BAB II
TINJAUAN PUSTAKA

2.1. Tinjauan Pustaka

2.1.1. Prediksi Harga *Cryptocurrency* Berbasis *Time series*

Pada pendekatan *time series*, metode statistik tradisional seperti ARIMA sering digunakan untuk meramalkan harga aset kripto. Penelitian prediksi pergerakan harga Ethereum [13] menggunakan ARIMA sebagai model statistik dibandingkan dengan model *Machine learning* yang lain, berhasil memberikan akurasi tertinggi dengan 61.17% akurasi prediksi arah harga Ethereum. Namun penelitian ini hanya berfokus pada satu aset kripto (Ethereum) dan menggunakan fitur input yang terbatas hanya pada OHLC (*Open High Low Close*). Meskipun ARIMA tergolong sederhana dan memiliki kebutuhan komputasi ringan, model ini masih dapat mengungguli model *machine learning* seperti Logistic Regression yang mendapatkan akurasi prediksi 56.94%.

Pada penelitian lain [14] ARIMA dikombinasikan dengan variable eksogen (ARIMAX) menggunakan sentiment media sosial (Twitter) untuk meminimalkan *error* prediksi harga bitcoin yang sangat fluktuatif. ARIMAX dengan data sentiment berhasil mengungguli model RNN-LSTM, dibuktikan dengan nilai *mean squre error* (MSE) yang lebih rendah pada ARIMAX sebesar 0.0003 dan RNN-LSTM 0.0013. Namun penelitian ini masih terbatas pada satu aset kripto (Bitcoin) dan menggunakan fitur input terbatas pada BTC weighted price. Meskipun ARIMAX tergolong sederhana, model ini dapat membaca pola musiman atau jangka pendek dengan cukup efektif bahkan lebih akurat daripada model *deep learning* seperti RNN.

Adapun penelitian serupa [15] membandingkan 3 algoritma ARIMA, FBProphet, dan XGBoosting untuk memprediksi harga Bitcoin berdasarkan data historis harian dari Kaggle. Hasil menunjukan ARIMA berhasil menjadi model prediksi terbaik dengan skor RMSE 322 dan skor MAE 227. Meskipun demikian, penelitian ini hanya berfokus pada satu aset kripto (Bitcoin) serta menggunakan

fitur terbatas tanpa indikator teknikal lain seperti RSI, MACD, atau Stochactic. Selain itu, belum dilakukan perbandingan terhadap model *deep learning* dan metrik evaluasi non-fungsional seperti waktu komputasi dan lain-lain.

Terdapat penelitian yang menganalisa model ARIMA untuk memprediksi harga bitcoin yang fluktuatif [16], [17]. Hasil penelitian ini menunjukkan bahwa model ARIMA(5,2,1) menunjukkan performa lebih baik untuk peramalan jangka pendek. Metrik evaluasi (RMSE, MAE, MAPE, MASE) menunjukkan bahwa *error* prediksi berhasil diminimalkan secara signifikan ketika horizon prediksi dipersingkat. Namun penelitian ini hanya berfokus pada pengujian model ARIMA dan belum melakukan perbandingan dengan model *Machine Learning* atau *deep learning* lainnya.

Dalam analisis deret waktu (*time series*), pendekatan ARIMA-GARCH sering digunakan untuk memodelkan keuangan yang bersifat fluktuatif dan memiliki *volatility clustering* [17]. Penelitian menunjukkan bahwa model ARIMA(12,1,12) – GARCH(1,1) memiliki performa prediksi lebih baik dengan nilai RMSE = 14,14, MAE = 12,34, MAPE 32,31%, dibandingkan model tunggal ARIMA(12,1,12) dengan nilai RMSE = 15,38, RMSE= 15,3, dan MAPE = 42,91%. Namun demikian, penelitian ini hanya menggunakan dataset dengan jangka waktu 3 tahun, mulai dari 17 Januari 2021 hingga 17 Desember 2023.

2.1.2. Prediksi Harga *Cryptocurrency* Berbasis *Machine Learning* dan *Deep learning*

Volatilitas tinggi Bitcoin mendorong berbagai penelitian untuk menggunakan pendekatan *Machine learning* (ML) dan *Deep learning* (DL) guna meminimalkan *error* prediksi harga. Terdapat penelitian yang membandingkan model seperti LSTM, GRU, CNN, XGBoost, dan LightBGM untuk memprediksi harga Bitcoin dengan mempertimbangkan indikator ekonomi global seperti harga minyak, indeks dollar, dan volatilitas pasar) [18]. Hasil nya menunjukkan bahwa model berbasis LSTM mampu menangkap pola temporal dengan baik, menghasilkan kesalahan prediksi yang lebih kecil dengan RMSE 0.07 dan MAPE

5.92%. Namun penelitian ini belum menggabungkan ekstraksi fitur seperti CNN untuk mengekstrasi pola harga jangka pendek sebelum dimasukkan ke jaringan LSTM, yang berpotensi meminimalkan *error* prediksi harga Bitcoin.

