

BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Tabel 2. 1 Penelitian Terdahulu

No.	Nama Jurnal	Jurnal	Tahun	Penulis	Hasil
1	Jurnal Informatika dan Sistem Informasi (JIFoSI) Vol. 1, No.2 [28]	Rancang Bangun Bot Auto Trade Cryptocurrency Berbasis Web	2020	Setiawan R, Parlita R, Mumpuni R	Hasil dari penelitian ini berupa aplikasi yang memberikan sebuah keuntungan bagi para trader. Dengan penelitian ini, trader tidak harus terus menerus memantau pergerakan harga <i>cryptocurrency</i> .
2	<i>Journal of Mathematical Finance</i> [29]	<i>Bitcoin Price Prediction Based on Deep Learning Methods</i>	2020	Jiang X	Hasil dari penelitian menunjukkan bahwa 2-layer LSTM akan memberikan hasil terbaik saat digunakan untuk menguji data <i>testing</i> .
3	<i>Finance Research Letters</i> [25]	<i>A Novel Cryptocurrency Price Trend Forecasting Model Based on LightGBM</i>	2020	Sun X, Liu M, Sima Z	Kesimpulan dari hasil penelitian adalah model LightGBM lebih disarankan untuk prediksi dalam kurun waktu 2 minggu. LightGBM lebih unggul dibandingkan dengan model lainnya, karena cocok dalam menangani jumlah data dengan fitur dan sampel yang besar.
4	<i>Intelligent Automation & Soft Computing</i> [24]	<i>Forecast of LSTM-XGBoost in Stock Price Based on Bayesian Optimization</i>	2021	Tian Liwei, Feng Li, Sun Yu, Guo Yuankai	Model hibrida memberikan kinerja, stabilitas dan kelayakan yang lebih unggul dalam memprediksi harga saham dibandingkan penggunaan model tunggal.
5	<i>Computers, Materials & Continua</i> [30]	<i>A Novel Cryptocurrency Prediction Method Using Optimum CNN</i>	2022	Hasan Syed, Hasan Salih, Ahmed M, Hasan Hamid	Model LSTM dengan <i>lookback</i> period 1 minggu menghasilkan performa yang optimal dan akurat untuk keadaan <i>short-term</i> , dengan nilai metrik, sebagai berikut untuk BTC dan ETH: Performa Prediksi BTC RMSE = 16.83 MAE = 9.64

					MSE: = 268.46 Performa Prediksi ETH RMSE = 19.64 MAE = 14.98 MSE = 431.99
6	Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi) [1]	Prediksi Harga Cryptocurrency Menggunakan Algoritma Long Short Term Memory (LSTM)	2022	Moch Farryz Rizkilloh, Sri Widiyanesti	Hasil model prediksi mengemukakan bahwa LSTM terbukti dapat digunakan terhadap <i>time series</i> . Hasil pengujian dari koin DOGE dan ADA memberikan nilai RMSE yang baik dengan angka sebagai berikut: DOGE = 0.0544 ADA = 0.1607
7	Seminar Nasional Teknologi Informasi dan Komunikasi STI&K (SeNTIK) [31]	Aplikasi Perbandingan Prediksi Harga Bitcoin Menggunakan Deep Learning dengan Metode ARIMA, SARIMA, LTSM, dan Gradient Boosting Regressor	2023	Agung Slamet Riyadi, Ire Puspa Wardhani, Irfan, Andi Perdana	Hasil penelitian menunjukkan bawah model SARIMA memberikan hasil RMSE yang paling akurat dibandingkan dengan LSTM ataupun <i>Gradient Boosting Regressor</i> . Berikut merupakan hasil yang diperoleh : ARIMA = 14490,55 SARIMA = 13692,22 LSTM = 19650,40 <i>Gradient Boosting Regressor</i> = 14573,20
8	IJCCS (Indonesian Journal of Computing and Cybernetics Systems) [32]	Improved LSTM Method of Predicting Cryptocurrency Price Using Short-Term Data	2023	Risna Sari, Kusrini Kusrini, Tonny Hidayat, Theofanis Orphanoudakis	Penelitian ini mengevaluasi efektivitas metode LSTM dalam memprediksi <i>cryptocurrency</i> jangka pendek dan terbukti akurasi yang didapatkan berbeda karena sangat bergantung dengan jenis <i>cryptocurrency</i> dan konfigurasi model. Penelitian ini juga mengungkapkan bahwa LSTM dapat memberikan hasil prediksi yang akurat jika disesuaikan Dengan karakteristik spesifik dari data (jenis <i>cryptocurrency</i>)

