

IMPLEMENTASI MODEL *GATED RECURRENT UNIT (GRU)* ATAU *EXTREME GRADIENT BOOSTING (XGBOOST)* UNTUK PREDIKSI HARGA *CRYPTOCURRENCY ETHEREUM*

Muhammad Fakhrol Reza¹, Ghufron²

^{1,2}Teknik Informatika, Universitas Islam Sultan Agung, Indonesia

mufaza192603@gmail.com¹, ghufron@unissula.ac.id²

Received:10-08-2025

Revised: 20-08-2025

Approved: 27-08-2025

ABSTRAK

Tingginya volatilitas harga *Ethereum* menciptakan tantangan signifikan bagi investor dan menuntut adanya model prediksi yang akurat. Tujuan penelitian ini adalah untuk mengimplementasikan dan membandingkan secara komprehensif dua paradigma *machine learning* yang berbeda *deep learning* sekuensial dan *ensemble learning* tabular untuk memprediksi harga *Ethereum*. Metode penelitian dimulai dari pengumpulan data historis harian (*OHLCV*), yang kemudian melalui tahap pra-pemrosesan, termasuk normalisasi *Min-Max Scaler*. Data dibagi secara kronologis menjadi 80% data latih dan 20% data uji. Model *Gated Recurrent Unit (GRU)* dilatih pada data sekuensial untuk menangkap dependensi temporal, sementara model *Extreme Gradient Boosting (XGBoost)* dilatih pada data tabular dengan fitur-fitur yang direkayasa dari data lampau. Kinerja kedua model dievaluasi menggunakan metrik *Root Mean Squared Error (RMSE)* dan *R-squared (R²)*. Hasil penelitian menunjukkan bahwa model *GRU* secara konsisten lebih unggul. *GRU* mencapai skor *RMSE* 101.37 dan *R²* 0.9718, sementara *XGBoost* memperoleh *RMSE* 107.83 dan *R²* 0.9656. Secara kualitatif, visualisasi prediksi *GRU* menunjukkan garis yang lebih halus dan menempel lebih dekat pada harga aktual, sedangkan prediksi *XGBoost* lebih reaktif. Hasil ini mengindikasikan bahwa kemampuan *GRU* dalam menangkap pola deret waktu secara otomatis lebih efektif daripada *XGBoost* yang berbasis rekayasa fitur. Kesimpulannya, model *GRU* terbukti lebih efektif dan andal untuk kasus prediksi harga *Ethereum* dalam penelitian ini.

Kata Kunci: *Prediksi, Ethereum, Gated Recurrent Unit, Extreme Gradient Boosting, Root Mean Squared Error.*

PENDAHULUAN

Perkembangan teknologi semakin menghasilkan inovasi yang canggih. Salah satunya adalah teknologi *blockchain* (Mufrihah & Najmudin, 2024). Teknologi *Blockchain* adalah teknologi inovatif yang memungkinkan pengguna untuk berkomunikasi tanpa memerlukan perantara yang tepercaya (Gupta, 2022). Teknologi *Blockchain* menarik minat komunitas penelitian pada tahun 2008 ketika sebuah white paper diterbitkan oleh Satoshi Nakamoto tentang masalah pengeluaran ganda dalam jaringan desentralisasi peer-to-peer (Kushwaha et al., 2022).

Dalam dunia investasi, teknologi *blockchain* ini juga digunakan dalam aset digital berupa *cryptocurrency* (Syarif Aminul Khoiri & Abdul Wahid, 2024). *Cryptocurrency* ini memiliki keunggulan, seperti sistem pengelolaan yang terdesentralisasi, tingkat keamanan tinggi yang menjaga kerahasiaan, dan kemudahan dalam proses transaksi (Satria Andromeda & Anisa Sri Winarsih, 2025). Berdasarkan Peraturan Badan Pengawas Perdagangan Berjangka Komoditi (Bappebti) Nomor 7 Tahun 2020, perdagangan aset kripto telah memperoleh legalitas di Indonesia, di mana peraturan tersebut menetapkan daftar aset kripto yang sah untuk ditransaksikan di pasar fisik (Rivaldo et al., 2025).

Tingginya minat masyarakat Indonesia terhadap investasi *cryptocurrency*, yang kini telah mencapai lebih dari 22,1 juta investor terdaftar di Bappebti, berhadapan langsung dengan masalah fundamental yaitu volatilitas harga yang ekstrim (Rahmat &

Iswardhani, 2025). Aset utama seperti Ethereum, meskipun populer, memiliki fluktuasi harga tajam yang menciptakan risiko kerugian signifikan, terutama bagi investor ritel yang sering kali tidak memiliki akses ke alat analisis pasar yang canggih (Arta Pangaribuan et al., 2023). Kesenjangan antara partisipasi pasar yang masif dengan minimnya alat bantu analisis inilah yang menjadi masalah utama dalam penelitian ini, menciptakan urgensi untuk mengembangkan model prediksi yang andal guna mendukung pengambilan keputusan investasi (Dhea Larasati & Primandari, 2021).

Untuk menjawab permasalahan tersebut, penelitian ini mengusulkan implementasi dan perbandingan dua paradigma *machine learning* yang berbeda sebagai solusi (Krittanawong et al., 2020). Pendekatan pertama adalah Deep Learning melalui model *Gated Recurrent Unit* (GRU), yang dirancang untuk secara otomatis mengenali pola dari urutan data deret waktu (Prayogi dkk., 2024). Pendekatan kedua adalah *Ensemble Learning* melalui *Extreme Gradient Boosting* (XGBoost), yang unggul dalam memodelkan hubungan kompleks dari fitur-fitur yang direkayasa secara eksplisit (Nusaiba Yulisa et al., 2023).

Dengan membandingkan kedua pendekatan ini, tujuan penelitian adalah untuk mengidentifikasi metodologi yang paling akurat dan efektif dalam memprediksi harga Ethereum (Satria Andromeda & Anisa Sri Winarsih, 2025). Hasil dari penelitian ini diharapkan dapat menjadi dasar pengembangan sebuah alat bantu analisis yang mampu memberikan gambaran tren harga di masa depan, sehingga dapat membantu investor dalam memitigasi risiko di pasar *cryptocurrency* yang sangat dinamis (Moch Farryz Rizkilloh & Sri Widiyanesti, 2022).

