

$$X = 5 \times (7 \times 20)$$

$$X = 700$$

$$Y = \text{skor terendah} \times (\text{jumlah pertanyaan} \times \text{jumlah responden})$$

$$Y = 1 \times (7 \times 20)$$

$$Y = 140$$

Maka didapatkan perhitungan persentase *UAT* sebagai berikut:

$$\text{Persentase UAT} = \frac{\text{Total Skor}}{X} \times 100\%$$

$$= \frac{586}{700} \times 100\% = 83,71\%$$

Berdasarkan hasil persentase *UAT* didapatkan sebesar 83,71%. Hal ini menunjukkan bahwa sistem prediksi harga emas termasuk dalam kategori “Sangat Layak” karena berada dalam *range* 81% – 100%.

BAB V

SIMPULAN DAN SARAN

5.1 Kesimpulan

Penelitian ini telah berhasil membangun dan mengimplementasikan sistem prediksi harga emas berbasis web menggunakan metode *Long Short-Term Memory* (LSTM), yang terbukti mampu memprediksi harga emas berdasarkan data historis secara akurat dan efisien. Beberapa kesimpulan utama dari penelitian ini antara lain:

1. Mengevaluasi kinerja model LSTM menggunakan metrik akurasi seperti MSE dan RMSE

Evaluasi dilakukan dengan menghitung nilai *Mean Squared Error* (MSE) dan *Root Mean Squared Error* (RMSE) dari hasil prediksi harga emas oleh model LSTM dibandingkan dengan data aktual. Nilai MSE dan RMSE yang rendah menunjukkan bahwa model LSTM memiliki tingkat kesalahan prediksi yang kecil. Berdasarkan pengujian yang dilakukan pada data uji, model LSTM menunjukkan performa yang baik dalam menangkap pola fluktuasi harga emas, yang tercermin dari nilai MSE dan RMSE yang berada dalam rentang toleransi yang dapat diterima dalam konteks prediksi harga komoditas. Model LSTM memberikan akurasi yang cukup baik dengan nilai MAPE sebesar 4.63% berarti tingkat akurasi sebesar 95.37%

2. Penerimaan Pengguna (UAT)

Berdasarkan pengujian *User Acceptance Test* (UAT) terhadap 20 responden, sistem memperoleh skor kelayakan sebesar 83,71%, yang masuk dalam kategori “Sangat Layak”. Hal ini menunjukkan bahwa sistem dapat

digunakan oleh pengguna non-teknis secara efektif dalam mengambil keputusan investasi.

3. Mengembangkan antarmuka sistem berbasis web

Antarmuka sistem dirancang dengan memperhatikan prinsip user-friendly dan responsive agar dapat digunakan secara luas oleh berbagai jenis perangkat, termasuk desktop dan mobile. Fitur-fitur utama seperti input data, grafik hasil prediksi, dan informasi historis harga emas ditampilkan secara interaktif. Pengujian fungsionalitas dilakukan untuk memastikan bahwa setiap komponen sistem berjalan sesuai dengan spesifikasi yang telah ditentukan, termasuk pemrosesan data prediksi secara real-time yang langsung ditampilkan ke pengguna setelah proses input atau permintaan prediksi dilakukan.

5.2 Saran

Berdasarkan hasil penelitian dan kesimpulan yang telah diperoleh, penulis memberikan beberapa saran sebagai berikut:

1. Pengayaan Variabel Model

Untuk meningkatkan akurasi dan ketepatan prediksi harga emas, disarankan agar pengembangan sistem selanjutnya mempertimbangkan penambahan variabel eksternal, seperti tingkat inflasi, nilai tukar mata uang asing, dan kondisi geopolitik global yang diketahui memiliki pengaruh signifikan terhadap fluktuasi harga emas.

2. Penggunaan Model Kombinasi (Hybrid)

Metode LSTM telah terbukti efektif, namun sistem dapat lebih ditingkatkan dengan menggabungkannya dengan metode lain seperti Support Vector Regression (SVR), ARIMA, atau Random Forest.

