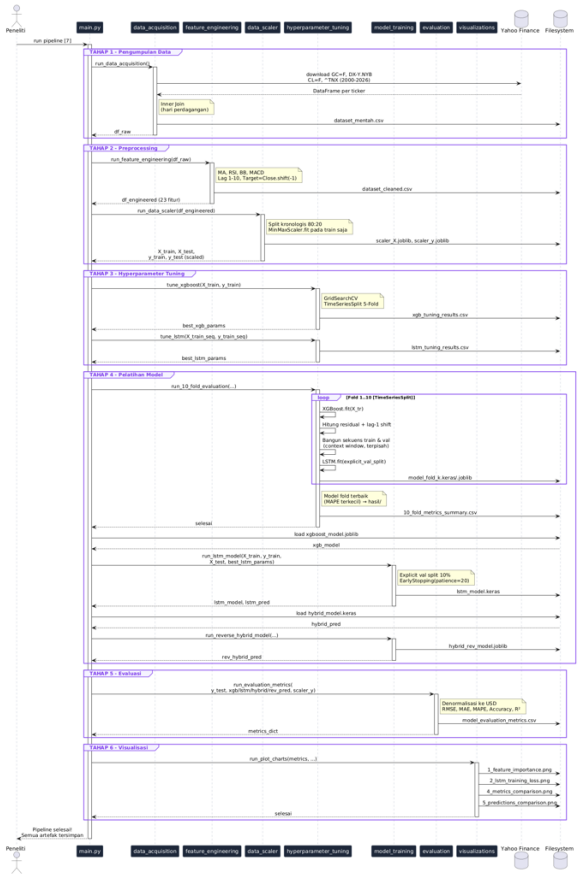
BAB IV
HASIL DAN PEMBAHASAN

4.1 Implementasi Penelitian

Implementasi penelitian ini dilaksanakan mengikuti alur pipeline yang telah dirancang pada BAB III, mulai dari pengumpulan data hingga visualisasi hasil. Seluruh tahapan diimplementasikan dalam bentuk modul Python yang terpisah (data_acquisition, feature_engineering, data_scaler, hyperparameter_tuning, model_training, evaluation, dan visualizations) yang dieksekusi secara berurutan oleh satu berkas orkestrasi utama (main.py). Gambar 4.1 menyajikan sequence diagram yang menggambarkan interaksi antar modul tersebut secara rinci, dan menjadi acuan utama dalam menjelaskan implementasi pada subbab ini.



Gambar 4. 1 Sequence Diagram Alur Eksekusi Program Prediksi Harga Emas XAU/USD

Berdasarkan Gambar 4.1 implementasi penelitian terbagi ke dalam enam tahapan berurutan sebagai berikut.

1. Tahap Pengumpulan Data. Modul data_acquisition memanggil fungsi run_data_acquisition() yang mengunduh data historis dari Yahoo Finance untuk empat ticker, yaitu GC=F (harga emas), DX-Y.NYB (indeks Dolar AS), CL=F (harga minyak mentah WTI), dan ^TNX (yield obligasi AS 10 tahun) untuk periode 2000 hingga 2026. Data per ticker yang diterima dalam bentuk DataFrame terpisah kemudian digabungkan menggunakan operasi inner join berdasarkan hari perdagangan (trading day) yang sama, sehingga hanya baris dengan tanggal yang tersedia pada seluruh sumber yang dipertahankan. Hasil penggabungan disimpan sebagai df_raw dan diekspor ke berkas dataset_mentah.csv.
2. Tahap Preprocessing. Modul feature_engineering menerima df_raw dan menghasilkan 23 fitur turunan (MA, RSI, Bollinger Band, MACD, dan lag feature), serta kolom Target yang merupakan hasil pergeseran (shift) Close price satu hari ke depan. DataFrame hasil rekayasa fitur (df_engineered) kemudian diteruskan ke modul data_scaler, yang melakukan pembagian data secara kronologis dengan rasio 80:20 dan menyesuaikan (fit) MinMaxScaler hanya pada data training untuk mencegah kebocoran data (data leakage). Hasilnya diekspor sebagai dataset_cleaned.csv beserta berkas scaler_X.joblib dan scaler_y.joblib.
3. Tahap Hyperparameter Tuning. Modul hyperparameter_tuning menjalankan fungsi tune_xgboost() menggunakan GridSearchCV dengan skema TimeSeriesSplit, serta tune_lstm() untuk mencari kombinasi parameter LSTM terbaik. Kedua fungsi mengembalikan parameter optimal (best_xgb_params dan best_lstm_params) yang digunakan pada tahap pelatihan berikutnya.
4. Tahap Pelatihan Model. Modul model_training menjalankan run_10_fold_evaluation(), yaitu iterasi walk-forward sebanyak 10 fold

menggunakan skema TimeSeriesSplit. Pada setiap fold, XGBoost dilatih terlebih dahulu, kemudian residualnya dihitung dan digeser satu langkah (lag-1 shift) untuk dibangun menjadi sekuens input LSTM. Model fold dengan nilai MAPE terkecil dipilih sebagai model_fold_terbaik dan disimpan pada direktori hasil. Setelah proses 10-fold selesai, model XGBoost, LSTM, model Hybrid XGBoost-LSTM, serta model Hybrid LSTM-XGBoost (arah terbalik, run_reverse_hybrid_model()) masing-masing dilatih ulang secara utuh pada keseluruhan data training untuk menghasilkan prediksi akhir (xgb_pred, lstm_pred, hybrid_pred, rev_hybrid_pred) pada data testing.