Terdapat penelitian yang menerapkan beberapa arsitektur *Deep learning* seperti CNN-LSTM, LSTNet, dan TCN untuk memprediksi arah pergerakan Bitcoin menggunakan *on-chain* data tahun 2013-2023 [19]. Hasilnya menunjukkan bahwa kombinasi Boruta + CNN–LSTM menjadi yang terbaik dengan akurasi prediksi arah sebesar 82.44%, F1-Score 0.819, dan AUC 0.824, mengungguli model lain termasuk ARIMA yang hanya mencapai sekitar 50% akurasi. Namun, studi ini masih berfokus pada prediksi arah (*classification*) bukan nilai harga (*regression*), dan belum menggunakan arsitektur yang mampu memperhitungkan dependensi dua arah (*bidirectional*).

Pada penelitian lain dilakukan perbandingan model ARIMA, XGBoost, XB-Prophet, LSTM serta model *hybrid* LSTM-GRU dan LSTM-1D CNN [20]. Menunjukkan hasil bahwa LSTM-GRU memiliki performa terbaik dengan MAE = 0.464 dan RMSE 0.323, hal ini menegaskan keunggulan pendekatan *hybrid* dalam menangkap pola temporal kompleks pada harga kripto yang sangat volatil. Namun, penelitian ini belum mengintegrasikan mekanisme *feature extraction* berbasis CNN yang dapat memperkaya representasi pola jangka pendek sebelum diproses oleh model sekuensial dan belum menggunakan arsitektur yang mampu memperhitungkan dependensi dua arah (*bidirectional*).

Penelitian serupa melakukan perbandingan performa empat model popular seperti XGBoost, LightGBM, ARIMA, dan GARCH [21]. Memprediksi harga tiga aset kripto utama, yaitu Bitcoin, Ethereum, dan Dogecoin, yang dilengkapi dengan berbagai fitur teknikal seperti SMA, MACD, Bollinger Bands, ATR, CCI, dan MFI. Hasil pengujian menunjukkan bahwa model berbasis *Machine learning*, khususnya XGBoost dan LightGBM, memberikan nilai *error* terkecil dengan nilai R^2 mendekati 0.99 dan MAPE di bawah 5%, jauh melampaui model statistik seperti ARIMA(6,1,7) dan GARCH, yang performanya menurun signifikan pada masa volatilitas tinggi (misalnya saat pandemi COVID-19). Hasil penelitian mengindikasikan bahwa model *boosting* lebih efektif dalam menangani *non-*

*linear*itas dan ketidakstabilan pasar kripto. Namun, penelitian ini masih terbatas pada model statistik dan *machine learning* tanpa melakukan perbandingan dengan model *deep learning*.

Penelitian lain menggunakan XGBoost Regressor dengan indikator teknikal seperti EMA, MACD, RSI, Momentum, dan Stochastic Oscillator untuk melakukan prediksi pada harga Bitcoin [22]. Hasil penelitian menunjukkan performa model yang sangat tinggi dengan MAE = 46.22, RMSE = 59.95, dan R^2 = 0.9999, hal ini mengonfirmasi bahwa model *boosting* seperti XGBoost sangat efektif menangani *non-linear*itas dan volatilitas ekstrem pasar kripto. Namun, penelitian ini masih berfokus pada regresi berbasis fitur statistik tanpa ekstraksi pola dan perlunya membandingkan model *deep learning* lainnya.

2.2. Celah Penelitian

Tinjauan terhadap berbagai penelitian terdahulu menunjukkan bahwa metode prediksi harga *cryptocurrency* telah berkembang pesat, mencakup pendekatan statistik klasik seperti ARIMA, algoritma *machine learning* seperti XGBoost, hingga arsitektur *deep learning hybrid* seperti CNN-BiLSTM.