Pendekatan *hybrid* ini dapat menghasilkan prediksi yang lebih akurat dengan memperkuat keunggulan masing-masing algoritma.

3. Perluasan Dataset Historis

Disarankan untuk menggunakan data historis yang lebih panjang dan beragam agar model mampu mengenali lebih banyak pola jangka panjang serta beradaptasi terhadap kondisi pasar yang dinamis.

4. Pengembangan Antarmuka dan Fitur Tambahan

Untuk meningkatkan kenyamanan dan kegunaan sistem bagi pengguna, dapat dikembangkan antarmuka pengguna (UI) yang lebih interaktif serta penambahan fitur seperti notifikasi tren harga, integrasi data real-time melalui API, dan rekomendasi prediksi berdasarkan preferensi pengguna.

5. Penerapan pada Komoditas Lain

Metodologi dan sistem yang dibangun dalam penelitian ini juga berpotensi diterapkan pada prediksi harga komoditas lain, seperti minyak bumi, perak, atau mata uang kripto. Oleh karena itu, penelitian lanjutan disarankan untuk mengeksplorasi penerapan sistem serupa pada komoditas berbeda guna memperluas manfaatnya dalam dunia investasi.

DAFTAR PUSTAKA

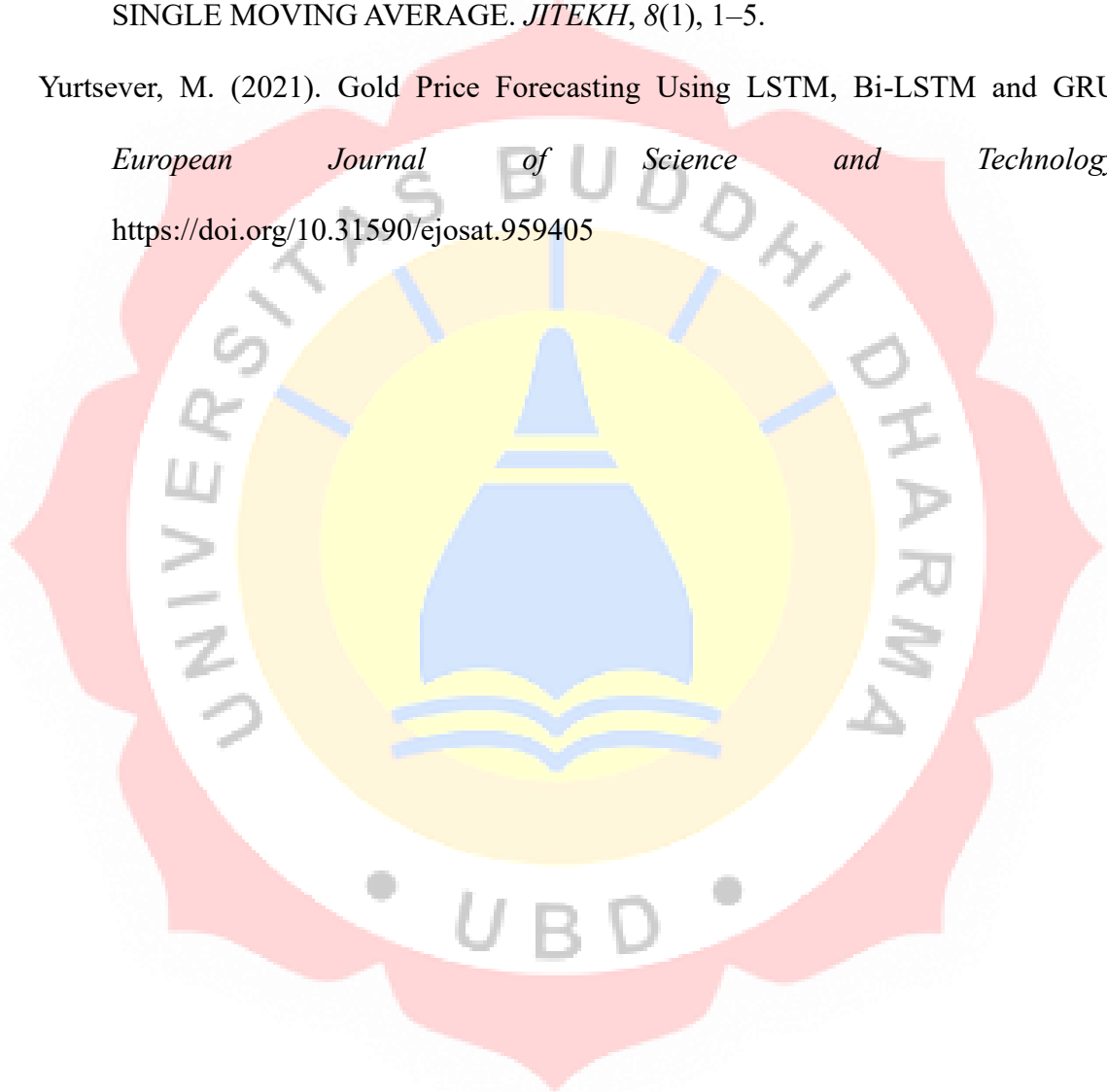
- Alfani, A., Rozi, F., & Sukmana, F. (2021). PREDIKSI PENJUALAN PRODUK UNILEVER MENGGUNAKAN METODE K-NEAREST NEIGHBOR. *JIPI (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika)*, 6(1).
- Al-Husaini, M., Rahman, R. M., Natsir, M. F., & Abdussalam, F. (2024). *Prediksi Harga Saham Menggunakan Algoritma Long short-Term Memory (LSTM) Recurrent Neural Networks (RNN)* (Efitra & W. Gustiani, Eds.). PT. Sonpedia Publishing Indonesia.
- Amryliana, S., Bahri, S., & Alfian, M. R. (2025). Optimization of Long Short Term Memory Model for Gold Price Prediction Using Adaptive Moment Estimation. *Jurnal Matematika, Statistika Dan Komputasi*, 21(3), 669–683. <https://doi.org/10.20956/j.v21i3.42872>
- Azis, N., Pribadi, G., & Nurcahya, M. S. (2020). Analisa dan Perancangan Aplikasi Pembelajaran Bahasa Inggris Dasar Berbasis Android. *Jurnal IKRA-ITH Informatika*, 3(3).
