

MODEL HYBRID LSTM-XGBOOST UNTUK PREDIKSI NILAI TUKAR RUPIAH TERHADAP DOLAR AMERIKA

Kevin¹, Riyadi J. Iskandar², Sandi Tendean.³

¹²³Teknik Informatika, Fakultas Teknologi Informasi Universitas Widya Dharma Pontianak
e-mail: ¹21421483_kevin@widyadharmia.ac.id, ²riyadi@widyadharmia.ac.id, ³sandi_t@widyadharmia.ac.id

Abstract

Exchange rate prediction is crucial for economic stability, investment, and international trade. The high volatility of the rupiah against the US dollar makes accurate prediction challenging. This study proposes a hybrid Long Short-Term Memory (LSTM) and eXtreme Gradient Boosting (XGBoost) model to improve exchange rate prediction accuracy. The LSTM captures long-term temporal dependencies in time-series data, while XGBoost models the residuals from LSTM predictions to capture non-linear patterns and macroeconomic influences. The dataset spans January 2020 to December 2024, including daily Rupiah exchange rates, BI 7-Day Reverse Repo Rate, inflation, and US Dollar Index (DXY). Results demonstrate that the hybrid model provides relatively better performance compared to individual LSTM and XGBoost models, which experienced significant prediction failures. While the hybrid approach shows potential for short-term forecasting with acceptable margin error, the model's ability to explain data variability remains limited. This indicates that although the hybrid method offers improvement over individual models, substantial enhancements in model architecture and data preprocessing are still required for practical applications.

Keywords: Long Short-Term Memory, eXtreme Gradient Boosting, Machine Learning, Prediction, Exchange Rate.

Abstrak

Prediksi nilai tukar sangat penting untuk stabilitas ekonomi, investasi, dan perdagangan internasional. Volatilitas tinggi rupiah terhadap dolar Amerika Serikat membuat prediksi yang akurat menjadi menantang. Penelitian ini mengusulkan model hibrid Long Short-Term Memory (LSTM) dan eXtreme Gradient Boosting (XGBoost) untuk meningkatkan akurasi prediksi nilai tukar. LSTM menangkap ketergantungan temporal jangka panjang dalam data deret waktu, sedangkan XGBoost memodelkan residual dari prediksi LSTM untuk menangkap pola non-linear dan pengaruh makroekonomi. Dataset mencakup periode Januari 2020 hingga Desember 2024, termasuk nilai tukar harian Rupiah, BI 7-Day Reverse Repo Rate, inflasi, dan US Dollar Index (DXY). Hasil menunjukkan bahwa model hibrid memberikan performa yang relatif lebih baik dibandingkan model LSTM dan XGBoost tunggal yang mengalami kegagalan prediksi signifikan. Meskipun pendekatan hibrid menunjukkan potensi untuk peramalan jangka pendek dengan margin error yang dapat diterima, kemampuan model dalam menjelaskan variabilitas data masih terbatas. Hal ini mengindikasikan bahwa meskipun metode hibrid menawarkan perbaikan dibanding model tunggal, peningkatan substansial dalam arsitektur model dan preprocessing data masih diperlukan untuk aplikasi praktis.

Kata Kunci: Long Short-Term Memory, eXtreme Gradient Boosting, Machine Learning, Prediksi, Nilai Tukar

1. PENDAHULUAN

Nilai tukar mata uang memiliki peran penting dalam perdagangan internasional, investasi, dan stabilitas ekonomi suatu negara. Fluktuasi nilai tukar dapat memengaruhi inflasi, harga barang impor, dan daya saing ekspor. Di Indonesia, depresiasi rupiah terhadap dolar Amerika Serikat (USD) sering diikuti dengan kenaikan tingkat inflasi, yang berdampak pada perekonomian secara keseluruhan.

Prediksi nilai tukar bersifat kompleks karena dipengaruhi oleh banyak faktor, antara lain perbedaan tingkat inflasi, perbedaan suku bunga, defisit transaksi berjalan, utang publik, ketentuan perdagangan, harga komoditas, stabilitas politik, serta peristiwa tak terduga seperti pandemi, bencana alam, dan gejolak geopolitik. Kompleksitas ini memerlukan metode pemodelan yang mampu mengolah hubungan antar variabel secara tepat.

Berbagai metode telah digunakan dalam analisis deret waktu (*time series*) untuk memodelkan pergerakan nilai tukar, salah satunya *Long Short-Term Memory* (LSTM). LSTM memiliki keterbatasan ketika jumlah data terbatas karena memerlukan volume data yang besar untuk menghasilkan prediksi yang akurat^[1].

Metode lain yang digunakan adalah *Extreme Gradient Boosting* (XGBoost), yang mampu mengolah hubungan non-linear dan variabel eksternal. XGBoost memiliki keterbatasan pada data deret waktu karena tidak dapat menangkap pola temporal secara efektif^[1]. XGBoost dapat memanfaatkan variabel eksternal seperti tingkat

inflasi, BI 7-Day Reverse Repo Rate, dan US Dollar Index (DXY), namun kurang mampu mengidentifikasi ketergantungan jangka panjang pada data berurutan.

Berdasarkan temuan tersebut, kombinasi metode LSTM dan XGBoost digunakan untuk mengatasi kelemahan masing-masing. LSTM memproses pola historis dan keterkaitan waktu pada data deret waktu, sedangkan XGBoost mengolah variabel eksternal secara lebih efektif. Pendekatan ini digunakan untuk menghasilkan model prediksi nilai tukar yang dapat memanfaatkan keunggulan keduanya dalam pemodelan data deret waktu dengan variabel eksternal.

2. METODE PENELITIAN

2.1 Pendekatan Penelitian

Penelitian ini menggunakan pendekatan kuantitatif dengan memanfaatkan data numerik historis yang diperoleh dari Bank Indonesia dan Investing.com, mencakup periode Januari 2014 hingga Desember 2024. Data utama terdiri dari kurs jual Rupiah terhadap Dolar Amerika Serikat, tingkat inflasi, suku bunga acuan BI 7-Day Reverse Repo Rate (BI7DRR), dan US Dollar Index (DXY).

2.1.1 Metode Pengumpulan Data

Data yang digunakan dalam penelitian ini diperoleh melalui pengumpulan data sekunder dari sumber-sumber resmi, yaitu Bank Indonesia (bi.go.id) dan Investing.com. Periode pengamatan mencakup Januari 2014 hingga Desember 2024. Jenis data yang dikumpulkan meliputi kurs harian Rupiah terhadap Dolar Amerika Serikat (USD/IDR), suku bunga acuan BI 7-Day Reverse Repo Rate, tingkat inflasi bulanan, dan US Dollar Index (DXY). Data kurs dan DXY berskala harian, sedangkan inflasi dan suku bunga dicatat dalam skala bulanan. Seluruh data kemudian diselaraskan dalam format deret waktu harian untuk keperluan pelatihan dan pengujian model prediksi.