5. Tahap Evaluasi. Modul evaluation menjalankan run_evaluation_metrics() yang mendenormalisasi seluruh hasil prediksi kembali ke skala USD menggunakan scaler_y, kemudian menghitung metrik RMSE, MAE, MAPE, Accuracy, dan R^2 untuk keempat model. Hasilnya disimpan sebagai model_evaluation_metrics.csv.
6. Tahap Visualisasi. Modul visualizations menjalankan run_plot_charts() yang menghasilkan berkas-berkas grafik pendukung analisis, di antaranya grafik perbandingan metrik evaluasi dan grafik perbandingan prediksi terhadap harga aktual.

4.2 Hasil Pengumpulan Data

Data yang digunakan dalam penelitian ini bersumber dari Yahoo Finance, mencakup harga harian emas dunia (ticker GC=F) sebagai variabel utama, serta tiga variabel makroekonomi pendukung, yaitu indeks Dolar AS (DX-Y.NYB), harga minyak mentah WTI (CL=F), dan yield obligasi pemerintah AS tenor 10 tahun (^TNX). Data dikumpulkan untuk periode 30 Agustus 2000 hingga 27 Februari 2026 melalui proses run_data_acquisition() sebagaimana dijelaskan pada Subbab 4.1, menghasilkan dua bentuk dataset yang diuraikan pada Subbab 4.2.1 dan 4.2.2 berikut.

4.3 Dataset Mentah

Dataset mentah merupakan hasil penggabungan (inner join) keempat sumber data berdasarkan hari perdagangan yang sama, sebelum melalui proses feature engineering maupun pembersihan lebih lanjut. Dataset ini berisi 6.386 baris data harian dengan sembilan kolom, yaitu Date, Open, High, Low, Close, Volume, DXY, WTI, dan TNX. Tabel 4.1 menyajikan cuplikan empat baris pertama dari dataset mentah.

Tabel 4. 1 Cuplikan Dataset Mentah Harga Emas (XAU/USD) beserta Fitur Makroekonomi

No	Date	Open	High	Low	Close	Volume	DXY	WTI	TNX
1	2000-08-30	2.738	2.738	2.738	2.738	0	11.213	33.400	5.800
2	2000-08-31	27.479	27.829	27.479	27.829	0	1.125	33.099	5.729
3	2000-09-01	277.00	277.00	277.00	277.00	0	11.141	33.380	5.675
4	2000-09-05	27.579	27.579	27.579	27.579	2	11.241	3.379	5.683
…	…	…	…	…	…	…	…	…	…
6384	2026-02-24	51.587	5159.00	51.127	51.557	88	9.787	6.562	40.329
6385	2026-02-25	5166.00	520.639	5166.00	520.639	1772	9.769	6.541	4.047
6386	2026-02-26	51.772	51.992	514.389	5176.05	1520	9.779	6.520	4.017

6387	2026-02-27	51.867	5280.00	51.767	5230.5	354	9.761	670.199	39.619

Berdasarkan Tabel 4.1, terlihat bahwa pada awal periode pengumpulan data (tahun 2000), harga emas (Close) berada pada kisaran 270-280 USD per troy ons, jauh lebih rendah dibandingkan harga pada akhir periode data (27 Februari 2026) yang mencapai 5.230,5 USD per troy ons sebagaimana tercatat pada baris terakhir dataset_mentah.csv. Rentang pergerakan harga yang sangat lebar ini (kurang lebih 19 kali lipat) mengonfirmasi urgensi permasalahan yang diangkat pada BAB I, yaitu volatilitas harga emas yang tinggi dalam jangka panjang. Pada tahap ini, data belum mengandung indikator teknikal maupun fitur lag sehingga belum dapat langsung digunakan sebagai input model.

4.4 Dataset Hasil Preprocessing

Dataset mentah kemudian diproses melalui tahap preprocessing sebagaimana dijabarkan pada BAB III, meliputi feature engineering (perhitungan MA, RSI, Bollinger Band, MACD, dan fitur lag), serta penghapusan baris yang mengandung nilai kosong (NaN) akibat perhitungan indikator berbasis window (misalnya MA-30 yang membutuhkan 30 hari data sebelumnya sehingga 30 baris pertama pada awal periode data tidak memiliki nilai MA-30 yang valid). Proses ini menghasilkan dataset_cleaned.csv sebanyak 6.356 baris data (berkurang 30 baris dari dataset mentah) dengan total 24 kolom, terdiri atas 23 fitur input dan satu kolom Target (Close price yang digeser satu hari ke depan).

Tabel 4.2 menyajikan cuplikan empat baris pertama dataset_cleaned.csv, ditampilkan pada sebagian kolom representatif dari setiap kelompok fitur (penjelasan lengkap seluruh 23 fitur disajikan pada Subbab 4.3).

Tabel 4. 2 Cuplikan Dataset Hasil Preprocessing

No	Date	Close	MA_7	RSI_14	MACD	Lag_1	Target

1	2000-10-11	270.5	270.69	50.43	-0.74	272.40	276.40
2	2000-10-12	276.4	271.37	58.33	-0.30	270.50	272.40
3	2000-10-13	272.4	271.67	47.10	-0.28	276.40	271.50
4	2000-10-16	271.5	271.80	46.00	-0.33	272.40	271.10
…	…	…	…	…	…	…	…
6354	2026-02-23	5204.7	5007.8	70.94	102.6	5059.29	5155.7
6355	2026-02-24	5155.7	5041.0	60.89	106.86	5204.7	5206.3
6356	2026-02-25	5206.3	5067.3	62.00	112.97	5155.7	5176.5
6357	2026-02-26	5176.5	5109.2	63.55	114.09	5206.39	5230.5

Berdasarkan Tabel 4.2, terlihat bahwa dataset hasil preprocessing telah dilengkapi dengan indikator teknikal (MA_7, RSI_14, MACD), fitur lag (Lag_1), dan kolom Target yang siap digunakan sebagai input dan label pada tahap normalisasi, pembagian data, serta pelatihan model pada subbab-subbab berikutnya. Sebagai contoh, kolom Target pada baris pertama (276,40) sama dengan nilai Close pada baris kedua (276,40), yang mengonfirmasi bahwa Target memang merupakan hasil pergeseran Close price satu hari ke depan (shift(-1)).

