

BAB II

KAJIAN PUSTAKA

Penelitian ini didasari oleh beberapa topik utama yang meliputi teknologi *cryptocurrency*, metode *deep learning*, serta penelitian terdahulu yang relevan dengan prediksi harga *cryptocurrency*. Beberapa topik utama yang akan dibahas meliputi:

2.1 TEORI PENUNJANG

2.1.1 *Cryptocurrency* dan Teknologi *Blockchain*

Cryptocurrency merupakan varian uang digital yang dibangun dengan kriptografi dan teknologi *blockchain* untuk memastikan keamanan, transparansi, serta desentralisasi. Keamanannya didukung oleh tanda tangan digital yang menggunakan kunci privat dan publik, sehingga hanya pemilik sah yang dapat melakukan transaksi. Selain itu, seluruh transaksi dicatat dalam *blockchain*, sebuah sistem pencatatan terdistribusi yang bersifat permanen dan tidak dapat diubah, serta setiap blok transaksi divalidasi melalui mekanisme konsensus yang melibatkan partisipasi node dalam jaringan. Kombinasi kriptografi dan *blockchain* ini menjadikan *cryptocurrency* sangat sulit dimanipulasi dan aman dari gangguan pihak ketiga (Nakamoto, 2008).

2.1.1.1 Bitcoin (BTC)

Bitcoin merupakan *cryptocurrency* pertama yang mengadopsi teknologi *blockchain* yang diciptakan oleh Satoshi Nakamoto pada tahun 2008 dan tetap menjadi aset digital paling bernilai karena memiliki pasokan yang terbatas, yaitu maksimal hanya 21 juta unit. Pembatasan jumlah ini diterapkan melalui mekanisme protokol yang dikenal sebagai *halving*, yaitu pengurangan imbalan penambangan setiap 210.000 blok atau sekitar empat tahun sekali, yang bertujuan untuk mengendalikan inflasi. Dalam perkembangannya hingga saat ini, mekanisme *halving* Bitcoin telah berlangsung beberapa kali pada tahun 2012, 2016, 2020, dan terakhir pada April 2024, yang menurunkan imbalan penambangan dari 6.25 BTC menjadi 3.125 BTC per blok. Selain itu, sifat desentralisasi Bitcoin yang tidak dikendalikan oleh otoritas pusat, serta adopsinya yang semakin luas sebagai *store of value* berfungsi layaknya emas digital, turut memperkuat posisinya sebagai *cryptocurrency* paling dominan hingga saat ini (Nakamoto, 2008).

2.1.1.2 Ethereum (ETH)

Ethereum merupakan platform berbasis *blockchain* yang pertama kali memperkenalkan konsep *smart contract*, yaitu program komputer yang berjalan secara otomatis di atas *blockchain* tanpa memerlukan pihak ketiga. Ethereum memungkinkan pengembangan dan *deployment* aplikasi desentralisasi (*dApps*) yang dapat beroperasi secara transparan, tahan gangguan, dan anti-penipuan. Tidak seperti Bitcoin yang berfokus sebagai mata uang digital, Ethereum dirancang sebagai platform *general-purpose*. Sebagai *cryptocurrency* ETH memiliki utilitas sebagai aset yang digunakan untuk membayar biaya transaksi (*gas fee*) dan mengamankan jaringan melalui mekanisme *Proof of Stake*. Selain itu, mekanisme pembakaran sebagian *base fee* melalui EIP-1559 dapat mengurangi suplai ETH ketika aktivitas jaringan meningkat (Buterin, 2015).

2.1.1.3 Solana (SOL)

Solana adalah *blockchain* dengan fokus pada skalabilitas tinggi dan kecepatan transaksi. Dikenal dengan mekanisme konsensus *Proof of History* (PoH), Solana dapat menangani sekitar 65.000 transaksi setiap detik dengan biaya yang relatif murah. PoH bekerja dengan menciptakan urutan waktu kriptografis untuk mencatat peristiwa dalam *blockchain*, sehingga mengurangi kebutuhan sinkronisasi waktu di antara node dan memungkinkan paralelisasi pemrosesan transaksi. Selain digunakan sebagai *cryptocurrency*, SOL berfungsi sebagai token utama dalam ekosistem Solana untuk membayar biaya transaksi (*gas fee*), melakukan *staking*, serta mendukung eksekusi *smart contract* dan berbagai aplikasi *Decentralized Finance* (DeFi), *Non-Fungible Token* (NFT), maupun aplikasi berbasis *blockchain*. Karakteristik tersebut menjadikan Solana sebagai salah satu *blockchain* yang banyak digunakan untuk aplikasi yang membutuhkan kecepatan transaksi tinggi dan latensi rendah (Li, 2022).