2.1.2 Teknik Perancangan Model

Perancangan model menggunakan *python 3.10.9* sebagai bahasa pemrograman, *tensorflow 2.19.0*, *keras 3.10.0*, *scikit-learn 1.6.1*, *matplotlib 3.10.3* sebagai pustaka dan *visual studio code* sebagai *code editor*.

2.2 Landasan Teori

2.2.1 Forecasting

Forecasting adalah prediksi nilai masa depan dari suatu deret waktu menggunakan nilai masa lalu yang diketahui dari deret waktu tersebut dan/atau beberapa variabel terkait lainnya. Hal ini sangat mirip dengan prediksi dalam ML di mana kita menggunakan model untuk memprediksi data yang belum pernah dilihat sebelumnya^[2]. *Forecasting* adalah salah satu cara memprediksi kejadian di masa depan dengan menggunakan metode ilmiah yang dilakukan secara sistematis. *Forecasting* bertujuan untuk menguji perkembangan saat ini dan keterkaitannya dengan masa mendatang^[3].

2.2.2 Machine Learning

Machine Learning adalah bagian dari AI yang melibatkan penerapan algoritma yang mampu belajar dari data atau contoh sebelumnya dan mampu melakukan tugas-tugas tanpa instruksi atau pemrograman yang eksplisit. Prosedur untuk belajar dari data melibatkan pengenalan pola Secara statistik dan menyesuaikan model untuk mengevaluasi data dengan lebih akurat dan memberikan hasil yang tepat^[4]. *Machine Learning* (ML) merupakan bagian dari *Artificial Intelligence* yang mempelajari algoritma komputer yang mampu meningkatkan performanya secara otomatis melalui pengalaman dan pemanfaatan data^[5].

2.2.3 Nilai Tukar (Kurs)

Kurs adalah rasio antara satuan unit suatu valuta asing lainnya yang bisa saling ditukarkan (dikonversikan) pada satu waktu dengan uang dalam negeri (rupiah) pada waktu tertentu^[6]. Kurs merupakan perbandingan nilai tukar mata uang suatu negara dengan mata uang negara asing atau perbandingan nilai tukar valuta antar Negara^[7].

2.2.4 Moving Average

Moving Average (MA) adalah indikator teknikal yang digunakan untuk menunjukkan arah tren harga dengan meratakan fluktuasi data. MA dihitung dengan menjumlahkan harga penutupan selama periode tertentu, lalu dibagi dengan jumlah periode:

$$(MA)_n = \frac{\sum_{i=t-n+1}^{t=n} C_i}{n} \quad (4)$$

MA jangka pendek lebih cepat merespons perubahan harga, sedangkan MA jangka panjang lebih stabil dan membantu mengurangi noise, meski menghasilkan sinyal yang lebih lambat^[8]. MA jangka pendek lebih cepat merespons perubahan harga, sedangkan MA jangka panjang lebih stabil dan membantu mengurangi noise, meski menghasilkan sinyal yang lebih lambat^[9].

2.2.5 Exponential Moving Average

Exponential Moving Average (EMA) adalah metode perhitungan rata-rata harga yang memberikan bobot lebih besar pada data terbaru. EMA lebih responsif terhadap perubahan harga terkini dibandingkan *Simple Moving*

Average (SMA), yang memberikan bobot yang sama untuk setiap data. Nilai awal EMA diperoleh dari SMA periode n , dengan rumus:

$$\text{Initial SMA} = \frac{n - \text{period sum}}{2^n} \quad (1)$$

$$\text{Multiplier} = \frac{2^n}{(n + 1)} \quad (2)$$

$$\text{EMA} = \{\text{Close} - \text{EMA}(\text{yesterday})\} \times \text{Multiplier} + \text{EMA}(\text{yesterday}) \quad (3)$$

Dengan semakin besar nilai n , respons EMA terhadap perubahan harga menjadi semakin halus namun tetap mempertahankan kepekaan terhadap tren terbaru^[8]. *Exponential Moving Average* (EMA) adalah jenis rata-rata bergerak yang memberikan bobot lebih pada harga terbaru dibandingkan harga sebelumnya, sehingga mengurangi keterlambatan dalam merespons perubahan harga^[10].

2.2.6 Long Short-Term Memory

Long Short-Term Memory (LSTM) mengatasi masalah *vanishing gradient* pada *Recurrent Neural Networks* (RNN) dengan menggunakan struktur cell state dan tiga gerbang utama: input, forget, dan output. Cell state diperbarui sebagai berikut:

$$\bar{c}_t^{(k)} = \bar{f} \odot \bar{c}_{t-1}^{(k)} + \bar{i} \odot \bar{c} \quad (5)$$

Sementara *hidden state* dihitung dengan:

$$\bar{h}_t^{(k)} = \bar{f} \odot \tanh(\bar{c}_t^{(k)}) \quad (6)$$

Simbol \odot menyatakan perkalian elemen-per-elemen (*Hadamard product*). Mekanisme ini menjaga stabilitas gradien dan memungkinkan pembelajaran jangka panjang yang efektif^[11]. *Long Short-Term Memory* (LSTM) adalah jaringan saraf yang diperkenalkan pada tahun 1997 dan tetap banyak digunakan hingga sekarang. LSTM menggunakan fungsi aktivasi *tanh*, yang memberikan non-linearitas serta turunan kedua yang bisa dipertahankan dalam urutan panjang, untuk menghindari masalah *exploding* dan *vanishing gradients*. LSTM terdiri dari serangkaian sel LSTM yang terhubung secara berurutan, dengan setiap sel memiliki empat mekanisme pengaturan untuk mengelola *forget*, *learn*, *remember*, dan *use* data urutan^[12].

2.2.7 eXtreme Gradient Boosting

XGBoost adalah algoritma *ensemble learning* yang kuat di bidang ilmu data. XGBoost, sebuah algoritma *gradient boosting* yang banyak digunakan dalam bidang ilmu data, merupakan implementasi dari *gradient boosting* yang dirancang untuk efisiensi, adaptabilitas, dan portabilitas yang luar biasa^[13]. *eXtreme Gradient Boosting* (XGBOOST) adalah suatu kerangka kerja yang berbasis pada metode *gradient boosting* yang mengintegrasikan keuntungan dari teknik *bagging* dan *boosting*^[14].