4.5 Hasil Feature Engineering

Feature engineering menghasilkan 23 fitur input yang terbagi ke dalam empat kelompok, sebagaimana ditampilkan pada struktur kolom dataset_cleaned.csv. Berikut penjelasan fungsi masing-masing kelompok fitur terhadap prediksi harga emas.

1. Moving Average (MA_7, MA_14, MA_30, MA_20). Keempat fitur ini merupakan rata-rata bergerak Close price pada periode 7, 14, 30, dan 20 hari. Sebagai ilustrasi, pada baris pertama dataset_cleaned.csv (tanggal 11 Oktober 2000) tercatat MA_7 sebesar 270,69 dan MA_30 sebesar 272,93, sedangkan Close price pada tanggal tersebut adalah 270,50. Nilai MA yang mendekati namun tidak identik dengan Close price ini menunjukkan fungsi MA sebagai

penghalus (smoother) fluktuasi harga jangka pendek, sehingga model dapat mengenali arah tren dasar tanpa terganggu oleh noise harian.

2. Relative Strength Index (RSI_14). RSI mengukur kekuatan momentum kenaikan atau penurunan harga dalam skala 0-100. Pada baris pertama dataset_cleaned.csv, RSI_14 tercatat sebesar 50,43, yang menunjukkan kondisi pasar berada pada momentum netral (tidak overbought maupun oversold). Fitur ini membantu model mengidentifikasi potensi pembalikan arah (reversal) harga.
3. Bollinger Band (BB_Upper, BB_Middle, BB_Lower). Ketiga fitur ini mengukur volatilitas harga di sekitar MA-20. Pada baris pertama dataset_cleaned.csv, BB_Upper tercatat 276,63 dan BB_Lower tercatat 267,63, dengan rentang pita (band width) sebesar 9,0 di sekitar BB_Middle sebesar 272,13. Lebar pita ini merepresentasikan tingkat volatilitas harga pada periode tersebut; semakin lebar pita, semakin tinggi volatilitas pasar.
4. MACD dan MACD_Signal. MACD dihitung sebagai selisih EMA-12 dan EMA-26, sedangkan MACD_Signal adalah EMA-9 dari MACD. Pada baris pertama, MACD tercatat -0,74 dan MACD_Signal -0,53, atau MACD berada di bawah garis sinyal, yang secara teknikal umumnya diinterpretasikan sebagai sinyal momentum bearish (tekanan jual). Fitur ini memberikan informasi arah dan kekuatan tren kepada model.
5. Fitur Lag (Lag_1, Lag_2, Lag_3, Lag_5, Lag_10). Fitur lag merepresentasikan nilai Close price pada t-1 hingga t-10 hari sebelumnya. Fitur ini secara eksplisit memberikan konteks historis langsung kepada model, sehingga baik XGBoost yang secara alami tidak memiliki memori urutan maupun LSTM dapat memanfaatkan informasi historis tersebut sebagai fitur input tabular.
6. Target. Target merupakan Close price yang digeser satu hari ke depan (shift(-1)), yaitu variabel yang hendak diprediksi oleh model untuk horizon prediksi jangka pendek (1 hari ke depan) sesuai batasan masalah yang ditetapkan pada BAB I.

4.6 Hasil Normalisasi dan Pembagian Data

Normalisasi data dilakukan menggunakan metode MinMaxScaler dari library scikit-learn, yang mentransformasi setiap fitur ke dalam rentang [0, 1] menggunakan rumus x' = (x - x_min) / (x_max - x_min). Normalisasi diperlukan karena LSTM menggunakan fungsi aktivasi (seperti sigmoid dan tanh) yang sensitif terhadap skala input; tanpa normalisasi, fitur dengan rentang nilai besar (misalnya Close price pada kisaran ratusan hingga ribuan USD) dapat mendominasi proses pembelajaran dibandingkan fitur dengan rentang kecil (misalnya RSI_14 pada skala 0-100).

Sesuai dengan alur pada Gambar 4.1, proses fit MinMaxScaler hanya dilakukan pada data training (scaler_X.joblib dan scaler_y.joblib), sementara data testing hanya ditransformasi (transform) menggunakan parameter x_min dan x_max yang telah dipelajari dari data training. Sebagai ilustrasi, nilai minimum dan maksimum Close price pada data training tercatat masing-masing sebesar 255,10 USD dan 2.051,50 USD. Dengan menggunakan kedua nilai ini, titik data pertama pada set testing (3 Februari 2021) yang memiliki Close price aktual sebesar 1.832,20 USD dinormalisasi menjadi 0,8779. Pendekatan fit-on-train-only ini penting untuk mencegah data leakage, yaitu kebocoran informasi dari data masa depan (testing) ke dalam proses pelatihan model.

Pembagian data dilakukan secara kronologis dengan rasio 80:20, di mana 80% data awal (5.084 baris, periode 11 Oktober 2000 hingga 2 Februari 2021) digunakan sebagai data training, dan 20% data terbaru (1.272 baris, periode 3 Februari 2021 hingga 26 Februari 2026) digunakan sebagai data testing. Pembagian secara kronologis dipilih karena data yang digunakan bersifat time series; jika pembagian dilakukan secara acak (random split), model berpotensi mempelajari informasi dari periode setelah titik prediksi yang seharusnya belum diketahui, sehingga hasil evaluasi menjadi bias dan tidak merepresentasikan kondisi prediksi dunia nyata.