Berdasarkan penelitian sebelumnya tentang penggunaan metode GRU untuk memprediksi harga saham Coca-Cola dengan data harga penutupan menunjukkan hasil yang baik. Penelitian ini menggunakan *window-width* 3, jumlah *neuron* 64, *batch size* 32, dan *epoch* sebanyak 30, dengan rasio data latih dan uji 80:20. Model dikompilasi menggunakan *optimizer* Adam, *loss function* *mean squared error* (MSE), dan evaluasi menggunakan *mean absolute error* (MAE). Hasilnya, model mencapai nilai MAE sebesar 0,42, RMSE 0,64, dan MSE 0,40 (Silalahi & Muljono, 2024).

Dalam penelitian lain teknik data mining yang diterapkan melalui algoritma *Random Forest* dan XGBoost menunjukkan hasil prediksi yang akurat dan konsisten. Penelitian ini menggunakan dataset klinis dan biokimia yang diperoleh dari situs Kaggle, terdiri dari 768 data dengan 9 indikator. Setelah melalui proses *preprocessing* termasuk penanganan data hilang, penanganan *outlier*, dan normalisasi diperoleh 688 data yang layak untuk dianalisis. Selanjutnya, dilakukan proses pelatihan dan pengujian menggunakan teknik *Cross Validation*, serta pencarian parameter terbaik untuk masing-masing model. Evaluasi kinerja model dilakukan menggunakan metrik akurasi, presisi, *recall*, dan *F1-score*. Hasil evaluasi menunjukkan bahwa model memiliki performa yang baik, dengan tingkat akurasi sebesar 74% untuk *Random Forest* dan 76% untuk XGBoost (Salsabil Muhammad et al., 2024).

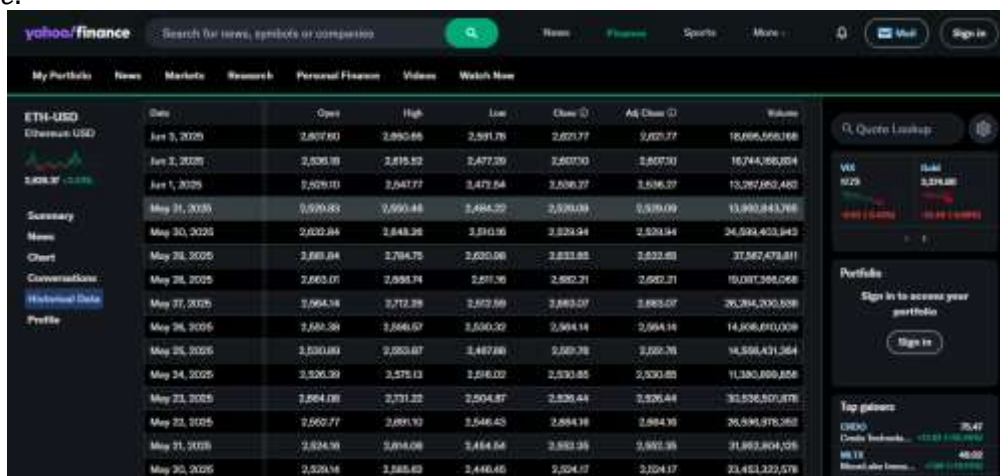
Pada penelitian lain tentang optimasi model XGBoost terhadap data harga saham PT United Tractors Tbk. menunjukkan bahwa model yang dihasilkan memiliki tingkat akurasi yang baik. Model terbaik diperoleh dengan kombinasi *hyperparameter* *gamma* sebesar 0,01, *learning rate* sebesar 0,05, *max_depth* sebesar 15, dan jumlah estimasi pohon (*n_estimators*) sebanyak 200. Berdasarkan hasil eksperimen, model ini menghasilkan nilai *Mean Absolute Percentage Error* (MAPE) sebesar 3,89%. Dengan nilai tersebut, model dapat dikategorikan sebagai model yang akurat dalam melakukan prediksi (Astutiningsih et al., 2023).

METODE PENELITIAN

Metode penelitian digunakan untuk merencanakan, melaksanakan, dan menganalisis penelitian. Metode penelitian ini dapat membantu dalam merancang prosedur yang tepat untuk menyusun data bermanfaat, ini adalah beberapa metode penelitian yang digunakan:

1. Pengumpulan Data

Pada proses ini adalah pengumpulan data yang diperlukan oleh peneliti. Sumber data pada penelitian ini menggunakan data historis harga ethereum yang diperoleh dari sumber *yahoo finance* atau bisa diakses di <https://finance.yahoo.com>. Data historis yang digunakan berisi informasi berupa data *date, open, high, low, close, dan volume*.



ETH-USD	Date	Open	High	Low	Close	Adj. Close	Volume
Ethereum USD	May 1, 2025	2,807.80	2,890.86	2,591.78	2,621.77	2,621.77	18,096,566,168
	May 2, 2025	2,836.18	2,815.52	2,477.28	2,607.90	2,607.90	15,744,588,024
	May 3, 2025	2,829.10	2,947.77	2,472.54	2,598.37	2,598.37	13,287,082,482
	May 4, 2025	2,820.85	2,893.48	2,484.22	2,593.08	2,593.08	13,993,843,788
	May 5, 2025	2,822.84	2,848.26	2,510.38	2,523.94	2,523.94	24,534,453,343
	May 6, 2025	2,881.84	2,784.75	2,620.08	2,633.88	2,633.88	27,587,673,811
	May 7, 2025	2,883.01	2,888.74	2,617.38	2,682.21	2,682.21	18,081,388,068
	May 8, 2025	2,884.18	2,712.38	2,612.88	2,683.07	2,683.07	20,284,200,888
	May 9, 2025	2,880.38	2,886.07	2,530.32	2,584.14	2,584.14	14,808,610,009
	May 10, 2025	2,830.88	2,883.87	2,487.88	2,588.28	2,588.28	14,888,431,384
	May 11, 2025	2,828.39	2,575.13	2,516.02	2,570.85	2,570.85	11,380,888,856
	May 12, 2025	2,884.08	2,731.22	2,504.87	2,538.44	2,538.44	21,836,820,878
	May 13, 2025	2,880.77	2,881.10	2,546.42	2,684.18	2,684.18	28,638,978,388
	May 14, 2025	2,824.38	2,814.08	2,484.54	2,582.28	2,582.28	21,883,884,028
	May 15, 2025	2,828.14	2,888.82	2,446.46	2,524.17	2,524.17	23,483,222,878