2.3 Pra-pemrosesan Data

2.3.1 Pemuatan Data

Tahap awal dalam proses Pra-Pemrosesan adalah pengambilan dan pemuatan data dari berbagai sumber eksternal. Pada penelitian ini, data diambil dari empat *file* utama yaitu, kurs harian USD/IDR dari Bank Indonesia, *BI 7-Day Reverse Repo Rate* (BI7DRR), data inflasi bulanan dari Bank Indonesia, serta *US Dollar Index* (DXY) yang diunduh dari Investing.com. Masing-masing *file* dibaca ke dalam format *DataFrame* menggunakan pustaka *pandas*, dengan memastikan kolom tanggal (*Date*) dikenali sebagai tipe data *datetime*. Langkah ini bertujuan untuk mengintegrasikan seluruh sumber data yang diperlukan sebelum dilakukan proses pembersihan dan penggabungan data. Berikut adalah jumlah baris dari setiap data yang dimuat.

2.3.2 Penyesuaian Nama Kolom

Penyesuaian dan standarisasi nama kolom pada setiap dataset. Proses ini bertujuan untuk menciptakan konsistensi penamaan variabel yang digunakan, sehingga memudahkan integrasi dan pemanggilan data pada tahap-tahap berikutnya. Kolom 'Kurs Jual' diubah menjadi 'Kurs_Jual' dan 'BI-7Day-RR' menjadi 'BI_7Day_RR' pada masing-masing dataset. Selain itu, kolom 'Data Inflasi' diubah menjadi 'Inflasi' dan 'Price' menjadi 'DXY_Price'. Standarisasi ini menghindari terjadinya kesalahan referensi antar *dataset* dan mendukung proses penggabungan data.

2.3.3 Pembersihan Kolom Persentase

Pembersihan nilai pada kolom persentase dilakukan dengan menghapus karakter '%' pada data inflasi dan BI7DRR, kemudian mengubah seluruh nilainya menjadi tipe data *float* agar dapat diolah secara numerik. Sebagai contoh, data yang awalnya berbentuk 7.5% dibersihkan menjadi 7.5.

2.3.4 Penghapusan Kolom Tidak Relevan

Penghapusan terhadap kolom-kolom yang tidak relevan atau tidak memiliki kontribusi terhadap analisis dan pemodelan data. Kolom seperti 'No' dan 'Nilai' seringkali merupakan sisa dari proses unduh data yang tidak memberikan informasi tambahan pada analisis yang dilakukan. Dengan menghapus kolom-kolom tersebut, struktur data yang dihasilkan menjadi lebih bersih dan hanya memuat fitur-fitur yang memang diperlukan. Tujuannya untuk meningkatkan efisiensi proses analisis dan meminimalisasi risiko *error* akibat adanya kolom dengan nilai yang tidak terpakai.

2.3.5 Penggabungan Data

Penggabungan seluruh dataset yang telah disiapkan menjadi satu struktur data terintegrasi berdasarkan kolom tanggal. Penggabungan dilakukan secara bertahap, dengan menggunakan teknik *left join* agar seluruh variabel dari masing-masing sumber data dapat digabungkan sesuai dengan tanggal yang sama. Hasil dari proses ini adalah sebuah DataFrame yang berisi seluruh variabel prediktor dan target pada setiap barisnya untuk setiap tanggal dalam rentang waktu penelitian. Dengan data yang telah tergabung, proses analisis *time series* dan pelatihan model akan menjadi lebih efektif karena seluruh variabel prediktor dan target berada dalam satu struktur data yang terintegrasi. Kolom yang terbentuk setelah penggabungan data antara lain: Date, Kurs_Jual, BI7Day_RR, Inflasi dan DXY_Price.

2.3.6 Penyesuaian Indeks menjadi Kalender Harian

Karena tidak semua data makroekonomi tersedia untuk setiap hari, dilakukan penyesuaian indeks menjadi kalender harian penuh. sehingga setelah proses penggabungan, terdapat tanggal-tanggal yang kosong. Untuk mengatasi hal ini, dilakukan penyesuaian indeks DataFrame menjadi kalender harian penuh, yaitu dengan menyesuaikan indeks agar mencakup seluruh tanggal dari awal hingga akhir periode penelitian tanpa ada yang terlewat. DataFrame hasil gabungan diindeks ulang ke rentang tanggal harian dari tanggal awal hingga tanggal akhir dataset. Dengan penyesuaian ini, data akan mencakup seluruh hari dalam periode penelitian, termasuk akhir pekan yang menjadi 4.017 baris data. Tujuan utama dari langkah ini adalah memastikan seluruh rentang waktu penelitian tercakup secara lengkap, sehingga model yang akan dibangun dapat belajar pola musiman, tren, maupun fluktuasi harian secara utuh. Selain itu, penyesuaian indeks ke kalender harian juga memungkinkan integrasi data antar variabel dengan frekuensi pencatatan yang berbeda tetap dapat dianalisis secara bersamaan.

2.3.7 Penyesuaian Panjang Data

Penyesuaian ke kalender harian menyebabkan jumlah titik data menjadi berbeda dari data sumber, terutama pada variabel dengan frekuensi pencatatan bulanan seperti BI7DRR dan inflasi. Perbedaan panjang data ini diatasi dengan menerapkan interpolasi spline untuk menghasilkan nilai pada tanggal-tanggal yang sebelumnya tidak memiliki catatan, sehingga pola perubahan tetap terjaga secara alami. Jika pada awal atau akhir deret waktu terdapat kekosongan akibat rentang data sumber yang berbeda, digunakan metode *forward fill* dan *backward fill*. Dengan langkah ini, seluruh variabel memiliki data lengkap pada setiap tanggal, membuat *dataset* konsisten dan siap digunakan untuk pemodelan.