4.7 Optimasi Hyperparameter

Optimasi hyperparameter dilakukan secara terpisah untuk model XGBoost dan model LSTM menggunakan skema Grid Search dengan validasi

TimeSeriesSplit, sesuai dengan rentang nilai yang telah ditetapkan pada Tabel 3.3 BAB III.

1. Pada model XGBoost, lima hyperparameter yang diuji adalah n_estimators, max_depth, learning_rate, subsample, dan colsample_bytree. Pencarian dilakukan menggunakan GridSearchCV yang dikombinasikan dengan TimeSeriesSplit, sehingga setiap kombinasi parameter divalidasi pada beberapa fold data yang tetap mempertahankan urutan kronologis, mencegah kebocoran informasi dari data validasi yang berada setelah data training pada fold tersebut.

Optimasi Hyperparameter - Model XGBoost

Tabel 1: Hyperparameter yang Dioptimasi

Parameter	Nilai / Rentang Uji
colsample_bytree	[0.8, 1.0]
learning_rate	[0.05, 0.1]
max_depth	[3, 5]
n_estimators	[50, 100]
subsample	[0.8, 1.0]

Tabel 2: Hasil Parameter Optimal

Parameter	Nilai Optimal
colsample_bytree	1.0
learning_rate	0.1
max_depth	3
n_estimators	100
subsample	1.0

Gambar 4. 2 Hasil Optimasi Hyperparameter Model XGBoost

Berdasarkan Gambar 4.2, parameter optimal yang dihasilkan adalah n_estimators = 100, max_depth = 3, learning_rate = 0,1, subsample = 1,0, dan colsample_bytree = 1,0. Pemilihan max_depth yang dangkal (3, dari rentang uji 3-5) mengindikasikan bahwa pohon keputusan yang lebih sederhana (weak learner) justru menghasilkan generalisasi yang lebih baik pada data harga emas dibandingkan pohon yang lebih dalam, yang cenderung overfitting terhadap noise historis. Nilai subsample dan colsample_bytree yang keduanya bernilai 1,0 (batas atas rentang uji) mengindikasikan bahwa penggunaan seluruh data dan seluruh fitur pada setiap pohon (tanpa random subsampling) justru menghasilkan performa validasi terbaik, kemungkinan karena volume data historis yang cukup besar (25 tahun) membuat risiko overfitting akibat penggunaan data penuh menjadi lebih kecil. Learning rate sebesar 0,1 (batas atas rentang uji) juga mengindikasikan konvergensi yang lebih

cepat masih dapat diterima tanpa mengorbankan stabilitas pelatihan pada jumlah pohon yang relatif sedikit (100).

2. Pada model LSTM, lima hyperparameter yang diuji adalah batch_size, dropout_rate, learning_rate, lstm_units_1 (lapisan pertama), dan lstm_units_2 (lapisan kedua).

Optimasi Hyperparameter - Model LSTM

Tabel 1: Hyperparameter yang Dioptimasi

Parameter	Nilai / Rentang Uji
batch_size	[16, 32]
dropout_rate	[0.1, 0.2]
learning_rate	[0.001, 0.002]
lstm_units_1	[64, 128]
lstm_units_2	[32, 64]

Tabel 2: Hasil Parameter Optimal

Parameter	Nilai Optimal
batch_size	16
dropout_rate	0.1
learning_rate	0.001
lstm_units_1	128
lstm_units_2	32

Gambar 4. 3 Hasil Optimasi Hyperparameter Model LSTM

Berdasarkan Gambar 4.3, parameter optimal yang dihasilkan adalah batch_size = 16, dropout_rate = 0,1, learning_rate = 0,001, lstm_units_1 = 128, dan lstm_units_2 = 32. Batch size kecil (16, dari rentang uji 16-32) menghasilkan pembaruan bobot yang lebih sering dalam satu epoch, yang pada data time series dengan pola non-stasioner seperti harga emas dapat membantu model beradaptasi lebih baik terhadap perubahan pola jangka pendek, meskipun dengan konsekuensi waktu pelatihan yang lebih lama. Learning rate optimal berada pada batas bawah rentang uji (0,001), menunjukkan bahwa pembelajaran yang lebih hati-hati (gradual) lebih sesuai untuk mencegah osilasi pada proses konvergensi model rekuren. Konfigurasi lstm_units_1 = 128 pada lapisan pertama dan lstm_units_2 = 32 pada lapisan kedua membentuk arsitektur funnel (mengerucut), di mana lapisan pertama

menangkap representasi temporal yang kaya, sedangkan lapisan kedua memadatkan representasi tersebut menjadi fitur yang lebih ringkas sebelum diteruskan ke lapisan output. Dropout rate optimal sebesar 0,1 (batas bawah rentang uji) mengindikasikan bahwa regularisasi yang terlalu agresif (dropout tinggi) justru merugikan performa model pada kasus ini, kemungkinan karena volume data yang besar sudah cukup untuk mencegah overfitting tanpa memerlukan dropout yang tinggi.

4.8 Pelatihan Model

Proses pelatihan dilakukan terhadap empat model untuk memungkinkan perbandingan yang komprehensif: XGBoost tunggal, LSTM tunggal, model Hybrid XGBoost-LSTM (arsitektur utama yang diusulkan), dan model Hybrid LSTM-XGBoost (arsitektur terbalik) yang disertakan sebagai eksperimen pembanding tambahan untuk menguji apakah urutan penggabungan kedua algoritma memengaruhi hasil akhir.

