

**RANCANG BANGUN APLIKASI PREDIKSI
HARGA ETHEREUM MENGGUNAKAN ALGORITMA
LONG SHORT-TERM MEMORY (LSTM)**

**AKMAL ZUHDY PRASETYA
H071191035**



**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS HASANUDDIN
MAKASSAR
2024**

**RANCANG BANGUN APLIKASI PREDIKSI
HARGA ETHEREUM MENGGUNAKAN ALGORITMA
LONG SHORT-TERM MEMORY (LSTM)**

**AKMAL ZUHDY PRASETYA
H071191035**

Skripsi

Diajukan sebagai salah satu syarat untuk mencapai gelar sarjana

Program Studi Sistem Informasi

pada



**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS HASANUDDIN
MAKASSAR**

2024

**RANCANG BANGUN APLIKASI PREDIKSI
HARGA ETHEREUM MENGGUNAKAN ALGORITMA
LONG SHORT-TERM MEMORY (LSTM)**

AKMAL ZUHDY PRASETYA
H071191035

Skripsi,

telah dipertahankan di depan Panitia Ujian Sarjana Sistem Informasi pada 19
Agustus 2024 dan dinyatakan telah memenuhi syarat kelulusan


pada

Program Studi Sistem Informasi
Departemen Matematika
Fakultas Matematika & Pengetahuan Alam
Universitas Hasanuddin
Makassar

Mengesahkan:

Pembimbing Utama,

Pembimbing Pendamping,


Dr. Eng. Armin Lawi, S.Si., M.Eng.
NIP. 197204231995121001


Muhammad Sadno, S.Si., M.Si.
NIP. 199008162022043001

Mengetahui:
Ketua Program Studi,


Prof. Drs. Jeffry Kusuma, Ph.D.
NIP. 196411121987031002



PERNYATAAN KEASLIAN SKRIPSI DAN PELIMPAHAN HAK CIPTA

Dengan ini saya menyatakan bahwa, skripsi berjudul "RANCANG BANGUN APLIKASI PREDIKSI HARGA ETHEREUM MENGGUNAKAN ALGORITMA LONG SHORT-TERM MEMORY (LSTM)" adalah benar karya saya dengan arahan dari pembimbing (Dr.Eng. Armin Lawi, S.Si., M.Eng. sebagai Pembimbing Utama dan Muhammad Sadno, S.Si., M.Si. sebagai Pembimbing Pendamping). Karya ilmiah ini belum diajukan dan tidak sedang diajukan dalam bentuk apapun kepada perguruan tinggi mana pun. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan maupun tidak diterbitkan dari penulis lain telah disebutkan dalam teks dan dicantumkan dalam Daftar Pustaka skripsi ini. Apabila di kemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan skripsi ini adalah karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut berdasarkan aturan yang berlaku.

Dengan ini saya melimpahkan hak cipta (hak ekonomis) dari karya tulis saya berupa skripsi ini kepada Universitas Hasanuddin.



UCAPAN TERIMA KASIH

Dengan segala hormat dan rasa syukur kepada Allah SWT Tuhan Yang Maha Esa, penulis menyampaikan rasa terima kasih karena telah diberi kesempatan dalam menyelesaikan penelitian yang berjudul “Rancang Bangun Aplikasi Prediksi Harga Ethereum Menggunakan Algoritma *Long Short-Term Memory* (LSTM)”. Penulis menyadari sepenuhnya bahwa penulisan suatu karya ilmiah bukanlah suatu hal yang mudah dan perlu dukungan dari banyak pihak. Oleh karena itu pada kesempatan ini dengan segala kerendahan hati penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Kedua orang tua yaitu ibu dan bapak, terima kasih telah mengajarkan penulis untuk bertanggung jawab atas segala hal, terima kasih telah memberikan dukungan serta doa kepada penulis, terima kasih telah mau mendengarkan keluh kesah penulis dan menuntun penulis hingga saat ini. Semoga Allah memberi kesehatan dan memberi umur panjang sehingga dapat melihat penulis menjadi insan yang diharapkan dan dapat dibanggakan.
2. Rektor Universitas Hasanuddin, Bapak Prof. Dr. Ir. Jamaluddin Jompa, M.Sc. beserta jajarannya.
3. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam, Dr. Eng. Amiruddin beserta jajarannya.
4. Ketua Departemen Matematika Fakultas Matematika Dan Ilmu Pengetahuan Alam Universitas Hasanuddin, Bapak Prof. Dr. Nurdin, S.Si., M.Si., atas seluruh ilmu dan saran-saran yang telah diberikan.
5. Ketua Program Studi Sistem Informasi, Bapak Dr. Khaeruddin, M.Sc. atas seluruh ilmu dan saran-saran yang telah diberikan.
6. Ketua Program Studi Sistem Informasi periode sebelumnya, Bapak Dr. Hendra, S.Si., M.Kom. yang telah senantiasa membantu dan memberikan arahan selama masa studi penulis.
7. Pembimbing Utama penulis Bapak Dr. Eng. Armin Lawi, S.Si., M.Eng. yang telah senantiasa membantu, membimbing, dan memberikan arahan selama masa studi penulis khususnya dalam masa penyusunan skripsi.
8. Pembimbing Pertama penulis Bapak Muhammad Sadno, S.Si., M.Si. yang telah senantiasa membantu, membimbing, dan memberikan arahan selama masa studi penulis khususnya dalam masa penyusunan skripsi ini.
9. Kedua dosen penguji, Ibu Rozalina Amran, S.T., M.Eng. dan Bapak Ir. Eliyah Acantha Manapa Sampetoding, S.Kom., M.Kom. yang telah memberikan kritik dan masukan yang bermanfaat dalam penelitian tugas akhir ini sehingga skripsi ini dapat tersusun dengan baik.
10. Bapak/Ibu Dosen Program Studi Sistem Informasi beserta seluruh tenaga pendidik yang telah memberikan ilmu dan mendidik penulis selama masa perkuliahan. Serta kepada seluruh staf dan pegawai Departemen Matematika yang telah membantu penulis terutama dalam segala proses administrasi.
11. Sahabat penulis yaitu Adelia Tahir, S.Psi., terima kasih atas kehadiran dan dukungan yang telah diberikan sejak awal hingga akhir penelitian ini, terima kasih telah menyelamatkan penulis dari berbagai peristiwa kritis dan negatif yang terjadi selama keberlangsungan penelitian ini. Semoga Allah memberikan kesehatan dan memberikan umur panjang sehingga dapat berkembang menjadi manusia yang diimpikan serta terus menyebarkan hal-hal positif kepada makhluk lain di dunia ini.

Makassar, 1 Agustus 2024



Akmal Zuhdy Prasetya
H071191035



ABSTRAK

AKMAL ZUHDY PRASETYA, **Rancang Bangun Aplikasi Prediksi Harga Ethereum Menggunakan Algoritma Long Short-Term Memory (LSTM)** (dibimbing oleh Dr. Eng. Armin Lawi, S.Si., M.Eng. dan Muhammad Sadno, S.Si., M.Si.).

Ethereum merupakan salah satu jenis *cryptocurrency* terbesar dalam persaingan pasar global yang berfungsi sebagai platform *open-source* berbasis *blockchain*, menawarkan kelebihan seperti fleksibilitas *smart contract* dan kemitraan strategis dengan perusahaan terkemuka. Dengan fluktuasi harga yang tinggi dan ketatnya persaingan dalam pasar global, kebutuhan akan model prediksi harga yang akurat serta kemudahan dalam menavigasi informasi relevan menjadi penting. Penelitian ini berfokus pada pemanfaatan algoritma *Long Short-Term Memory* (LSTM), sebuah metode *machine learning* untuk membuat model yang dapat melakukan prediksi harga Ethereum kemudian mengintegrasikannya ke dalam suatu aplikasi web yang interaktif dengan bantuan *framework* Streamlit. Metode kuantitatif digunakan dalam penelitian ini untuk menganalisis data numerik harga Ethereum melalui tiga tahap *preprocessing*: seleksi fitur, normalisasi data, dan segmentasi data. Penelitian menghasilkan sebuah aplikasi web interaktif dengan model LSTM yang diterapkan menunjukkan performa yang baik ketika diuji menggunakan metrik evaluasi *Root Mean Squared Error* (RMSE) dengan nilai akurasi masing-masing sebesar 97% pada data latih dan 98% pada data uji.

Kata kunci: *Cryptocurrency*, Ethereum, *Blockchain*, *Long Short-Term Memory*, LSTM, *Machine Learning*, Streamlit, *Preprocessing*, *Root Mean Squared Error*, RMSE



ABSTRACT

AKMAL ZUHDY PRASETYA, **Design and Development of an Ethereum Price Prediction Application Using the Long Short-Term Memory (LSTM) Algorithm** (supervised by Dr. Eng. Armin Lawi, S.Si., M.Eng. dan Muhammad Sadno, S.Si., M.Si.).

Ethereum is one of the largest cryptocurrencies in the global market, functioning as an open-source blockchain-based platform that offers advantages such as smart contract flexibility and strategic partnerships with leading companies. Due to high price volatility and intense competition in the global market, the need for an accurate price prediction model and ease of navigating relevant information is crucial. This study focuses on utilizing the Long Short-Term Memory (LSTM) algorithm, a machine learning method, to create a model capable of predicting Ethereum prices and then integrating it into an interactive web application using the Streamlit framework. A quantitative method is employed in this research to analyze numerical data of Ethereum prices through three preprocessing stages: feature selection, data normalization, and data segmentation. The study results in an interactive web application where the applied LSTM model demonstrates strong performance when evaluated using the Root Mean Squared Error (RMSE) metric, achieving accuracy rates of 97% on training data and 98% on test data.