9	<i>Journal of Computer & Electrical and Electronics Engineering Sciences</i> [33]	<i>Cryptocurrency Analysis Using Machine Learning and Deep Learning Approaches</i>	2023	Farah Muhammed Sakran	Hasil dari penelitian yang dilakukan menunjukkan keunggulan dari model <i>deep learning</i> , yaitu CNN dalam memprediksi harga <i>cryptocurrency</i> , bahkan lebih baik dari model ANN. Berikut merupakan hasil MAE dan RMSE dari kedua model: ANN: MAE = 0.0612 RMSE = 0.093494 CNN: MAE = 0.032474 RMSE = 0.054376
10	<i>Journal of Advances in Information Technology</i> [17]	<i>Gradient Boosting and LSTM Based Hybrid Ensemble Learning for Two Step Prediction of Stock Market</i>	2023	Pratyush Ranjan Mohapatra, Ajaya Kumar Parida, Santosh Kumar Swain, dan Santi Swarup Basa	Penelitian ini membuktikan bahwa model gabungan (<i>hybrid model</i>) yang diusulkan, yaitu CNN berbasis XGB-LSTM unggul dalam memprediksi harga saham dibandingkan dengan model lainnya, bahkan lebih baik dari model tunggal. <i>Hybrid model</i> juga menawarkan efisiensi waktu dan komputasi. Adanya gabungan dari metode-metode dengan implementasi teknik optimasi dapat secara signifikan meningkatkan hasil performa dan prediksi.

Tabel 2.1 merupakan tabel penelitian terdahulu yang memberikan informasi dan pengetahuan lebih dalam mengenai topik yang paralel dengan penelitian. Dimana teknik *deep learning* dengan algoritma LSTM menjadi unggul dalam prediksi *cryptocurrency* dengan kemampuannya dalam menangkan pola jangka panjang. Penelitian terdahulu juga menunjukkan adanya potensi yang dapat dikembangkan dari CNN dalam meningkatkan akurasi dan performa model dalam memproses data yang besar dan kompleks. Keunikan dalam penelitian ini adalah adanya perbandingan antara kedua model *deep learning* tersebut (LSTM dan CNN) untuk mengetahui model yang sebenarnya lebih

optimal dan nantinya model terunggul akan diintegrasikan dengan algoritma *machine learning* pilihan, yaitu XGBoost dan LightGBM. *Hybrid* model yang nantinya dikembangkan diharapkan dapat mengatasi tantangan prediksi harga *cryptocurrency* dengan menggunakan model *hybrid* yang optimal.

Penelitian pertama oleh Setiawan et al., (2020) mengembangkan aplikasi BOT auto trade menggunakan metode prediksi *Stochastic* dan *Exponential Moving Average* (EMA). Masalah utama yang dihadapi adalah fluktuasi harga yang tinggi. Gap dari penelitian ini adalah tidak menggabungkan model *machine learning* atau *deep learning* yang lebih canggih untuk meningkatkan keakuratan prediksi harga.

Penelitian kedua oleh Jiang X., (2020) mengeksplorasi berbagai metode *deep learning* seperti MLP, LSTM, dan GRU untuk prediksi harga Bitcoin. LSTM terbukti memberikan prediksi terbaik dengan menggunakan metode normalisasi berdasarkan jendela. Gap yang ditemukan adalah penelitian ini tidak menggabungkan model *hybrid* yang mengkombinasikan *machine learning* dan *deep learning* untuk menghasilkan performa prediksi yang lebih optimal dan akurat.

Penelitian ketiga oleh Sun X et al., (2020) menggunakan model LightGBM untuk prediksi tren harga *cryptocurrency*. LightGBM menunjukkan performa yang baik dalam memprediksi harga *cryptocurrency*. Namun, penelitian ini fokus pada satu algoritma *machine learning* tanpa kombinasi model *deep learning* lainnya untuk hasil prediksi yang lebih optimal.

Penelitian keempat oleh Liwei et al., (2021) mengembangkan model *hybrid* LSTM-XGBoost untuk prediksi harga saham. Model *hybrid* ini menunjukkan akurasi yang lebih baik dibandingkan model tunggal. Namun, penelitian ini fokus pada prediksi harga saham, bukan pada *cryptocurrency*, sehingga perlu dilakukan pengujian serupa pada data *cryptocurrency* untuk evaluasi lebih lanjut.

Penelitian kelima oleh Hassan et al., (2022) menggunakan metode CNN optimal untuk prediksi *cryptocurrency*. CNN menunjukkan performa terbaik di antara model *deep learning* lainnya. Gap yang ada adalah penelitian ini tidak

adanya kombinasi *deep learning* dan *machine learning* (model *hybrid*) untuk memanfaatkan kekuatan dari kedua pendekatan tersebut.