Sesuai dengan Gambar 4.1, tahap pelatihan diawali dengan proses evaluasi 10-fold menggunakan skema TimeSeriesSplit. Pada setiap fold, XGBoost dilatih terlebih dahulu menggunakan potongan data training pada fold tersebut (XGBoost.fit(X_tr)). Selanjutnya, residual dihitung sebagai selisih antara prediksi XGBoost dan nilai aktual, kemudian residual tersebut digeser satu langkah ke belakang (lag-1 shift) agar pada saat prediksi, model hanya menggunakan informasi residual yang telah tersedia (tanpa melihat residual pada hari yang sama dengan target prediksi). Fitur gabungan (fitur asli, prediksi XGBoost, dan residual yang telah di-lag) kemudian disusun menjadi sekuens (context window) yang terpisah untuk data training dan validasi pada fold tersebut, lalu dilatih pada LSTM (LSTM.fit dengan explicit validation split). Proses ini konsisten dengan mekanisme dua tahap yang dijelaskan pada Subbab 3.5.3 BAB III, yaitu XGBoost sebagai feature extractor pada tahap pertama dan LSTM sebagai temporal modeler pada tahap kedua.

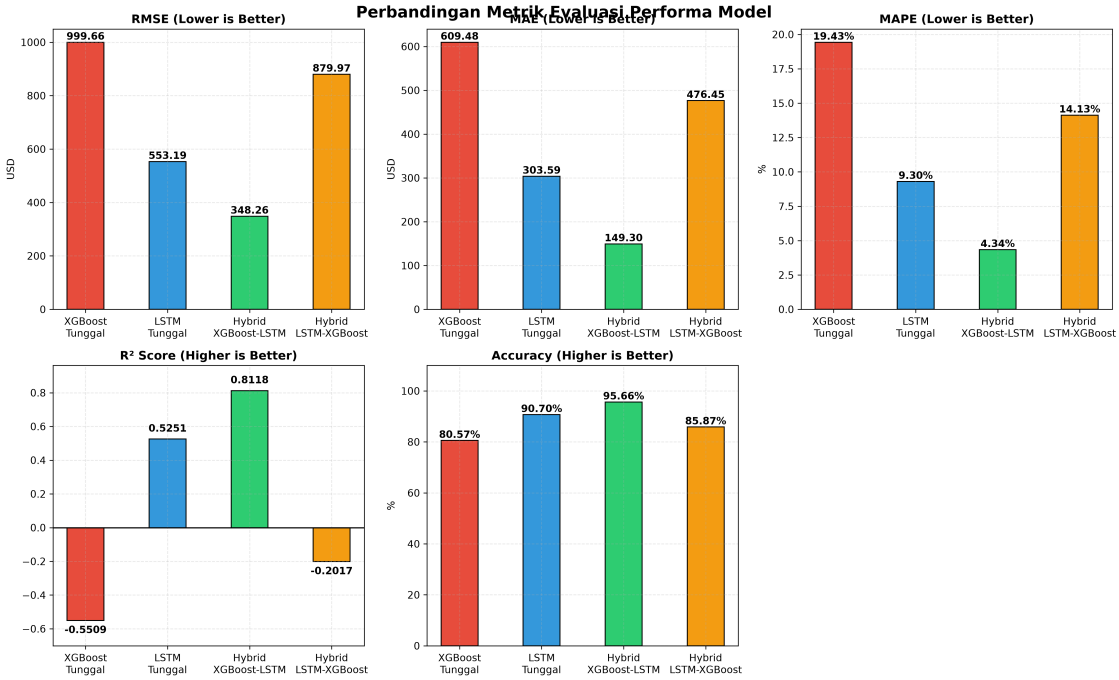
Dari sepuluh fold yang dievaluasi, model pada fold dengan nilai MAPE terkecil dipilih sebagai model_fold_terbaik dan disimpan sebagai representasi performa model pada kondisi data yang paling optimal. Sebagai contoh, Gambar 4.6

pada Subbab 4.8 menunjukkan hasil prediksi pada fold ke-9, yang mencatatkan MAPE sebesar 0,53% dan akurasi 99,47% pada segmen data validasi fold tersebut.

Setelah proses evaluasi 10-fold selesai, keempat model (XGBoost, LSTM, Hybrid XGBoost-LSTM, dan Hybrid LSTM-XGBoost) masing-masing dilatih ulang secara utuh menggunakan keseluruhan data training (80% data awal) dan diuji pada keseluruhan data testing (20% data terbaru, 1.272 baris) untuk menghasilkan hasil evaluasi akhir yang dibandingkan pada Subbab 4.7. Pelatihan model LSTM pada tahap ini menggunakan explicit validation split sebesar 10% dari data training beserta callback EarlyStopping dengan patience 20 epoch, sesuai dengan Subbab 3.6.1 BAB III. Pada model Hybrid LSTM-XGBoost, urutan pemrosesan dibalik: LSTM dilatih terlebih dahulu untuk memodelkan pola temporal, dan hasil keluarannya (beserta residualnya) kemudian dijadikan fitur tambahan bagi XGBoost pada tahap kedua.

4.9 Evaluasi Model

Evaluasi dilakukan pada data testing (1.272 baris data, periode 3 Februari 2021 hingga 26 Februari 2026) untuk keempat model yang telah dilatih, menggunakan lima metrik: RMSE, MAE, MAPE, Accuracy (dihitung sebagai 100% - MAPE), dan R^2 (koefisien determinasi). Seluruh hasil prediksi telah didenormalisasi kembali ke skala harga asli (USD per troy ons) sebelum dihitung metriknya.