2.3.8 Rekayasa Fitur

Tahap rekayasa fitur melibatkan penciptaan variabel-variabel baru yang lebih informatif dari data yang sudah ada, bertujuan untuk meningkatkan daya prediktif model. Fitur-fitur yang direkayasa meliputi:

a. MA200

Kolom ini berisi rata-rata kurs jual selama 200 hari terakhir. Fitur ini menggambarkan tren jangka panjang dan mengurangi fluktuasi jangka pendek. Perhitungan MA dapat dilihat pada persamaan (4). Pengimplementasian rumus perhitungan MA untuk fitur ini adalah sebagai berikut:

$$MA_{200} = \frac{12.303 + 12.287 + 12.288,33 + \dots + 11.635}{200} = 11.779,66$$

b. EMA10

Kolom ini berisi nilai *Exponential Moving Average* (EMA) dari kurs jual dengan periode 10 hari. Fitur ini berfungsi agar model lebih sensitif dan responsif terhadap perubahan harga jangka pendek atau volatilitas kurs yang tiba-tiba, membantu model LSTM untuk agar model lebih sensitif dan responsif terhadap perubahan harga jangka pendek atau volatilitas kurs yang tiba-tiba. Perhitungan untuk EMA dapat dilihat pada persamaan (1), (2) dan (3). Pengimplementasian rumus perhitungan EMA untuk fitur ini adalah sebagai berikut:

$$\begin{aligned} Multiplier &= \frac{2}{(10 + 1)} = 0,18 \\ EMA_{10} &= \{12.288,33 - 12.300,09\} \times 0,18 + 12.300,09 \\ EMA_{10} &= 12.297,95 \end{aligned}$$

Dalam fitur ini, jumlah periode yang digunakan untuk perhitungan EMA adalah sebanyak 10 hari.

c. is_weekend

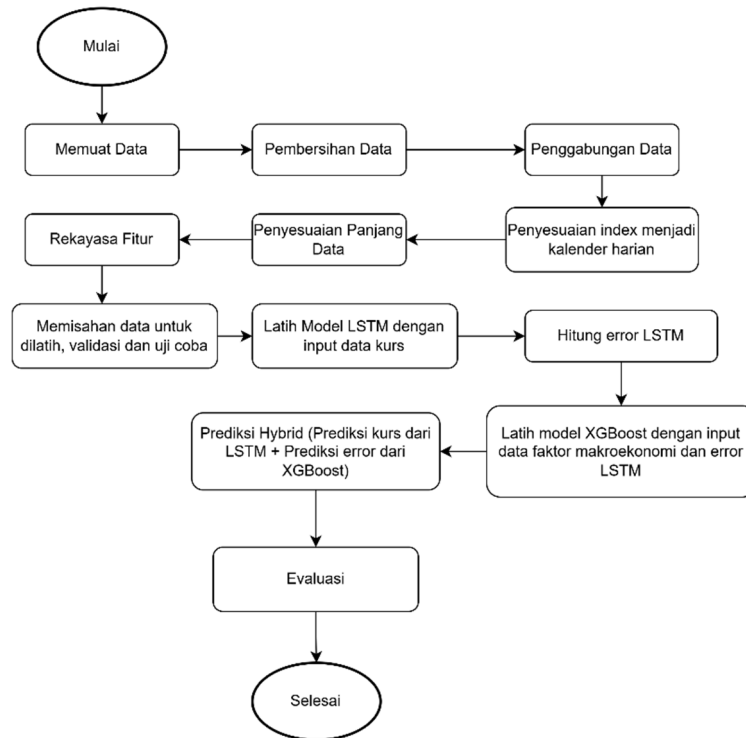
Penambahan fitur *is_weekend*, yaitu fitur biner (*boolean*) yang menandai apakah suatu tanggal merupakan akhir pekan atau bukan. Fitur ini sangat penting pada dataset *time series* keuangan, karena aktivitas pasar biasanya berhenti pada akhir pekan sehingga pola pergerakan kurs dapat berbeda dibandingkan hari kerja. Berikut adalah potongan kode untuk menambahkan fitur *is_weekend*:

```
merged_df['MA200'] = merged_df['Kurs_Jual'].rolling(window=200,
min_periods=1).mean()
merged_df['EMA10'] = merged_df['Kurs_Jual'].ewm(span=10,
adjust=False).mean()
merged_df['is_weekend'] = merged_df.index.weekday >= 5
```

2.4 Perancangan Model *Hybrid* LSTM-XGBoost

2.4.1 Flowchart

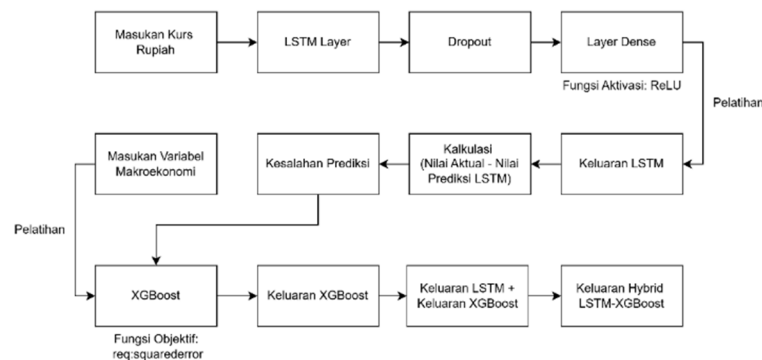
Pada Gambar 1 digambarkan proses kerja model yang dimulai dengan memuat data dari berbagai sumber seperti kurs USD/IDR, BI 7-Day Reverse Repo Rate, inflasi, dan DXY. Selanjutnya dilakukan pembersihan dan penyesuaian nama kolom, serta penghapusan kolom tidak relevan. Data kemudian digabungkan berdasarkan tanggal untuk membentuk satu dataset multivariat. Setelah integrasi, dilakukan penyesuaian ke kalender harian, yang menimbulkan nilai kosong akibat perbedaan frekuensi data. Nilai kosong diisi menggunakan interpolasi spline serta metode *forward-fill* dan *backward-fill*. Data yang lengkap kemudian diperkaya melalui rekayasa fitur: MA200, EMA10, dan *is_weekend*. Setelah itu, data dibagi menjadi data pelatihan, validasi, dan pengujian. Model LSTM dilatih terlebih dahulu menggunakan data kurs, dan selisih prediksinya (*error*) dihitung. *Error* ini, bersama data makroekonomi (BI7DRR, inflasi, DXY), digunakan untuk melatih model XGBoost. Hasil prediksi akhir adalah gabungan prediksi LSTM dan koreksi dari XGBoost. Terakhir, model dievaluasi menggunakan metrik MAPE, MAE, RMSE, dan R^2 untuk mengukur akurasi prediksi terhadap data aktual.



Gambar 1. Flowchart Perancangan model

2.4.2 Arsitektur Model

Model *hybrid* yang digunakan dalam penelitian ini didesain untuk memanfaatkan kemampuan dua pendekatan *machine learning* berbeda secara sinergis, yaitu LSTM (Long Short-Term Memory) dan XGBoost (eXtreme Gradient Boosting). Kombinasi ini bertujuan agar prediksi nilai tukar Rupiah terhadap Dolar menjadi lebih akurat, karena memadukan keunggulan LSTM dalam menangkap pola historis data *time series* dengan keunggulan XGBoost dalam mengeksplorasi pengaruh variabel makroekonomi. Gambaran arsitektur dari model *hybrid* LSTM-XGBoost dapat dilihat pada Gambar 2.