2.1.1.4 Binance Coin (BNB)

Binance Coin (BNB) merupakan *cryptocurrency* yang pertama kali diterbitkan oleh Binance pada tahun 2017 sebagai *utility token* untuk memberikan potongan biaya perdagangan (*trading fee*) di platform Binance. Seiring perkembangan ekosistem Binance, fungsi BNB tidak lagi terbatas sebagai token utilitas, tetapi juga menjadi aset utama pada BNB Smart Chain (BSC). BNB digunakan untuk membayar biaya transaksi

(*gas fee*), melakukan *staking*, serta mendukung berbagai layanan *Decentralized Finance* (DeFi). BNB Smart Chain menawarkan biaya transaksi yang relatif rendah dan waktu konfirmasi yang cepat, sehingga banyak dimanfaatkan dalam pengembangan aplikasi berbasis *blockchain*. Selain itu, Binance secara berkala melakukan mekanisme BNB *Auto-Burn*, yaitu pengurangan jumlah token yang beredar secara permanen untuk menjaga kelangkaan aset dan mendukung stabilitas nilai BNB (Mallick, 2020).

2.1.1.5 Ripple (XRP)

Ripple merupakan perusahaan teknologi finansial yang mengembangkan solusi pembayaran lintas negara berbasis *blockchain*, sedangkan XRP merupakan *cryptocurrency* yang beroperasi pada XRP Ledger (XRPL). XRP dirancang untuk memfasilitasi transfer nilai secara cepat dengan biaya transaksi yang relatif rendah, sehingga dapat digunakan sebagai aset perantara dalam transaksi lintas negara. XRP Ledger menggunakan mekanisme konsensus yang relatif tersentralisasi karena proses validasi transaksi dilakukan oleh sejumlah validator yang saling dipercaya. Sebagai *cryptocurrency*, harga XRP dipengaruhi oleh permintaan dan penggunaan jaringan, kondisi pasar, sentimen investor, serta perkembangan regulasi dan adopsi teknologi pembayaran berbasis *blockchain*. Karakteristik tersebut menjadikan XRP sebagai *cryptocurrency* dengan likuiditas yang relatif tinggi, tetapi tetap memiliki volatilitas harga yang dipengaruhi oleh dinamika pasar *cryptocurrency* (Moreno-Sánchez, 2020).

2.1.1.6 Dogecoin (DOGE)

Dogecoin awalnya dikembangkan sebagai *cryptocurrency* berbasis meme. Meskipun berawal dari lelucon, Dogecoin berhasil membangun komunitas yang kuat dan menjadi populer karena biaya transaksinya yang rendah dan kecepatan konfirmasi yang tinggi. Dogecoin menunjukkan bagaimana kekuatan komunitas dan jaringan sosial dapat mempengaruhi nilai dan penggunaan *cryptocurrency* secara luas (Yadulla, 2024).

2.1.1.7 Regulasi Cryptocurrency

Di Indonesia, Otoritas Jasa Keuangan (OJK) kini memiliki kewenangan untuk mengawasi perdagangan *cryptocurrency*, setelah sebelumnya di bawah pengawasan Badan Pengawas Perdagangan Berjangka Komoditi (BAPPEBTI). Pengalihan ini resmi berlaku sejak 10 Januari 2025 berdasarkan Peraturan Pemerintah Nomor 49 Tahun 2024

dan POJK Nomor 27 Tahun 2024 yang mengatur secara khusus tata kelola serta kerangka hukum perdagangan aset digital di Indonesia (OJK, 2024).

2.1.2 Machine Learning

Machine learning memungkinkan komputer mempelajari pola dari data dan membuat prediksi atau keputusan tanpa diprogram secara eksplisit untuk setiap permasalahan. Model *machine learning* dibangun melalui proses pelatihan (*training*) menggunakan data historis sehingga mampu mengenali hubungan antarvariabel dan menghasilkan prediksi terhadap data baru. Secara umum, *machine learning* dibagi menjadi tiga kategori, yaitu *supervised learning*, *unsupervised learning*, dan *reinforcement learning* (Goodfellow, 2016). Pada penelitian ini digunakan pendekatan *supervised learning*, karena model dilatih menggunakan pasangan data historis harga *cryptocurrency* dan nilai target untuk mempelajari hubungan antara data masukan dan keluaran.

2.1.3 Deep Learning

Deep learning merupakan cabang dari *machine learning* yang memanfaatkan jaringan neural berlapis-lapis untuk mengekstraksi informasi dari data secara hierarkis. Prosesnya dimulai dari lapisan awal yang mengenali fitur sederhana hingga lapisan yang lebih dalam yang mengenali pola kompleks (Bre, 2017). Dalam konteks prediksi deret waktu seperti harga *cryptocurrency*, *deep learning* sangat efektif karena mampu mengenali pola tersembunyi dan hubungan non-linear dalam data historis.