Gambar 1 Data Historis Ethereum USD di Yahoo Finance

Pada gambar 1 merupakan tampilan antarmuka dari website yahoo finance. Seperti yang ditunjukkan pada gambar, halaman yang diakses adalah bagian "*Historical Data*", yang menyajikan data deret waktu harian. Setiap baris pada tabel mewakili satu hari perdagangan dan terdiri dari beberapa kolom (fitur) esensial yang digunakan dalam penelitian ini, yaitu *Date, Open, High, Low, dan Volume*.

2. Preprocessing Data

Tahapan ini adalah proses mengolah data mentah menjadi data yang siap digunakan untuk pemodelan dengan tujuan untuk menghasilkan data yang lebih akurat (Aprianto et al., 2025). Data *preprocessing* meliputi tahapan sebagai berikut :

a. Penanganan *Missing Value*

Untuk mengidentifikasi keberadaan nilai yang hilang secara sistematis, penelitian ini menggunakan pustaka *pandas* pada Python. Prosesnya adalah dengan menerapkan fungsi `.isnull().sum()`. Fungsi ini akan memindai seluruh *DataFrame* dan menghasilkan sebuah objek baru dengan ukuran yang sama, di mana setiap sel yang berisi nilai hilang akan ditandai sebagai `True`, dan yang berisi data akan ditandai sebagai `False`. Hasilnya adalah sebuah daftar yang menunjukkan nama setiap kolom beserta jumlah total nilai yang hilang di dalamnya.

b. Penanganan *Data Duplikat*

Tujuan dari tahap penanganan data duplikat ini adalah untuk memastikan bahwa setiap baris data dalam dataset bersifat unik. Keberadaan data duplikat dapat menyebabkan bias pada model *machine learning*, karena model akan memberikan bobot yang tidak semestinya pada data yang berulang. Hal ini juga dapat mengarah pada hasil evaluasi yang terlalu optimis dan tidak mencerminkan kinerja model di

dunia nyata. Implementasi untuk menangani data duplikat dilakukan menggunakan pustaka pandas dengan fungsi `.duplicated().sum()`.

c. Normalisasi Data

Normalisasi data merupakan teknik krusial dalam tahap pra-pemrosesan, terutama karena setiap variabel dalam dataset umumnya memiliki skala nilai yang berbeda. Tujuan dari normalisasi adalah untuk mengubah skala nilai dari berbagai fitur ke dalam rentang yang seragam. Dalam penelitian ini, peneliti menerapkan metode *min-max scaler* untuk melakukan normalisasi, yaitu dengan mengubah nilai-nilai data ke dalam rentang tertentu umumnya antara 0 hingga 1 agar setiap variabel berada pada skala yang sebanding dan dapat dibandingkan secara seimbang (Salsabil Muhammad et al., 2024). Transformasi ini dilakukan dengan menggunakan rumus matematis sebagai berikut:

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

Dimana :

X_{scaled} adalah nilai data setelah dinormalisasi.

X adalah nilai data asli.

X_{min} adalah nilai minimum dari seluruh data pada fitur tersebut.

X_{max} adalah nilai maksimum dari seluruh data pada fitur tersebut.

d.Split Data

Tahap ini merupakan prosedur fundamental dalam pengembangan model *machine learning* untuk mengevaluasi kinerja model secara objektif. Tujuan utama dari pembagian data adalah untuk melatih model pada satu set data (data latih) dan kemudian menguji seberapa baik model tersebut dapat melakukan generalisasi pada data baru yang belum pernah dilihat sebelumnya (data uji). Untuk data deret waktu (*time series*) seperti harga Ethereum, pembagian data tidak boleh dilakukan secara acak. Pembagian harus dilakukan secara kronologis untuk menjaga dependensi temporal data. Hal ini mensimulasikan skenario dunia nyata di mana kita menggunakan data masa lalu untuk memprediksi data di masa depan. Dalam penelitian ini, rasio pembagian yang digunakan adalah 80:20, dengan rincian yaitu 80% sebagai data latih (*training data*) dan 20% sebagai data uji (*testing data*).

2. Pembuatan Model

Tahap pembuatan model merupakan inti dari penelitian ini karena pada bagian ini dilakukan implementasi dua pendekatan pemodelan berbeda, yaitu model GRU (*Gated Recurrent Unit*) yang berbasis *deep learning* dan model XGBoost (*Extreme Gradient Boosting*) yang merupakan algoritma *machine learning* berbasis pohon keputusan. Tujuan dari tahap ini adalah membangun kedua model dengan arsitektur dan parameter yang sesuai.

a. Pembuatan Model GRU

Pada tahap ini merupakan langkah untuk pembuatan model *Gated Recurrent Unit* (GRU) setelah proses *preprocessing data*. Berikut merupakan langkah pembuatan model GRU yaitu :

- 1) Membangun Arsitektur Model Arsitektur model dibangun secara sekuensial menggunakan library seperti Keras. Urutan lapisannya adalah sebagai berikut:
 - Lapisan GRU (*GRU Layer*): Sebagai lapisan inti, lapisan ini bertugas untuk mempelajari pola dan dependensi temporal dari data sekuens. Jumlah unit (neuron) di dalamnya ditentukan sebagai *hyperparameter*.
 - Lapisan *Dropout* (*Dropout Layer*): Lapisan ini sering ditambahkan setelah lapisan

GRU untuk regularisasi. Tujuannya adalah mencegah *overfitting* dengan menonaktifkan sebagian *neuron* secara acak selama pelatihan.