Gambar 2. Arsitektur Model Hybrid LSTM-XGBoost

Model *hybrid* LSTM-XGBoost dirancang untuk menggabungkan keunggulan LSTM dalam mengenali pola time series dengan kemampuan XGBoost dalam memodelkan variabel makroekonomi. Arsitektur model terdiri dari dua tahap: prediksi awal oleh LSTM, dan koreksi error oleh XGBoost. Data kurs Rupiah digunakan sebagai input LSTM, yang dilengkapi dengan *dropout* dan *dense layer* untuk menghindari overfitting dan menghasilkan output prediksi awal. Selisih antara prediksi dan nilai aktual (error) kemudian menjadi input bagi XGBoost, bersama dengan variabel eksternal seperti inflasi, BI7DRR, DXY, serta indikator teknikal EMA10 dan MA200. XGBoost dilatih untuk meminimalkan *Mean Squared Error* dan menghasilkan koreksi prediksi. Hasil akhir diperoleh dari penjumlahan output LSTM dan koreksi dari XGBoost, membentuk sistem prediksi hybrid yang lebih akurat dalam memproyeksikan nilai tukar Rupiah terhadap Dolar

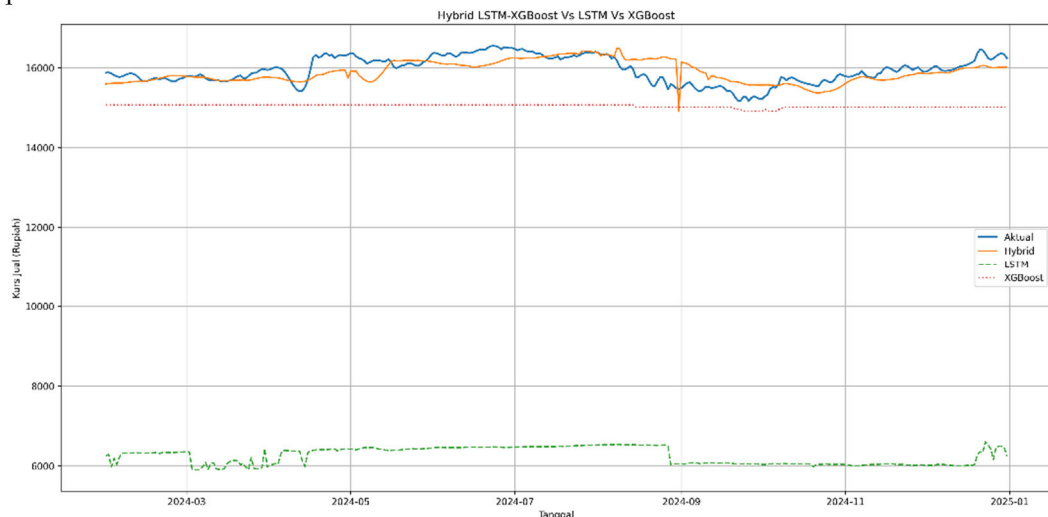
2.4.2 Pembagian Data

Pembagian data dilakukan untuk menghindari *data leakage* dan memastikan evaluasi model yang objektif. Dataset dibagi menjadi tiga bagian, yaitu data pelatihan (70 %) dari 1 Januari 2014 hingga 31 Desember 2021, yang digunakan untuk membangun model; data validasi (20 %) dari 1 Januari 2022 hingga 31 Desember 2023, yang digunakan untuk tuning parameter dan pemilihan model terbaik; serta data pengujian (10 %) dari Januari hingga Desember 2024, yang digunakan untuk menguji kemampuan model terhadap data yang benar-benar baru. Pembagian ini dirancang agar model dapat diuji dalam skenario prediksi nyata secara lebih akurat dan terukur.

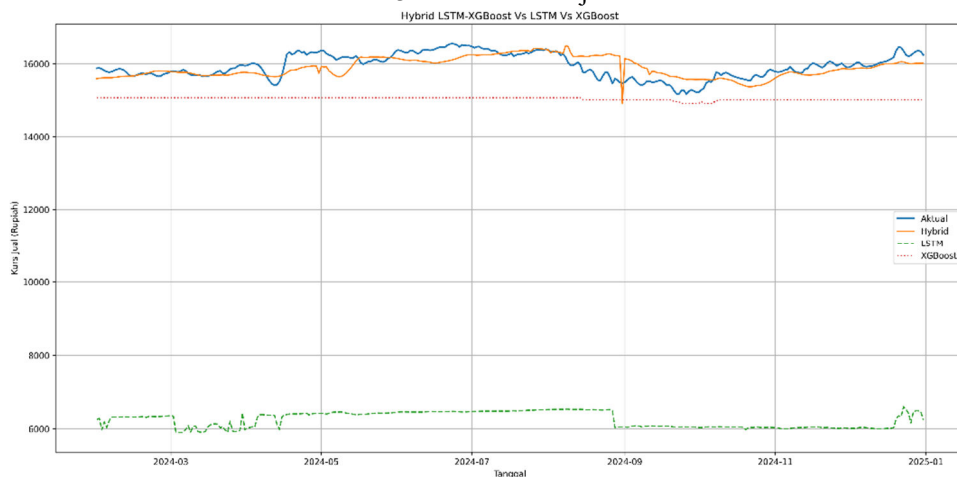
3. HASIL DAN PEMBAHASAN

3.1 Evaluasi Kinerja Model

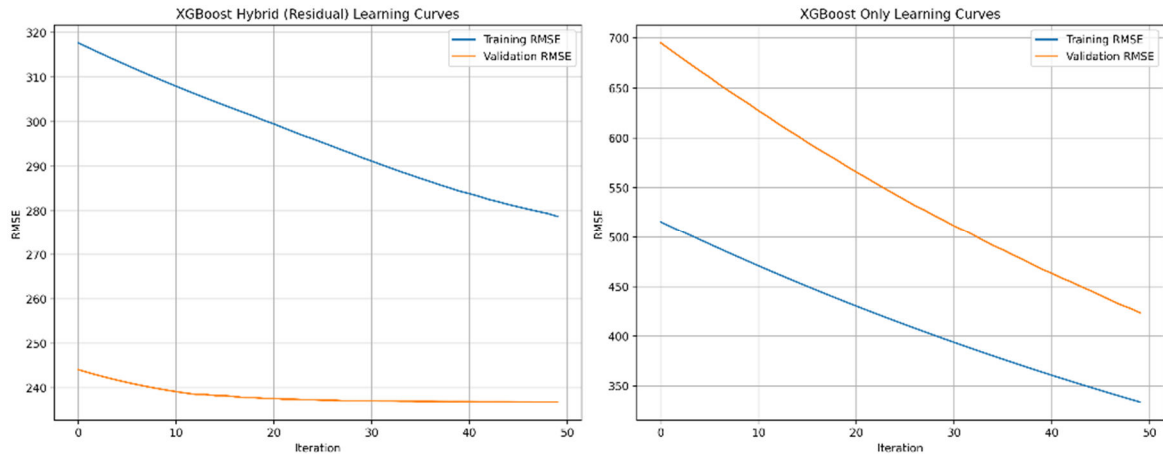
Setelah seluruh model berhasil dilatih dan dibentuk, Pengujian dilakukan pada data yang belum pernah dilihat model sebelumnya, dan hasilnya dievaluasi menggunakan metrik MAPE, MAE, RMSE dan *R-squared* untuk mengukur akurasi prediksi serta menganalisis kemampuan model dalam memprediksi nilai tukar rupiah terhadap dolar amerika.