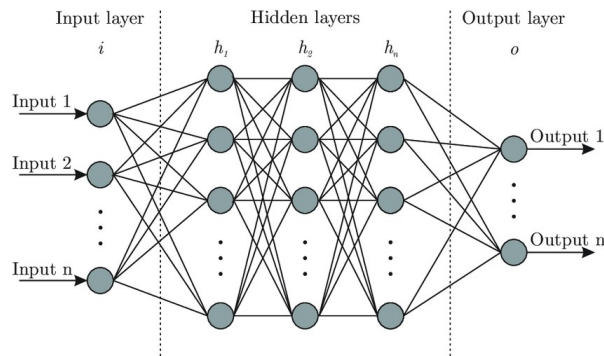
2.1.3.1 Tahapan Deep Learning

Penerapan *deep learning* dalam prediksi harga *cryptocurrency* melibatkan beberapa tahapan sistematis agar model mampu mempelajari pola data yang kompleks dan memberikan hasil prediksi yang akurat. Mulai dari pengumpulan data historis lewat API dan *preprocessing* data historis seperti pembersihan *missing values* dan normalisasi untuk memudahkan pelatihan model, Pembentukan dataset pelatihan digunakan untuk melatih bobot model dan pengujian untuk menilai kemampuan generalisasi model pada data yang benar-benar baru. Selanjutnya adalah tahap perancangan arsitektur model dengan memilih *layer* yang akan digunakan kemudian dilatih dengan pengaturan *hyperparameter* tertentu seperti jumlah neuron, *learning rate*, *epoch*, *kernel*

size, dan *dropout rate* guna mencegah *overfitting*. Tahap selanjutnya adalah pelatihan model, yaitu proses pembelajaran di mana model memperbarui bobot secara iteratif berdasarkan *loss function* menggunakan algoritma optimisasi seperti Adam. Pada tahap ini, model secara bertahap mempelajari pola temporal dalam data historis dengan tujuan meminimalkan kesalahan prediksi. Selanjutnya, model dievaluasi untuk menilai tingkat akurasi dan performa prediksi yang dihasilkan menggunakan metrik evaluasi. Tahapan ini dirancang agar model dapat belajar secara optimal, mengenali pola kompleks dalam data, serta menghasilkan prediksi yang akurat dan adaptif terhadap dinamika pasar (Sindhuri, 2025).

2.1.3.2 Artificial Neural Networks (ANN)

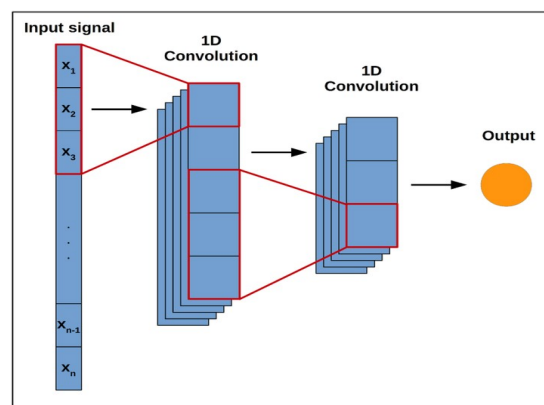
ANN merupakan model pembelajaran mesin yang meniru pola kerja otak manusia dalam mengolah informasi. ANN dibangun dari neuron-neuron buatan yang saling terikat dalam sebuah jaringan, dan dirancang untuk mempelajari serta mengenali pola-pola kompleks dalam data secara otomatis. Secara umum, ANN terdiri dari beberapa *layer*, yaitu *input layer* yang merupakan lapisan pertama yang menerima data, *hidden layer* yang melakukan komputasi utama, dan *output layer* yang merupakan lapisan terakhir untuk menghasilkan prediksi. Konsep umum arsitektur ANN tersebut dapat dilihat pada Gambar 2.1. Salah satu arsitektur ANN yang paling umum adalah *feed-forward neural network* (FFNN), yang mengalirkan informasi secara searah dari input menuju *hidden layer* hingga *output* tanpa adanya umpan balik. FFNN memungkinkan proses pembelajaran yang stabil dan efisien melalui mekanisme *backpropagation*, di mana bobot jaringan diperbarui berdasarkan perhitungan error antara prediksi dan nilai sebenarnya. Selain FFNN, terdapat arsitektur lain seperti *recurrent neural network* (RNN) yang memiliki koneksi umpan balik untuk menyimpan informasi sebelumnya, sehingga cocok digunakan untuk data berurutan seperti deret waktu. RNN dapat menangkap dependensi temporal dan hubungan jangka pendek dalam data, meskipun sering menghadapi masalah *vanishing gradient* pada urutan panjang (Bre, 2017).