Keywords: Cryptocurrency, Ethereum, Blockchain, Long Short-Term Memory, LSTM, Machine Learning, Streamlit, Preprocessing, Root Mean Squared Error, RMSE



DAFTAR ISI

	Halaman
HALAMAN JUDUL.....	i
HALAMAN PENGANTAR.....	ii
HALAMAN PENGESAHAN.....	iii
PERNYATAAN KEASLIAN SKRIPSI DAN PELIMPAHAN HAK CIPTA.....	iv
UCAPAN TERIMA KASIH.....	v
ABSTRAK.....	vi
ABSTRACT.....	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR.....	x
DAFTAR TABEL.....	xi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Maksud dan Tujuan Penelitian.....	2
1.4 Manfaat Penelitian.....	2
1.5 Teori.....	2
1.6 Penelitian Terkait.....	14
BAB II METODE PENELITIAN.....	16
2.1 Lokasi Penelitian.....	16
2.2 Teknik Penelitian.....	16
2.3 Desain Lingkungan Sistem Penelitian.....	16
2.4 Alur Kerja Penelitian.....	17
2.5 Pengambilan Data.....	18
2.6 Arsitektur Sistem.....	18
2.7 Instrumen Penelitian.....	20
BAB III HASIL DAN PEMBAHASAN.....	21
3.1 Hasil Pengambilan Data.....	21
3.2 <i>Preprocessing Data</i>	21
3.3 Pemodelan dan Implementasi Algoritma LSTM.....	24
Model.....	25
an Aplikasi Web.....	25
IV DAN SARAN.....	30
.....	30
.....	30



DAFTAR PUSTAKA.....31



Optimized using
trial version
www.balesio.com

DAFTAR GAMBAR

Gambar 1.1 <i>Design theory</i>	3
Gambar 1.2 Kategori <i>machine learning</i> dan beberapa contoh algoritma.....	4
Gambar 1.3 Contoh neuron ANN.....	5
Gambar 1.4 <i>Feed forward neural network</i>	5
Gambar 1.5 <i>Multi layered feed forward neural network</i>	6
Gambar 1.6 <i>Recurrent neural network</i>	6
Gambar 1.7 Modul berulang pada RNN memiliki satu lapisan.....	7
Gambar 1.8 Modul berulang pada LSTM memiliki 4 lapisan.....	7
Gambar 1.9 Keterangan simbol diagram.....	7
Gambar 1.10 <i>Cell state</i> pada LSTM.....	8
Gambar 1.11 <i>Gates</i> pada LSTM.....	8
Gambar 1.12 Langkah pertama proses kerja LSTM.....	8
Gambar 1.13 Langkah kedua proses kerja LSTM.....	9
Gambar 1.14 Langkah ketiga proses kerja LSTM.....	9
Gambar 1.15 Popularitas beberapa bahasa pemrograman selama 5 tahun terakhir.....	10
Gambar 2.1 Desain lingkungan sistem penelitian.....	17
Gambar 2.2 Diagram alur kerja penelitian.....	18
Gambar 2.3 Diagram <i>use case</i> arsitektur sistem.....	19
Gambar 2.4 <i>Acitivity diagram</i> untuk menentukan waktu prediksi dan jenis prediksi.....	19
Gambar 2.5 <i>Acitivity diagram</i> untuk eksplorasi infografis.....	20
Gambar 3.1 Dataset Ethereum.....	21
Gambar 3.2 Kolom Avg.....	22
Gambar 3.3 <i>Pearson Correlation Heatmap</i>	22
Gambar 3.4 Sebelum dan setelah proses normalisasi data.....	23
Gambar 3.5 Ukuran data latih dan data uji.....	23
Gambar 3.6 Ukuran data latih dan data uji terbaru.....	23
Gambar 3.7 Detail arsitektur model LSTM.....	24
Gambar 3.8 Proses pelatihan model LSTM.....	24
Gambar 3.9 Skor RMSE untuk model pada data latih dan data uji.....	25
Gambar 3.10 Tampilan keseluruhan aplikasi web.....	25
Gambar 3.11 Tampilan <i>Historical Price Trend</i>	26
Gambar 3.12 Tampilan <i>Volume Traded</i>	26
Gambar 3.13 Tampilan <i>Moving Averages</i>	27
Gambar 3.14 Tampilan <i>Sidebar</i>	28
Gambar 3.15 Tampilan <i>Prediction Chart</i>	29



DAFTAR TABEL

Tabel 1.1 Perbandingan Penelitian Sebelumnya	15
Tabel 2.1 Tahapan Penelitian	16



BAB I

PENDAHULUAN

1.1 Latar Belakang

Cryptocurrency adalah mata uang digital atau virtual yang digunakan sebagai bentuk pertukaran dan transfer aset digital. Lee (2015) mengemukakan bahwa *cryptocurrency* termasuk ke dalam kelompok mata uang digital, yang berarti *cryptocurrency* menjadi jenis yang signifikan karena ditandai dengan karakteristik yang berbeda, yakni pemanfaatan teknologi *blockchain* yang dapat mengatasi masalah pengeluaran ganda dan tidak memerlukan perantara pihak ketiga. Garcia dkk. (2014) menjelaskan bahwa *cryptocurrency* menggunakan kriptografi untuk melakukan aset dengan aman, mengontrol dan mengatur penambahan mata uang kripto, dan mengamankan transaksi. Oleh karena itu, *cryptocurrency* dapat disebut sebagai mata uang.

Kim dkk. (2021) mengemukakan studi tentang prediksi harga *cryptocurrency* telah secara aktif dilakukan di berbagai disiplin ilmu seiring dengan meningkatnya minat pada *cryptocurrency* serta algoritma fundamental. Berdasarkan survey yang dilakukan oleh CoinMarketCap pada tahun 2021, Bitcoin dan Ethereum telah tercatat sebagai *cryptocurrency* paling representatif dengan volume perdagangan terbesar dan memiliki kapitalisasi pasar tertinggi. Per April 2021, Bitcoin telah mencatat volume perdagangan USD 64 miliar dan kapitalisasi pasar USD 1.304 miliar, dan Ethereum telah mencapai volume perdagangan \$38 miliar USD dan kapitalisasi pasar sebesar \$265 miliar USD.

Menurut Lee dkk. (2018) salah satu *cryptocurrency* selain Bitcoin adalah Ethereum, yaitu platform *open-source* berbasis *blockchain* yang mendukung *Turing-Complete smart contracts*. Token nilai blockchain Ethereum ini disusun pada tahun 2013 oleh Vitalik Buterin dan kemudian dikembangkan dengan dana sebesar \$18 juta USD dalam bentuk Bitcoin yang diperoleh melalui penjualan Ethereum publik online pada tahun 2014.

Sebagai platform *open-source blockchain*, Ethereum memiliki beberapa kelebihan dan kekurangan. Dilansir dari CryptoEQ pada tahun 2023, kelebihan dari Ethereum adalah termasuk sebagai *liquid digital asset* terbesar kedua setelah Bitcoin, serta fleksibilitas fitur *smart contract* Ethereum memungkinkan banyak proyek yang dibangun pada Ethereum *blockchain* sekaligus berpartisipasi pada *Web 3.0*. Kelebihan yang lain adalah, Ethereum memiliki kemitraan dengan perusahaan yang terkemuka, sehingga dapat memiliki efek *networking* yang luar biasa. Adapun kekurangan pada Ethereum saat ini terletak pada proses peningkatan besar-besaran yang saat ini masih berjalan dan belum selesai, yaitu proyek ini memiliki resiko yang sangat besar. Selain itu, terdapat masalah penskalaan yang membatasi kemampuan pengembangan dan proyek lain dalam menciptakan pengalaman pengguna yang memuaskan. Kekurangan terakhir yakni persaingan yang ketat dengan jenis *blockchain* lain seperti Cosmos, Solana, dan Cardano yang dapat mengganggu popularitas dari Ethereum.

Prediksi harga *cryptocurrency* adalah salah satu area yang sedang tren di kalangan peneliti. Menurut Khedr dkk. (2021), fluktuasi harga *cryptocurrency* yang luas memotivasi kebutuhan mendesak untuk model yang akurat untuk memprediksi harganya. Sejalan dengan hal tersebut, Vaddi (2020) mengatakan bahwa algoritma *machine learning* dapat diterapkan pada data *cryptocurrency* untuk memprediksi pergerakan harga di masa depan.

Bonetto & Latzko (2020) mengemukakan bahwa *machine learning* (ML) adalah istilah umum yang mencakup banyak teori dan algoritma heterogen (misalnya, pembelajaran statistik, *Bayesian Network*, *Self-Organizing Maps*, dll.) yang dikembangkan selama rentang waktu 70 tahun. Tujuan dari setiap strategi tersebut adalah sebagai mengingat sebagian pengamatan sistem dan model parametrik yang diprediksi berdasarkan keadaan sistem yang diamati, serta menemukan parameter terbaik untuk model yang diberikan untuk memaksimalkan akurasi prediksi sehubungan dengan



Terdapat beberapa model ML yang dapat digunakan untuk melakukan prediksi harga Ethereum, antara lain penelitian yang dilakukan oleh McNally dkk. (2018) menggunakan *Recurrent Neural Network* (RNN) dan *Long Short-Term Memory* (LSTM), penelitian dari Liu dkk. (2017) menggunakan *Artificial Neural Network* (ANN) dari berbagai arsitektur, serta penelitian oleh Peng dkk. (2018) yang menggunakan *Support Vector Machine* (SVM) dan lain-lain.

Penelitian yang dilakukan oleh McNally dkk. (2018) berfokus pada prediksi tren Bitcoin (masalah klasifikasi) dengan menggunakan RNN dan LSTM. Penelitian tersebut menggunakan harga periode sebelumnya dan beberapa indikator teknis seperti rata-rata pergerakan sederhana untuk memprediksi. Keakuratan yang diperoleh untuk memprediksi perubahan arah tren Bitcoin berada pada kisaran akurasi 51-52% untuk kedua metode tersebut.

Hochreiter & Schmidhuber (2018) mengemukakan bahwa LSTM adalah algoritma improvisasi yang neuronnya dapat menyimpan memori di salurannya untuk mengurangi masalah gradien yang hilang secara efektif. Menurut Li dkk. (2019) hal tersebut membuat LSTM menjadi populer dalam memprediksi deret waktu. Keuntungan utama dari LSTM menurut Ma & Mao (2021) terletak pada mengatasi beberapa kekurangan RNN seperti gradien yang hilang dan meledak dalam proses pelatihan.

Berdasarkan beberapa penelitian yang telah disebutkan di atas mengenai prediksi *cryptocurrency* utamanya Ethereum, terdapat kurangnya sistem yang dapat mempermudah dalam menavigasi proses prediksi sehingga penulis ingin merancang aplikasi prediksi harga Ethereum dengan menggunakan algoritma LSTM berbasis web yang interaktif.

1.2 Rumusan Masalah

Berdasarkan penjelasan pada latar belakang, permasalahan yang ingin dibahas dalam penelitian ini, yaitu:

- Bagaimana *preprocessing* data harga Ethereum dalam melatih model LSTM?
- Bagaimana kinerja dari model LSTM dalam memprediksi harga Ethereum di masa mendatang?
- Bagaimana cara mengintegrasikan model ke dalam bentuk aplikasi web yang interaktif?

1.3 Maksud dan Tujuan Penelitian

Berdasarkan penjelasan pada latar belakang, tujuan dari penelitian ini, yaitu:

- Untuk menerapkan *preprocessing* data harga Ethereum dalam melatih model LSTM.
- Untuk menganalisis kinerja dari model LSTM dalam memprediksi harga Ethereum di masa mendatang.
- Untuk mengintegrasikan model ke dalam bentuk aplikasi web yang interaktif.