Penelitian keenam oleh Moch Farryz Rizkilloh et al., (2022) memfokuskan pada prediksi harga *cryptocurrency* menggunakan algoritma LSTM. Pembagian data yang berbeda mempengaruhi akurasi prediksi. Penelitian ini hanya menggunakan satu pendekatan *deep learning* (LSTM) dan tidak mempertimbangkan model *hybrid* atau metode *deep learning* lainnya untuk prediksi harga.

Penelitian ketujuh oleh Agung Slamet Riyadi et al., (2023) membandingkan algoritma *deep learning* (ARIMA, SARIMA, LSTM) dan Gradient Boosting Regressor untuk prediksi harga Bitcoin. SARIMA menunjukkan performa terbaik dengan RMSE terendah. Namun, penelitian ini tidak mengeksplorasi model *hybrid* atau teknik *deep learning* yang lebih canggih seperti CNN untuk meningkatkan akurasi prediksi lebih lanjut.

Penelitian kedelapan oleh Risna Sari et al., (2023) menunjukkan bahwa data jangka pendek dapat meningkatkan efektivitas metode LSTM dalam prediksi harga *cryptocurrency*. Namun, penelitian ini tidak mempertimbangkan model *hybrid* yang menggabungkan LSTM dengan algoritma lain untuk peningkatan keakuratan prediksi.

Penelitian kesembilan oleh Farah Muhammed Sakran (2023) mengevaluasi berbagai model *machine learning* dan *deep learning* untuk prediksi harga Bitcoin. CNN menunjukkan performa terbaik di antara model *deep learning*, sementara RF terbaik di antara model *machine learning*. Namun, penelitian ini tidak mengkombinasikan model *machine learning* dan *deep learning* (*hybrid*) untuk memanfaatkan kekuatan dari kedua pendekatan dalam prediksi harga.

Penelitian kesepuluh oleh Mohapatra et al., (2023) mengembangkan model *ensemble hybrid* menggunakan CNN, LSTM, dan XGBoost untuk prediksi pasar saham. Metode *ensemble hybrid* menunjukkan peningkatan akurasi dibandingkan model tunggal. Namun, penelitian ini fokus pada prediksi harga saham, sehingga perlu validasi lebih lanjut pada data *cryptocurrency* untuk memastikan keefektifan metode *hybrid* yang digunakan.

Melalui perbandingan kedua algoritma *deep learning* dan evaluasi dari hasil integrasi *deep learning* dengan *machine learning*, penelitian bertujuan untuk mengembangkan model yang optimal, melihat dari segi performa. Dengan demikian, penelitian ini tidak hanya memperdalam pemahaman algoritma pilihan secara tersendiri melainkan juga bagaimana kombinasi (model *hybrid*) dari algoritma yang telah disaring dapat mengoptimalkan prediksi harga cryptocurrency, serta diharapkan juga penelitian dapat memberikan kontribusi yang berarti secara literatur dan membukakan peluang untuk pengembangan penelitian lebih lanjut.

2.2 Tinjauan Teori

2.2.1 *Cryptocurrency*

Perkembangan teknologi yang semakin pesat, terutama pada Revolusi Industri 4.0 memberikan inovasi-inovasi baru, terutama pada dunia perekonomian dan investasi. *Cryptocurrency* merupakan salah satu inovasi yang sedang naik daun. *Cryptocurrency* adalah instrumen investasi jual beli uang virtual (token). *Cryptocurrency* ini tersimpan dalam dompet digital yang dimana token tersebut dapat diperjual belikan kembali ataupun ditukar dengan uang (secara virtual) atau token jenis lainnya [34]. Jika dilihat dari kata-katanya, *cryptocurrency* terbagi atas 2 kata, yaitu “*cryptography*” dan “*currency*”, dimana secara berurutan dapat diartikan sebagai kode rahasia dan mata uang. Terdapat hampir lebih dari 10,000 jenis *cryptocurrency* yang diperjualbelikan pada pasar *cryptocurrency*.

Terdapat 4 Jenis *cryptocurrency* yang terkenal dalam kalangan investor, yaitu Bitcoin (BTC), Ethereum (ETH), Litecoin (LTC), Dogecoin. Bitcoin menjadi aset digital pertama (*cryptocurrency*) dan saat ini memegang harga tertinggi diantara token lainnya. Dilanjutkan dengan Ethereum, token ini merupakan token yang dicetuskan pada tahun 2015, token ini menjadi salah satu token yang menarik bagi para investor dikarenakan Ethereum dapat digunakan pada berbagai aset digital dan memiliki fitur *smart contract*. Berikutnya terdapat Litecoin (LTC), token yang terkenal karena kemiripannya dengan Bitcoin yang memproses blok dalam kurun waktu yang cepat. Terakhir Dogecoin, token ini

menjadi unggulan bagi para investor muda karena token ini berawal sebagai lelucon atau sebagai “*meme coin*”, tidak hanya itu namun token ini juga didukung oleh Elon Musk.