Gambar 2.1 Konsep ANN (Bre, 2017)

2.1.3.3 *Convolutional Neural Network (CNN)*

CNN merupakan salah satu jenis *feed-forward neural network* yang terdiri dari beberapa tahap konvolusi untuk melakukan ekstraksi fitur secara otomatis dari data input. Dalam CNN, lapisan konvolusi menggunakan *kernel* atau *filter* berukuran kecil yang bergerak secara *sliding* pada data input untuk menangkap pola lokal. Setelah proses konvolusi, CNN umumnya menerapkan *pooling layer* yang berfungsi untuk mereduksi dimensi data, mengurangi kompleksitas komputasi, dan meningkatkan ketahanan model terhadap perubahan kecil pada input. Setelah beberapa lapisan konvolusi dan *pooling*, CNN biasanya diakhiri dengan *fully connected layer* yang menggabungkan fitur untuk menghasilkan prediksi akhir. CNN awalnya banyak digunakan di bidang *computer vision* dan kini juga banyak diterapkan pada data satu dimensi seperti *time series*. Pada data *time series cryptocurrency*, CNN mampu mendeteksi pola lokal naik-turun yang sering muncul dalam pergerakan harga (Shenfield, 2020). Konsep umum arsitektur CNN tersebut dapat dilihat pada Gambar 2.2.

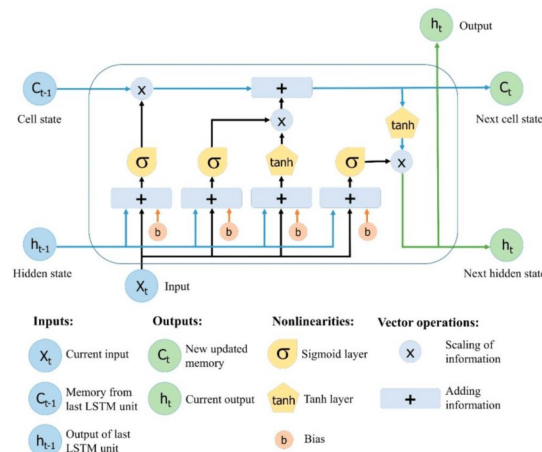


Gambar 2.2 Konsep CNN (Shenfield, 2020)

2.1.3.4 Long Short-Term Memory (LSTM)

LSTM yaitu tipe *recurrent neural network* yang menyelesaikan masalah keterbatasan menyimpan informasi jangka panjang akibat *vanishing gradient* (Greff, 2017). Mekanisme dan komponen utama LSTM tersebut dapat dilihat pada Gambar 2.3. LSTM terdiri dari:

1. *Forget gate* menentukan bagian informasi dari *cell state* sebelumnya yang harus dibuang dengan mempelajari bobot melalui fungsi aktivasi *sigmoid*.
2. *Input gate* mengontrol informasi baru yang akan dimasukkan ke dalam *cell state*. *Input gate* bekerja bersama *candidate state* yang dihasilkan oleh fungsi aktivasi *tanh* untuk memperbarui memori jangka panjang.
3. *Output gate* untuk mengatur informasi mana dari *cell state* yang akan dikeluarkan sebagai *hidden state* pada langkah waktu berikutnya.
4. *Cell state* sebagai representasi jangka panjang jalur informasi utama yang mengalir sepanjang urutan waktu.
5. *Hidden state* sebagai representasi jangka pendek yang terus diperbarui pada setiap langkah waktu.

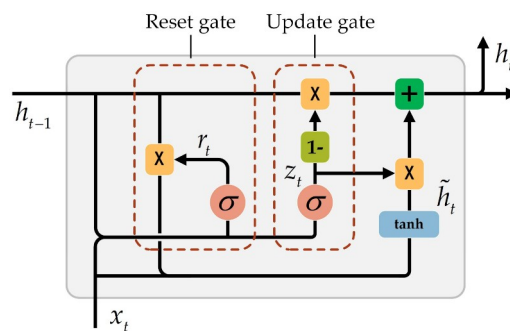


Gambar 2.3 Konsep LSTM (Adhinata, 2021)

2.1.3.5 Gated Recurrent Unit (GRU)

GRU adalah salah satu varian dari RNN yang diperkenalkan untuk mengatasi permasalahan *vanishing gradient* dan sulitnya menangkap ketergantungan jangka

panjang pada data deret waktu. GRU menggunakan dua mekanisme utama, yaitu *update gate* untuk menentukan seberapa besar bagian *hidden state* sebelumnya yang perlu dibawa ke langkah waktu saat ini dan *reset gate* untuk mengatur seberapa banyak informasi lama yang harus diabaikan ketika model memproses input baru yang memungkinkan model merespons perubahan pola secara lebih cepat. Melalui kombinasi kedua gerbang ini, GRU dapat secara adaptif memilih informasi mana yang perlu dipertahankan dan mana yang perlu dilupakan, sekaligus mengatasi permasalahan *vanishing gradient* secara lebih efisien. Secara umum, GRU lebih sederhana dan lebih ringan dibandingkan LSTM karena hanya menggunakan satu struktur memori (*hidden state*) alih-alih dua struktur terpisah seperti pada LSTM (*cell state* dan *hidden state*), sehingga jumlah parameter yang harus dipelajari cenderung lebih sedikit. Meskipun lebih sederhana, GRU pada banyak kasus menunjukkan performa yang setara bahkan lebih baik dari LSTM, terutama pada dataset dengan struktur sekuensial yang tidak terlalu kompleks. Dalam model *hybrid* LSTM untuk prediksi harga *cryptocurrency*, GRU lebih unggul dalam menangkap dependensi temporal jangka panjang dengan kompleksitas komputasi yang lebih rendah, sedangkan CNN lebih efektif dalam mengekstraksi fitur lokal dari urutan data (Alvi, 2024). Konsep dan mekanisme utama GRU dapat dilihat pada Gambar 2.4.