Meskipun hasil-hasil penelitian tersebut memberikan kontribusi signifikan terhadap pengembangan model prediktif, masih terdapat sejumlah kesenjangan (research gap) yang belum sepenuhnya terjawab, yaitu:

1) Keterbatasan pada satu jenis aset dan periode analisis: Sebagian besar penelitian terdahulu hanya berfokus pada satu aset, terutama Bitcoin, tanpa memperluas pengujian terhadap aset besar lainnya seperti Ethereum (ETH) dan Binance Coin (BNB). Selain itu, periode data yang digunakan umumnya relatif pendek (1 - 2 tahun), sehingga belum mampu merepresentasikan dinamika harga *cryptocurrency* yang sangat volatil pada rentang waktu panjang (multi-tahun).
2) Kurangnya penelitian komparatif lintas pendekatan (statistik–ML–DL): Banyak penelitian hanya membandingkan dua kategori model, misalnya ARIMA vs LSTM atau Random Forest vs XGBoost. Hingga saat ini masih terbatas studi yang melakukan analisis komparatif

langsung antara tiga pendekatan sekaligus (ARIMA, XGBoost, dan CNN-BiLSTM) dengan kondisi dataset dan indikator yang seragam. Padahal, perbandingan seperti ini penting untuk menentukan model paling adaptif terhadap karakteristik non-stasioner dan pola *non-linear* pada data crypto.

3) Minimnya evaluasi non-fungsional (komputasi dan efisiensi): Sebagian besar literatur hanya menitikberatkan pada metrik fungsional seperti MAE, MAPE, RMSE, atau R^2. Aspek non-fungsional seperti waktu pelatihan (*training time*), waktu inferensi (*inference speed*), serta efisiensi sumber daya komputasi belum banyak dikaji, padahal faktor tersebut krusial untuk implementasi sistem prediksi *real-time*.

Oleh karena itu, penelitian ini diarahkan untuk mengisi kesenjangan tersebut dengan melakukan analisis komparatif yang komprehensif terhadap tiga model utama, yaitu ARIMA, XGBoost, dan CNN-BiLSTM, menggunakan dataset serta indikator teknikal yang seragam untuk memastikan validitas perbandingan. Penelitian ini juga memanfaatkan data multi-aset (Bitcoin, Ethereum, dan Binance Coin) dengan rentang waktu 2020–2025 agar hasil yang diperoleh lebih representatif terhadap dinamika pasar kripto jangka panjang. Selain menilai aspek fungsional seperti MAPE, RMSE, R^2, penelitian ini turut mengkaji aspek non-fungsional seperti waktu pelatihan (*training time*) dan waktu inferensi (*inference time*) guna menilai efisiensi komputasi dari masing-masing model. Ringkasan

perbandingan penelitian terdahulu dapat dilihat pada Tabel 2.1.

Tabel 2. 1 Tabel Perbandingan Penelitian

Penulis	Objek/Data	Model
Chen dkk [13]	Ethereum, data OHLC harian	ARIMA, Naive Bayes, SVM, RNN
Serafini dkk [14]	Bitcoin, harga + sentimen Twitter	ARIMAX, LSTM
Yang Si [16]	Bitcoin, harga penutupan harian (2017–2022)	ARIMA
Phung Duy dkk [17]	Bitcoin, harga harian (2021–2023)	ARIMA-GARCH
Iqbal dkk [15]	*Cryptocurrency* market, multi-aset	Logistic Regression, SVM, XGBoost, MLP
Kose dkk [18]	Bitcoin, indikator ekonomi global	LSTM, GRU, CNN, TFT, XGBoost, SVR
Omole & Enke [19]	Bitcoin, *on-chain* + strategi trading	CNN, LSTM, Autoencoder
Hota dkk [20]	Bitcoin, harga historis harian	SVM, MLP, KNN, Decision Tree
Shen & Wang [21]	BTC, ETH, DOGE (multi-aset), 2021–2023	ARIMA, Linear Reg., Decision Tree
Hafid dkk [22]	Bitcoin, teknikal indikator (2020–2023)	XGBoost, RSI, MACD, SMA, EMA

2.3. Landasan Teori

2.3.1. Bitcoin & Crptocurrency

Bitcoin adalah salah satu bentuk aset digital yang diperkenalkan pertama kali oleh Satoshi Nakamoto melalui *whitepaper* berjudul "Bitcoin: A Peer-to-Peer Electronic Cash System" pada tahun 2008 [1]. Teknologi dasar dari Bitcoin adalah *blockchain*, yaitu sistem pencatatan transaksi terdesentralisasi yang bekerja secara *peer-to-peer* tanpa memerlukan otoritas pusat. Setiap transaksi dicatat dalam blok yang saling terhubung dan divalidasi oleh jaringan *node* melalui algoritma konsensus *Proof-of-Work* (PoW).