Cryptocurrency bekerja hampir sama dengan mata uang yang digunakan dalam kehidupan kita. Token atau sering juga disebut sebagai koin dapat dibaca dan diproses dengan adanya *blockchain*. Pencatatan transaksi pada rantai blok (*blockchain*) akan mencatat dan menampung kode tersendiri (hash kriptografia), informasi mengenai transaksi dan waktu. Transaksi yang dilakukan masuk ke dalam rekaman data yang dapat diakses oleh publik.

2.2.2 *Machine Learning*

Machine Learning merupakan peranakan dari *artificial intelligence* (AI) yang dapat diartikan sebagai mesin yang dapat melakukan proses pembelajaran layaknya seorang manusia. Mesin dapat melakukan proses pembelajaran dengan adanya dukungan dari ilmu lainnya seperti *data mining*, statistika dan matematika. Cara kerja dari teknologi ini adalah dengan menganalisa data agar dapat mengidentifikasi pola agar dapat membuat prediksi ataupun sebuah keputusan dari hasil data yang telah dianalisa. Dalam kata lain, mesin ini merefleksikan pola-pola dari data yang telah diperoleh untuk dapat menghasilkan sebuah model [35]. Teknik ini sangat berbeda dengan teknologi pemrograman lainnya (pemrograman konvensional) yang memerlukan berbagai macam instruksi dan proses yang berulang pada proseliniergram. Umumnya ML bekerja dengan terlebih dahulu mengoleksi data (*collect data*), dilanjutkan dengan pemilihan fitur (*feature selection*), pelatihan model (*training model*), evaluasi (*evaluation*), dan terakhir pengujian model (*testing model*) [36], alur dapat dilihat pada Gambar 2.1.



Gambar 2. 1 Implementasi Metodologi Umum Machine Learning

Sumber: [37]

Terdapat 3 tipe *machine learning*, yaitu *supervised learning*, *unsupervised learning*, dan *reinforcement learning*. Ketiga tipe *machine learning* digunakan menyesuaikan dengan situasi dan kasus spesifik yang akan ditangani. *Supervised learning* mengelola dan mempelajari model sebelumnya (terdahulu) dengan mencocokkan data *input* kepada target label. Tipe *machine learning* ini juga dapat digunakan untuk mengolah data yang sudah terstruktur maupun belum dengan menggunakan berbagai teknik *Linear Regression*, *Decision Trees*, *K-Nearest Neighbor*, dan *Naive Bayes*. *Unsupervised learning* ada teknik pembelajaran dimana mesin memberikan perhatian khusus pada pola dan struktur dari data. Teknik ini mampu memproyeksikan sebuah pola data yang dapat membantu dalam memberikan pemahaman yang lebih. Algoritma yang umumnya digunakan pada teknik ini adalah *Clustering* dan *K-Means*. Terakhir yaitu *Reinforcement learning* yang dapat beroperasi dengan data yang tidak di label (*unlabeled data*). Tipe ini mampu melakukan proses pembelajaran hanya dengan berinteraksi dengan sekelilingnya (lingkungan) dan mengoptimalkan perilakunya (sistem dalam model) agar dapat mencapai target tertentu.

2.2.3 Deep Learning

Deep learning merupakan metode pendekatan dari *artificial intelligence* (AI) yang merupakan sub bagian dari *machine learning*, namun tidak kalah jauh dari *machine learning*. Terinspirasi oleh jaringan saraf otak manusia, metode ini menggunakan *Artificial Neural Networks* dalam mempelajari hubungan kompleks dan pola yang terdapat dalam secara optimal dan efektif. Model *Deep Learning* terbukti dapat memberikan hasil yang efektif disaat dipadukan dengan data yang cukup kompleks dan berukuran besar, seperti data teks, angka, audio, dan gambar. Sebagai contoh, metode *deep learning* yang lebih maju seperti *Convolutional Neural Networks* (CNN) dan *Long Short-Term Memory* (LSTM) yang digunakan untuk memprediksi pasar *cryptocurrency* dengan kinerja yang lebih baik dibandingkan metode tradisional [38]. Kemampuan *deep learning* untuk menangani hubungan nonlinier yang kompleks membuatnya sangat efektif dalam prediksi *time-series*, seperti harga *cryptocurrency* yang sangat dinamis [39]. Pendekatan ini juga populer dikarenakan keuntungan yang dibawakannya

dalam meminimalisir, bahkan menghilangkan adanya keperluan untuk modifikasi data atau pengolahan data saat proses pembelajaran berlangsung.