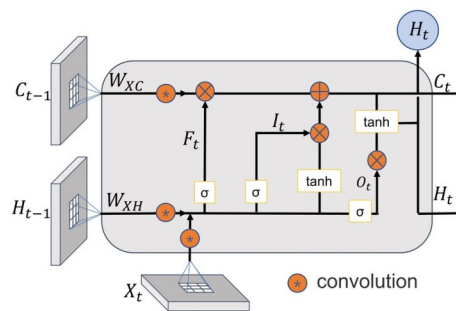


Gambar 2.4 Konsep GRU (Li, 2020)

2.1.3.6 Convolutional LSTM

Convolutional LSTM adalah arsitektur yang menggabungkan mekanisme memori jangka panjang dari LSTM dengan kemampuan ekstraksi fitur lokal dari CNN. *Convolutional LSTM* pada awalnya dirancang untuk memproses input seperti citra. Namun, konsep tersebut juga dapat diterapkan pada data *time series* satu dimensi dengan menempatkan lapisan *Conv1D* sebelum LSTM. *Conv1D* berfungsi untuk mengekstraksi pola lokal dari data input. Hasil ekstraksi fitur ini kemudian diteruskan

ke LSTM agar dapat menangkap hubungan urutan waktu yang lebih panjang dan kompleks. Pendekatan ini memungkinkan model menangkap pola lokal dan ketergantungan temporal secara bersamaan sehingga lebih sesuai digunakan untuk data *time series* yang memiliki pola kompleks, seperti harga *cryptocurrency*. *Hyperparameter* yang mempengaruhi akurasi *Convolutional LSTM* antara lain jumlah neuron, *learning rate*, *kernel size*, *dropout rate*, jumlah *epoch*, dan *batch size* (Shi, 2022). Konsep arsitektur *Convolutional LSTM* dapat dilihat pada Gambar 2.5.



Gambar 2.5 Konsep *Convolutional LSTM* (Shi, 2022)

2.1.3.7 Hybrid Deep Learning

Hybrid deep learning merupakan pendekatan yang menggabungkan dua atau lebih arsitektur *deep learning* dalam satu model untuk memanfaatkan keunggulan masing-masing arsitektur. Model yang mengombinasikan dua atau lebih arsitektur *deep learning* disebut sebagai *Hybrid Deep Learning* (HDL), yang dikembangkan untuk meningkatkan kemampuan pembelajaran dibandingkan penggunaan satu arsitektur secara terpisah (Jena, 2021). Pendekatan ini banyak diterapkan pada berbagai permasalahan seperti klasifikasi citra, pemrosesan bahasa alami, dan prediksi *time series* karena mampu menggabungkan kemampuan beberapa arsitektur *deep learning* dalam menyelesaikan karakteristik data yang kompleks.

2.1.3.8 Hybrid Deep Learning Convolutional LSTM

Pendekatan *hybrid deep learning Convolutional LSTM* merupakan salah satu bentuk penerapan konsep *hybrid deep learning* yang menggabungkan *Convolutional Neural Network* (CNN) dan *Long Short-Term Memory* (LSTM) dalam satu arsitektur. Pada pendekatan ini, lapisan konvolusi satu dimensi (*Conv1D*) digunakan untuk mengekstraksi fitur lokal dari data masukan, sedangkan LSTM mempelajari hubungan

temporal dan ketergantungan jangka panjang dari fitur yang telah diekstraksi. Kombinasi kedua arsitektur tersebut memungkinkan model memanfaatkan kemampuan ekstraksi fitur CNN sekaligus kemampuan LSTM dalam memodelkan data deret waktu, sehingga diharapkan menghasilkan performa prediksi yang lebih baik dibandingkan penggunaan CNN atau LSTM secara terpisah.