1.4 Manfaat Penelitian

Adapun penelitian ini diharapkan dapat menghasilkan yang hal bermanfaat yaitu sebagai berikut:

- Dapat menjadi landasan rujukan untuk peneliti selanjutnya dengan tema penelitian yang serupa.
- Dapat memberikan informasi mengenai *preprocessing* data harga Ethereum dengan memanfaatkan model LSTM dan proses integrasinya ke dalam bentuk aplikasi web pada pembaca.

1.5 Teori

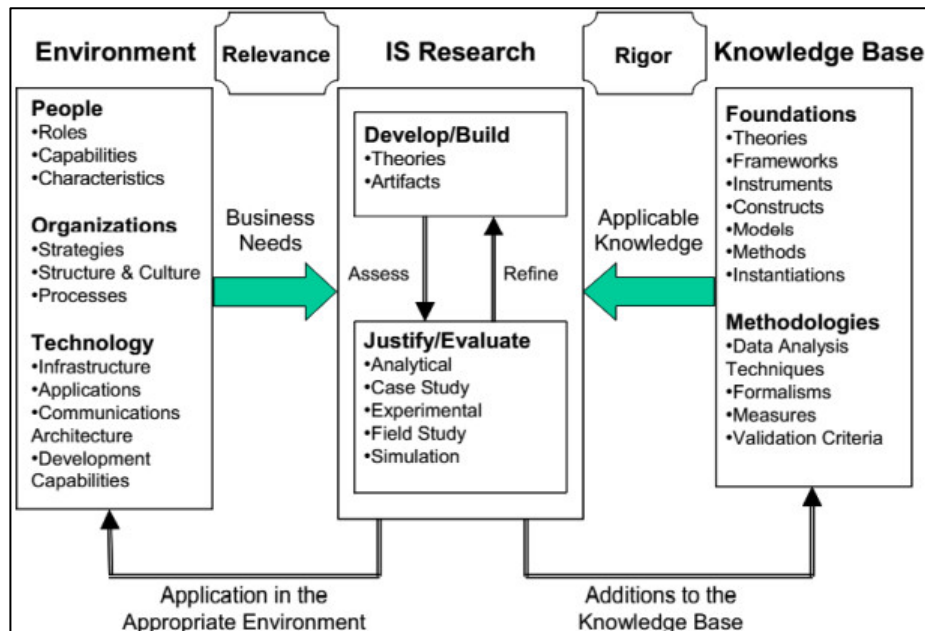
Adapun penelitian ini diharapkan dapat menghasilkan yang hal bermanfaat yaitu sebagai berikut:

1.5.1 Lingkungan Sistem Informasi

Hevner dkk., (2004) mendeskripsikan batas-batas *design theory* dalam penelitian sistem informasi melalui konsep *framework* yang didesain untuk memahami dan sebagai pedoman untuk melakukan dan mengevaluasi penelitian sistem informasi. Hevner dkk., (2004) mengemukakan konsep *framework* untuk memahami, mengeksekusi dan mengevaluasi penelitian pada sistem yang menggabungkan ilmu perilaku dan paradigma ilmu desain, *Environment* merupakan *ig* menggambarkan mengenai ruang masalah pada penelitian sistem informasi yang *people, organizations* atau bisnis dan teknologi baik yang sudah ada maupun yang masih an. *Business needs* terdiri dari tujuan, tugas, masalah dan peluang yang arkan kebutuhan yang dirasakan oleh orang-orang dalam organisasi. *Knowledge base* bagian yang menyediakan bahan sumber dan metode dalam menjalankan penelitian



sistem informasi. Bagian tersebut terdiri dari landasan teori dan metodologi. *Rigor* atau ketelitian merupakan bagian yang dapat dicapai ketika fondasi dan metodologi yang digunakan diterapkan dengan baik. Adapun bentuk dari konsep *framework*, dapat di lihat pada gambar di bawah.



Gambar 1.1 Design theory (IS Theory, 2017)

1.5.2 Cryptocurrency

a. Definisi Cryptocurrency

Adapun beberapa definisi *cryptocurrency* adalah sebagai berikut.

- 1) *Cryptocurrency* menurut Kim dkk. (2016) merupakan sarana pertukaran kriptografi untuk mengamankan transaksi dan mengontrol pembuatan unit baru dari mata uang yang sama.
- 2) *Cryptocurrency* menurut Milutinović (2018) dapat disebut sebagai aset digital yang berfungsi sebagai media pertukaran yang menggunakan kriptografi sehingga semua transaksi diamankan dan dikendalikan oleh sistem sendiri.
- 3) *Cryptocurrency* menurut Farsangi dkk. (2020) merupakan bentuk uang elektronik atau mata uang digital yang menggunakan kriptografi untuk transfer di internet.
- 4) *Cryptocurrency* menurut Mariana (2021) didefinisikan sebagai representasi digital dari nilai atau hak kontraktual yang menggunakan beberapa bentuk teknologi registri terdistribusi serta dapat ditransfer, disimpan atau diperdagangkan secara elektronik.
- 5) *Cryptocurrency* menurut Siddharth dkk. (2022) adalah pertukaran terkomputerisasi berbasis jaringan yang membuat peniruan dan pembelanjaan ganda hampir tidak mungkin.

b. Ethereum

Buterin (2014) mengemukakan bahwa ethereum mewakili blockchain dengan *built-in Turing-complete smart contracts*, yaitu lapisan abstrak yang memungkinkan siapa pun untuk membuat aturan mereka sendiri untuk kepemilikan, format transaksi, dan fungsi transisi status. *Smart contracts* merupakan seperangkat aturan kriptografi yang dieksekusi hanya jika kondisi tertentu



dkk. (2018) juga mengemukakan bahwa ethereum merupakan platform *open-source blockchain* yang mendukung *Turing-Complete smart contracts*. Token nilai *blockchain* ini disusun pada tahun 2013 oleh Vitalik Buterin dan kemudian dikembangkan dengan

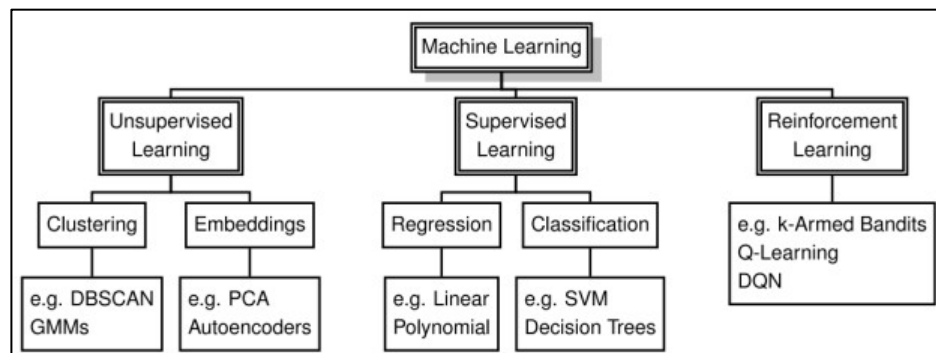
dana sebesar US\$18 juta dalam bentuk Bitcoin yang diperoleh melalui penjualan Ethereum publik *online* pada tahun 2014.

Menurut Angela & Sun (2020), Ethereum adalah salah satu *altcoin* yang paling banyak ditambang dan berada di posisi kedua setelah Bitcoin pada periode 2017-2019. Per Oktober 2019, pasar *cryptocurrency* didominasi oleh Bitcoin dan Ethereum yang nilainya melebihi setengah dari total pasar *cryptocurrency* di seluruh dunia, yaitu Ethereum mendominasi 65% pasar *cryptocurrency* setelah Bitcoin.

1.5.3 Machine Learning (ML)

Bonetto & Latzko (2020) menjelaskan bahwa ML mencakup banyak teori dan algoritma heterogen seperti pembelajaran statistik, *Bayesian Network*, *Self-Organizing Maps*, dan lain-lain. Dalam rentang waktu 70 tahun, ML dikembangkan dengan tujuan strategi pembelajaran mesin yang dapat dirumuskan dalam beberapa bentuk. Pertama, mengingat sebagian pengamatan sistem serta model parametrik yang menghasilkan prediksi berdasarkan keadaan sistem yang diamati. Kedua, menemukan parameter terbaik untuk model yang diberikan untuk memaksimalkan akurasi prediksi sehubungan dengan tugas dan menetapkan parameter tersebut.

Algoritma ML dapat dikelompokkan ke dalam tiga kategori makro, yaitu *Unsupervised Learning*, *Supervised Learning*, dan *Reinforcement Learning*. *Unsupervised Learning* mengacu pada kumpulan algoritma yang mengekstraksi informasi dari data yang tidak berlabel seperti, data yang klasifikasi spesifiknya tidak diketahui atau diperlukan. Contoh dari tipe ML ini adalah *Clustering*. *Clustering* mengacu pada pengelompokan data ke dalam sejumlah set (*nonoverlapping*) berdasarkan beberapa ukuran kesamaan. Kategori selanjutnya adalah *Supervised Learning*, yaitu kategori yang mencakup kumpulan algoritma dengan beberapa *input* yang memprediksi beberapa karakteristiknya seperti, menggambarkan lalu memprediksi digit sebenarnya yang diwakili oleh gambar tersebut. Dan yang terakhir adalah *Reinforcement Learning*, yaitu algoritma yang mengatasi masalah dalam mengajar agen untuk berinteraksi dengan lingkungan berdasarkan pengamatan sesuai kondisi. Secara terstruktur, kategori ML dapat dilihat seperti yang ditunjukkan sebagai berikut:



Gambar 1.2 Kategori *machine learning* dan beberapa contoh algoritma (Caldegren, 2018)

1.5.4 Neural Network (NN)

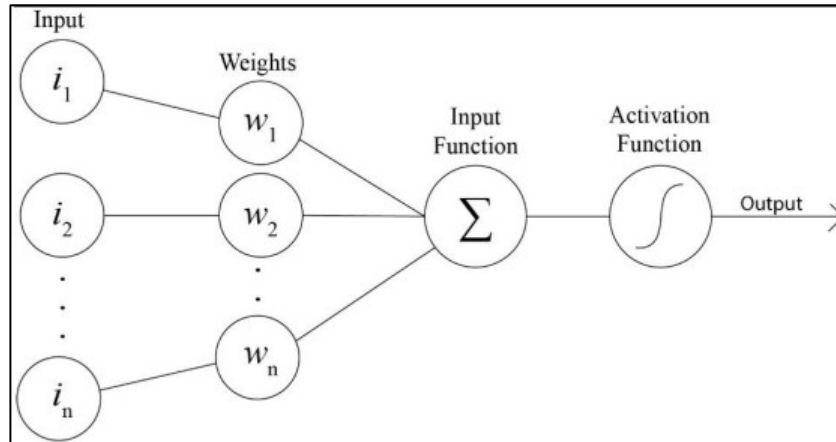
a. Definisi Neural Network

Neural Network, juga disebut *Artificial Neural Network* (ANN) menurut Haykin dkk. (1999) adalah algoritma yang mencoba mereplikasi fungsi otak manusia. Lebih lanjut Caldegren (2018) menyatakan bahwa ANN dapat mensimulasikan bagian kecil dari otak dengan membuat model jaringan neuron. Pada model neuron tersebut, terdapat beberapa lapisan yaitu lapisan input, lebih lapisan tersembunyi, lapisan output, dan sinapsis atau koneksi.