- Aziz, Z. A., Abdulqader, D. N., Sallow, A. B., & Omer, H. K. (2021). Python Parallel Processing and Multiprocessing: A Review . *Academic Journal of Nawroz University (AJNU)*, 10(3).
- Azizah, S., Febriadi, S. R., & Srisusilawati, P. (2020). Analisis Maqashid Syariah tentang Menjaga Harta terhadap Penangguhan Penyerahan Jaminan Logam Mulia Kolektif. *Prosiding Hukum Ekonomi Syariah*, 53. <https://doi.org/10.29313/syariah.v0i0.19380>

- Bhandari, H. N., Rimal, B., Pokhrel, N. R., Rimal, R., Dahal, K. R., & Khatri, R. K. C. (2022). Predicting stock market index using LSTM. *Machine Learning with Applications*, 9, 100320. <https://doi.org/10.1016/j.mlwa.2022.100320>
- Bimanjaya, A., Handayani, H. H., & Rachmadi, R. F. (2023). Penggunaan Deep Learning dan Post-Processing Algoritma Douglas-Peucker untuk Ekstraksi Jaringan Jalan pada Area Urban dari Orthophoto. *IOP Conference Series : Earth & Environmental Science*, 1127(1).
- Budiman, I., Saori, S., Anwar, R. N., Fitriani, & Pangestu, M. Y. (2021). ANALISIS PENGENDALIAN MUTU DI BIDANG INDUSTRI MAKANAN (Studi Kasus: UMKM Mochi Kaswari Lampion Kota Sukabumi). *Jurnal Inovasi Penelitian*, 10.
- Fitriyah Salsabilah, A., Sabili Nurilhaq, M., Dwi Wira Y, P. G., Arbi Hanafi, A., & Daffa Atillah, M. (2024). IMPLEMENTASI ALGORITMA REGRESI LINEAR UNTUK MEMPREDIKSI HARGA EMAS. *Jurnal Informatika Dan Sistem Informasi (JIFoSI)*, 5(3), 2722–130. <https://doi.org/10.33005/jifosi.v5i3.461>
- Hanifa, A., Fauzan, S. A., Hikal, M., & Ashfiya, M. B. (2021). PERBANDINGAN METODE LSTM DAN GRU (RNN) UNTUK KLASIFIKASI BERITA PALSU BERBAHASA INDONESIA. *DINAMIKA REKAYASA*, 17(1). <https://covid19.go.id/p/hoax-buster>.
- Hidayatullah, B. K., Kallista, M., Setianingsih, C., S1, P., & Komputer, T. (2022). PREDIKSI INDEKS STANDAR PENCEMAR UDARA MENGGUNAKAN METODE LONG SHORT-TERM MEMORY BERBASIS WEB (STUDI KASUS PADA KOTA JAKARTA). *E-Proceeding of Engineering*, 9(3). <https://data.jakarta.go.id/>