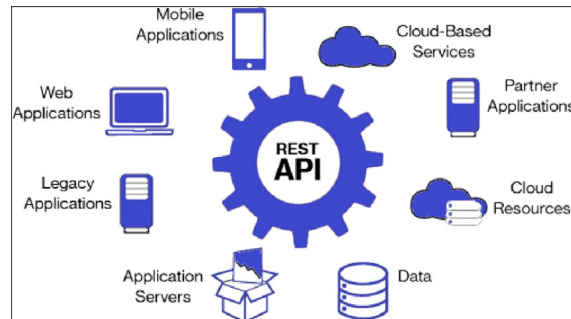
2.1.3.9 Optuna

Hyperparameter merupakan parameter yang ditentukan sebelum proses pelatihan model dimulai dan tidak diperbarui selama proses pembelajaran, tetapi berperan dalam mengendalikan proses pelatihan, seperti *learning rate*, jumlah neuron, jumlah *filter* CNN, *kernel size*, *batch size*, jumlah *epoch*, dan *dropout rate*. Pemilihan *hyperparameter* yang kurang tepat dapat menyebabkan model mengalami *underfitting* atau *overfitting*, sehingga diperlukan proses optimasi untuk memperoleh konfigurasi terbaik. Salah satu *framework* yang banyak digunakan adalah Optuna, yang menerapkan pendekatan *Bayesian Optimization* untuk mencari kombinasi *hyperparameter* secara efisien berdasarkan hasil evaluasi dari percobaan sebelumnya. Optuna menggunakan konsep *study* dan *trial*, di mana setiap *trial* merepresentasikan satu kombinasi *hyperparameter* yang dievaluasi menggunakan *objective function*, serta dilengkapi mekanisme *pruning* untuk menghentikan percobaan yang kurang menjanjikan sehingga mengurangi waktu komputasi (Akiba, 2019). Pada penelitian ini, Optuna digunakan untuk mengoptimalkan *hyperparameter* pada setiap arsitektur *hybrid deep learning Convolutional LSTM*, meliputi jumlah neuron LSTM, *learning rate*, *dropout rate*, jumlah *filter* CNN, *kernel size*, *batch size*, dan jumlah *epoch*, sehingga diperoleh konfigurasi yang mampu menghasilkan performa dan kemampuan generalisasi model dalam memprediksi harga *cryptocurrency*.

2.1.4 RESTful API

RESTful API merupakan pendekatan arsitektur layanan web yang digunakan untuk memfasilitasi komunikasi antara klien dan *server* menggunakan HTTP. Dalam pengembangan website prediksi harga *cryptocurrency*, *RESTful API* berperan sebagai jembatan yang memungkinkan klien, seperti React, berinteraksi dengan *server* yang menggunakan *framework* Flask. Melalui *RESTful API*, data hasil prediksi dapat diambil

dari *server* dan ditampilkan secara dinamis di sisi klien, sehingga *user* dapat memperoleh informasi secara langsung (Kurniawan, 2020). Konsep *RESTful API* dapat dilihat pada Gambar 2.6.



Gambar 2.6 Konsep *RESTful Api* (Kurniawan, 2020)

2.1.5 Python

Python merupakan bahasa pemrograman komputer yang mudah dipahami berkat sintaksnya yang sederhana dan bersifat *human-readable*, sehingga menjadikannya pilihan utama bagi pemula maupun peneliti profesional. Bahasa pemrograman python juga memiliki ekosistem *library* yang sangat kaya untuk kebutuhan komputasi ilmiah dan pengembangan model *deep learning* seperti TensorFlow, PyTorch, dan Keras. Selain itu, Python menyediakan *library* pendukung seperti NumPy, Pandas, dan Matplotlib yang memudahkan proses manipulasi data, analisis statistika, dan visualisasi hasil. Python juga mendukung pengembangan *backend* aplikasi melalui *framework* seperti Flask yang memungkinkan integrasi antara model prediksi dan antarmuka pengguna (Mohialden, 2024).

2.1.6 Metrik Evaluasi

Proses evaluasi terhadap model prediksi bertujuan untuk melihat tingkat kesesuaian antara hasil prediksi dan nilai sebenarnya. Metrik evaluasi yang digunakan adalah R^2 , MAE, dan RMSE.

2.1.6.1 *Coefficient of Determination* (R^2)

R^2 merupakan metrik evaluasi yang menunjukkan sejauh mana model dapat memaparkan variasi yang terdapat pada data target. Nilai R^2 mendekati 1 mengindikasikan bahwa model dapat merepresentasikan proporsi variasi data target secara signifikan, yang mengindikasikan kinerja prediktif yang tinggi. Sebaliknya, nilai

R^2 yang mendekati nol menandakan model tidak mampu menjelaskan variasi data (Chicco, 2021).

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} \quad 2-1$$

R^2 dihitung sebagai 1 dikurangi rasio jumlah kuadrat selisih antara nilai aktual dan prediksi terhadap jumlah kuadrat selisih antara nilai aktual dan rata-rata, sehingga menunjukkan proporsi variasi data yang dapat dijelaskan oleh model.

2.1.6.2 Mean Absolute Error (MAE)

MAE adalah metrik evaluasi yang menggambarkan rata-rata selisih absolut antara nilai prediksi model dan nilai aslinya dalam data. Semakin rendah nilai MAE, maka semakin tinggi akurasi prediksi yang dihasilkan oleh model (Chicco, 2021).