Cryptocurrency secara umum merujuk pada aset digital yang menggunakan kriptografi untuk menjamin keamanan dan transparansi transaksi. Selain Bitcoin, terdapat berbagai mata uang digital lain seperti Ethereum, Binance Coin, dan Dogecoin yang memiliki karakteristik dan tujuan penggunaan berbeda. Ethereum, misalnya, mendukung *smart contract* dan aplikasi terdesentralisasi (*dApps*), sementara Binance Coin banyak digunakan dalam ekosistem *exchange* Binance. Karakteristik utama dari pasar *cryptocurrency* adalah volatilitas harga yang tinggi, likuiditas global, serta fluktuasi yang sangat sensitif terhadap sentimen pasar dan peristiwa ekonomi dunia [2].

Dalam konteks keuangan modern, *cryptocurrency* memiliki potensi sebagai instrumen investasi alternatif, namun juga menyimpan risiko tinggi akibat kurangnya regulasi dan sifatnya yang sangat spekulatif. Fenomena ini menjadi tantangan sekaligus peluang dalam dunia riset, khususnya pada topik prediksi harga aset digital berbasis *time series*, *machine learning*, dan *deep learning*.

2.3.2. Indikator Teknikal

Indikator teknikal merupakan perangkat analisis yang digunakan dalam perdagangan aset keuangan untuk membantu memprediksi arah pergerakan harga berdasarkan data historis [23]. Indikator ini biasanya dihitung dari informasi harga seperti *open, high, low, close*, dan *volume*. Dalam konteks prediksi harga *cryptocurrency*, indikator teknikal berperan penting sebagai variabel input dalam model prediktif karena dapat menggambarkan tren, momentum, dan kondisi pasar secara kuantitatif.

Beberapa indikator teknikal yang umum digunakan dalam penelitian ini antara lain Stochastic Oscillator, Moving Average Convergence Divergence (MACD), dan Volume. Masing-masing indikator memiliki formula dan fungsi analitis tersendiri dalam mendeteksi potensi sinyal beli atau jual. Penjelasan detail mengenai perhitungan dan prinsip kerja indikator-indikator tersebut akan dibahas pada subbagian selanjutnya.

2.3.3. Stochastic

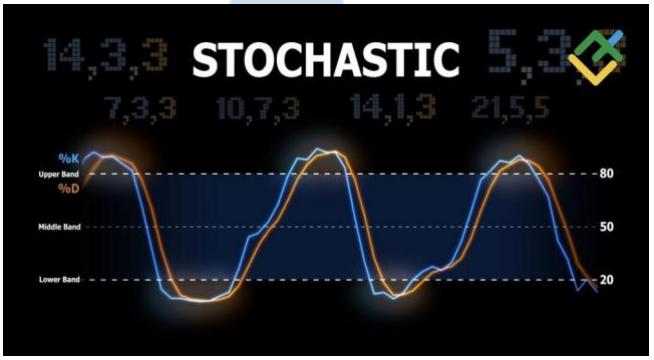
Stochastic Oscillator adalah indikator momentum yang digunakan untuk membandingkan harga penutupan aset terhadap rentang harga dalam periode waktu tertentu. Indikator ini dikembangkan oleh George C. Lane pada tahun 1950-an dan digunakan untuk mengidentifikasi kondisi *overbought* atau *oversold* di pasar. Stochastic terdiri dari dua garis, yaitu %K dan %D, ditunjukan pada persamaan 1 dan persamaan 2 berikut:

$$\%K = \frac{Current\ Close - Lowest\ Low}{Highest\ High - Lowest\ Low} * 100 \quad (1)$$

$$\%D = 3\ day\ moving\ average\ of\ \%K \quad (2)$$

Di mana Close adalah harga penutupan, Low_n adalah harga terendah dalam n hari, dan High_n adalah harga tertinggi dalam n hari. Biasanya, nilai n yang digunakan adalah 14 hari [24]. Nilai indikator berkisar antara 0 hingga 100. Ketika nilai Stochastic berada di atas 80, pasar dianggap berada dalam kondisi *overbought*, sedangkan jika berada di bawah 20, dianggap *oversold*.

Indikator ini sangat berguna dalam perdagangan jangka pendek karena dapat memberikan sinyal pembalikan arah harga dengan cukup cepat. Namun, seperti indikator teknikal lainnya, Stochastic Oscillator sebaiknya tidak digunakan secara tunggal, melainkan dikombinasikan dengan indikator lain untuk menghindari sinyal palsu. Indikator Stochastic terdiri dari %K dan %D, seperti ditunjukkan pada Gambar 2.1.



Gambar 2. 1. Ilustrasi Indikator *Stochactic* (Sumber: www.litefinance.org, 2026)

2.3.4. MACD

Moving Average Convergence Divergence (MACD) adalah indikator teknikal yang digunakan untuk mengidentifikasi perubahan arah tren serta kekuatan momentum. Indikator ini diperkenalkan oleh Gerald Appel pada akhir tahun 1970-an dan telah menjadi salah satu indikator populer di kalangan analis teknikal [25].