- Huseynov, S. (2023). *Estimating term premium with long and short memory term structure models*. <https://ssrn.com/abstract=4436976>
- Indika, A., Warusamana, N., Welikala, E., & Deegalla, S. (2021). *Ensemble Stock Market Prediction using SVM,LSTM, and Linear Regression*. <https://doi.org/10.36227/techrxiv.16626019.v1>
- Karyadi, Y., & Santoso, H. (2022). Prediksi Kualitas Udara Dengan Metoda LSTM, Bidirectional LSTM, dan GRU. *Jurnal Teknik Informatika Dan Sistem Informasi*, 9(1), 671–684.
- Kristiana, L., & Miyanto, D. (2023). Penambahan Parameter PM2.5 dalam Prediksi Kualitas Udara : Long Short Term Memory. *Multimedia Artificial Intelligent Networking Database*, 8(2), 188–202. <https://doi.org/10.26760/mindjournal.v8i2.188-202>
- Kurniansyah, J., Kurnia Gusti, S., Yanto, F., & Affandes, M. (2025). Implementasi Model Long Short Term Memory (LSTM) dalam Prediksi Harga Saham. *Bulletin of Information Technology (BIT)*, 6(2), 79–86. <https://doi.org/10.47065/bit.v5i2.1783>
- Li, S. (2024). *Evaluating the Directional-Weighted Mean Absolute Error in Long Short-Term Memory Models for Stock Price Prediction*. 171–177. <https://doi.org/10.5220/0012814700003885>
- Manjula, K. A., & Karthikeyan, P. (2019). Gold Price Prediction using Ensemble based Machine Learning Techniques. *Proceedings of the Third International Conference on Trends in Electronics and Informatics*.
- Merdiansah, R., Wulandari, K., Hasibuan, M., & Umaidah, Y. (2024). Perbandingan Kinerja Model RNN, LSTM, dan BLSTM dalam Memprediksi Jumlah Gempa