- Lapisan *Dense* (*Dense Layer*): Ini adalah lapisan *output* yang menghasilkan nilai prediksi akhir. Karena tujuannya adalah memprediksi satu nilai harga, lapisan ini hanya memiliki 1 unit *neuron*.
- 2) Kompilasi Model Setelah arsitektur di bangun, model perlu dikompilasi. Proses ini melibatkan penentuan dua komponen penting:
 - *Optimizer*: Algoritma yang digunakan untuk memperbarui bobot model berdasarkan data kesalahan. Adam adalah pilihan yang populer dan efisien.
 - *Loss Function*: Fungsi yang mengukur seberapa besar kesalahan model. Untuk masalah regresi seperti prediksi harga, *Mean Squared Error* (MSE) adalah pilihan yang umum digunakan.
 - 3) Melatih Model (*Training*) Langkah terakhir adalah melakukan *training* model menggunakan data *training* yang telah disiapkan. Proses ini dilakukan dengan memanggil fungsi `.fit()`, di mana beberapa parameter ditentukan:
 - *Epochs*: Jumlah berapa kali seluruh dataset latih akan dilewatkan melalui model.
 - *Batch Size*: Jumlah sampel data yang diproses sebelum model memperbarui bobotnya.

b. Pembuatan Model XGBoost

Pada tahap ini merupakan langkah untuk pembuatan model *Extreme Gradient Boosting* (XGBoost) setelah proses preprocessing data. Berikut merupakan langkah pembuatan model XGBoost yaitu :

- 1) Persiapan Data untuk XGBoost: Berbeda dengan GRU yang memproses data sekuensial, XGBoost memerlukan data dalam format tabular (baris dan kolom). Oleh karena itu, data deret waktu perlu diubah menjadi format *supervised learning*. Proses ini biasanya melibatkan pembuatan fitur (*feature engineering*) dari data masa lalu, seperti menggunakan harga penutupan hari sebelumnya (t-1, t-2, dst.) dan nilai indikator teknikal sebagai fitur input untuk memprediksi harga hari ini (t).
- 2) Inisialisasi Model XGBoost: Model XGBoost tidak dibangun lapis demi lapis seperti jaringan saraf. Sebagai gantinya, sebuah objek model diinisialisasi, yaitu *XGBRegressor*, karena tugasnya adalah prediksi regresi (memprediksi nilai kontinu).
- 3) Menentukan *Hyperparameter*: Sejumlah *hyperparameter* utama ditentukan untuk mengontrol kinerja dan kompleksitas model. Beberapa di antaranya adalah:
 - *n_estimators*: Jumlah total pohon keputusan (*decision tree*) yang akan dibangun.
 - *learning_rate*: Mengontrol laju belajar model untuk mencegah *overfitting*.
 - *max_depth*: Kedalaman maksimum dari setiap pohon untuk membatasi kompleksitas.
 - *objective*: Fungsi tujuan yang dioptimalkan, misalnya 'reg:squarederror' untuk masalah regresi.
- 4) Melatih model: Model dilatih menggunakan fungsi `.fit()` pada data latih. Seringkali, teknik *Early Stopping* digunakan selama pelatihan, di mana performa model dipantau pada data validasi. Pelatihan akan berhenti jika tidak ada peningkatan performa setelah beberapa iterasi tertentu untuk mendapatkan model yang optimal.

3. Evaluasi Model

Tahap evaluasi model merupakan bagian krusial dalam proses penelitian karena di sinilah efektivitas kedua model, GRU dan XGBoost, benar-benar diuji

dalam memberikan prediksi harga ethereum. Evaluasi ini tidak hanya bertujuan untuk mengukur akurasi dari masing-masing model, tetapi juga untuk memahami bagaimana kemampuan masing-masing algoritma dalam menangani pola-pola kompleks pada data keuangan yang bersifat *time-series* dan sangat fluktuatif (Nugraha & Ariatmanto, 2025). Metrik evaluasi yang digunakan untuk mengukur seberapa baik model tersebut digunakan dalam memprediksi yaitu dengan menggunakan RMSE dan R²Score. Dari hasil pengujian ini maka akan diberikan kesimpulan untuk masing-masing model yang di uji.

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

Dimana :

y_i : nilai aktual

\hat{y}_i : nilai prediksi

$\frac{1}{n}$: menghitung rata-rata dari kuadrat error

\sum : menjumlahkan semua error dari semua data

Dari perhitungan tersebut memberikan pengertian bahwa semakin kecil RMSE yang dihasilkan maka akurasi model yang digunakan semakin tinggi dan sebaliknya apabila hasil RMSE semakin besar maka nilai akurasi pada model semakin kecil. RMSE ini dipilih karena keunggulan yang dimiliki oleh RMSE dibandingkan dengan perhitungan akurasi yang lain. RMSE dinilai cocok untuk digunakan dalam pengolahan data yang besar (Meriani & Rahmatulloh, 2024).

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y}_i)^2}$$

Dimana :

n jumlah total data

y_i adalah nilai harga aktual pada data ke- i

\hat{y}_i adalah nilai harga prediksi yang dihasilkan oleh model pada data ke- i

\bar{y}_i adalah nilai rata-rata dari seluruh harga aktual

$\sum_{i=1}^n (y_i - \hat{y}_i)^2$ adalah jumlah kuadrat selisih antara nilai aktual dan prediksi (sama seperti pada perhitungan MSE)

$\sum_{i=1}^n (y_i - \bar{y}_i)^2$ adalah jumlah kuadrat selisih total dari data aktual terhadap nilai rata-ratanya

Selain RMSE, penelitian ini juga menggunakan metrik *R-squared* (R^2) atau Koefisien Determinasi untuk mengukur kecocokan model (*goodness-of-fit*). Metrik ini berfungsi untuk mengukur seberapa besar proporsi variasi dari harga aktual Ethereum yang berhasil dijelaskan oleh model prediksi. Nilai R^2 berkisar antara 0 hingga 1 (atau 0% hingga 100%), di mana skor yang lebih tinggi menunjukkan performa yang lebih baik. Secara matematis, R^2 dihitung dengan membandingkan jumlah kuadrat kesalahan prediksi model terhadap jumlah kuadrat total dari data, sehingga memberikan gambaran komprehensif tentang kemampuan prediktif model (Sanjaya, 2024).