Gambar 4. 4 Grafik Perbandingan Metrik Evaluasi Performa Model

Tabel 4.3 merekapitulasi hasil kelima metrik evaluasi untuk keempat model yang dibandingkan.

Tabel 4. 3 Rekapitulasi Komparatif Hasil Metrik Evaluasi Kinerja Model

Model	RMSE (USD)	MAE (USD)	MAPE (%)	Accuracy (%)	R^2
XGBoost Tunggal	999,66	609,48	19,43	80,57	-0,5509
LSTM Tunggal	553,19	303,59	9,30	90,70	0,5251
Hybrid XGBoost-LSTM	348,26	149,30	4,34	95,66	0,8118
Hybrid LSTM-XGBoost	879,97	476,45	14,13	85,87	-0,2017

1. RMSE. XGBoost tunggal mencatatkan RMSE tertinggi (999,66 USD), diikuti Hybrid LSTM-XGBoost (879,97 USD), LSTM tunggal (553,19 USD), dan Hybrid XGBoost-LSTM terendah (348,26 USD). Karena RMSE memberi penalti kuadratik terhadap kesalahan besar, nilai RMSE XGBoost yang jauh

lebih tinggi mengindikasikan model ini kerap menghasilkan kesalahan prediksi ekstrem pada periode-periode dengan pergerakan harga yang tajam.

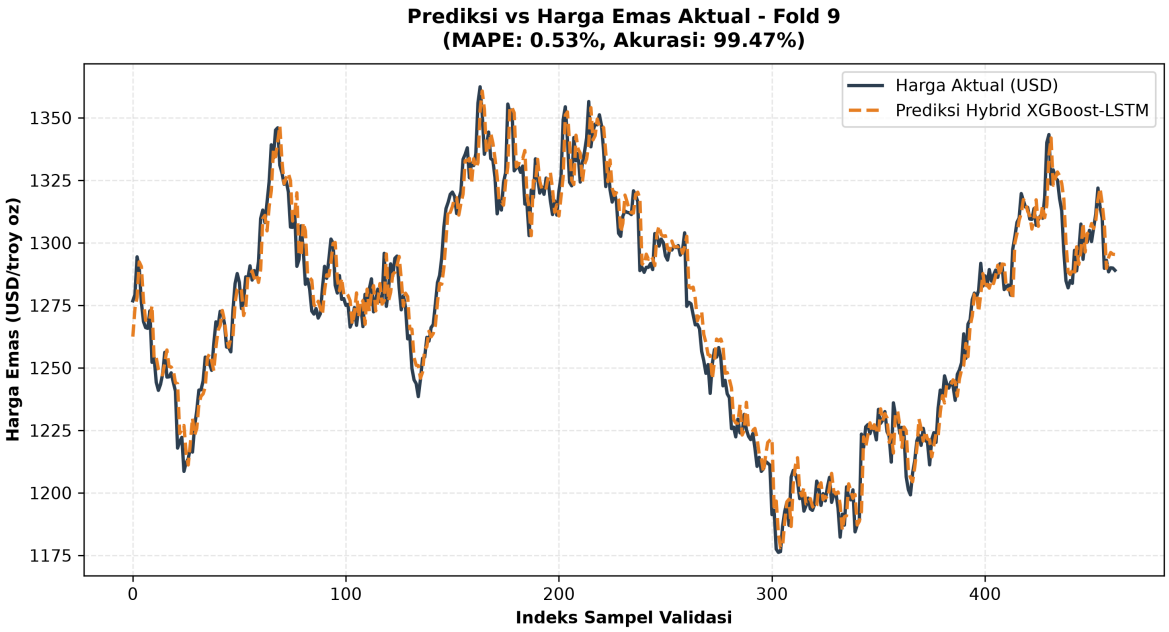
2. MAE. Pola yang sama terlihat pada MAE, di mana Hybrid XGBoost-LSTM mencatatkan kesalahan rata-rata terendah (149,30 USD), kurang dari seperempat MAE XGBoost tunggal (609,48 USD). Karena MAE tidak memberi penalti tambahan pada outlier, konsistensi urutan performa antar-model pada RMSE dan MAE mengindikasikan bahwa keunggulan Hybrid XGBoost-LSTM bukan semata karena berkurangnya kesalahan ekstrem, melainkan karena kesalahan prediksi yang secara konsisten lebih kecil di sepanjang data testing.
3. MAPE dan Accuracy. Hybrid XGBoost-LSTM mencatatkan MAPE terendah sebesar 4,34% (accuracy 95,66%), yang menurut kategori interpretasi MAPE pada Tabel 2.1 BAB II tergolong sangat akurat (di bawah 10%). Sebaliknya, XGBoost tunggal dengan MAPE 19,43% berada pada ambang batas antara kategori akurat dan cukup akurat.
4. R^2. Nilai R^2 memberikan gambaran paling mencolok mengenai perbedaan performa antar-model. Hybrid XGBoost-LSTM mencatatkan R^2 sebesar 0,8118, yang berarti model ini mampu menjelaskan sekitar 81,18% variansi pergerakan harga aktual pada data testing. Sebaliknya, XGBoost tunggal (-0,5509) dan Hybrid LSTM-XGBoost (-0,2017) mencatatkan R^2 negatif, yang berarti kedua model tersebut secara keseluruhan berkinerja lebih buruk dibandingkan model naif yang hanya memprediksi rata-rata harga aktual pada seluruh periode testing. Hal ini konsisten dengan keterbatasan XGBoost yang dijelaskan pada Subbab 2.6.3 BAB II, yaitu ketidakmampuannya memproses urutan temporal secara alami; pada horizon testing yang mencakup tren kenaikan harga emas yang sangat tajam (2021-2026), keterbatasan ini berdampak signifikan terhadap akurasi prediksi.
5. Perbandingan XGBoost, LSTM, dan Hybrid XGBoost-LSTM. Hybrid XGBoost-LSTM secara konsisten unggul pada seluruh metrik dibandingkan kedua model tunggalnya. Dibandingkan XGBoost tunggal, Hybrid XGBoost-LSTM menurunkan RMSE sebesar 65,17% dan MAPE sebesar 77,66%.