2.3 Teori Framework / Algoritma / Metode Tuning

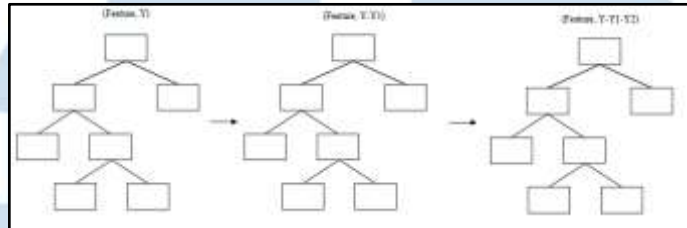
2.3.1 Framework CRISP-DM

Cross-Industry Standard Process for Data Mining (CRISP-DM) merupakan salah satu metodologi *data mining* yang umumnya digunakan dalam sebuah penelitian [40]. Gabungan antara penggunaan teknik dan metodologi yang tepat dapat menjadi alat yang esensial saat proses *data mining*, karena dengan itu pemahaman prosedural mengenai penelitian menjadi lebih jelas. Metodologi memberikan hasil yang terstruktur dan panduan yang lengkap untuk menjalankan sebuah penelitian. Panduan atau arahan yang jelas pada penelitian sangatlah krusial karena dapat mempermudah proses penelitian dan secara keseluruhan menjadi *blueprint* untuk mendapatkan hasil yang terbaik, dengan meminimalisir adanya *error*. Terdapat 6 tahapan dalam kerangka CRISP-DM yang harus dieksekusi dalam melakukan proses *data mining*, yaitu pengertian bisnis (*Business Understanding*), pengertian data (*Data Understanding*), persiapan data (*Data Preparation*), modeling data (*Data Modelling*), evaluasi model (*Model Evaluation*) dan penyebaran (*Deployment*) [41].

2.3.2 Extreme Gradient Boosting (XGBoost)

XGBoost merupakan algoritma berbasis *machine learning* yang merupakan peningkatan dari *Decision Tree/Regression Tree*. Algoritma ini menjadi istimewa karena menghasilkan optimisasi yang luar biasa, mulai dari efisiensi algoritma dalam meningkatkan kinerja sistem melalui parallelism hingga meningkatkan efisiensi perangkat keras (*hardware*) secara signifikan [42]. Algoritma ini bekerja dengan cara memahami pola dari *training data* dari hasil pohon pertama untuk mendapatkan patokan hasil awal, lalu dilanjutkan dengan pohon kedua, dimana pohon kedua akan mempelajari dari *training data* pada pohon pertama. Nilai yang akan didapatkan pada pohon kedua adalah $|Y - Y_1|$ yang merupakan perbedaan dari label sesungguhnya dengan hasil prediksi label yang didapatkan dari hasil pohon pertama. Pohon terakhir juga akan mempelajari proses dan melakukan estimasi

yang akan memberikan nilai Y_3 dari hasil gabungan antara pohon pertama hingga pohon terakhir, demikian dapat meminimalisir adanya *error* ataupun *anomalies*. Proses kerja yang telah disampaikan sebelumnya dapat dilihat pada Gambar 2.2.



Gambar 2. 2 Proses Kerja XGBoost

Sumber: [43]

Hasil akhir prediksi dari XGBoost tentunya akan dihitung menggunakan formula yang umum digunakan. Formula ini memberikan *output* penjumlahan dari semua pohon regresi yang telah dilatih oleh data. Gabungan antara berbagai macam model yang saling melatih hasil satu sama lain akan memberikan hasil yang lebih padu dan akurat. Pada Formula 2.1, $f_k(xi)$ merupakan kontribusi pohon ke- k terhadap hasil prediksi (akhir), sedangkan xi merupakan nilai dari fitur yang menjadi patokan. Terakhir yaitu F yang merupakan ruang fungsi dari semua pohon yang telah dilakukan pelatihan disaat *boosting* dan K yang merupakan total dari jumlah pohon di dalam model XGBoost.

$$y = \sum_{k=1}^K f_k(xi), f_k \in F$$

Formula 2. 1 Rumus Jumlah Skor Hasil XGBoost

Sumber : [42]