4. Visualisasi Hasil

Tahapan visualisasi hasil berfungsi sebagai analisis kualitatif untuk memberikan pemahaman yang intuitif mengenai kinerja model, yang melengkapi metrik kuantitatif seperti RMSE. Implementasinya adalah dengan membuat sebuah grafik garis

menggunakan pustaka seperti Matplotlib, yang membandingkan secara langsung antara pergerakan harga aktual dengan garis prediksi historis yang dihasilkan model pada data uji. Grafik ini juga secara khusus menandai titik prediksi untuk hari berikutnya untuk memberikan konteks visual. Dilengkapi dengan tampilan yang jelas, visualisasi ini secara efektif mempermudah interpretasi akurasi model dan membantu mengidentifikasi di mana model berhasil atau gagal.

HASIL DAN PEMBAHASAN

1. Hasil Pengumpulan Data

Data yang dikumpulkan mencakup rentang waktu dari tanggal 1 Januari 2021 hingga 1 Juli 2025. Rentang waktu yang panjang ini dipilih untuk memastikan model dapat mempelajari berbagai kondisi pasar, termasuk periode tren naik (*bull market*), tren turun (*bear market*), dan periode konsolidasi. Jumlah data yang digunakan adalah sebanyak 1.642 data yang nantinya akan digunakan sebagai data training dan testing.

Setiap baris data mewakili satu hari dan terdiri dari beberapa atribut atau fitur utama yang disajikan pada Tabel 1.

Tabel 1 Dataet Harga Historis Ethereum

Date	Open	High	Low	Close	Volume
	ETH-USD	ETH-USD	ETH-USD	ETH-USD	ETH-USD
01/01/2021	737.7	749.2	719.79	730.36	13652004358
02/01/2021	730.4	786.79	718.1	774.53	19740771179
03/01/2021	774.51	1006.56	771.56	975.5	45200463368
04/01/2021	977.05	1153.18	912.3	1040.23	56945985763
05/01/2021	1041.49	1129.37	986.81	1100.0	41535932781
06/01/2021	1101.0	1209.42	1064.23	1207.11	44699914188
07/01/2021	1208.07	1282.57	1167.44	1225.67	40468027280
08/01/2021	1225.96	1273.82	1076.08	1224.19	44334826666

Fitur-fitur yang diambil meliputi:

- *Open*: Harga pembukaan Ethereum pada hari tersebut.
- *High*: Harga tertinggi pada hari tersebut.
- *Low*: Harga terendah pada hari tersebut.
- *Close*: Harga penutupan pada hari tersebut, yang akan menjadi target prediksi utama.
- *Volume*: Jumlah total Ethereum yang diperdagangkan pada hari tersebut.

1. Hasil Preprocessing Data

a. Mengecek Missing Value

Pengecekan *missing value* dilakukan dengan menghitung jumlah nilai kosong (NaN atau null) pada setiap kolom fitur yang digunakan, yaitu *Open*, *High*, *Low*, *Close*, dan *Volume*. Hasil dari proses pengecekan disajikan pada tabel 2 di bawah ini.

Tabel 2 Handling Missing Value

Price	Ticker	Handling Missing Value
Close	ETH-USD	0
High	ETH-USD	0
Low	ETH-USD	0
Open	ETH-USD	0
Volume	ETH-USD	0

Berdasarkan Tabel 2, dapat dilihat bahwa hasil pengecekan menunjukkan angka 0 untuk semua kolom fitur. Hal ini mengindikasikan bahwa tidak ditemukan adanya nilai yang hilang dalam dataset penelitian. Dengan demikian, tidak diperlukan tindakan lebih lanjut seperti imputasi atau penghapusan data, dan dataset dapat dianggap lengkap serta siap untuk tahap pra-pemrosesan.

b. Mengecek Duplikasi Data

Tabel 3 Mengecek duplikasi data

Price	Ticker	Duplicate Data
Close	ETH-USD	0
High	ETH-USD	0
Low	ETH-USD	0
Open	ETH-USD	0
Volume	ETH-USD	0

Berdasarkan tabel 3 pada tahap ini proses untuk melihat apakah ada duplikasi data dari dataset yang akan digunakan. Tahap ini menghasilkan bahwa data yang akan digunakan tidak terdapat duplikasi data sehingga data tersebut bisa digunakan untuk proses selanjutnya.

c. Normalisasi Data

Dalam penelitian ini, metode *Min-Max Scaler* digunakan dalam proses normalisasi data pada model GRU. Metode ini melakukan penskalaan ulang data ke dalam rentang nilai tertentu, umumnya antara 0 dan 1. Implementasi teknisnya menggunakan fungsi *MinMaxScaler* dari *library Scikit-learn* di *Python*. Penting untuk dicatat bahwa proses fit (mempelajari nilai *Xmin* dan *Xmax*) hanya dilakukan pada data latih (*training data*). Hal ini dilakukan untuk mencegah kebocoran informasi (*data leakage*) dari data validasi dan data uji ke dalam model. Selanjutnya, *scaler* yang sama digunakan untuk mentransformasi data validasi dan data uji.

d. Split data

Tahap ini merupakan prosedur standar dalam pengembangan model *machine learning* yang bertujuan untuk mengevaluasi kinerja model secara objektif pada data yang belum pernah dilihat sebelumnya (*unseen data*).

Tabel 4 Hasil *Splitting Data*

Ticker	Jumlah Data	Splitting Data	
		Data Training	Data Testing
ETH-USD	1.642	1.313	329

Pada tabel 4.6 menunjukkan hasil pembagian data dengan perbandingan 80% untuk data latih dan 20% untuk data uji. Pembagian ini tidak dilakukan secara acak, melainkan berdasarkan urutan waktu. Pendekatan ini sangat penting untuk data deret waktu (*time series*) seperti harga ethereum, karena menjaga dependensi temporal data. Model dilatih menggunakan data dari masa lalu dan diuji kemampuannya untuk memprediksi data di masa depan, yang mensimulasikan skenario penggunaan di dunia nyata. Secara spesifik, 80% data pertama dalam urutan kronologis dialokasikan sebagai

data latih, sementara 20% data sisanya yang merupakan data paling akhir digunakan sebagai data uji.