Dibandingkan LSTM tunggal, Hybrid XGBoost-LSTM menurunkan RMSE sebesar 37,04% dan MAPE sebesar 53,33%. Hasil ini mengonfirmasi skema hybrid yang dirancang pada Subbab 3.5.3 BAB III: XGBoost berperan mengekstraksi dan membobot fitur multi-variabel (OHLCV, indikator teknikal, dan variabel makroekonomi) menjadi prediksi awal dan residual yang terstruktur, sementara LSTM kemudian mempelajari pola temporal jangka panjang dari residual tersebut yang tidak tertangkap oleh XGBoost saja. Kombinasi kedua kemampuan inilah yang menghasilkan penurunan error yang jauh lebih besar dibandingkan peningkatan yang diperoleh LSTM tunggal atas XGBoost tunggal.

6. Perbandingan Hybrid XGBoost-LSTM dan Hybrid LSTM-XGBoost. Menariknya, model Hybrid LSTM-XGBoost (arah terbalik) justru berkinerja jauh lebih buruk daripada Hybrid XGBoost-LSTM, bahkan lebih buruk daripada LSTM tunggal pada seluruh metrik (RMSE 879,97 USD berbanding 553,19 USD, R^2 -0,2017 berbanding 0,5251). Temuan ini mengindikasikan bahwa urutan penggabungan kedua algoritma sangat menentukan efektivitas skema hybrid. Ketika LSTM diproses terlebih dahulu, model kehilangan kesempatan untuk memanfaatkan kemampuan XGBoost dalam menyeleksi dan membobot fitur multi-variabel sejak awal proses, sehingga LSTM harus mempelajari pola temporal langsung dari fitur mentah yang belum terseleksi. Sebaliknya, ketika residual LSTM diteruskan ke XGBoost pada tahap kedua, XGBoost yang secara alami memperlakukan setiap observasi secara independen tidak mampu memanfaatkan struktur temporal residual tersebut secara optimal. Hasil ini memperkuat justifikasi desain arsitektur Hybrid XGBoost-LSTM (XGBoost sebagai feature extractor pada tahap pertama, LSTM sebagai temporal modeler pada tahap kedua) yang telah ditetapkan pada BAB III sebagai skema yang lebih tepat dibandingkan urutan sebaliknya.

4.10 Analisis Prediksi

Untuk memberikan gambaran visual kualitas prediksi model Hybrid XGBoost-LSTM, Gambar 4.5 menampilkan kurva perbandingan antara harga aktual dan hasil prediksi pada fold ke-9 dari proses evaluasi 10-fold TimeSeriesSplit yang dijelaskan pada Subbab 4.6.



Gambar 4. 5 Analisis Visual Pergerakan Prediksi vs Aktual XAU/USD - Fold 9

Perlu ditegaskan bahwa metrik pada Gambar 4.5 (MAPE 0,53%, akurasi 99,47%) merupakan hasil pada segmen validasi fold ke-9 yang dipilih sebagai model_fold_terbaik dalam proses evaluasi 10-fold, dan berbeda dari metrik agregat pada keseluruhan data testing yang disajikan pada Tabel 4.3 (MAPE 4,34%, akurasi 95,66%). Perbedaan ini wajar mengingat fold ke-9 merupakan satu dari sepuluh segmen data yang dievaluasi secara walk-forward, sehingga performanya merepresentasikan kondisi data pada satu jendela waktu tertentu, bukan keseluruhan periode testing yang mencakup rentang waktu lebih panjang dan lebih beragam kondisi pasarnya.

Berdasarkan Gambar 4.5, dapat diamati beberapa karakteristik kualitas prediksi sebagai berikut.

1. Kesesuaian kurva. Kurva prediksi (garis oranye putus-putus) mengikuti kurva harga aktual (garis biru tua) dengan sangat rapat sepanjang keseluruhan indeks sampel validasi (0 hingga lebih dari 450), baik pada segmen harga menurun (indeks 0-30, harga turun dari sekitar 1.295 ke 1.210 USD) maupun pada segmen harga menaik tajam (indeks 130-160, harga naik dari sekitar 1.240 ke 1.365 USD).
2. Deviasi dan titik error terbesar. Deviasi antara kurva prediksi dan aktual tampak paling terlihat pada titik-titik puncak (peak) dan lembah (trough) tajam, misalnya di sekitar indeks 160 dan 300, di mana kurva prediksi sedikit tertinggal (lag) dari kurva aktual sesaat sebelum dan sesudah titik balik arah. Pola keterlambatan pada titik balik ini merupakan karakteristik umum model berbasis sekuensial seperti LSTM, yang cenderung mempelajari kelanjutan tren berdasarkan pola historis sehingga membutuhkan sedikit waktu untuk 'menyadari' pembalikan arah yang tiba-tiba.
3. Kemampuan mengikuti tren. Model menunjukkan kemampuan yang baik dalam mengikuti tren jangka menengah, terlihat dari kesesuaian kurva pada tren penurunan panjang di sekitar indeks 260-300 (harga turun dari sekitar 1.300 ke di bawah 1.180 USD) maupun tren kenaikan panjang pada indeks 340-430 (harga naik dari sekitar 1.185 ke 1.345 USD).
4. Kemampuan menangkap volatilitas. Pada segmen dengan fluktuasi harian yang tinggi (indeks 150-220), di mana harga aktual berosilasi tajam antara 1.300 hingga 1.365 USD dalam rentang indeks yang relatif pendek, kurva prediksi tetap mampu mengikuti osilasi tersebut meskipun dengan amplitudo yang sedikit lebih halus (smoothed) dibandingkan pergerakan aktual. Hal ini mengindikasikan bahwa mekanisme residual XGBoost yang diteruskan ke LSTM cukup efektif dalam menangkap komponen non-linear jangka pendek, meskipun belum sepenuhnya menyamai volatilitas ekstrem harga aktual.