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad 2-2$$

MAE dihitung sebagai rata-rata dari nilai absolut selisih antara nilai aktual dan nilai prediksi untuk seluruh sampel, sehingga menunjukkan tingkat kesalahan prediksi model secara langsung.

2.1.6.3 Root Mean Square Error (RMSE)

RMSE merupakan metrik evaluasi yang dipakai untuk menilai tingkat kesalahan antara hasil prediksi model dan data aktual. RMSE diperoleh dengan menghitung akar kuadrat dari rata-rata selisih kuadrat antara nilai prediksi dan nilai aslinya. Semakin rendah RMSE nya semakin bagus akurasi prediksi (Chicco, 2021).

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad 2-3$$

RMSE dihitung sebagai akar kuadrat dari rata-rata kuadrat selisih antara nilai aktual dan nilai prediksi untuk seluruh sampel, sehingga menunjukkan besarnya kesalahan prediksi model secara keseluruhan.

2.1.7 Unified Modelling Language (UML)

UML berfungsi untuk membantu proses perancangan sistem perangkat lunak. UML menyediakan berbagai jenis diagram yang mempermudah desain sistem. Diagram yang sering digunakan seperti *use case diagram* dan *activity diagram*.

2.1.7.1 Use Case Diagram

Use case diagram menggambarkan fungsi dari sistem informasi yang akan dirancang yang menunjukkan hubungan antara aktor yang dapat berupa pengguna, proses, atau sistem lain dan sistem itu sendiri. Elemen utama dalam *use case diagram* meliputi aktor, yaitu entitas yang melakukan interaksi dengan sistem, serta *use case* yang menggambarkan fungsi-fungsi yang ditawarkan oleh sistem. Simbol utama diagram mencakup aktor, *use case*, dan garis relasi di antara keduanya (OMG, 2017).

2.1.7.2 Activity Diagram

Activity diagram dirancang untuk memvisualisasikan alur kerja dalam suatu sistem. Elemen utama dalam *activity diagram* meliputi *activities*, yaitu langkah-langkah dalam proses yang biasanya digambarkan sebagai kotak atau oval, *decision nodes* berupa simbol berlian yang menunjukkan titik keputusan di mana alur dapat bercabang, *start node* sebagai simbol lingkaran hitam, *end node* sebagai simbol lingkaran hitam dengan lingkaran luar, *flows* yang berupa panah menunjukkan aliran dari satu aktivitas ke aktivitas lainnya, *Fork* dan *join* merepresentasikan aktivitas paralel dalam proses. *Fork* membagi satu alur menjadi beberapa jalur eksekusi paralel, sedangkan *join* berfungsi untuk menyatukan kembali jalur-jalur tersebut. *Activity diagram* digunakan untuk memodelkan alur kerja dalam sistem, sehingga membantu memahami bagaimana aktivitas dilakukan dan keputusan diambil dalam sistem tersebut (OMG, 2017).

2.2 PENELITIAN TERKAIT

Berdasarkan penelitian oleh (Ahmed, 2023) berfokus pada prediksi harga Bitcoin dengan memanfaatkan arsitektur *hybrid 1D-Convolutional Neural Network* (1D-CNN) dan LSTM. Model ini menggunakan 1D CNN untuk mengekstraksi fitur lokal dari data historis dan LSTM untuk menangkap ketergantungan temporal. Dengan dataset harga Bitcoin yang mencakup periode 2014 hingga 2022, model ini menunjukkan performa terbaik pada 150 *epoch*. Namun, model ini hanya terbatas pada

Bitcoin. Digunakan *hyperparameter* 100 unit LSTM, *kernel size* sebanyak 2, 64 *filter*, dan *batch size* sebanyak 32. Perbaikan utama dari pendekatan adalah pemisahan proses ekstraksi fitur 1D-CNN dan pemodelan temporal LSTM secara modular untuk meningkatkan akurasi prediksi.

(Alonso-Monsalve, 2020) menggunakan *Convolutional LSTM* untuk mengklasifikasi arah tren harga enam *cryptocurrency* utama yaitu Bitcoin, Dash, Ether, Litecoin, Monero, XRP dengan hasil menunjukkan bahwa model *Convolutional LSTM* secara signifikan unggul dalam memprediksi harga *cryptocurrency*. Namun, terbatas pada data satu tahun yang relatif stabil. Nilai *learning rate* 0.001, *dense layer* 200 neuron, dan 150 unit LSTM. Perbaikannya terletak pada kombinasi lima lapisan konvolusional awal untuk mengekstraksi fitur secara mendalam, diikuti oleh LSTM yang menangkap dinamika temporal, serta *dense layer* lalu *output layer* untuk meningkatkan akurasi prediksi.