MACD terdiri dari tiga komponen utama: MACD line, Signal line, dan Histogram. Rumus MACD ditunjukan pada persamaan 3, 4, dan 5 sebagai berikut:

$$MACD\ Line = EMA(12) - EMA(26)\ (3)$$

$$Signal\ Line = EMA\ (9)\ dari\ MACD\ Line\ (4)$$

$$Histogram = MACD\ Line - Signal\ Line\ (5)$$

EMA mengacu pada Exponential Moving Average dengan periode tertentu. Sinyal beli biasanya muncul saat MACD line memotong signal line dari bawah, dan sinyal jual muncul ketika MACD line memotong signal line dari atas.

MACD sangat efektif dalam mendeteksi crossover dan divergence dalam tren harga. Namun, indikator ini cenderung memberikan sinyal yang terlambat dalam kondisi pasar yang sangat volatil. Komponen utama MACD berupa MACD Line, signal line, dan histogram ditunjukkan pada Gambar 2.2.



Gambar 2. 2. Ilustrasi Indikator *MACD* (Sumber: www.investopedia.com, 2026)

2.3.5. Volume

Volume adalah indikator teknikal yang menunjukkan jumlah total transaksi yang terjadi dalam suatu periode waktu tertentu. Volume mencerminkan minat dan partisipasi pasar. Semakin tinggi volume, semakin kuat sinyal tren harga yang sedang terjadi [23].

2.3.6. *Open Interest*

Open interest (OI) adalah jumlah total kontrak derivatif (misalnya *futures*, *swaps*) yang masih terbuka (belum dieksekusi, dieksekusi, atau diselesaikan) pada suatu waktu tertentu. OI merepresentasikan ukuran posisi terbuka yang masih berjalan. Dalam konteks pasar tradisional (misalnya komoditas atau indeks *futures*) OI menunjukkan bahwa ketika naik bersamaan dengan kenaikan harga, itu bisa menjadi konfirmasi tren yang kuat. Sebaliknya, jika OI naik namun harga turun, bisa mengindikasikan tekanan jual atau potensi pembalikan tren [26].

Di pasar kripto, OI memiliki relevansi besar dikarenakan banyak perdagangan yang terjadi di pasar derivatif menggunakan *leverage*. OI dapat menjadi indikator jumlah dana yang terikat dalam posisi terbuka, dan berpotensi menjadi indikator likuiditas pasar yang akurat.

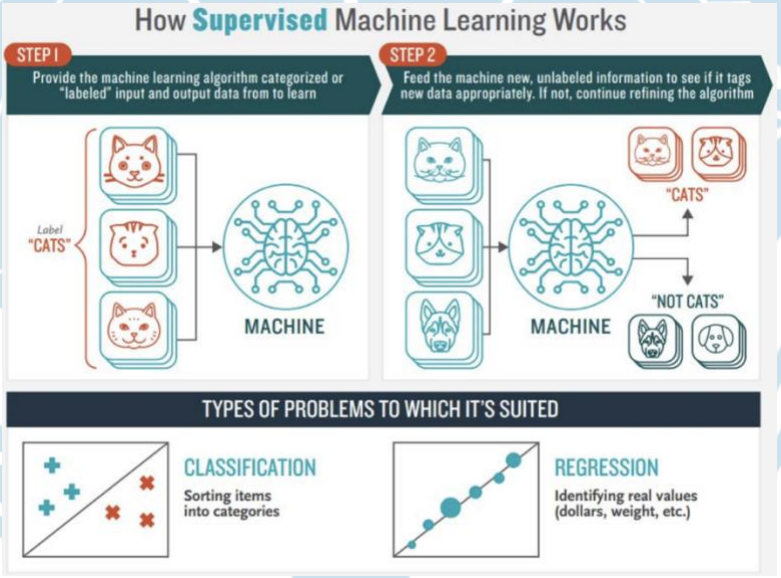
2.3.7. *Supervised Learning*

Supervised learning adalah pendekatan pembelajaran mesin di mana model dilatih menggunakan dataset yang telah diberi label, yaitu data input yang sudah diketahui hasil output-nya. Tujuan dari *supervised learning* adalah membangun fungsi atau pemetaan dari input ke output sehingga dapat digunakan untuk memprediksi hasil dari data baru yang belum pernah dilihat sebelumnya.