- Bulanan di Indonesia. *Jurnal Penelitian Rumpun Ilmu Teknik*, 3(1), 262–277.
<https://doi.org/10.55606/juprit.v3i1.3466>
- Mohammad, K. (2019). PENERAPAN METODE K-NEAREST NEIGHBORS UNTUK PREDIKSI PENJUALAN BERBASIS WEB PADA BOUTIQ DEALOVE BONDOWOSO. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 3(2).
- Muhammad, R., & Yulianto, S. (2023). PENERAPAN PEMROGRAMAN PYTHON DALAM MENENTUKAN WAKTU OVERHOUL KONDENSOR TURBIN UAP. *Jurnal Konversi Energi Dan Manufaktur*, 8(1).
- Pradana, F. N. S. P. M., & Papilaya, F. S. (2023). Analisa Prediksi Harga Emas Dengan Kemungkinan Terjadinya Resesi Menggunakan Metode Svr. *Sintech Journal*, 6(1).
- Rangkuti, A., & Yahfizham. (2023). Pengenalan Algoritma Pemrograman Dasar Dalam Konteks Pembelajaran Pemrograman Awal. *Konstanta : Jurnal Matematika Dan Ilmu Pengetahuan Alam*, 1(4).
- RENALDY. (2022). *APLIKASI PREDIKSI HARGA AYAM DENGAN METODE NAIVES BAYES PADA SUPPLIER AYAM POTONG*.
- Rosyd, A., Irma Purnamasari, A., & Ali, I. (2024). PENERAPAN METODE LONG SHORT TERM MEMORY (LSTM) DALAM MEMPREDIKSI HARGA SAHAM PT BANK CENTRAL ASIA. *Jurnal Mahasiswa Teknik Informatika*, 8(1).
- Safitri, C. M. S. (2023). *PENERAPAN DATA MINING UNTUK MEMPREDIKSI HARGA EMAS DI INDONESIA MENGGUNAKAN METODE LINEAR REGRESSION*. Universitas Buddhi Dharma.
- Saptadi, N. T. S., Kristiawan, H., & Nugroho, A. Y. (2025). *Deep Learning: Teori, Algoritma, dan Aplikasi* (N. Azizah, Ed.). Sada Kurnia Pustaka.

- Sari, I. P., Siska, S. T., & Budiman, A. (2021). Perancangan Aplikasi Pelayanan Gangguan TV Kabel Berbasis Web dan SMS Gateway. *Jurnal Pusat Akses Kajian Teknologi Artificial Intelligence*, 1(1).
- Sitorus, L. (2015). *Algoritma dan pemrograman*. Andi.
- Suwandi, A. (2020). PREDIKSI HARGA EMAS MENGGUNAKAN METODE SINGLE MOVING AVERAGE. *JITEKH*, 8(1), 1–5.
- Yurtsever, M. (2021). Gold Price Forecasting Using LSTM, Bi-LSTM and GRU. *European Journal of Science and Technology*.
<https://doi.org/10.31590/ejosat.959405>



DAFTAR RIWAYAT HIDUP



Data Pribadi

Nama Lengkap : Elvian Presley
NIM : 20211000028
Tempat, Tanggal Lahir : Jakarta, 24 Januari 2002
Jenis Kelamin : Laki – laki
Alamat : Poris Gaga
Agama : Kristen
No Kontak : 089506704230
eMail : elvian2480@gmail.com

Pendidikan Formal

2008-2014 : Sekolah Dasar Poris Indah
2015-2017 : Sekolah Menengah Pertama Poris Indah
2018-2020 : Sekolah Menengah Kejuruan Mutiara Bangsa 1
2021-2025 : Universitas Buddhi Dharma

Pengalaman Kerja

Magang di PT AeroTRANS, : Des 2018 - Jan 2019
Tangerang
Admin Toko Online di Star : Mar 2021 - Jun 2023
Top Jaya, Tangerang
Application Support : Mei 2024 - Present
Analyst di PT Pikkat QTA
Retail Servisindo, Jakarta

Tangerang, 05-08-2025

Elvian Presley

Lampiran 1 Kartu Bimbingan Skripsi



UNIVERSITAS BUDDHI DHARMA

Jl. Imam Bonjol No. 41 Karawaci Ilir, Tangerang

021 5517853 / 021 5586822 admin@buddhidharma.ac.id

KARTU BIMBINGAN TA/SKRIPSI

NIM : 20211000028
Nama Mahasiswa : ELVIAN PRESLEY
Fakultas : Sains dan Teknologi
Program Studi : Teknik Informatika
Jenjang : Strata Satu
Tahun Akademik/Semester : 2024/2025 Genap
Dosen Pembimbing : Rudy Arijanto, S.Kom., M.Kom.
Judul Skripsi : RANCANGAN BANGUN WEBSITE PREDIKSI HARGA EMAS
MENGUNAKAN METODE LONG SHORT-TERM MEMORY
(LSTM)

Tanggal	Catatan	Paraf
2025-03-04	Konsultasi dan persetujuan judul	
2025-03-11	Latar belakang penelitian	
2025-03-17	Tujuan dan Manfaat	
2025-04-08	Metodologi Penelitian	
2025-04-08	Analisa Kebutuhan	
2025-04-22	Demo program	
2025-05-06	Spesifikasi hardware dan software	
2025-05-16	Demo program	
2025-05-20	Simpulan dan Saran	

Mengetahui

Ketua Program Studi



Hartana Wijaya, M.Kom

Tangerang, 04 June 2025

Pembimbing

Rudy Arijanto, S.Kom., M.Kom.