2. Hasil Pembuatan Model *Gated Recurrent Unit* (GRU) dan *Extreme Gradient Boosting* (XGBoost)
 - a. Pembuatan Model GRU

Tabel 5 Arsitektur model GRU

MODEL	Arsitektur Model		
	<i>Neuron</i>	<i>Drop out</i>	<i>Dense</i>
<i>Gated Recurrent Unit</i> (GRU)	50	0.2	1

Arsitektur model GRU yang dibuat dalam penelitian ini disusun secara sekuensial, dimulai dengan lapisan inti GRU yang terdiri dari 50 unit neuron. Lapisan fundamental ini dirancang untuk memproses data deret waktu dan secara efektif menangkap pola serta dependensi temporal yang kompleks dari data sekuens harga sebelumnya, di mana kapasitas 50 unit dipilih untuk memberikan kemampuan belajar yang memadai tanpa menjadi terlalu rumit. Untuk memastikan model tidak mengalami *overfitting* dan mampu melakukan generalisasi dengan baik pada data baru, setelah lapisan GRU ditambahkan sebuah lapisan *Dropout* dengan laju (rate) 0.2. Lapisan regularisasi ini bekerja dengan cara menonaktifkan secara acak 20% neuron dari lapisan GRU pada setiap iterasi pelatihan, sehingga memaksa jaringan untuk belajar representasi fitur yang lebih kuat. Sebagai tahap akhir, arsitektur ini ditutup dengan sebuah lapisan *Dense* yang berfungsi sebagai lapisan output. Lapisan ini hanya memiliki 1 unit *neuron*, sesuai dengan tujuan penelitian yaitu untuk menghasilkan satu nilai prediksi tunggal yang merepresentasikan harga penutupan (*Close Price*) Ethereum untuk hari berikutnya.

Tabel 6 *Hyperparameter* model GRU

MODEL	<i>Hyperparameter</i>		
	<i>Optimizer</i>	<i>Batch Size</i>	<i>Verbose</i>
<i>GATED RECURRENT UNIT</i> (GRU)	Adam	32	1

Proses pelatihan model dilakukan dengan memanggil fungsi *fit()* pada arsitektur yang telah dikompilasi, dengan konfigurasi parameter yang telah ditentukan secara cermat. Model dilatih selama 70 epoch, yang berarti keseluruhan data latih dilewatkan melalui jaringan saraf sebanyak 70 kali untuk memungkinkan model belajar secara iteratif dan menyempurnakan bobot internalnya guna meminimalkan kesalahan prediksi. Dalam setiap epoch, proses pembelajaran ini dibagi ke dalam batch dengan ukuran 32, artinya pembaruan bobot model dilakukan setelah setiap 32 sampel data diproses, yang membuat prosesnya lebih efisien secara komputasi. Untuk memantau jalannya pelatihan secara langsung, parameter *verbose* diatur ke 1, yang berfungsi untuk menampilkan progress bar di setiap epoch agar kemajuan dan penurunan nilai loss dapat diamati secara real-time.

b. Pembuatann Model XGBoost

Tabel 7 Hyperparameter model XGBoost

MODEL	Hyperparameter					
	objective	n_estimator	learning_rate	max_depth	sub_sample dan colsample_bytree	random_state
XGBoost	reg:squarederror	1000	0.05	5	0.8	42

Model XGBoost diinisialisasi sebagai XGBRegressor karena tugas yang diemban adalah regresi untuk memprediksi nilai kontinu, dengan objective diatur ke 'reg:squarederror' agar model secara eksplisit mengoptimalkan Mean Squared Error (MSE). Untuk membangun model ini, jumlah pohon keputusan (*n_estimators*) ditetapkan sebanyak 1000, dengan laju belajar (*learning_rate*) sebesar 0.05 yang memungkinkan model belajar secara perlahan namun lebih robust. Guna mengontrol kompleksitas dan mencegah overfitting, kedalaman maksimum setiap pohon (*max_depth*) dibatasi hingga 5. Selain itu, teknik regularisasi lebih lanjut diterapkan dengan mengatur parameter *subsample* dan *colsample_bytree* ke 0.8. Hal ini berarti setiap pohon hanya menggunakan 80% sampel data dan 80% fitur secara acak, sehingga menambah keragaman dan kemampuan generalisasi model. Terakhir, *random_state* diatur ke 42 untuk memastikan bahwa hasil pelatihan dapat diulang kembali dengan konsisten.

3. Evaluasi Model

a. Evaluasi Model GRU

Tabel 8 Hasil evaluasi model GRU

Model	Ticker	RMSE	R ²
GRU	ETH-USD	101.37	0.9718

Berdasarkan hasil evaluasi pada data uji, model GRU menunjukkan tingkat performa yang sangat tinggi dan akurat dalam memprediksi harga ethereum (ETH-USD) untuk hari berikutnya. Model ini berhasil mencapai skor *Root Mean Squared Error* (RMSE) sebesar 101.37, yang mengindikasikan bahwa rata-rata kesalahan atau selisih antara harga prediksi dengan harga aktual adalah sekitar \$101.37. Kinerja yang kuat ini dipertegas oleh skor *R-squared* (R²) yang mencapai 0.9718, yang berarti model mampu menjelaskan sekitar 97.18% dari variasi pergerakan harga aktual. Kombinasi dari nilai RMSE yang relatif rendah (menganingat rentang harga Ethereum) dan nilai R² yang sangat mendekati 1 secara bersamaan menunjukkan bahwa model GRU tidak hanya akurat dalam besaran prediksi, tetapi juga sangat andal dalam menangkap pola dan tren data, sehingga membuktikan efektivitasnya dalam prediksi ini.