Menurut Caldegren (2018) jaringan yang memiliki lebih dari satu lapisan tersembunyi, dikenal sebagai *deep neural network*. Setiap lapisan dalam jaringan saraf memiliki jumlah neuron yang berubah-ubah. Setiap neuron terhubung pada neuron di tingkat berikutnya dengan



sinapsis berbobot. Bobot sinapsis inilah yang mempengaruhi hasil akhir. ANN menggunakan *backpropagation* untuk meningkatkan hasil pada masalah yang belum terlihat oleh jaringan. Satu lapisan neuron dengan sinapsis yang terhubung dapat dilihat pada gambar di bawah ini.



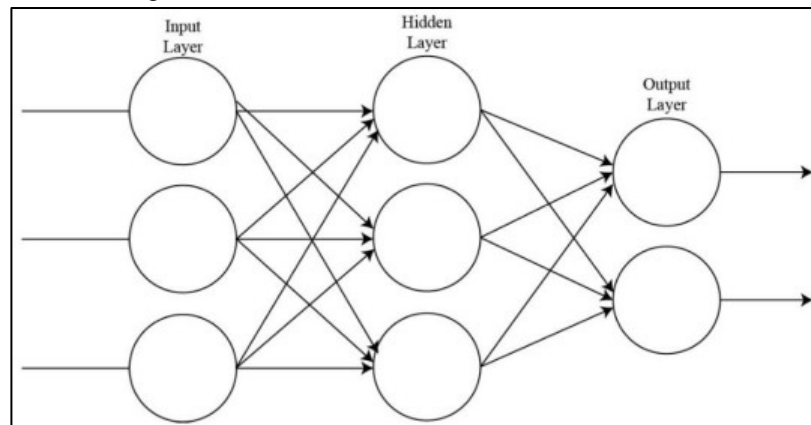
Gambar 1.3 Contoh neuron ANN (Caldegren, 2018)

b. Tipe Neural Network

1) *Feed Forward Neural Networks*

Bentuk sederhana dari ANN yang dikemukakan oleh Lippmann (1988) adalah *feed forward neural network*. Jaringan ini mengambil satu atau beberapa *input* dan memasukkannya melalui sinapsis ke lapisan tersembunyi pertama, kemudian melalui lapisan tersembunyi yang tersisa jika ada, dan akhirnya sampai ke lapisan output.

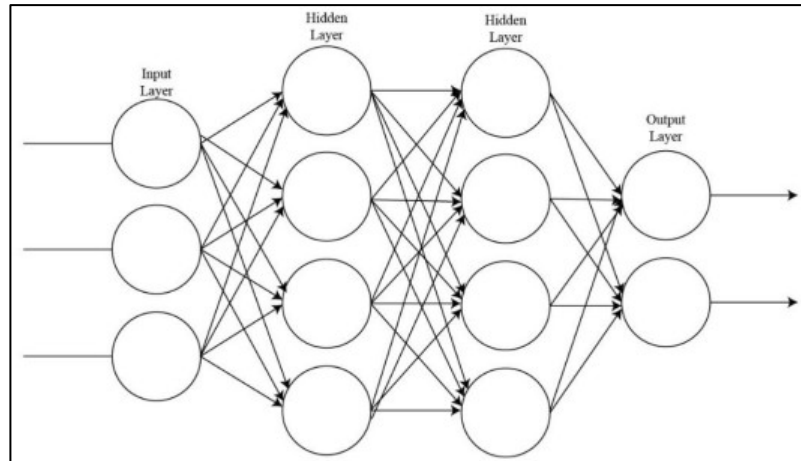
Pada sebagian besar jaringan saraf, mungkin ada sejumlah lapisan tersembunyi dengan sejumlah sinapsis di masing-masing, sejumlah neuron input dan sejumlah output. Pada jaringan saraf, terdapat proses *forward pass* yaitu ketika lapisan input menerima tiga input lalu meneruskannya pada satu lapisan tersembunyi yang berisi tiga neuron. Setelah itu, lapisan tersembunyi meneruskan input menuju dua output. Gambaran jelas mengenai proses *forward pass* dapat dilihat sebagai berikut.



Gambar 1.4 *Feed forward neural network* (Caldegren, 2018)



Keunggulan jaringan ini mampu mengklasifikasikan data tetapi tidak sebaik dalam buat prediksi yaitu ketika hubungan data tersebar sepanjang waktu. Adapun jenis lain bentuk ANN ini dapat dilihat pada gambar berikut.

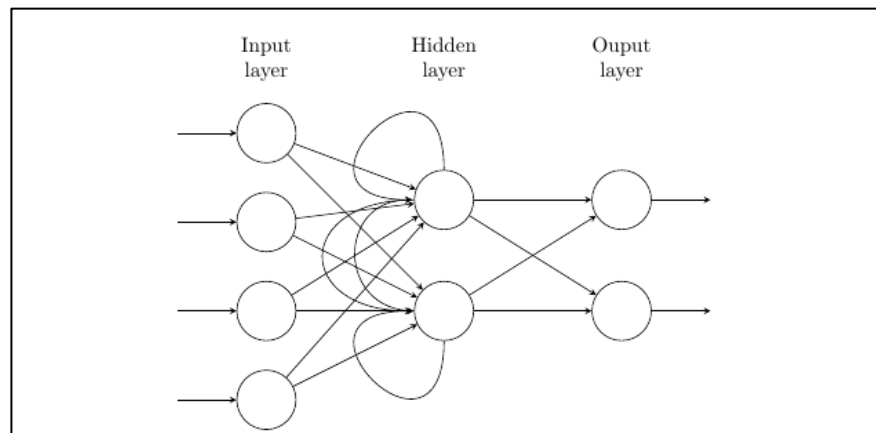


Gambar 1.5 Multi layered feed forward neural network (Johnson, 2018)

Berdasarkan gambar 2.4 dapat dipahami bahwa *multi layered deep feed forward neural network* memiliki dua lapisan yaitu ketika setiap neuron di lapisan tersembunyi pertama terhubung ke setiap neuron di lapisan kedua.

2) *Recurrent Neural Networks (RNN)*

Pada RNN beberapa output dari lapisan tersembunyi kembali dan menjadi input. Egeli dkk., (2019) menyatakan bahwa hal ini menjadikan RNN pilihan yang tepat untuk prediksi deret waktu selain karena memiliki memori internal. RNN telah digunakan secara luas di bidang-bidang seperti prediksi pasar saham. Oleh karena itu, menurut Caldegren (2018) RNN mungkin menjadi pilihan yang lebih cocok, dibandingkan dengan *feed forward neural network*, ketika hubungan yang diperiksa ada dari waktu ke waktu. Gambaran mengenai RNN dapat dilihat sebagai berikut.



Gambar 1.6 Recurrent neural network (Johnson, 2018)

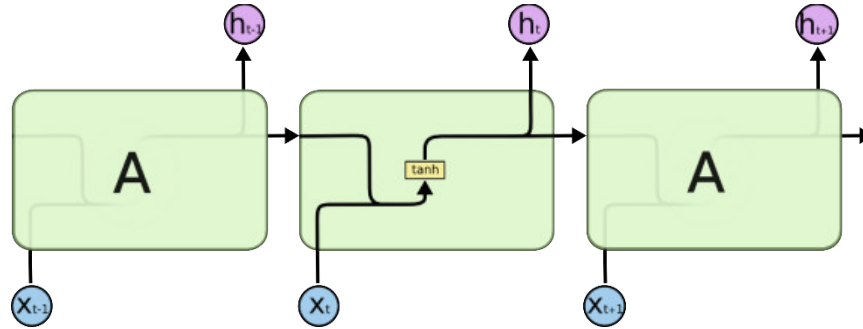
3) *Long Short-Term Memory Networks (LSTM)*



Jaringan *Long Short-Term Memory*, umumnya dikenal sebagai "LSTM", mewakili jenis dari RNN dengan kemampuan untuk memahami dan belajar dari ketergantungan jarak dalam data. LSTM berdasarkan Hochreiter & Schmidhuber (1997) adalah jenis NN yang menggunakan data dan melakukan pemrosesan data melalui algoritma pembelajaran asarkan gradien. Hal ini menghasilkan sebuah algoritma yang meningkatkan kinerjanya an inklusi lebih banyak parameter, secara sederhana LSTM belajar saat menerima data ahan.

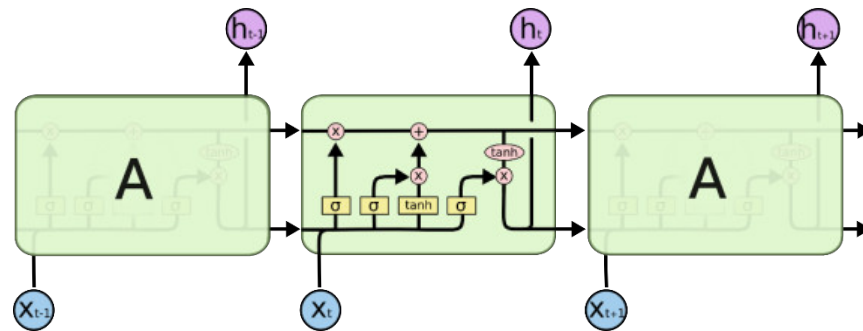
LSTM menurut Olah (2015) secara alamiah dirancang untuk mengatasi masalah ketergantungan jangka panjang. Kemampuan bawaan mereka untuk menyimpan informasi dalam jangka waktu yang panjang adalah karakteristik fundamental, bukan keterampilan yang perlu mereka pelajari melalui pelatihan.

Setiap RNN terdiri dari urutan modul jaringan saraf yang berulang. Pada RNN konvensional, modul-modul berulang ini biasanya memiliki arsitektur dasar, seringkali terdiri hanya dari satu lapisan *tanh*.



