprediksi data yang telah diuji.

```
[ ] # Mean Absolute Percentage Error (MAPE)
mape = np.mean(np.abs((y_test_denorm - y_pred_denorm) / y_test_denorm)) * 100
print("Mean Absolute Percentage Error (MAPE): {:.2f}%".format(mape))
```

➤ Mean Absolute Percentage Error (MAPE): 1.45%

Gambar 14. Evaluasi Nilai MAPE

SIMPULAN DAN SARAN

Model *deep learning* menggunakan LSTM berhasil melakukan prediksi harga saham pada PT Astra International Tbk yang dibuktikan dengan visualisasi data yang menunjukkan pola tren prediksi dapat diprediksi dengan cukup akurat. Evaluasi performa model menggunakan nilai *Mean Absolute Error* (MAE) sebesar 87.69 dengan persentase 1.44% yang menunjukkan rata-rata kesalahan prediksi model sangat kecil. Nilai *Root Mean Squared Error* (RMSE) didapatkan nilai sebesar 116.87 dengan persentase RMSE sebesar 1.92% yang menunjukkan bahwa kesalahan prediksi relatif sangat kecil. Sementara itu untuk *Mean Absolute Percentage Error* (MAPE) nilai yang diperoleh adalah 1.45% yang menunjukkan model prediksi sangat akurat.

Prediksi harga saham menggunakan Model LSTM memiliki beberapa kekurangan yang diantaranya, menambah jumlah data dari kolom sahamnya agar model lebih banyak belajar dari informasi historisnya. Dengan menambah lebih banyak data, model akan memiliki kesempatan untuk menangkap pola dan tren yang lebih kompleks, sehingga memiliki kesempatan untuk mendapatkan akurasi prediksi yang lebih maksimal. Selain itu, juga bisa mempertimbangkan untuk menggunakan ticker kode perusahaan lain untuk memperluas kemampuan belajar model LSTM. Selain menggunakan model LSTM, disarankan untuk mencoba model lain seperti GRU (*Gated Recurrent Unit*), *Prophet*, *ARIMA* (*AutoRegressive Integrated Moving Average*), untuk mengevaluasi apakah ada peningkatan performa dalam akurasi prediksi.

DAFTAR PUSTAKA

- Adi, R. M. S., & Sudianto, S. (2022). Prediksi harga komoditas pangan menggunakan algoritma long short-term memory (LSTM). *Building of Informatics, Technology and Science (BITS)*, 4(2), 1137–1145. doi.org:10.47065/bits.v4i2.2229
- Akbar, R., Santoso, R., & Warsito, B. (2023). Prediksi tingkat temperatur Kota Semarang menggunakan metode long short-term memory (LSTM). *Jurnal Gaussian*, 11(4), 572–579. doi.org:10.14710/j.gauss.11.4.572-579
- Budiprasetyo, G., Hani'ah, M., & Aflah, D. Z. (2023). Prediksi harga saham syariah menggunakan algoritma long short-term memory (LSTM). *Jurnal Nasional Teknologi Dan Sistem Informasi*, 8(3), 164–172. doi.org:10.25077/teknosi.v8i3.2022.164-172
- Herlin, F., Veronica, D., Rosita, R., & Neldawaty, R. (2022). Determinan pertumbuhan laba industri otomotif di Indonesia (studi empiris pada PT Astra Internasional Tbk.). *Paradigma Ekonomika*, 17(1), 1-15.
- Irahadi, D.R, Sianturi, M.S., & Kim, S. S. (2022). Penggunaan indikator analisa teknikal pada pasar saham di Indonesia. *Jurnal Ilmiah Manajemen Bisnis Dan Inovasi*, 9(2), 808–827.
- Julian, R., & Pribadi, M. R. (2021). Peramalan harga saham pertambangan pada Bursa Efek Indonesia (BEI) menggunakan long short term memory (LSTM). *Jurnal Teknik Informatika dan Sistem Informasi*, 8(3), 1596–1606. doi.org:10.35957/jatisi.v8i3.1159
- Merida, & Sari, P.A. (2022). Pengaruh profitabilitas dan likuiditas terhadap harga saham dengan dividen per share sebagai variabel moderating. *Journal Intelektual*, 1(2), 217–230. doi.org:10.61635/jin.v1i2.106
- Nabipour, M., Nayyeri, P., Jabani, H., Mosavi, A., Salwana, E., & Shahab, S. (2020). Deep learning for stock market prediction. *Entropy*, 22(8), 1-23. doi.org:10.3390/e22080840
- Rizkillloh, M.F. & Widiyanesti, S. (2022). Prediksi harga cryptocurrency menggunakan algoritma long short-term memory (LSTM). *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 6(1), 25–31. doi.org:10.29207/resti.v6i1.3630
- Nurashila, S. S., Hamami, F., & Kusumasari, T. F. (2023). Perbandingan kinerja algoritma recurrent neural network (RNN) dan long short-term memory (LSTM): Studi kasus prediksi kemacetan lalu lintas jaringan PT XYZ. *Jurnal Ilmiah Penelitian dan Pembelajaran Informatika (JIPI)*, 8(3), 864-877.
- Patriya, E., Latif, A., & Handayani. (2023). Peramalan harga saham penutupan indeks harga saham gabungan (IHSG) menggunakan algoritma long short term memory (LSTM). *Jurnal Ilmiah Ekonomi Bisnis*, 28(2), 304–314. doi.org:10.35760/eb.2023.v28i2.7964
- Permana, D., Hartoko, S., & Dewi, I. R. (2023). Faktor-faktor yang memengaruhi kesediaan berinvestasi saham (studi kasus pada Galeri Bursa Efek Indonesia Politeknik Negeri Samarinda dan Universitas Muhammadiyah Kalimantan Timur). *Jurnal EKSIS*, 19(2), 26–38. doi.org:10.46964/eksis.v19i2.552
- Pradnyawati, N. L. P., & Sinarwati, N. (2022). Analisis keputusan investasi pada generasi milenial di pasar modal saat pandemi Covid-19. *Jurnal Manajemen*, 8(2), 1–10. Retrieved from: <https://ejournal.undiksha.ac.id/index.php/BISMA-JM/article/view/42838>
- Puteri, D. I. (2023). 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*, 11(1), 35–43. doi.org:10.34312/euler.v11i1.19791
- Ramadhan, P. R. (2020). Analisis determinan harga saham perusahaan sektor agriculture yang terdaftar di Bursa Efek Indonesia. *Jurnal Riset Akuntansi Dan Bisnis*, 20(2), 2623–2650. doi.org/10.30596/jrab.v20i2.5569
- Raup, A., Ridwan, W., Khoeriyah, Y., Supiana, & Zaqiah, Q. Y. (2022). Deep learning dan penerapannya dalam pembelajaran. *Jurnal Ilmiah Ilmu Pendidikan (JIIP)*, 5(9), 3258-3267.
- Riyantoko, A.P., Fahrudin, T.M., Hindrayani, K.M., & Safitri, E.M. (2020). Analisis prediksi harga saham sektor perbankan menggunakan algoritma long-short term memory (LSTM). *Prosiding Seminar Nasional Informatika (SEMNASIF)*, 1(1), 427–435.
- Rosyd, A., Purnamasari, A. I., & Ali, I. (2024). Penerapan metode long short term memory (LSTM) dalam memprediksi harga saham PT Bank Central Asia. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(1), 501.