4.11 Pembahasan

Hasil penelitian ini secara umum mendukung kerangka teori yang disintesiskan pada Subbab 2.10 BAB II, yaitu bahwa LSTM unggul dalam menangkap dependensi temporal jangka panjang namun lemah dalam seleksi fitur multi-variabel, sementara XGBoost unggul dalam pengolahan fitur terstruktur namun tidak mampu memproses urutan temporal. Superioritas Hybrid XGBoost-LSTM (MAPE 4,34%, R^2 0,8118) atas kedua model tunggalnya pada seluruh metrik evaluasi membuktikan bahwa mekanisme dua tahap yang dirancang pada Subbab 3.5.3 BAB III XGBoost sebagai feature extractor dan LSTM sebagai temporal modeler berhasil menutupi keterbatasan masing-masing algoritma secara saling melengkapi, sebagaimana diprediksikan secara teoritis.

Dibandingkan dengan penelitian terdahulu yang dirangkum pada Tabel 1.1 BAB I, hasil penelitian ini juga menunjukkan pola yang konsisten. Selayanti et al. (2025) melaporkan bahwa model hybrid LSTM-XGBoost menghasilkan RMSE 117,89 dan MAE 92,45 pada prediksi harga saham BBRI, lebih unggul dari kedua model individualnya; penelitian ini juga menemukan bahwa model hybrid (dengan urutan XGBoost-LSTM) mengungguli kedua model tunggalnya, meskipun pada instrumen dan rentang harga yang berbeda (harga emas pada skala ratusan hingga ribuan USD dibandingkan harga saham BBRI). Idham (2025) menyimpulkan bahwa pendekatan hybrid LSTM-XGBoost secara umum menghasilkan performa lebih baik dibandingkan model tunggal pada data time series; kesimpulan umum ini didukung oleh penelitian ini, meskipun temuan pada Subbab 4.7 menambahkan nuansa penting bahwa urutan penggabungan kedua algoritma bukan hal yang dapat dipertukarkan begitu saja, karena Hybrid LSTM-XGBoost pada penelitian ini justru berkinerja lebih buruk daripada LSTM tunggal.

Sementara itu, dibandingkan dengan capaian LSTM tunggal pada penelitian Rahman et al. (2025) yang melaporkan akurasi 99,55% pada prediksi harga emas ANTAM menggunakan 4.545 data harian, LSTM tunggal pada penelitian ini hanya mencapai akurasi 90,70%. Perbedaan ini kemungkinan disebabkan oleh perbedaan karakteristik data: harga emas ANTAM di pasar domestik Indonesia cenderung

memiliki volatilitas harian yang lebih rendah dan pergerakan yang lebih tertata dibandingkan harga emas dunia XAU/USD yang digunakan pada penelitian ini, yang mencakup periode-periode gejolak pasar global seperti pandemi COVID-19 (2020) dan lonjakan harga emas yang ekstrem pada 2024-2026 (harga naik dari sekitar 1.800 USD menjadi lebih dari 5.000 USD dalam rentang data testing). Rentang volatilitas yang jauh lebih ekstrem inilah yang menjadi tantangan utama bagi model tunggal XGBoost dan LSTM pada penelitian ini, sekaligus menjadi konteks yang memperkuat urgensi pendekatan hybrid yang diusulkan.

Perbandingan dengan capaian XGBoost tunggal pada penelitian Pratama (2024), yang melaporkan R^2 89,09% dan MAPE 3,21% untuk prediksi harga saham, juga menunjukkan hasil yang berbeda dengan XGBoost tunggal pada penelitian ini (R^2 -0,5509, MAPE 19,43%). Perbedaan yang signifikan ini kembali mengonfirmasi keterbatasan mendasar XGBoost yang dijelaskan pada Subbab 2.6.3 BAB II, yaitu ketidakmampuannya memproses urutan temporal secara alami setiap observasi diperlakukan secara independen sehingga tidak dapat menangkap long-range temporal dependency. Keterbatasan ini tampaknya berdampak jauh lebih besar pada data harga emas dengan tren jangka panjang yang kuat dan volatilitas tinggi (seperti pada rentang data testing penelitian ini) dibandingkan pada data harga saham yang digunakan Pratama (2024), yang kemungkinan memiliki karakteristik pergerakan yang berbeda.

Secara keseluruhan, hasil penelitian ini memperkuat justifikasi ilmiah penggunaan model hybrid XGBoost-LSTM yang diuraikan pada Subbab 1.1 Latar Belakang dan mengisi celah penelitian (research gap) yang diidentifikasi pada Subbab 3.2 Studi Literatur BAB III, yaitu belum tersedianya penelitian yang secara khusus mengimplementasikan model hybrid XGBoost-LSTM untuk memprediksi harga emas dunia (XAU/USD). Temuan tambahan mengenai pentingnya urutan penggabungan algoritma (Hybrid XGBoost-LSTM versus Hybrid LSTM-XGBoost) juga memberikan kontribusi baru yang belum secara eksplisit dibahas pada penelitian-penelitian terdahulu yang dirujuk pada BAB I dan BAB II