Lampiran 2 Kode Model

```
import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

from sklearn.preprocessing import MinMaxScaler

from sklearn.metrics import mean_absolute_percentage_error

import tensorflow as tf

from keras import Model

from keras.layers import Input, Dense, Dropout, LSTM

from tensorflow.keras.callbacks import EarlyStopping, ReduceLROnPlateau

from tensorflow.keras.optimizers import Adam

import joblib

import os

from sklearn.metrics import mean_squared_error

import numpy as np

# Load dataset

df = pd.read_csv('emas_2020_2022.csv')

# Membersihkan dataset

df.drop(['Vol.', 'Perubahan%'], axis=1, errors='ignore', inplace=True)

df['Date'] = pd.to_datetime(df['Date'], dayfirst=True)

df.sort_values(by='Date', ascending=True, inplace=True)

df.reset_index(drop=True, inplace=True)
```

```
# Format nilai numerik

for col in ['Price', 'Open', 'High', 'Low']:

    df[col] = df[col].astype(float)


# Normalisasi data

scaler = MinMaxScaler()

df['Scaled_Price'] = scaler.fit_transform(df[['Price']])


# Tentukan ukuran test (10% terakhir)

test_size = int(len(df) * 0.1)


# Siapkan data untuk LSTM

window_size = 60

data = df['Scaled_Price'].values.reshape(-1, 1)

train_data = data[:-test_size]

test_data = data[-test_size - window_size:]


X_train, y_train, X_test, y_test = [], [], [], []

for i in range(window_size, len(train_data)):

    X_train.append(train_data[i-window_size:i, 0])

    y_train.append(train_data[i, 0])

for i in range(window_size, len(test_data)):

    X_test.append(test_data[i-window_size:i, 0])

    y_test.append(test_data[i, 0])
```

```
X_train, X_test = np.array(X_train), np.array(X_test)
```

```
y_train, y_test = np.array(y_train), np.array(y_test)
```

```
# Reshape untuk LSTM
```

```
X_train = np.reshape(X_train, (X_train.shape[0], X_train.shape[1], 1))
```

```
X_test = np.reshape(X_test, (X_test.shape[0], X_test.shape[1], 1))
```

```
# Arsitektur model LSTM (stabil dan terbukti)
```

```
def define_model():
```

```
    input1 = Input(shape=(window_size, 1))
```

```
    x = LSTM(units=64, return_sequences=True)(input1)
```

```
    x = Dropout(0.2)(x)
```

```
    x = LSTM(units=64, return_sequences=True)(x)
```

```
    x = Dropout(0.2)(x)
```

```
    x = LSTM(units=64)(x)
```

```
    x = Dropout(0.2)(x)
```

```
    x = Dense(32, activation='relu')(x)
```

```
    output = Dense(1, activation='linear')(x)
```

```
    model = Model(inputs=input1, outputs=[output])
```

```
    model.compile(loss='mean_squared_error', optimizer=Adam(learning_rate=0.001))
```

```
    return model
```

```
model = define_model()
```

```
# Callback training

early_stopping = EarlyStopping(monitor='val_loss', patience=15,
restore_best_weights=True)

reduce_lr = ReduceLROnPlateau(monitor='val_loss', factor=0.5, patience=7, min_lr=1e-5)

# Train model

model.fit(X_train, y_train, epochs=150, batch_size=32, validation_split=0.1, verbose=1,
callbacks=[early_stopping, reduce_lr])

# Evaluasi

y_pred = model.predict(X_test)

# Inverse transform hasil prediksi dan ground truth

y_test_inverse = scaler.inverse_transform(y_test.reshape(-1, 1))
y_pred_inverse = scaler.inverse_transform(y_pred)