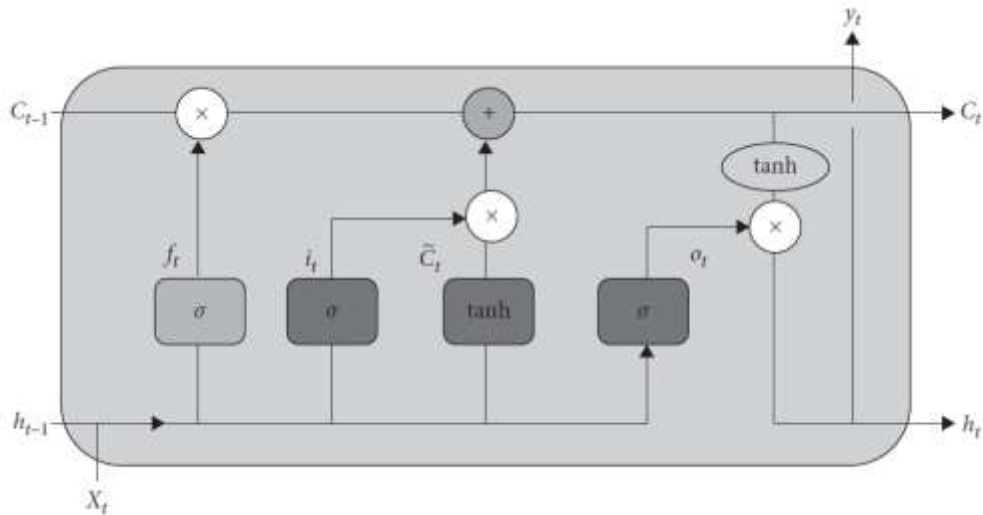
Deskripsi symbol:

1. y = Jumlah prediksi akhir dari setiap pohon
2. \sum = Penjumlahan (kontribusi setiap pohon)
3. K = Total jumlah pohon pada XGBoost
4. k = Indeks pohon ke- k (adanya penambahan untuk setiap pohon)
5. f_k = Fungsi prediksi hasil dari pohon ke- k (individual)
6. F = Ruang fungsi (mengandung semua pohon yang sudah dilatih)

7. x_i = Vektor fitur dari sampel data ke- i (data yang diprediksi)

2.3.3 Long Short-Term Memory (LSTM)

Long Short-Term Memory merupakan model dari *Recurrent Neural Network* (RNN) yang ditemukan pada bidang Deep Learning. Model ini di temukan oleh Hochreiter & Schmidhuber pada tahun 1997, lalu kemudian dikembangkan dipopulerkan oleh para ahli lainnya. Algoritma ini memerlukan data dalam jumlah yang besar. Dengan adanya algoritma ini, kita dapat memprediksi dan mendapat gambaran (simpulan) mengenai informasi jangka panjang dari data yang diteliti. LSTM sendiri bekerja dalam bentuk modul yang menjalankan proses berulang kali, berikut merupakan contoh gambaran infrastruktur dalam LSTM:



Gambar 2. 3 Infrastruktur LSTM

Sumber: [21]

Metode Long-Short Term Memory (LSTM) digunakan dalam kasus peramalan data (*forecasting*), dimana prediksi yang dapat dihasilkan merupakan prediksi yang akurat. Prediksi yang akurat dan baik dapat dilihat secara langsung dari tingkat kesalahan yang terjadi dalam prediksi, dimana jika tingkat kesalahan kecil, maka dengan metode dalam prediksi dapat dikatakan tepat. Perlu diketahui bahwa di dalam metode ini terdapat cell atau tempat dimana informasi akan dikumpulkan sebagai tempat temporary. Pada cell akan dilakukan sebuah

manipulasi memori yang dijalankan oleh komponen gate. Terdapat 3 jenis gate pada LSTM, yaitu *Forget gate*, *Input gate*, dan terakhir yaitu *Output gate*, yang digunakan dalam proses penghapusan ataupun penambahan informasi pada LSTM sel [44]. Secara kelebihan LSTM dapat memperoleh prediksi secara kronologis dan dapat memprediksikan depensi jarak jauh. LSTM juga lebih unggul dalam memori jarak pendek.

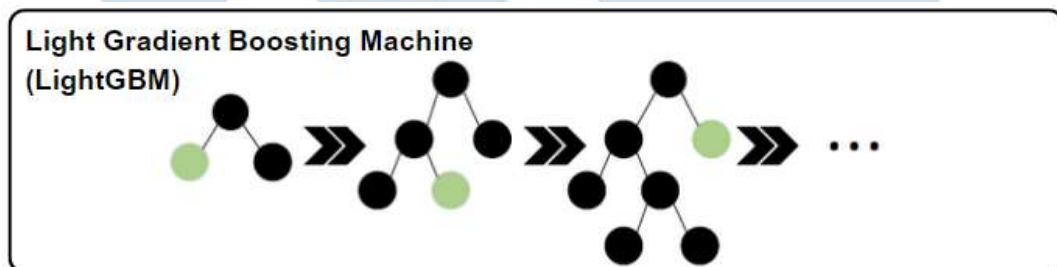
2.3.4 Convolutional Neural Networks (CNN)