Gambar 3. Grafik Hasil Uji Model



Gambar 4. Grafik Training dan Validation Loss



Gambar 5. Grafik Learning Curve XGBoost

3.1.1 Evaluasi Kinerja Model Hybrid LSTM-XGBoost

Model *hybrid* LSTM-XGBoost menunjukkan performa terbaik dalam memprediksi nilai tukar Rupiah terhadap Dolar Amerika. Hasil prediksi pada Tabel 1 memiliki rata-rata *margin error* sebesar 0,012249, yang menunjukkan bahwa model memiliki tingkat kesalahan yang relatif rendah terhadap nilai kurs aktual. Rata-rata *margin error* tersebut merepresentasikan deviasi sekitar 1,22 persen. Jika dibandingkan dengan model LSTM tunggal, model *hybrid* memang lebih konsisten dalam mengikuti fluktuasi jangka pendek, meskipun performanya belum sepenuhnya optimal.

Tabel 1. Sampel Data Hasil Uji Model Hybrid LSTM-XGBoost

Tanggal	Kurs Aktual	Kurs Prediksi	Margin Error ((Aktual-Prediksi)/Terakhir)
01-02-2024	15.882,01	15.610,48	0,015353
02-02-2024	15.853,88	15.615,81	0,012523
03-02-2024	15.824,73	15.617,87	0,010483
04-02-2024	15.795,59	15.617,03	0,009476
05-02-2024	15.766,44	15.616,62	0,010575
06-02-2024	15.783,52	15.623,25	0,011979

Stabilitas proses pembelajaran ditunjukkan oleh grafik *loss* selama pelatihan, baik *training loss* maupun *validation loss* menunjukkan penurunan yang konsisten tanpa indikasi *overfitting*. Kedua kurva menunjukkan nilai *loss* yang sudah rendah dan stabil sejak awal pelatihan hingga akhir *epoch*, seperti yang ditunjukkan pada Gambar 4.

3.1.2 Evaluasi Kinerja Model LSTM Tunggal

Model LSTM menunjukkan performa yang cukup baik dalam mengikuti pola historis kurs harian, namun kurang responsif terhadap perubahan tren ekstrem. *Margin error* harian cenderung lebih besar dibandingkan model *hybrid*, berkisar antara 0,005 hingga 0,012. Model LSTM tunggal menunjukkan performa yang kurang memuaskan dalam mengikuti pola kurs jual harian. Model ini mengalami kesulitan signifikan dalam merespon perubahan tren, terutama pada lonjakan atau penurunan tajam seperti yang ditunjukkan pada Gambar 4.

Tabel 2. Sampel Data Hasil Uji Model LSTM Tunggal

Tanggal	Kurs Aktual	Kurs Prediksi	Margin Error ((Aktual-Prediksi)/Terakhir)
01-02-2024	15.882,01	6.281,47	0,604492
02-02-2024	15.853,88	5.969,89	0,623443

03-02-2024	15.824,73	6.185,35	0,608865
04-02-2024	15.795,59	6.032,26	0,617808
05-02-2024	15.766,44	6.218,47	0,605588
06-02-2024	15.783,52	6.313,93	0,599967

Berdasarkan Tabel 2 model LSTM tunggal menghasilkan prediksi dengan deviasi yang sangat besar dari nilai aktual, terlihat dari jarak antara prediksi dengan nilai kurang lebih 6.000 dan nilai aktual dengan nilai kurang lebih 15.800. Hal ini menunjukkan bahwa model LSTM tunggal mengalami kegagalan dalam mempelajari pola data kurs secara efektif.

3.1.3 Evaluasi Kinerja Model XGBoost Tunggal

Model XGBoost menunjukkan prediksi yang cenderung stagnan, kurang adaptif terhadap perubahan harian nilai tukar. Hal ini disebabkan oleh keterbatasan model dalam memahami pola sekuensial time series, meskipun fitur teknikal dan makro telah disertakan.

Tabel 3. Sampel Data Hasil Uji Model XGBoost Tunggal

Tanggal	Kurs Aktual	Kurs Prediksi	Margin Error ((Aktual-Prediksi)/Terakhir)
01-02-2024	15.882,01	15.067,27	0,051300
02-02-2024	15.853,88	15.067,27	0,049616
03-02-2024	15.824,73	15.067,27	0,047210
04-02-2024	15.795,59	15.067,27	0,045368
05-02-2024	15.766,44	15.067,27	0,044346

Berdasarkan Tabel 4.11, model XGBoost menunjukkan kelemahan fundamental dengan prediksi yang hampir konstan pada nilai 15.067,27 untuk seluruh periode pengujian. Hal ini mengindikasikan bahwa model gagal menangkap variabilitas dalam data nilai tukar harian dan cenderung memberikan *output* yang statis. Rata-rata margin error sebesar 0,1325 merepresentasikan deviasi sekitar 13,25 persen, meskipun lebih baik dibandingkan model LSTM tunggal yang mengalami kegagalan total, namun masih jauh dari performa model *hybrid*. Grafik *learning curve* XGBoost pada Gambar 5 menunjukkan indikasi *overfitting* yang jelas, di mana *training* RMSE terus menurun sementara *validation* RMSE meningkat, mengindikasikan model kehilangan kemampuan generalisasi terhadap data baru.

3.2 Analisis Performa Model

Penelitian ini membandingkan performa tiga model prediksi nilai tukar Rupiah terhadap Dolar Amerika, yaitu LSTM, XGBoost, dan *hybrid* LSTM-XGBoost. Evaluasi dilakukan menggunakan metrik umum yaitu MAPE, MAE, RMSE, dan R^2 , dengan hasil yang dirangkum pada Tabel 4.