Dalam *supervised learning*, dua tugas utama yang umum digunakan adalah klasifikasi dan regresi. Klasifikasi digunakan ketika output yang diprediksi berupa kategori atau kelas, sedangkan regresi digunakan ketika output berupa nilai kontinu,

seperti prediksi harga. Model akan belajar dari pasangan input-output dan meminimalkan *error* melalui fungsi *loss*.

Beberapa algoritma populer dalam *supervised learning* meliputi *Linear Regression*, *Decision Tree*, *Support Vector Machine* (SVM), *k-Nearest Neighbors* (k-NN), dan ensemble methods seperti *Random Forest* dan *XGBoost*. Dalam konteks prediksi harga *cryptocurrency*, metode regresi seperti *XGBoost* banyak digunakan karena kemampuannya dalam menangani data *non-linear* dan memberikan performa tinggi pada dataset yang kompleks [27]. Ilustrasi proses *supervised learning* direpresentasikan pada Gambar 2.3.



Gambar 2. 3. Ilustrasi *Supervised learning* (Sumber: www.medium.com, 2026)

2.3.8. ARIMA

Autoregressive Integrated Moving Average (ARIMA) adalah model statistik klasik yang digunakan untuk memodelkan dan meramalkan data deret waktu. ARIMA cocok untuk data univariat yang memiliki tren atau musiman namun memerlukan stasioneritas [6]. Model ini terdiri dari tiga komponen:

- **AR (p)**: Autoregressive, memodelkan hubungan antara nilai saat ini dan nilai masa lalu.

- **I (d)**: Integrated, menyatakan berapa kali diferensiasi data diperlukan agar stasioner.
- **MA (q)**: Moving Average, menggunakan kesalahan residual dari masa lalu sebagai prediktor.

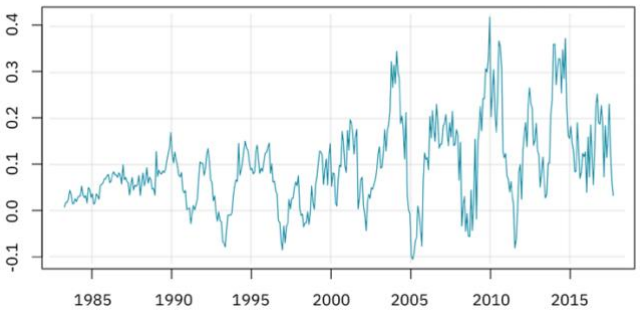
Bentuk umum ARIMA ditulis sebagai ARIMA(p, d, q). Contoh: ARIMA(1,1,1) berarti satu lag AR, satu diferensiasi, dan satu lag MA. Rumus model ARIMA secara matematis ditunjukan pada persamaan 6 berikut:

$$Y_t = c + \phi_1 Y_{t-1} + \cdots + \phi_p Y_{t-p} + \theta_1 \epsilon_{t-1} + \cdots + \theta_q \epsilon_{t-q} + \epsilon_t \quad (6)$$

Dimana:

- Y_t : nilai aktual pada waktu t
- ε_t : *error white noise*
- ϕ, θ: parameter model

Kelebihan utama ARIMA adalah kesederhanaannya dan performa yang baik pada data linier dengan tren jangka pendek. Selain itu, ARIMA relatif ringan secara komputasi dan mudah diinterpretasikan. Namun, ARIMA memiliki keterbatasan dalam menangani data non-stasioner yang kompleks atau bersifat non-linier. Model ini juga tidak cocok untuk memproses data multivariat atau data dengan variabilitas tinggi seperti harga *cryptocurrency* kecuali dilakukan modifikasi atau integrasi model tambahan (misalnya ARIMA-GARCH atau ARIMAX) [8]. Ilustrasi model arima (p, d, q) ditunjukan pada Gambar 2.4.



Gambar 2. 4. Ilustrasi *ARIMA* (Sumber: www.ibm.com, 2026)

2.3.9. *XGBoost*

Extreme Gradient Boosting (XGBoost) adalah algoritma *ensemble learning* berbasis pohon keputusan yang dirancang untuk memaksimalkan kecepatan dan efisiensi prediksi. Algoritma ini bekerja dengan membangun banyak pohon keputusan secara iteratif, di mana masing-masing pohon baru bertujuan memperbaiki kesalahan dari model sebelumnya berdasarkan turunan gradien dari fungsi loss [9]. Rumus umum prediksi pada XGBoost, ditunjukkan pada persamaan 7 berikut:

$$\hat{y}_i = \sum_{k=1}^{K} \mathrm{f_k}(x_i),\ f_k \in F \quad (7)$$

Dengan:

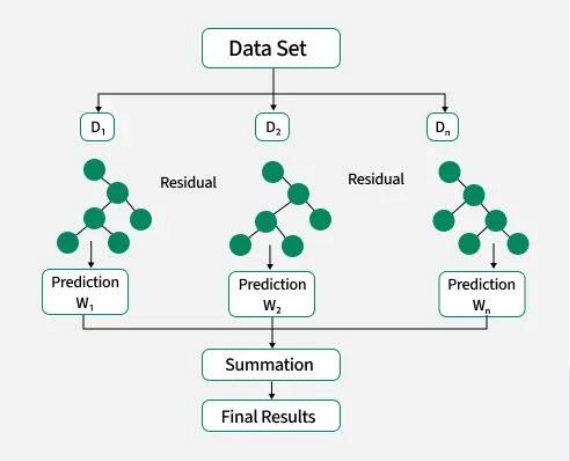
$\hat{y}_i$: output prediksi akhir untuk observasi i

f_k : fungsi dari pohon ke-k

F : ruang fungsi semua pohon keputusan

XGBoost menawarkan berbagai fitur unggulan seperti regularisasi (L1 dan L2) untuk mengurangi *overfitting*, kemampuan menangani data missing secara otomatis, dan efisiensi eksekusi tinggi lewat paralelisasi. Dalam konteks prediksi *cryptocurrency*, XGBoost sering digunakan karena mampu memodelkan hubungan *non-linear* antara variabel teknikal seperti indikator pasar dengan harga aset. Model ini juga fleksibel karena dapat digunakan untuk regresi maupun klasifikasi.

Kelemahan XGBoost antara lain adalah kompleksitas dalam tuning parameter (seperti learning rate, max depth, subsample, dan colsample_bytree) serta potensi *overfitting* jika tidak dikontrol dengan baik, terutama pada dataset kecil. Proses boosting yang membangun banyak pohon secara berurutan pada XGBoost dapat dilihat pada Gambar 2.5.



Gambar 2. 5. Ilustrasi *XGBoost* (Sumber: www.medium.com, 2026)

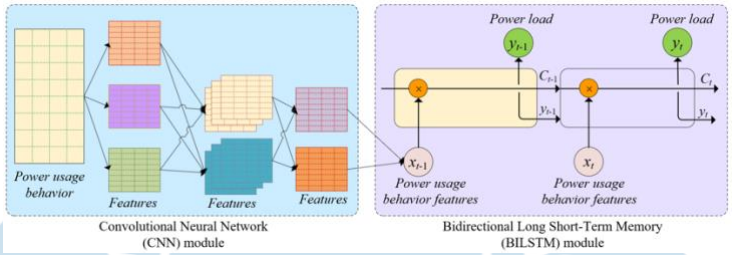
2.3.10. CNN-BiLSTM

CNN-BiLSTM adalah arsitektur *hybrid* dalam *deep learning* yang menggabungkan *Convolutional Neural Network* (CNN) dan *Bidirectional* Long Short-Term Memory (BiLSTM). Model ini didesain untuk mengatasi kelemahan model tradisional dalam memahami pola kompleks pada data deret waktu dengan memperkuat kemampuan deteksi fitur spasial dan temporal secara simultan [28].

CNN digunakan pada tahap awal untuk mengekstrak fitur lokal penting dari input *time series*, seperti pola tren jangka pendek. CNN bekerja dengan menerapkan filter konvolusi ke dalam matriks input, dan output-nya berupa representasi fitur yang dirangkum lewat operasi *pooling*. Setelah tahap ekstraksi fitur oleh CNN, data dilanjutkan ke jaringan BiLSTM. BiLSTM merupakan versi LSTM dua arah yang memproses data dari masa lalu ke depan (*forward*) dan masa depan ke belakang (*backward*). Hal ini memberikan konteks temporal lebih luas yang sangat bermanfaat dalam menangkap dinamika pasar *cryptocurrency* yang cepat berubah.

CNN-BiLSTM sangat unggul dalam mempelajari dependensi jangka pendek dan panjang secara simultan. Selain itu, model ini lebih *robust* terhadap *noise* dan ketidakteraturan data karena sifat adaptif dari CNN dan kemampuan memori jangka panjang dari LSTM. Namun, kelemahannya terletak pada kebutuhan data yang besar, waktu pelatihan yang lama, dan beban komputasi tinggi [10].

Arsitektur *hybrid* CNN-BiLSTM yang digunakan dalam penelitian ini diilustrasikan pada Gambar 2.6.