Gambar 1.7 Modul berulang pada RNN memiliki satu lapisan (Olah, 2015)

LSTM memiliki arsitektur berbentuk rantai juga, tetapi modul berulang dari LSTM memiliki desain yang berbeda, dibandingkan hanya satu lapisan jaringan saraf, LSTM terdiri dari empat lapisan yang berinteraksi dengan cara yang unik.



Gambar 1.8 Modul berulang pada LSTM memiliki 4 lapisan (Olah, 2015)

Dalam ilustrasi yang diberikan, setiap garis berfungsi sebagai saluran untuk vektor lengkap, melakukan transfer informasi dari *output* satu *node* ke *input node* lainnya. Lingkaran merah muda melambangkan operasi elemen demi elemen, seperti penambahan vektor, sementara kotak kuning mewakili lapisan jaringan saraf yang beradaptasi dan belajar. Garis yang bertemu menunjukkan penggabungan, sedangkan garis yang membelah mengimplikasikan duplikasi isinya dan mengirim salinan ke tujuan yang berbeda.

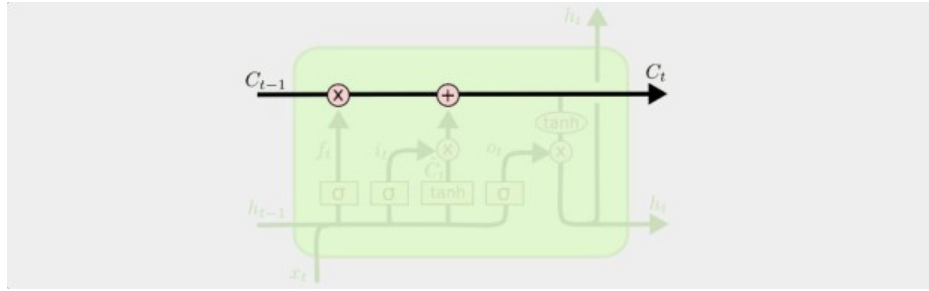


Gambar 1.9 Keterangan simbol diagram (Olah, 2015)



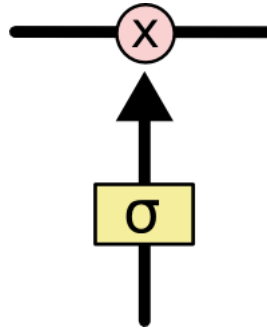
Kunci dari LSTM adalah *cell state*, yang digambarkan sebagai garis horizontal yang melintasi bagian atas diagram. *Cell state* dapat dibayangkan seperti sebuah *conveyor belt*

yang kadang-kadang mengalami interaksi linear kecil. Di sepanjang *conveyor belt* ini, informasi dapat mengalir dengan lancar dan sebagian besar dari informasi tidak berubah.



Gambar 1.10 Cell state pada LSTM (Olah, 2015)

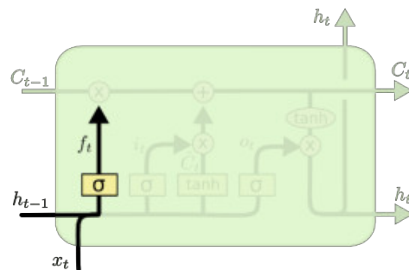
LSTM memiliki kemampuan untuk mengurangi atau memasukkan data ke dalam *cell state*, yang dikendalikan dengan cermat oleh komponen yang disebut *gates*. Gerbang-gerbang ini berfungsi sebagai mekanisme untuk memungkinkan informasi melewati secara selektif, yang terdiri dari lapisan jaringan saraf *sigmoid* yang digabungkan dengan operasi perkalian titik.



Gambar 1.11 Gates pada LSTM (Olah, 2015)

Lapisan *sigmoid* menghasilkan nilai yang berkisar dari nol hingga satu, menunjukkan sejauh mana setiap elemen seharusnya diizinkan untuk melewati. Nilai nol mengindikasikan "tidak mengizinkan apa pun", sementara nilai satu mengindikasikan "mengizinkan semuanya". LSTM menggabungkan tiga gerbang seperti ini untuk melindungi dan mengatur *cell state*.

Dalam proses kerja LSTM, langkah awal melibatkan penentuan informasi mana yang harus dibuang dari *cell state*. Penentuan ini dilakukan melalui penggunaan lapisan *sigmoid* yang dikenal sebagai *forget gate layer*. Lapisan ini menganalisis baik h_{t-1} dan x_t dan menghasilkan nilai yang berkisar dari 0 hingga 1 untuk setiap elemen dalam *cell state* C_{t-1} . Nilai 1 menandakan "dipertahankan sepenuhnya", sementara nilai 0 menandakan "dibuang sepenuhnya".

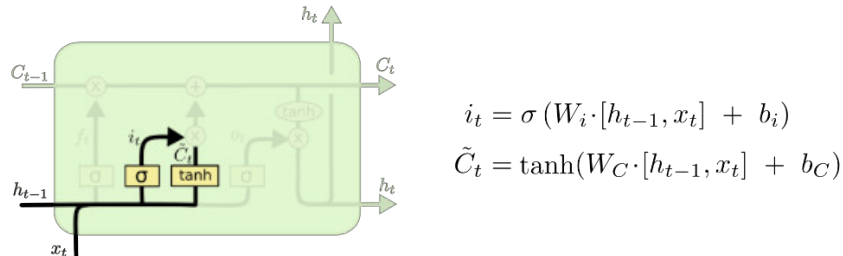


$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

Gambar 1.12 Langkah pertama proses kerja LSTM (Olah, 2015)

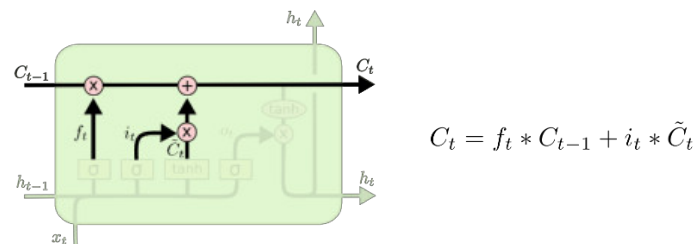


Tahap berikutnya melibatkan penentuan data baru yang akan dimasukkan ke dalam *cell state*, yang terdiri dari dua komponen utama. Pertama, sebuah lapisan *sigmoid* yang dikenal sebagai *input gate layer* menentukan nilai-nilai mana yang harus mengalami modifikasi. Setelah itu, sebuah lapisan *tanh* menghasilkan vektor nilai potensial baru yang disebut sebagai \tilde{C}_t , yang dapat dimasukkan ke dalam *cell state*. Pada tahap berikutnya, akan dilakukan penggabungan kedua elemen ini untuk menghasilkan pembaruan ke dalam *cell state*.



Gambar 1.13 Langkah kedua proses kerja LSTM (Olah, 2015)

Saatnya beralih dari *cell state* sebelumnya, C_{t-1} , ke *cell state* yang telah diperbarui, C_t . Hal ini dapat dicapai dengan mengalikan *cell state* lama dengan f_t , secara efektif menghapus informasi yang kita pilih untuk lupakan sebelumnya. Kemudian, kita menambahkannya dengan $i_t * \tilde{C}_t$, yang mewakili nilai calon baru, yang disesuaikan berdasarkan keputusan kita tentang seberapa besar pembaruan pada setiap nilai *cell state* tersebut.



Gambar 1.14 Langkah ketiga proses kerja LSTM (Olah, 2015)

Terakhir, kita perlu memutuskan *output* apa yang akan dihasilkan. *Output* ini akan berasal dari *cell state*, tetapi akan menjadi versi yang telah mengalami proses filter. Pada awalnya, akan diterapkan lapisan *sigmoid* untuk membuat penentuan tentang aspek-aspek mana dari *cell state* yang harus kita hasilkan sebagai *output*. Selanjutnya, *cell state* akan dikirim melewati fungsi *tanh* (untuk membatasi nilai-nilai dalam rentang -1 hingga 1) dan kemudian mengalikannya dengan hasil gerbang *sigmoid*. Hal ini memastikan bahwa proses hanya menghasilkan komponen-komponen yang telah kita pilih untuk dihasilkan.

1.5.5 Pemrograman

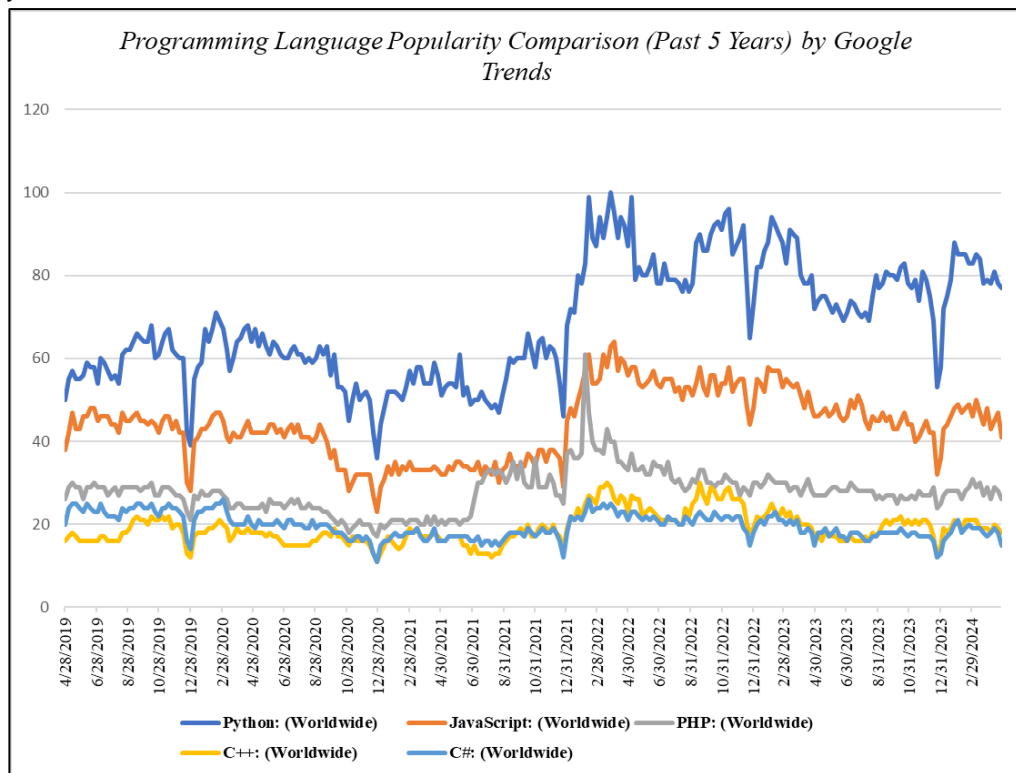
a. Bahasa Pemrograman Python

Berdasarkan penelitian oleh Thangarajah dkk. (2019), Guido van Rossum mengembangkan Python pada akhir tahun 1980-an dan awal tahun 1990-an di *National Research Institute for Mathematics and Computer Science* di Belanda. Python terinspirasi dari beberapa bahasa pemrograman lain seperti *ABC*, *Modula-3*, *C*, *C++*, *Algol-68*, *Smalltalk*, *Unix shell*, dan berbagai skrip lainnya. Python memiliki hak cipta tetapi kode sumbernya (*source code*) tersedia di bawah *GNU General Public License (GPL)*, mirip dengan *Perl*. Meskipun sekarang tim inti yang mengelola Python, Guido van Rossum tetap memainkan peran penting dalam mengarahkan kemajuannya.