Tabel 4. Perbandingan Hasil Evaluasi Model

Model	MAPE	MAE	RMSE	R^2
Hybrid	1,359	216,488	271,776	0,333
LSTM	60,793	9.693,469	9.696,883	-847,912
XGBoost	5,637	904,861	957,301	-7,274

Model *hybrid* LSTM-XGBoost menunjukkan performa terbaik relatif dengan nilai MAPE terendah sebesar 1,359 persen dibandingkan kedua model lainnya. Meskipun nilai R^2 sebesar 0,333 mengindikasikan bahwa hanya sekitar 33,3 persen variabilitas data dapat dijelaskan oleh model, hasil ini masih jauh lebih baik dibandingkan model LSTM tunggal dan XGBoost tunggal yang menunjukkan nilai R^2 negatif.

Model LSTM tunggal menunjukkan kegagalan signifikan dengan MAPE sebesar 60,793 persen dan R^2 sebesar -847,912, mengindikasikan bahwa model berkinerja jauh lebih buruk dibandingkan menggunakan rata-

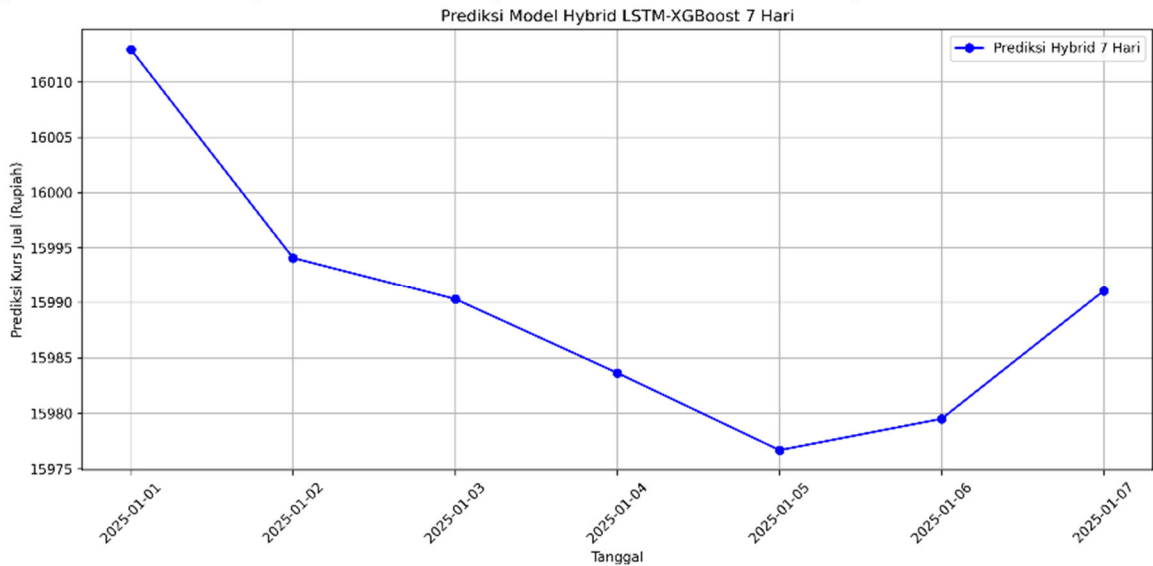
rata sederhana sebagai prediksi. Hal ini menunjukkan bahwa model LSTM tunggal gagal mempelajari pola data secara efektif, kemungkinan disebabkan oleh masalah preprocessing atau arsitektur yang tidak sesuai.

Sementara itu, model XGBoost tunggal memiliki performa yang lebih baik dari LSTM dengan MAPE sebesar 5,637 persen, namun masih menunjukkan keterbatasan signifikan dengan R^2 sebesar -7,274. Hal ini mengkonfirmasi bahwa XGBoost tunggal tidak mampu menangkap dinamika temporal dalam data nilai tukar secara efektif.

Implementasi model *hybrid* terbukti memberikan peningkatan signifikan dibandingkan kedua model tunggal. Meskipun masih memiliki ruang perbaikan yang besar, pendekatan *hybrid* berhasil mengintegrasikan kemampuan temporal LSTM dengan kapabilitas non-linear XGBoost untuk mempelajari residual, menghasilkan prediksi yang relatif lebih stabil dan akurat dalam konteks prediksi nilai tukar Rupiah terhadap Dolar Amerika.

3.3 Prediksi Nilai Tukar 7 Mendatang

Setelah model *hybrid* LSTM-XGBoost terbukti memberikan performa terbaik pada data uji, model ini kemudian digunakan untuk melakukan prediksi nilai tukar Rupiah terhadap Dolar Amerika Serikat selama 7 hari mendatang, dimulai dari 1 Januari 2025 hingga 7 Januari 2025. Prediksi ini bertujuan untuk mengevaluasi potensi penggunaan model *hybrid* dalam prakiraan nilai tukar ke depan, yang dapat dimanfaatkan dalam pengambilan keputusan bisnis atau kebijakan ekonomi. Hasil prediksi tersebut divisualisasikan pada Gambar 6:



Gambar 6. Grafik Prediksi Kurs Model Hybrid

Pada Gambar 6, grafik menunjukkan pola prediksi yang dinamis selama periode 7 hari tersebut. Model memprediksi nilai tukar tertinggi pada hari pertama sebesar sekitar 16.012 Rupiah, kemudian mengalami tren penurunan bertahap hingga mencapai titik terendah pada hari kelima di sekitar 15.976 Rupiah. Setelah itu, model memprediksi adanya pemulihan dengan kenaikan nilai tukar pada dua hari terakhir, mencapai sekitar 15.990 Rupiah pada hari Pola fluktuasi ini menunjukkan kemampuan model dalam menangkap dinamika pasar yang tidak statis, dengan rentang prediksi sekitar 36 Rupiah. selama periode tersebut, yang menunjukkan volatilitas yang relatif terkendali dalam jangka pendek.

Tabel 5. Sampel Data Prediksi Kurs Rupiah

Tanggal	Hasil yang Diharapkan	Kurs Prediksi	Margin Error ((Aktual-Prediksi)/Terakhir)
02-01-2025	16.237,78	16.012,93	0,019802
03-01-2025	16.317,18	15.994,07	0,018885
06-01-2025	16.298,08	15.990,29	0,018578
07-01-2025	16.273,97	15.983,63	0,020109

Dari Tabel 5, terlihat bahwa prediksi dihasilkan dengan rata-rata *margin error* sebesar 0,019343, yang menunjukkan bahwa model memiliki tingkat kesalahan yang relatif rendah terhadap hasil yang diharapkan. Margin error ini merepresentasikan deviasi sekitar 1,93 persen, atau perbedaan nominal sebesar 309,50 Rupiah. Meskipun terdapat selisih, tingkat akurasi ini menunjukkan performa yang dapat diterima untuk prediksi jangka pendek,

dengan tren pergerakan yang tetap dapat diidentifikasi secara akurat oleh model *hybrid*. Hal ini menunjukkan potensi model *hybrid* untuk digunakan dalam prediksi jangka pendek, dengan kombinasi LSTM dan XGBoost yang mampu menghasilkan prediksi yang relatif stabil untuk periode 7 hari ke depan.