b. Hasil evaluasi model XGBoost

Tabel 9 Hasil evaluasi model XGBoost

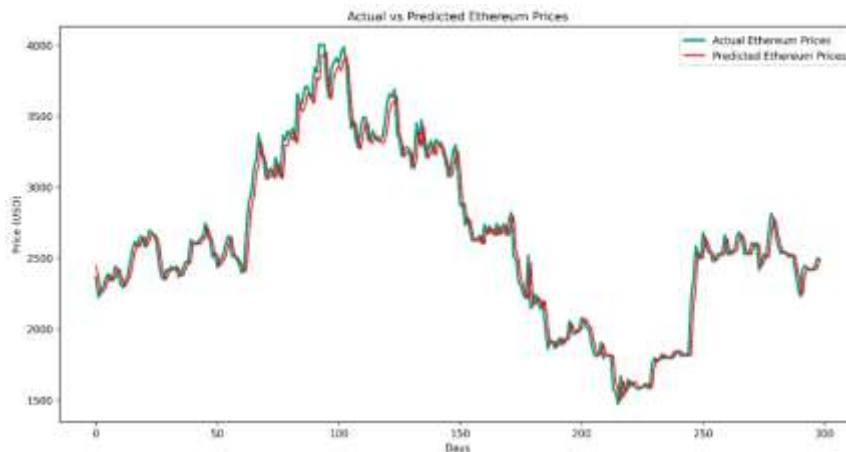
Model	Ticker	RMSE	R ²
XGBoost	ETH-USD	107.83	0.9656

Berdasarkan hasil evaluasi pada data uji, model XGBoost menunjukkan performa prediktif yang baik dalam memprediksi harga ethereum (ETH-USD) untuk hari

berikutnya. Model ini mencatatkan skor *Root Mean Squared Error* (RMSE) sebesar 107.83, yang menginterpretasikan bahwa rata-rata kesalahan prediksi yang dihasilkan adalah sekitar \$107.83 dari harga aktualnya. Selain itu, kinerja model yang kuat ini juga dikonfirmasi oleh skor *R-squared* (R^2) yang tinggi, yaitu sebesar 0.9656. Nilai ini menunjukkan bahwa model XGBoost mampu menjelaskan sekitar 96.56% dari variabilitas pergerakan harga Ethereum. Secara kolektif, kedua metrik ini membuktikan bahwa model XGBoost memiliki tingkat akurasi yang tinggi dan merupakan model yang kompeten, karena tidak hanya memiliki rata-rata kesalahan yang terkendali tetapi juga mampu menangkap sebagian besar pola tren dari data historis.

4. Visualisasi Hasil

a. Visualisasi hasil model GRU



Gambar 2 Hasil grafik prediksi GRU

Gambar 4.1 di atas menyajikan perbandingan visual antara harga penutupan aktual Ethereum (garis hijau) dengan harga yang diprediksi oleh model GRU (garis merah) pada data uji. Secara keseluruhan, grafik tersebut menunjukkan kemampuan model GRU yang sangat baik dalam menangkap dan mengikuti tren pergerakan harga, di mana garis prediksi (merah) secara konsisten bergerak sangat dekat mengikuti pola umum dari garis aktual (hijau), baik saat pasar naik, turun, maupun menyamping, yang mengindikasikan tingkat akurasi yang tinggi. Bahkan pada periode volatilitas tinggi dengan fluktuasi yang tajam, model mampu beradaptasi dengan baik, meskipun pada beberapa puncak dan lembah yang ekstrim dapat diamati adanya sedikit keterlambatan atau *lag* yang wajar terjadi pada model peramalan deret waktu. Visualisasi ini secara efektif mengonfirmasi bahwa model GRU telah berhasil mempelajari dependensi temporal yang kompleks dari data historis, sehingga mampu menghasilkan prediksi yang sangat mendekati pergerakan harga di dunia nyata.

b. Visualisasi hasil model XGBoost



Gambar 3 Hasil grafik prediksi XGBoost

Gambar 4.2 di atas menampilkan perbandingan visual antara harga asli Ethereum (garis biru solid) dan hasil prediksi dari model XGBoost (garis merah putus-putus) pada data uji. Analisis grafik menunjukkan bahwa model XGBoost berhasil menangkap arah pergerakan pasar secara umum, di mana garis prediksinya mampu mengikuti tren utama baik saat pasar naik maupun turun. Meskipun demikian, pada beberapa puncak dan lembah harga yang tajam, terlihat adanya sedikit jarak antara nilai prediksi dan nilai aktual, yang mengindikasikan bahwa model terkadang sedikit *overshoot* atau *undershoot*. Garis prediksi yang dihasilkan juga tampak lebih reaktif atau "patah-patah", yang merupakan karakteristik umum dari model berbasis pohon keputusan.

KESIMPULAN

Kedua model, GRU (*Gated Recurrent Unit*) dan XGBoost (*Extreme Gradient Boosting*), berhasil diimplementasikan dan mampu melakukan prediksi harga Ethereum untuk hari selanjutnya dengan mengikuti tren pergerakan harga aktual secara baik, sebagaimana ditunjukkan pada visualisasi hasil prediksi pada kedua model. Berdasarkan hasil evaluasi menggunakan metrik *Root Mean Squared Error* (RMSE), terdapat perbedaan performa yang jelas antara kedua model. Model GRU menghasilkan nilai RMSE sebesar 101.37 dan nilai R2 sebesar 0.9718, sementara model XGBoost menghasilkan nilai RMSE sebesar 107.83 dan nilai R2 sebesar 0.9656. Nilai RMSE yang lebih rendah pada model GRU menunjukkan bahwa model ini memiliki tingkat akurasi yang lebih tinggi dan rata-rata kesalahan prediksi yang lebih kecil dibandingkan model XGBoost. Sementara itu hasil evaluasi dengan menggunakan nilai R2 pada model GRU juga lebih tinggi untuk mendekati 1 dibandingkan dengan model XGBoost, yang berarti akurasi dari model GRU lebih bagus berdasarkan nilai R2Score. Dengan demikian, dapat disimpulkan bahwa model GRU lebih unggul dan efektif untuk memprediksi harga Ethereum dalam kerangka penelitian ini. Keunggulan ini kemungkinan besar disebabkan oleh kemampuan arsitektur GRU yang lebih baik dalam menangkap pola dan dependensi temporal jangka panjang pada data deret waktu.