Menurut Abu Rayhan & Gross David (2023), Python merupakan bahasa pemrograman tingkat tinggi yang berarti kode dari pemrograman dapat langsung dieksekusi tanpa melakukan proses *compile* atau kompilasi terlebih dahulu (*interpreted*). Python merupakan bahasa pemrograman dinamis dan serba guna, serta memiliki sistem pengolahan sampah otomatis. Python juga sering dijelaskan dengan istilah "*batteries included*", yang berarti bahwa bahasa pemrograman ini dilengkapi dengan beragam modul (*module*) dalam pustaka (*library*) standar yang dapat digunakan untuk berbagai tugas.

Abu Rayhan & Gross David lanjut menjelaskan beberapa alasan mengapa Python berkembang menjadi bahasa pemrograman yang populer yaitu pertama, kemudahan dalam memahami cara kerja bahasa pemrograman berkat *syntax* yang sederhana dan tidak berbelit-belit (*straightforward*) bagi pemula. Kedua, kecanggihannya bahasa dalam memungkinkan penggunaan di berbagai bidang seperti pengembangan web, *data science*, *machine learning*, dan komputasi ilmiah. Ketiga, Python memiliki komunitas pengguna dan pengembang (*developer*) yang besar dan aktif, hal ini menjamin tersedianya sumber daya yang melimpah untuk pembelajaran dan penyelesaian masalah.



Gambar 1.15 Popularitas beberapa bahasa pemrograman selama 5 tahun terakhir (Google Trends, 2024)

Penjelasan lebih lanjut oleh Abu Rayhan & Gross David dalam penelitiannya menyebutkan bahwa terdapat beberapa tren terkini yang memanfaatkan bahasa pemrograman Python, salah satu dari tren tersebut adalah penggunaannya dalam bidang *machine learning* dan *artificial intelligence*. Python sangat cocok untuk pekerjaan-pekerjaan yang ada di dalam bidang-bidang tersebut karena kemudahan penggunaannya serta memiliki banyak perpustakaan (*library*) yang tersedia untuk bidang-bidang tersebut.



Pemrograman

dalam artikel yang ditulis oleh Melitzer (2023), *library* dalam konteks pemrograman merujuk pada sekumpulan kode yang telah ditulis sebelumnya dan dapat digunakan kembali oleh

pengguna untuk mengoptimalkan pekerjaan dan pembuatan aplikasi dengan lebih efisien. Adapun *library* yang digunakan di dalam penelitian ini adalah sebagai berikut.

1) NumPy

NumPy, singkatan dari *Numerical Python*, menurut Ascher dkk. (2001) merupakan ekstensi Python yang memungkinkan manipulasi *array* dengan efisien serta kumpulan objek besar dengan struktur grid. *Array* ini dapat memiliki jumlah dimensi yang bervariasi, mirip dengan urutan standar Python atau matriks aljabar linear. Alasan utama mengapa NumPy diperlukan adalah karena memanipulasi angka dengan koleksi yang besar dalam Python menggunakan struktur data standar seperti *list*, *tuple*, atau *class* terasa lambat dan memakan terlalu banyak ruang.

2) Pandas

Pandas berdasarkan Unpingco (2021) merupakan modul yang kuat yang telah dioptimalkan di atas NumPy, menyajikan sejumlah struktur data yang sangat sesuai untuk analisis data, terutama data seri waktu, dengan tata letak menyerupai *spreadsheet*. Pandas menyediakan objek seperti *dataframe* yang dibangun di atas platform NumPy untuk mempermudah manipulasi data (khususnya untuk data seri waktu) dalam proses statistik. Keunggulan utama Pandas mencakup manipulasi data dan penyesuaian yang cepat, alat untuk pertukaran data antar format dan basis data SQL yang berbeda, penanganan data yang hilang, serta pembersihan data yang tidak teratur.

3) Scikit-learn

Scikit-learn berdasarkan Hao & Ho (2019) merupakan salah satu paket ML yang paling lengkap dan tersedia secara terbuka dalam bahasa pemrograman Python. Scikit-learn membahas empat topik utama dalam ML yaitu transformasi data, *supervised learning*, *unsupervised learning*, serta evaluasi dan pemilihan model. Scikit-learn juga mencakup beragam metode ML yang diimplementasikan secara efisien, dijelaskan dengan baik serta dikelola oleh komunitasnya. Ditambah dengan memanfaatkan berbagai aplikasi luas dari Python, Scikit-learn menjadi semakin terkenal untuk penggunaan terkait ML.

4) TensorFlow

TensorFlow berdasarkan Pattanayak (2023) adalah *library* yang bersifat *open source* dari Google yang secara khusus difokuskan pada *deep learning*. TensorFlow menggunakan graf aliran data komputasional untuk menggambarkan struktur jaringan saraf yang kompleks. Simbol-simbol (*nodes*) dalam graf tersebut mewakili perhitungan matematis, yang juga disebut sebagai *ops* (operasi), sementara tepi-tepi di graf menunjukkan transfer *tensor* data di antara mereka. Selain itu, gradien yang relevan disimpan di setiap node dalam graf komputasional, dan selama tahap *backpropagation*, mereka digabungkan untuk mendapatkan gradien terhadap setiap bobot. *Tensor* sendiri adalah *array* data multidimensi yang digunakan oleh TensorFlow.

5) Keras

Keras berdasarkan Manaswi (2018) adalah salah satu *library* dari Python untuk *deep learning* yang dapat berjalan di atas TensorFlow. Dengan menggunakan TensorFlow sebagai *end*, Keras memungkinkan pengguna untuk fokus pada konsep utama *deep learning*, rti pembuatan lapisan jaringan saraf serta mengatasi detail *tensor*, detail bentuk, dan l matematika secara bersamaan. Ada dua jenis API utama dalam Keras, yaitu sekuensial, didasarkan pada konsep rangkaian lapisan, dan fungsional. API sekuensial merupakan gunaan Keras yang paling umum digunakan.



6) Matplotlib

Matplotlib berdasarkan Unpingco (2021) merupakan alat utama untuk visualisasi grafik ilmiah dalam bahasa pemrograman Python, dengan keunggulan utamanya terletak pada kelengkapannya, yaitu hampir semua jenis plot ilmiah dua dimensi dapat dihasilkan. Matplotlib pertama kali diciptakan oleh John Hunter yang bermula dari kebutuhan pribadi ketika menggunakan Matlab untuk visualisasi ilmiah, namun, saat mulai mengintegrasikan data dari berbagai sumber dengan Python, muncul kesadaran bahwa perlunya solusi Python untuk visualisasi. Oleh karena itu, John menulis versi awal dari Matplotlib.

7) Streamlit

Nápoles-Duarte dkk., (2022) menjelaskan bahwa Streamlit merupakan *open-source* Python *coding framework* untuk membuat aplikasi web dan digunakan oleh peneliti untuk membagikan data set besar dari studi yang telah di terbitkan. Streamlit dapat membantu peneliti untuk menciptakan aplikasi web hanya dengan beberapa baris kode Python, tanpa membutuhkan kemampuan dalam *front-end coding*.

1.5.6 Moving Averages (MA)

Berdasarkan Acquire.Fi, MA adalah indikator statistik yang umum digunakan untuk memberikan wawasan yang jelas tentang tren kinerja aset selama periode waktu tertentu. Rumus untuk menghitung rata-rata bergerak melibatkan penjumlahan harga penutupan aset selama sejumlah periode waktu yang ditentukan, kemudian membaginya dengan jumlah periode yang sama. Proses ini diulang untuk setiap periode waktu berikutnya, menghasilkan garis halus yang mewakili nilai rata-rata aset selama rentang waktu yang ditentukan.

Untuk menghitung MA, perlu ditentukan jumlah periode waktu yang akan digunakan dan kemudian menjumlahkan harga penutupan aset. Jumlah tersebut kemudian dibagi dengan jumlah periode waktu. Nilai yang dihasilkan mencerminkan rata-rata bergerak untuk periode waktu tertentu. Perhitungan ini diulang untuk setiap periode waktu, menghasilkan serangkaian rata-rata bergerak yang dapat dipetakan pada grafik untuk mengidentifikasi tren secara visual.

MA berguna untuk menghilangkan sinyal palsu yang disebabkan oleh pergerakan harga jangka pendek. *Trader* berpengalaman menggunakan rata-rata ini untuk menentukan area kunci seperti *support*, *resistance*, atau momen yang tepat untuk membuka atau menutup posisi. Tujuan dari mengidentifikasi tren melalui MA sangat penting bagi investor jangka panjang yang ingin mengenali tren di tahap awal dan bertindak sesuai dengan itu.

Support adalah kondisi ketika harga suatu aset turun dan mendekati MA dari atas, hal ini berarti harga rata-rata selama periode waktu tertentu memberikan sebuah tingkatan yaitu ketika pembeli dapat membeli aset tersebut sehingga dapat mencegah harga jatuh lebih jauh. Hal ini berguna bagi analisis untuk mengamati pantulan harga ketika mencapai atau mendekati MA dari atas yang mengindikasikan potensi *support*.

Resistance, sebaliknya adalah ketika harga suatu aset naik dan mendekati MA dari bawah, yang berarti bahwa harga rata-rata selama periode waktu tertentu memberikan sebuah tingkatan yaitu ketika penjual dapat memasuki pasar, sehingga dapat mencegah kenaikan harga lebih lanjut. Hal ini berguna dalam analisis untuk melihat pembalikan harga atau perlambatan dalam momentum kenaikan ketika harga mencapai atau mendekati MA dari bawah yang mengindikasikan potensi *resistance*.