4. KESIMPULAN

Dari penelitian ini, Penulis dapat menarik kesimpulan sebagai berikut:

- a. Model *hybrid* LSTM-XGBoost memberikan performa terbaik relatif dalam memprediksi nilai tukar Rupiah terhadap Dolar Amerika Serikat dibandingkan kedua model tunggal. Hal ini ditunjukkan oleh nilai MAPE terendah sebesar 1,359 persen, MAE sebesar 216,488, dan RMSE sebesar 271,776. Meskipun nilai R^2 sebesar 0,333 mengindikasikan bahwa hanya sekitar 33,3 persen variabilitas data dapat dijelaskan oleh model, hasil ini masih jauh lebih baik dibandingkan model LSTM tunggal dan XGBoost tunggal yang menunjukkan nilai R^2 negatif dan tingkat kesalahan yang sangat tinggi
- b. Model LSTM tunggal mengalami kegagalan signifikan dalam memprediksi nilai tukar, ditunjukkan oleh MAPE yang sangat tinggi sebesar 60,793 persen dan R^2 sebesar -847,912. Hal ini mengindikasikan bahwa model berkinerja jauh lebih buruk dibandingkan menggunakan rata-rata sederhana sebagai prediksi, kemungkinan disebabkan oleh arsitektur model yang tidak sesuai dengan karakteristik data.
- c. Model *hybrid* menunjukkan potensi untuk prediksi jangka pendek dengan rata-rata margin error sebesar 1,93 persen pada prediksi 7 hari ke depan. Meskipun masih memiliki ruang perbaikan yang signifikan, pendekatan *hybrid* berhasil mengintegrasikan kemampuan temporal LSTM dengan kapabilitas non-linear XGBoost, menghasilkan prediksi yang relatif lebih stabil dibandingkan implementasi model tunggal..

5. SARAN

Beberapa saran agar model dapat dikembangkan dan dioptimasi lebih baik sebagai acuan untuk melakukan pengembangan atau penelitian yang akan datang adalah sebagai berikut:

- a. Model *hybrid* ini masih dalam tahap prototipe dan dapat dikembangkan lebih lanjut untuk integrasi dalam sistem prediksi *real-time* atau aplikasi keuangan.
- b. Variabel input yang digunakan masih terbatas pada DXY, inflasi, dan BI *7-Day Reverse Repo Rate*. Ketiganya belum cukup merepresentasikan kompleksitas pergerakan nilai tukar. Disarankan penambahan indikator global seperti harga komoditas, indeks volatilitas, dan sentimen ekonomi agar model lebih adaptif terhadap faktor eksternal.
- c. Arsitektur model *hybrid* saat ini masih memiliki ruang perbaikan yang signifikan dengan R^2 hanya 0,333, menunjukkan bahwa 66,7% variabilitas data belum dapat dijelaskan. Disarankan implementasi *hyperparameter tuning* yang komprehensif menggunakan Bayesian Optimization untuk mengoptimalkan performa kedua komponen model.

UCAPAN TERIMA KASIH

Pada kesempatan ini penulis mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang telah memberi bantuan, dukungan, bimbingan, petunjuk, saran maupun dorongan moril, sehingga penelitian ini dapat terselesaikan dengan baik.

DAFTAR PUSTAKA

- [1] Abed, Ahmed. M., Ahmed Fathy, Radwa A. El Behairy dan Tamer S Gaafar. (2025). Jidoka-DT Simulator Programmed By Hybridize XGBoost-LSTM To Evaluate Helmets Quality Produced By Rice-Straw-Alumina Plastic Dough To Resist Shocks And Impenetrable. *Results in Eginering*, 2-18.
- [2] Joseph, Manu. (2022). *Modern Time Series Forecasting with Python*. Birmingham: Packt Publishing.
- [3] Herawati, Sri, Novi Prastiti, dan Imamah. (2024). *Forecasting Bisnis Teori dan Penerapannya*. Tasikmalaya: Perkumpulan Rumah Cemerlang Indonesia.
- [4] Geetha, T. V., dan S. Sendhilkumar. (2023). *Machine Learning: Concepts, Techniques and Applications*. Boca Raton: CRC Press.
- [5] Mirtaheri, Seyedeh Leili, dan Reza Shahbazian. (2022). *Machine Learning: Theory to Applications*. Boca Raton: CRC Press.
- [6] Setyawan, Setu. (2021). *Akuntansi Keuangan Lanjutan: Pengukuran, Pencatatan, dan Pelaporan Transaksi Khusus*. Malang: UMMPress.

- [7] Harahap, Muhammad Arfan, dan Muhammad Hafizh. (2020). *Manajemen Keuangan: Konsep Dasar dan Prinsip Prinsip*. Medan: Merdeka Kreasi Group.
- [8] Kapoor, Vivek, dan Shubhamoy Dey (2021). *Genetic Algorithms and Applications for Stock Trading Optimization*. Pennsylvania: IGI Global.
- [9] Gera, Venkateswara Rao, Padamata Ramesh Babu, Kalyankumar Dasari, dan Shaik Mohammed Jany (2024). *Data Science and Analytics*. Tamil Nadu : Leilani Katie Publication.
- [10] Altekari, dan Rahul V. (2023). *Supply Chain Management: Concepts and Cases, Second Edition*. Delhi: PHI Learning Pvt. Ltd.
- [11] Aggarwal, Charu. C. (2023). *Neural Networks and Deep Learning: A Textbook*. New York: Springer.
- [12] Chin, Kin Ee. (2023). *The Deep Learning Architect's Handbook: Build and Deploy Production-ready DL Solutions Leveraging the Latest Python Techniques*. Britania Raya: Packt Publishing.
- [13] Natarajan, Gnanasankaran, Peturu Raj Chelliah, dan Sundaravadivazhagan Balasubramanian. (2024). *Intelligent Robots and Drones for Precision Agriculture*. Berlin: Springer.
- [14] Deng, Der Jiunn., Han Chieh Chao, dan Jyh Cheng Chen. (2023). *Smart Grid and Internet of Things: 6th EAI International Conference*. Taiwan: ICST Institute for Computer Sciences, Social Informatics and Telecommunications Engineering.