DAFTAR PUSTAKA

- Aprianto, K., Mahdiyah, U., & Wulanningrum, R. (2025). Analisis Perbandingan Model PSO-LSTM dan LSTM Konvensional untuk Prediksi Harga Bitcoin di Market Cryptocurrency 1*. In *INOTEK* (Vol. 9).
- Arta Pangaribuan, R., Ufrida Rahmi, N., & prima Indonesia, U. (2023). Risk and Return

- Investment Analysis on Bitcoin, Ethereum, Dogecoin, Litecoin, XRP and LQ45 Share After The Covid-19 Pandemic Analisis Risk Dan Return Investasi Pada Bitcoin, Ethereum, Dogecoin, Litecoin, XRP Dan Saham Saham LQ45 Setelah Pandemi Covid-19. In *Management Studies and Entrepreneurship Journal* (Vol. 4, Issue 3). <http://journal.yrpiaku.com/index.php/msej>
- Astutiningsih, T., Saputro, D. R. S., & Sutanto. (2023). Optimasi Algoritme Xtreme Gradient Boosting (XGBoost) pada Harga Saham PT. United Tractors Tbk. *SPECTA Journal of Technology*, 7(3), 632–641. <https://doi.org/10.35718/specta.v7i3.1031>
- Dhea Larasati, K., & Primandari, A. H. (2021). FORECASTING BITCOIN PRICE BASED ON BLOCKCHAIN INFORMATION USING LONG-SHORT TERM METHOD. In *Parameter: Journal of Statistics* (Vol. 1). www.blockchain.com
- Gupta, B. (2022). Understanding Blockchain Technology: How It Works and What It Can Do. *Metaverse Basic and Applied Research*, 1, 18. <https://doi.org/10.56294/mr202218>
- Krittanawong, C., Virk, H. U. H., Bangalore, S., Wang, Z., Johnson, K. W., Pinotti, R., Zhang, H. J., Kaplin, S., Narasimhan, B., Kitai, T., Baber, U., Halperin, J. L., & Tang, W. H. W. (2020). Machine learning prediction in cardiovascular diseases: a meta-analysis. *Scientific Reports*, 10(1). <https://doi.org/10.1038/s41598-020-72685-1>
- Kushwaha, S. S., Joshi, S., Singh, D., Kaur, M., & Lee, H. N. (2022). Ethereum Smart Contract Analysis Tools: A Systematic Review. In *IEEE Access* (Vol. 10, pp. 57037–57062). Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/ACCESS.2022.3169902>
- Meriani, A. P., & Rahmatulloh, A. (2024). PERBANDINGAN GATED RECURRENT UNIT (GRU) DAN ALGORITMA LONG SHORT TERM MEMORY (LSTM) LINEAR REFRESSION DALAM PREDIKSI HARGA EMAS MENGGUNAKAN MODEL TIME SERIES. *Jurnal Informatika Dan Teknik Elektro Terapan*, 12(1). <https://doi.org/10.23960/jitet.v12i1.3808>
- Moch Farryz Rizkilloh, & Sri Widiyanesti. (2022). Prediksi Harga Cryptocurrency Menggunakan Algoritma Long Short Term Memory (LSTM). *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 6(1), 25–31. <https://doi.org/10.29207/resti.v6i1.3630>
- Mufrihah, W. M., & Najmudin, N. (n.d.). *Transformasi Hukum Dagang Internasional Di Era Teknologi Blockchain Dan Cryptocurrency*. <https://doi.org/10.37817/IKRAITH-EKONOMIKA>
- Nugraha, D. M., & Ariatmanto, D. (2025). Meningkatkan Akurasi Prediksi Harga Bitcoin dengan Algoritma GRU-LSTM Hibrida. 11(1). <https://journal.fkom.uniku.ac.id/index.php/buffer>
- Nusaiba Yulisa, P., Al Haris, M., & Rismawati Arum, P. (2023). Peramalan Nilai Ekspor Migas di Indonesia dengan Model Long Short Term Memory (LSTM) dan Gated Recurrent Unit (GRU). In *J Statistika* (Vol. 16, Issue 1).
- Prayogi, K., Gata, W., & Kussanti, D. P. (n.d.). *Prediksi Harga Saham Bank Central Asia Menggunakan Algoritma Deep Learning GRU*.
- Rahmat, M. R. A., & Iswardhani, I. (2025). Volatilitas Cryptocurrency di Indonesia: Investasi Keuangan Pada Bitcoin Sebelum dan Sesudah Peresmian Bursa Kripto. *SYNERGY: Jurnal Bisnis Dan Manajemen*, 5(1), 65–69. <https://doi.org/10.52364/synergy.v5i1.65>
- Rivaldo, O. :, Tungka, C. A., Sondakh, J., & Mamengko, R. S. (2025). *TINJAUAN HUKUM CRYPTOCURRENCY SEBAGAI ALAT PERTUKARAN DI INDONESIA 1* (Vol. 15, Issue 3).

- Salsabil Muhammad, Azizah Lutvi Nuril, & Ade Eviyanti. (2024). Implementasi Data Mining Dalam Melakukan Prediksi Penyakit Diabetes Menggunakan Metode Random Forest Dan Xgboost. *Jurnal Ilmiah Komputasi*, 23(1). <https://doi.org/10.32409/jikstik.23.1.3507>
- Sanjaya, K. D. (2024). *PREDIKSI HARGA RUMAH DENGAN METODE REGRESI LINEAR DAN SUPPORT VECTOR REGRESSION DI DAERAH TEBET JAKARTA SELATAN* (Vol. 19, Issue 2). Versi Cetak.
- Satria Andromeda, R., & Anisa Sri Winarsih, N. (2025). *Sistemasi: Jurnal Sistem Informasi Perbandingan Kinerja Metode LSTM dan GRU dalam Prediksi Harga Close Cryptocurrency Performance Comparison of LSTM and GRU Methods in Predicting Cryptocurrency Closing Prices*. <http://sistemasi.ftik.unisi.ac.id>
- Silalahi, R. N., & Muljono, M. (2024). Perbandingan Kinerja Metode Linear Regression, LSTM dan GRU Untuk Prediksi Harga Penutupan Saham Coco-Cola. *Komputika : Jurnal Sistem Komputer*, 13(2), 201–211. <https://doi.org/10.34010/komputika.v13i2.12265>
- Syarif Aminul Khoiri, & Abdul Wahid. (2024). Analisis Kinerja Algoritma Machine Learning dalam Prediksi Harga Cryptocurrency. *JUSTINDO (Jurnal Sistem Dan Teknologi Informasi Indonesia)*, 9(2), 133–141. <https://doi.org/10.32528/justindo.v9i2.1965>