Korelasi Pearson

asarkan artikel yang ditulis oleh Turney (2022), Koefisien korelasi Pearson (*Pearson Coefficient/PCC*) mengukur hubungan linier antara dua variabel kontinu, dengan nilai antara -1 dan 1. Nilai -1 menunjukkan korelasi negatif sempurna, 0 menunjukkan tidak ada korelasi, dan 1 menunjukkan korelasi positif sempurna. PCC dihitung sebagai kovarians dua variabel dibagi dengan produk deviasi standarnya, umumnya dilambangkan dengan r .

Adapun interpretasi koefisien korelasi Pearson berdasarkan ini adalah sebagai berikut:

- Korelasi Sempurna: Nilai ± 1 menunjukkan korelasi sempurna, yaitu peningkatan (atau penurunan) satu variabel diikuti oleh variabel lainnya.
- Korelasi Kuat: Nilai antara ± 0.5 dan ± 1 menunjukkan korelasi kuat.
- Korelasi Sedang: Nilai antara ± 0.3 dan ± 0.49 menunjukkan korelasi sedang.
- Korelasi Lemah: Nilai di bawah ± 0.29 dianggap sebagai korelasi lemah.
- Tidak Ada Korelasi: Nilai 0 menunjukkan tidak ada korelasi.

Berdasarkan *Newcastle University*, berikut adalah cara menghitung koefisien korelasi Pearson:

- Buat diagram sebar terlebih dahulu untuk mendeteksi outlier. Jika outlier tidak dikeluarkan dari perhitungan, koefisien korelasi bisa jadi keliru. Melihat distribusi data akan memberikan gambaran yang jelas tentang kekuatan korelasi sebelum menghitung koefisien korelasi.
- Selanjutnya perlu diperiksa apakah data memenuhi semua kriteria perhitungan. Variabel-variabel tersebut harus:
 - Diukur pada skala interval/rasio (misalnya tinggi dalam inci dan berat dalam kilogram), ini dapat diperiksa dengan melihat satuan variabel yang diukur.
 - Terdistribusi normal, dapat diperiksa dengan melihat boxplot data. Jika boxplot hampir simetris, kemungkinan data tersebut terdistribusi normal.
 - Korelasi linear, periksa hasil uji signifikansi untuk hipotesis nol dan alternatif.
- Terakhir, dapat menghitung koefisien korelasi menggunakan rumus berikut:

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}}$$

dengan:

- x_i dan y_i adalah titik data,
- \bar{x} adalah rata-rata nilai x ,
- dan \bar{y} adalah rata-rata nilai y

1.5.8 Metrik Evaluasi

Metrik evaluasi berdasarkan Hodson (2022) merupakan sebuah konsep yang penting dalam ML untuk mengukur kinerja dari sebuah model, serta pilihan metrik evaluasi bergantung pada masalah yang diberikan dan model yang digunakan. Karena penelitian ini mencakup permasalahan peramalan atau prediksi, oleh karena itu akan digunakan metrik evaluasi *Root Mean Squared Error* (RMSE).

RMSE merupakan salah satu metrik standar yang digunakan dalam proses evaluasi model. Untuk sampel n observasi y ($y_i, i = 1, 2, \dots, n$) dan n prediksi model yang sesuai, RMSE adalah sebagai berikut.

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

Sesuai dengan namanya, RMSE merupakan akar kuadrat dari *Mean Squared Error* (MSE).

Penyamaan pengambilan akar kuadrat mempertahankan peringkat relatif model sambil menyediakan unit satuan yang sama dengan variabel terikat y . Metrik ini berfungsi sebagai ukuran standar untuk kesalahan yang mengikuti distribusi normal.

Sebagai alasan lain mengenai RMSE dilansir dari Greelance pada tahun 2023 menjelaskan bahwa metrik ini banyak digunakan dalam analisis regresi untuk melakukan evaluasi keakuratan sebuah model. RMSE mengukur jarak antara rata-rata nilai prediksi dan nilai aktual atau nilai sebenarnya, sehingga memberikan ukuran performa model yang komprehensif.



Berdasarkan (P3MPI, 2024), RMSE lebih baik digunakan dalam menilai akurasi prediksi karena memiliki beberapa keunggulan utama seperti pertama, RMSE lebih sensitif terhadap kesalahan besar dibandingkan kesalahan kecil yang merupakan hal penting dalam analisis yaitu ketika besarnya kesalahan sangat diperhatikan. Kedua, RMSE memberikan ukuran akurasi yang jelas dan mudah diinterpretasikan. Ketiga, RMSE dapat dibedakan, sehingga cocok untuk algoritma optimasi dalam pelatihan model.

Adapun cara melakukan perhitungan dari RMSE dapat dilakukan dengan mudah berkat bantuan bahasa pemrograman Python dengan *library* Scikit-learn.

1.6 Penelitian Terkait

Beberapa penelitian sebelumnya telah dilakukan di bidang mencoba untuk memprediksi harga masa depan *cryptocurrency* yang berbeda. Pendekatan berbeda telah digunakan seperti melihat postingan-postingan komunitas, mencoba menganalisis pendapat masyarakat umum tentang *cryptocurrency*, serta mencoba menganalisis sejarah pasar masa lalu dari *cryptocurrency* yang dipilih dalam rangka mencoba membuat prediksi tentang *cryptocurrency*.

Kim dkk. (2016) dalam penelitiannya yang berjudul “*Predicting fluctuations in cryptocurrency transactions based on user comments and replies*”, mencoba memprediksi fluktuasi *cryptocurrency* Bitcoin, Ethereum dan Ripple. Peneliti menganalisis komentar dan balasan pengguna yang relevan pada postingan komunitas dan menetapkan tag yang menunjukkan apakah postingan tersebut positif atau negatif terhadap *cryptocurrency*. Bekerja dari hipotesis bahwa komentar dan posting pengguna mungkin memiliki pengaruh pada harga *cryptocurrency*, peneliti mulai maju dengan pelan untuk data tersebut di komunitas yang relevan. Tag yang diberikan pada postingan-postingan tersebut berkisar dari sangat negatif, negatif, netral, positif, dan sangat positif. Peneliti menggunakan algoritma yang disebut VADER untuk mencapai hasil tersebut. Pada akhirnya, mereka menyimpulkan bahwa jumlah komentar dan postingan yang dibuat di komunitas memengaruhi jumlah transaksi yang akan dilakukan. Selanjutnya, metode mereka terbukti berguna dalam perdagangan *cryptocurrency*. Mereka juga menemukan jeda dari saat pengguna membuat postingan, dan saat pasar mengikuti. Jeda tersebut berkisar antara 6 atau 7 hari.

Dalam penelitian “*Bayesian regression and Bitcoin*” yang ditulis oleh Shah & Zhang (2014) menggunakan regresi *Bayesian* untuk membuat strategi perdagangan Bitcoin. Strategi ini memprediksi harga rata-rata masa depan Bitcoin selama interval 10 detik. Jika rata-rata ini mencapai ambang batas tertentu, mereka membeli, menjual, atau menahan aset mereka. Data dikumpulkan selama beberapa bulan, dengan interval 10 detik antara setiap titik data. Dengan data yang terkumpul, mereka membuat tiga vektor data, satu yang berisi data dengan lama waktu 30 menit, satu dengan 60 menit dan satu dengan lama waktu 120 menit. Ketika prediksi dibuat, ketiga vektor ini digunakan untuk regresi Bayesian. Untuk hasilnya, mereka mensimulasikan strategi mereka selama periode satu setengah bulan. Mereka bisa mendapatkan perkiraan pengembalian 89% selama 50 hari simulasi. Ini menyamai keuntungan 3362 yuan dari investasi 3781 yuan. Strategi ini bahkan berjalan dengan baik ketika pasar sedang dalam kondisi paling fluktuatif.

Penulis Kamijo & Tanigawa (1990) mengambil pendekatan RNN dalam mencoba mengenali pola harga pasar saham dalam penelitian mereka yang berjudul “*Stock Price Pattern Recognition: A Recurrent Neural Network Approach*” . Setelah mengumpulkan dan menganalisis data saham di masa lalu tiga tahun, mereka dapat menemukan pola segitiga, yang, menurut, merupakan indikasi penting ke mana arah saham.

Dalam penelitian lainnya oleh Egeli dkk. (2019) yang berjudul “*Stock market prediction using artificial neural networks*” para peneliti menggunakan jaringan ANN untuk memprediksi masa depan bursa saham menggunakan struktur *multi layered perceptron* ke jaringan yang dibuat, serta struktur yang digeneralisasi. *Supervised learning* digunakan untuk melatih jaringan melalui kumpulan riwayat harga. Sebagai kesimpulan mereka menentukan bahwa kedua struktur memiliki kinerja yang baik daripada *moving average*, dan *feed forward* yang digeneralisasi memiliki kinerja



Adapun rangkuman mengenai penelitian-penelitian sebelumnya dan perbedaannya dengan penelitian yang akan penulis lakukan dapat dilihat pada tabel di bawah.

Tabel 1.1 Perbandingan Penelitian Sebelumnya

No	Judul Penelitian	Nama Penulis dan Tahun Terbit	Metodologi	Perbedaan
1	<i>Predictiong Fluctuations in Cryptocurrency Transactions Based on User Comments and Replies”</i>	Y. Bin Kim dkk. (2016)	Kuantitatif	Penelitian yang akan penulis lakukan tidak berdasar pada postingan yang berkaitan tentang <i>cryptocurrency</i> pola naik turun harga pada website Yahoo Finance.
2	<i>Bayesian Regression and Bitcoin</i>	Shah & Zhang (2014)	Kuantitatif	Penelitian yang akan penulis lakukan tidak berkaitan dengan membuat strategi perdagangan namun hanya mencoba memprediksi harga di masa mendatang dalam rentang waktu tertentu, serta penelitian penulis berfokus pada Ethereum bukan Bitcoin.
3	<i>Stock Price Pattern Recognition: A Recurrant Neural Network Approach</i>	Kamijo & Tanigawa (1990)	Kuantitatif	Penelitian yang akan penulis lakukan akan memiliki GUI yang akan memudahkan user dalam berinteraksi dengan informasi prediksi yang telah diolah.
4	<i>Stock Market Prediction Using Artificial Neural Networks</i>	Egeli dkk. (2019)	Kualitatif	Penelitian yang akan penulis lakukan tidak berfokus untuk melakukan perbandingan melainkan berfokus untuk membuat suatu GUI.



BAB II METODE PENELITIAN

2.1 Lokasi Penelitian

Penelitian ini dilakukan pada awal bulan Oktober 2023 hingga akhir Mei 2024 di Laboratorium Rekayasa Perangkat Lunak, Program Studi Sistem Informasi, Departemen Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Hasanuddin.