(Li, 2020) memprediksi harga Bitcoin menggunakan model gabungan CNN dan LSTM, yang meningkatkan akurasi dalam meramalkan pergerakan harga Bitcoin periode 2016 hingga 2018. Model CNN digunakan untuk mengekstrak fitur lokal dari data harga Bitcoin, sementara LSTM digunakan untuk menangani dependensi temporal atau urutan waktu dalam data harga. Namun, model yang dikembangkan masih terbatas oleh satu *cryptocurrency* saja yaitu Bitcoin. Digunakan *hyperparameter learning rate* sebesar 0.01, 30 unit LSTM, dan *batch size* sebanyak 5. Perbaikan dari pendekatan ini adalah pemisahan proses ekstraksi fitur 2 *layer* CNN dan pemodelan temporal oleh LSTM ke dalam dua tahap yang lebih fleksibel dan terkontrol.

(Livieris, 2021) menggunakan arsitektur *hybrid 1D-CNN + LSTM multi-input* untuk prediksi harga dan arah pergerakan beberapa *cryptocurrency* secara bersamaan yaitu Bitcoin, Ethereum, XRP. Setiap jenis *cryptocurrency* diproses oleh lapisan konvolusi terpisah untuk mengekstrak fitur spesifik, kemudian dilanjutkan ke LSTM untuk menangkap ketergantungan temporal. Hasilnya dari masing-masing jalur digabung untuk satu *output* prediksi. Dataset terdiri dari 3 tahun data harian ketiga *cryptocurrency* terbesar. Model ini memanfaatkan data gabungan secara efisien, mengurangi *overfitting* dan menurunkan beban komputasi dibanding jaringan *fully-connected*. Hasil tertinggi $R^2 = 0.964$ yang hanya dicapai pada Ethereum,

sehingga menunjukkan adanya variasi performa antar *cryptocurrency* dan potensi peningkatan untuk mencapai akurasi yang lebih konsisten. Digunakan *hyperparameter* 16 *filter*, 50 unit LSTM, *dropout rate* sebesar 0.4 dan jumlah *dense layer* 64 neuron. Perbaikan utama dari pendekatan ini adalah strategi *multi-input* yang memungkinkan setiap aset diproses secara terpisah pada CNN untuk mengekstraksi fitur spesifik, kemudian dilanjutkan ke LSTM untuk menangkap ketergantungan temporal sebelum digabungkan pada tahap *output*.

Tabel 2.1 Perbandingan Hasil Penelitian

No	Peneliti & Tahun	Metode / Model	Hasil Penelitian	Kelemahan / Gap	Kaitan dengan Penelitian Ini
1	Ahmed (2023)	1D-CNN + LSTM (<i>Hybrid</i>)	Model berhasil memprediksi harga Bitcoin dengan akurasi cukup baik	Hanya menggunakan data Bitcoin	Penelitian ini akan menguji performa <i>Convolutional LSTM</i> di 6 <i>cryptocurrency</i> utama
2	Alonso-Monsalve (2020)	<i>Convolutional LSTM</i>	<i>Convolutional LSTM</i> unggul memprediksi tren 6 <i>cryptocurrency</i>	Data terbatas hanya 1 tahun dan kondisi pasar stabil	Penelitian ini akan menggunakan data historis multi-tahun untuk melihat generalisasi model
3	Li (2020)	CNN + LSTM (<i>Hybrid</i>)	CNN menangani fitur lokal, LSTM menangani urutan waktu, hasil cukup baik untuk Bitcoin	Hanya memfokuskan pada Bitcoin	Penelitian ini memperluas cakupan ke 6 <i>cryptocurrency</i> sekaligus untuk model <i>hybrid</i>
4	Livieris (2021)	1D-CNN + LSTM <i>Multi-Input</i>	1D-CNN + LSTM <i>Multi-Input</i> menghasilkan R^2 tertinggi 0.964 hanya pada Ethereum	Akurasi tidak konsisten di semua <i>cryptocurrency</i>	Penelitian ini akan mengoptimasi <i>hyperparameter</i> agar akurasi lebih merata di berbagai <i>cryptocurrency</i>

Secara keseluruhan, penelitian-penelitian terdahulu pada Tabel 2.1 menunjukkan bahwa model *hybrid deep learning Convolutional LSTM* (gabungan CNN

dan LSTM) memiliki potensi besar dalam memprediksi harga *cryptocurrency*. Namun, masih ditemukan berbagai keterbatasan seperti cakupan data yang terbatas, fokus prediksi hanya pada satu jenis *cryptocurrency* seperti Bitcoin, serta ketidakmerataan akurasi antar jenis *cryptocurrency*. Penelitian ini bertujuan untuk menjawab keterbatasan yang ditemukan dalam studi terdahulu, dengan menerapkan model *hybrid deep learning Convolutional LSTM* yang telah dioptimasi, menggunakan cakupan data historis yang lebih luas, serta fokus pada enam jenis *cryptocurrency* dengan kapitalisasi pasar terbesar guna memperoleh hasil prediksi yang semakin tepat, stabil, dan generalis.