Algoritma *Convolutional Neural Networks* (CNN) merupakan algoritma jaringan syaraf yang efektif digunakan saat menganalisa data array yang berstruktur. Algoritma yang merupakan salah satu tipe pengembangan dari *Artificial Neural Network* ini kerap digunakan untuk *image recognition* ataupun dalam memproses mencerna gambar agar dapat mengetahui pola. Algoritma ini membutuhkan data dalam jumlah makro agar dapat bekerja secara maksimal. Salah satu kelebihan dari CNN adalah, algoritma ini mampu bekerja dengan sendirinya tanpa adanya gangguan ataupun bantuan dari manusia [45]. Terdapat 3-layer utama yang dibangun pada pemodelan algoritma CNN yang jika diterapkan dapat memberikan akurasi yang tinggi, yaitu *convolution layer*, *pooling layer* dan layer terakhir disebut dengan *fully connected layer* [46].

2.3.5 Light Gradient Boosting Machine (LightGBM)

Algoritma *Light Gradient Boosting Machine* (LightGBM) atau juga dikenal dengan nama *Lightweight Efficient Gradient Boosting Tree*, merupakan algoritma *machine learning* yang dikembangkan oleh seorang peneliti di Microsoft pada tahun 2017 [25]. LightGBM merupakan algoritma yang efisien dan inovatif, bahkan menurut penelitian Ji Z et al., algoritma ini dapat meningkatkan efisiensi pelatihan secara signifikan karena kemampuannya untuk mengetahui titik pemisahan fitur yang tepat dan optimal menggunakan histogram [47]. Penelitian Cui et al., menambahkan bahwa LightGBM merupakan algoritma berperforma tinggi yang kemampuannya dalam meningkatkan kecepatan pelatihan hingga mencapai 10 kali lipat dan mengurangi penggunaan memori hingga 3 kali lipat [26]. LightGBM bekerja dengan mengembangkan pohon secara vertikal, cara kerja

inilah yang membedakannya dengan algoritma dasar lainnya [48]. Algoritma ini bekerja dengan mengimplementasikan strategi pertumbuhan pohon (*decision tree*) berbasis *Leaf-Wise*. Strategi ini mengurangi adanya *overhead* dan kompleksitas waktu antar data, serta memastikan agar kualitas dan keutuhan data terjamin. Dari Gambar 2.4, dapat disimpulkan bahwa fokus utama dari algoritma ini adalah dengan memilih daun yang berpotensi meningkatkan kinerja. Proses selektif berlangsung dengan membagi daun yang hanya memiliki keuntungan terbesar agar nantinya dapat menghasilkan pohon yang optimal.



Gambar 2. 4 Perkembangan Layer LightGBM

Sumber: [26]

2.3.6 Tuning

Tuning merupakan proses mengoptimalkan dan menyesuaikan model yang sedang dilatih. Proses ini diperlukan agar dapat memberikan hasil terbaik dan optimal. *Tuning* dapat dilakukan pada suatu model yang sebelumnya telah dilatih, namun dalam kasus-kasus tertentu *tuning* tidak harus dilakukan [49]. Dalam penelitian ini, terdapat 2 teknik *tuning* yang akan dilakukan, yaitu:

1. *Hyperband*: Teknik optimasi *tuning* untuk *hyperparameter* yang menggunakan *early stopping* (penghentian dini) dan alokasi sumber daya *bandit-based* untuk mengoptimalkan pelatihan. Tahapan penyaringan juga dilakukan dalam teknik *tuning hyperband*, dimana konfigurasi dengan performa terbaik akan lolos seleksi dan dilanjutkan, sedangkan konfigurasi *hyperparameter* yang tidak menjanjikan akan dihentikan. Dengan demikian,

penggunaan teknik ini tidak hanya menghemat waktu, namun secara efisien menghasilkan konfigurasi *hyperparameter* terbaik [50].

2. *Optuna*: Teknik pilihan optimasi *hyperparameter* melalui API berbasis *define-by-run* yang memberikan kemampuan untuk digunakan dengan pemrograman modular dan memungkinkan adanya konfigurasi spasial parameter secara fleksibel. Teknik *Optuna* menjadi solusi inovatif untuk konfigurasi *hyperparameter* yang adaptif dengan mengimplementasikan strategi *pruning* yang dinamis dan efektif dalam mengeliminasi percobaan dengan performa rendah, dengan demikian dapat secara signifikan meningkatkan efisiensi model [51], [52].