Gambar 2. 6. Ilustrasi *CNN-BiLSTM* (Sumber: www.mdpi.com, 2026)

2.3.11. Metrik Evaluasi

Dalam pemodelan prediksi regresi seperti harga *cryptocurrency*, evaluasi model sangat penting untuk mengukur seberapa baik hasil prediksi mendekati nilai sebenarnya [29]. Pemilihan metrik evaluasi yang tepat memungkinkan analisis menyeluruh terhadap performa model secara fungsional. Beberapa metrik yang umum digunakan meliputi *R-squared* (R^2), *Root Mean Squared Error* (RMSE), *Mean Absolute Error* (MAE), *Mean Absolute* Percentage *Error* (MAPE), serta *Execution Time* sebagai metrik non-fungsional.

2.3.12. *R-squared* (R^2)

R^2 atau koefisien determinasi digunakan untuk mengukur proporsi variansi dalam variabel dependen yang dapat dijelaskan oleh model prediksi. Nilai R^2 berkisar antara 0 hingga 1, di mana semakin mendekati 1 menunjukkan bahwa model mampu menjelaskan sebagian besar variasi dari data target [29]. Rumus perhitungan R^2 ditunjukan pada persamaan 8 berikut.

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \bar{y})^2} \quad (8)$$

Di sisi lain, R^2 dapat menyesatkan pada model *non-linear* atau jika terjadi *overfitting*, karena model dengan performa buruk tetap bisa menghasilkan nilai R^2 tinggi jika datanya terlalu sederhana. Oleh karena itu, penggunaannya sebaiknya didampingi dengan metrik *error* lainnya.

2.3.13. *Root Mean Squared Error* (RMSE)

RMSE adalah akar dari rata-rata kuadrat selisih antara nilai aktual dan prediksi [29]. RMSE memberikan penalti yang lebih besar terhadap kesalahan besar (*outliers*), sehingga cocok digunakan saat kesalahan ekstrem harus diminimalkan. Rumus perhitungan RMSE ditunjukan pada persamaan 9 berikut.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\mathrm{y_i} - \widehat{y_i})^2} \quad (9)$$

RMSE mempertahankan satuan dari variabel target dan memudahkan interpretasi secara kuantitatif. Namun, sensitivitas terhadap *outlier* menjadi kelemahan utama jika data memiliki *noise* tinggi.

2.3.14. *Mean Absolute Error* (MAE)

MAE menghitung rata-rata dari selisih absolut antara nilai prediksi dan nilai actual [29]. Metode ini tidak memberi penalti lebih besar untuk kesalahan ekstrem, sehingga lebih stabil terhadap *outlier* dibanding RMSE. Rumus perhitungan MAE ditunjukan pada persamaan 10 berikut.

$$MAE = \frac{1}{n}\sum_{i=1}^{n} | \, y_i - \widehat{y_i} \, | \quad (10)$$

MAE sangat mudah diinterpretasikan karena menggunakan satuan yang sama dengan data aktual. Meskipun begitu, MAE kurang sensitif dalam menyoroti kesalahan besar dan tidak mengindikasikan arah kesalahan.

2.3.15. *Mean Absolute Percentage Error* (MAPE)

MAPE adalah metrik yang mengukur rata-rata kesalahan dalam bentuk persentase terhadap nilai actual [29]. MAPE populer digunakan karena mudah dipahami secara intuitif dan membantu perbandingan performa antar model di berbagai skala data. Rumus perhitungan MAPE ditunjukan pada persamaan 11 berikut.

$$MAPE = \frac{100\%}{n} \sum_{i=1}^{n} \left| \frac{\mathrm{y_i} - \widehat{y_i}}{y_i} \right| \quad (11)$$

Namun, MAPE memiliki kelemahan signifikan saat nilai aktual mendekati nol, karena akan menghasilkan nilai *error* yang sangat besar atau tak terdefinisi.

2.3.16. *Execution Time*

Execution time adalah metrik non-fungsional yang mengukur waktu yang dibutuhkan model untuk menjalankan pelatihan (*training*) dan prediksi (*inference*) [27]. Metrik ini sangat penting pada sistem *real-time* seperti pasar aset digital, di mana keterlambatan prediksi dapat berdampak signifikan terhadap keputusan investasi.

Faktor-faktor yang memengaruhi execution time meliputi kompleksitas model, jumlah data, parameter tuning, serta kapasitas komputasi. Misalnya, model *deep learning* seperti CNN-BiLSTM cenderung memiliki execution time yang lebih tinggi dibanding model linear atau *tree-based* karena jumlah layer dan parameter yang jauh lebih besar. Rumus perhitungan execution time ditunjukan pada persamaan 12 berikut.

$$T_{total} = T_{training} + T_{inference} \quad (12)$$