MAPE = mean_absolute_percentage_error(y_test, y_pred) * 100
Accuracy = 100 - MAPE

print(f"Test Accuracy: {Accuracy:.2f}%")
print(f"MAPE (Mean Absolute Percentage Error): {MAPE:.2f}%")

# Visualisasi

plt.figure(figsize=(15, 6))

dates = df['Date'].iloc[-test_size:]

plt.plot(dates, scaler.inverse_transform(y_test.reshape(-1, 1)), color='blue', lw=2)
plt.plot(dates, scaler.inverse_transform(y_pred.reshape(-1, 1)), color='red', lw=2)
```

```
plt.title('Gold Price Prediction', fontsize=15)
```

```
plt.xlabel('Date', fontsize=12)
```

```
plt.ylabel('Price', fontsize=12)
```

```
plt.legend(['Actual Price', 'Predicted Price'])
```

```
plt.grid(True)
```

```
plt.tight_layout()
```

```
plt.show()
```

```
# Balikkan scaling
```

```
actual_prices = scaler.inverse_transform(y_test.reshape(-1, 1)).flatten()
```

```
predicted_prices = scaler.inverse_transform(y_pred.reshape(-1, 1)).flatten()
```

```
# Buat DataFrame evaluasi
```

```
eval_table = pd.DataFrame({
```

```
    'Date': dates.values,
```

```
    'Actual Price': actual_prices,
```

```
    'Predicted Price': predicted_prices
```

```
})
```

```
# Hitung APE dan SE
```

```
eval_table['Absolute Percentage Error'] = np.abs(eval_table['Actual Price'] -  
eval_table['Predicted Price']) / eval_table['Actual Price']
```

```
eval_table['Squared Error'] = (eval_table['Actual Price'] - eval_table['Predicted Price'])**2
```

```
# Formatting tabel
```

```

eval_table.index += 1 # Mulai index dari 1

eval_table['Date'] = pd.to_datetime(eval_table['Date']).dt.strftime("%d/%m/%Y")

eval_table['Actual Price'] = eval_table['Actual Price'].apply(lambda x: f'{x:,.0f}')

eval_table['Predicted Price'] = eval_table['Predicted Price'].apply(lambda x: f'{x:,.0f}')

eval_table['Absolute Percentage Error'] = eval_table['Absolute Percentage Error'].apply(lambda x: f'{x:.6f}')

eval_table['Squared Error'] = eval_table['Squared Error'].apply(lambda x: f'{x:,.2f}')

# Tampilkan tabel
print("\nTabel Evaluasi Model:")
print(eval_table.to_string(index=True))

# ===== Tambahan untuk Simpan ke Excel =====
# Untuk Excel: kembali ke format numerik
eval_table_excel = eval_table.copy()

# Kembalikan harga jual dan harga beli ke angka
eval_table_excel['Actual Price'] = eval_table_excel['Actual Price'].str.replace(',',
).astype(float)

eval_table_excel['Predicted Price'] = eval_table_excel['Predicted Price'].str.replace(',',
).astype(float)

# APE dikembalikan ke float
eval_table_excel['Absolute Percentage Error'] = eval_table_excel['Absolute Percentage Error'].astype(float)

```

Squared Error: tetap angka, tidak perlu diubah dari string

```
eval_table_excel['Squared Error'] = eval_table_excel['Squared Error'].str.replace(',',  
").astype(float)
```

Simpan file

```
eval_table_excel.to_excel('tabel_evaluasi_model.xlsx', index=True)
```

```
print("\nTabel Evaluasi Model juga telah disimpan sebagai 'tabel_evaluasi_model.xlsx'.")
```

Simpan model dan scaler

```
model.save("app/model_emas_final_2020.h5")
```

```
joblib.dump(scaler, "app/model_emas_2020.pkl")
```

```
print("Model dan scaler berhasil disimpan.")
```

```
print("Working directory saat ini:", os.getcwd())
```