2.4 Teori Pemilihan Alat (*Tools*)

2.4.1 *Jupyter Notebook*

Jupyter Notebook dapat diartikan sebagai buku catatan elektronik yang berisikan bahasa pemrograman. Bahasa yang umumnya digunakan dalam *Jupyter Notebook* adalah *Python*. *Jupyter Notebook* bersifat terbuka (*open source*), dimana dapat digunakan dengan mudah oleh programmer untuk menganalisis dan mendeskripsikan data, bahkan memproyeksikannya dalam grafik yang menarik [53]. Alat ini juga memiliki keunggulan lainnya, yaitu fitur-fitur perpustakaan yang dapat menunjang proses pemrograman. Selain itu, *Jupyter Notebook* juga dapat digunakan dengan bahasa pemrograman *MATLAB* ataupun *Java*. Dengan fitur-fitur perpustakaan dan kelengkapan dalam bahasa yang digunakan, *Jupyter Notebook* menjadi alat yang sederhana dan efektif untuk seorang programmer.

2.4.2 *Python*

Python merupakan bahasa pemrograman yang sudah tidak lagi asing dalam dunia *Big Data Analytics*. Bahasa pemrograman ini sering digunakan dalam proses pembuatan suatu aplikasi dan dalam melakukan analisis pada data tertentu. Secara fungsional, bahasa pemrograman ini memiliki tujuan untuk membuat program dan menyelesaikan berbagai permasalahan. Penggunaan bahasa pemrograman ini sangat tempat diimplementasikan dalam penelitian yang berlangsung dikarenakan bahasa

Python ini interpretatif multiguna dan secara keseluruhan menggabungkan fungsi pustaka (*library*), fungsionalitasnya bahasa yang luas dan dirancang untuk meringankan tugas para *programmer* dengan memprioritaskan pada kejelasan dan keefisienan [54]. *Library* yang mendominasi ini merupakan salah satu alasan mengapa bahasa pemrograman ini dapat terpilih. Tidak hanya pada *library* yang sangat meringankan beban penelitian, namun tampilan visual yang mudah untuk ditelaah dan jelas dalam menunjukkan perubahan pada pola tertentu. Secara keseluruhan bahasa pemrograman ini dirancang untuk meringankan tugas para *programmer* dengan tingkat adaptasi bahasa yang tinggi pada berbagai sistem dan penggunaan dari bahasa ini juga dapat menghemat waktu.

2.4.3 Pengukuran Performa

2.4.3.1 MAE

Mean Absolute Error (MAE) merupakan pengukuran yang memberikan hasil rata-rata nilai absolut dari perbedaan nilai sebenarnya dengan nilai prediksi. Nilai sebenarnya dilambangkan dengan y_i dan nilai prediksi dilambangkan dengan f_i .

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - f_i|$$

Formula 2. 2 Rumus MAE

Sumber : [55]

2.4.3.2 RMSE

Root Mean Square Error (RMSE) merupakan nilai kuadrat dari perbedaan yang terdapat pada nilai sebenarnya dengan nilai prediksi. Formula ini memberikan kita hasil, yaitu seberapa besar tingkat *error* atau kesalahan pada prediksi yang telah dikembangkan.

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - f_i)^2}$$

Formula 2. 3 Rumus RMSE

Sumber : [55]

2.4.3.3 MAPE

Mean Absolute Percentage Error (MAPE) merupakan metrik yang digunakan dalam mengukur sebuah akurasi dalam model prediksi. Umumnya pengukuran ini digunakan dalam menangani kasus regresi. MAPE secara singkat digunakan untuk melihat kesalahan atau *error* pada prediksi dalam persentase.

$$MAPE = \frac{100}{N} \sum_{i=1}^N \left| \frac{(y_i - f_i)}{y_i} \right|$$

Formula 2. 4 Rumus MAPE

Sumber : [55]

Terdapat tabel yang digunakan dalam penelitian untuk memahami performa model prediksi dengan pengukuran MAPE. Tabel 2.1 menunjukkan tingkat keoptimalan dari performa sebuah model dari nilai ini. Tabel akan digunakan dalam tahapan evaluasi performa model dan dijadikan sebagai patokan ataupun panduan dalam menentukan keoptimalan performa model prediksi yang telah dibuat.

Tabel 2. 2 Tingkat Performa Model Berdasarkan Nilai MAPE

Sumber : [56], [57]

<i>Nilai MAPE</i>	Performa
< 10%	Sangat Optimal
10% - 20%	Cukup Optimal
20% - 50%	Standar
50%	Tidak Optimal