Tabel 2.1 Tahapan Penelitian

Tahapan	Rencana Kegiatan Bulan	2023						2024				
		Maret	April	September	Oktober	November	Desember	Januari	Februari	Maret	April	Mei
Tahap Persiapan	Identifikasi masalah											
	Penyusunan proposal											
Tahap Pelaksanaan	Pengambilan data											
	<i>Preprocessing</i> data											
	Pelatihan model											
	Pengujian kinerja model											
	Integrasi model ke dalam aplikasi web											
Tahap Evaluasi	Pengujian performa aplikasi web											
	Perbaikan masalah pada aplikasi											
Tahap Penyusunan Laporan	Penyusunan laporan											

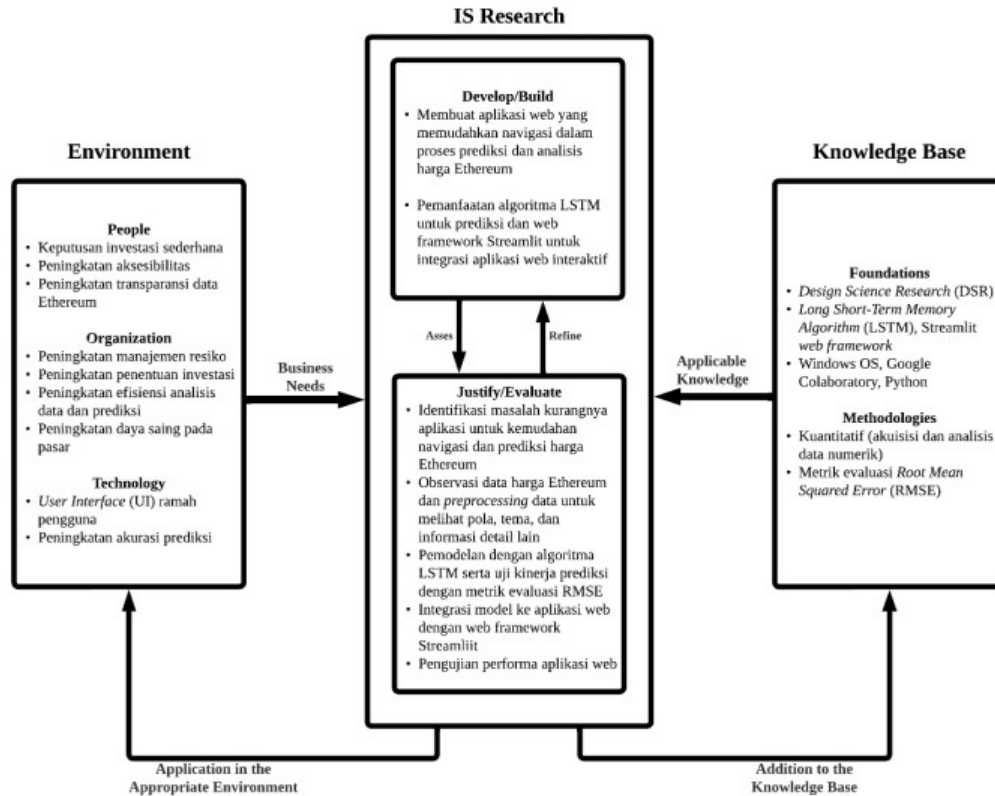
2.2 Teknik Penelitian

Penelitian ini menggunakan metode penelitian kuantitatif, penelitian kuantitatif menurut Bhanadrari (2020) impulkan dan menganalisis data numerik. Metode ini dapat digunakan untuk menemukan membuat prediksi, menguji hubungan sebab akibat, dan menggeneralisasi hasil ke populasi



gan Sistem Penelitian

iliki desain lingkungan sistem penelitian sebagai berikut.



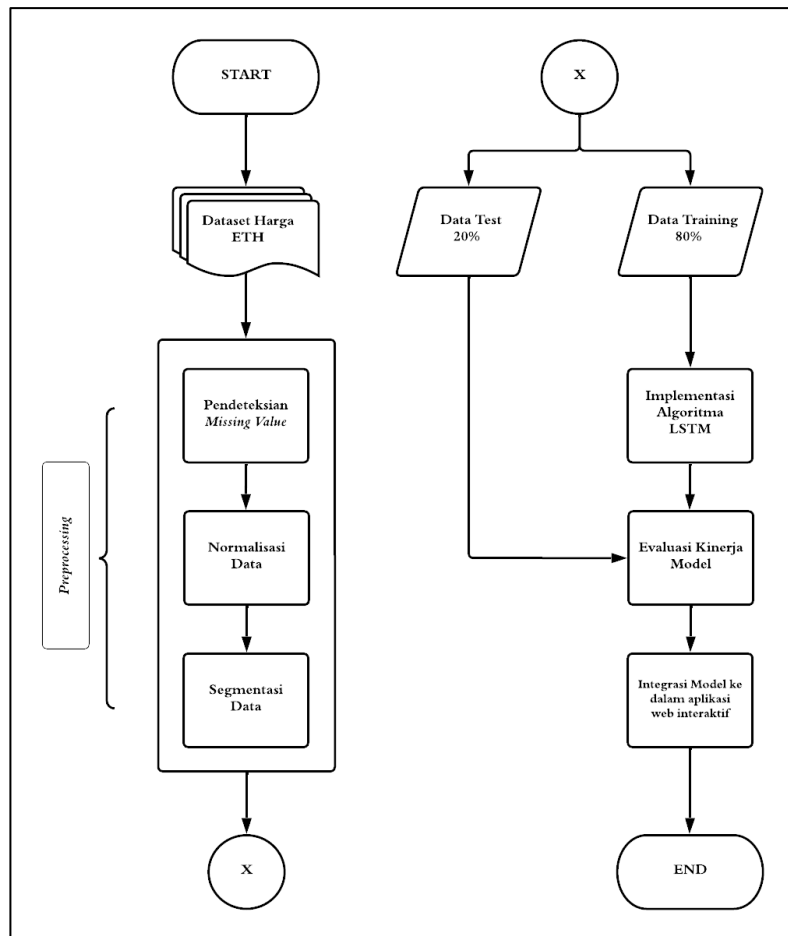
Gambar 2.1 Desain lingkungan sistem penelitian

Diagram di atas menjelaskan bahwa penelitian ini bertujuan untuk membuat suatu sistem atau artefak (*artifact*) yang memanfaatkan algoritma LSTM untuk analisis dan prediksi harga Ethereum serta web *framework* Streamlit sebagai media integrasi model ML ke dalam suatu aplikasi web yang memudahkan navigasi serta interaktif. Diagram menunjukkan bahwa penelitian ini menggunakan metode kuantitatif serta menggunakan metrik evaluasi RMSE dalam pengujian kinerja model ML yang akan dibuat, metode tersebut didukung oleh alat dan bahan pemrograman serta landasan kerangka *Design Science Research* (DSR), yaitu perancangan sebuah artefak teknologi informasi yang dapat memecahkan masalah dunia nyata atau mengembangkan solusi yang telah ada. Hal-hal ini kemudian diterapkan kepada teori yang digunakan dan artefak yang ingin dibuat sehingga diharapkan dapat memenuhi kebutuhan-kebutuhan yang ada pada lingkungan dalam konteks pasar *cryptocurrency*, bisnis, maupun teknologi sistem informasi dan ML.

2.4 Alur Kerja Penelitian

Alur kerja dari penelitian ini merupakan langkah-langkah yang akan dilakukan untuk menyelesaikan penelitian ini, yaitu: Pengumpulan *dataset*, *preprocessing* data, implementasi model atau arsitektur pada *dataset*, pelatihan model atau arsitektur, dan evaluasi kinerja model yang dihasilkan.





Gambar 2.2 Diagram alur kerja penelitian

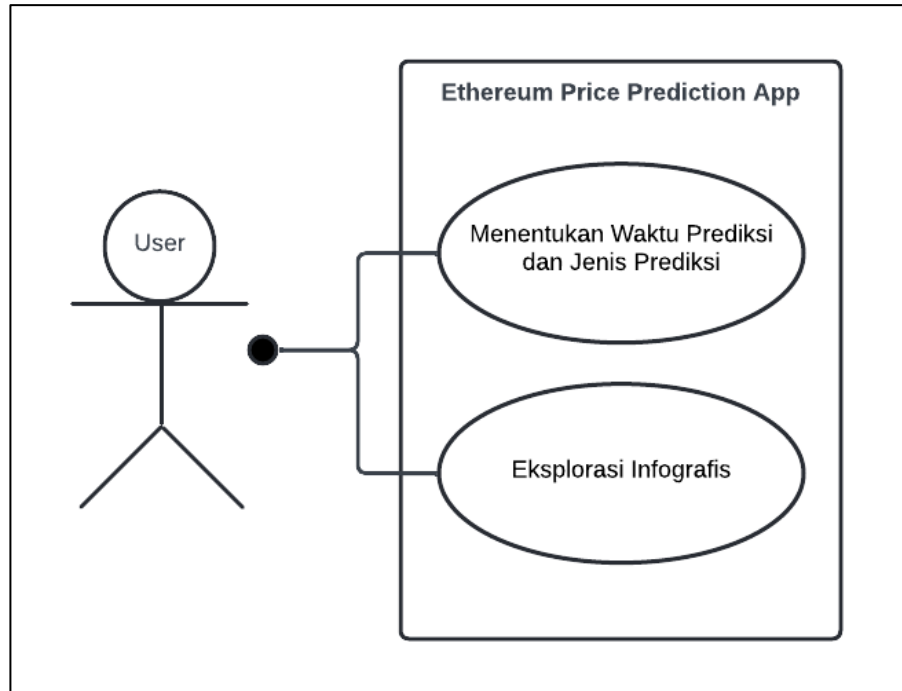
2.5 Pengambilan Data

Pengambilan data dilakukan dalam skrip Python menggunakan bantuan *library yfinance*. Skrip ini menggunakan API Yahoo Finance untuk mengumpulkan data dari situs Yahoo Finance pada periode waktu tertentu. Proses ini akan menghasilkan sebuah *dataframe* yang terdiri dari 6 kolom. Kolom *Date* yang menunjukkan keterangan tanggal, *Open* dan *Close* yang menunjukkan keterangan harga pembukaan dan penutupan saham, *High* dan *Low* yang menunjukkan harga saham tertinggi dan terendah, serta *Volume* yang menunjukkan total volume saham yang diperdagangkan. *Dataframe* berisi entri data untuk setiap hari dimulai untuk entri data tertua yaitu sejak 9 November 2017 hingga entri data terbaru.

2.6 Arsitektur Sistem

Adapun gambaran komponen mengenai sistem aplikasi website yang akan dibuat digambarkan oleh diagram *use case* berikut.

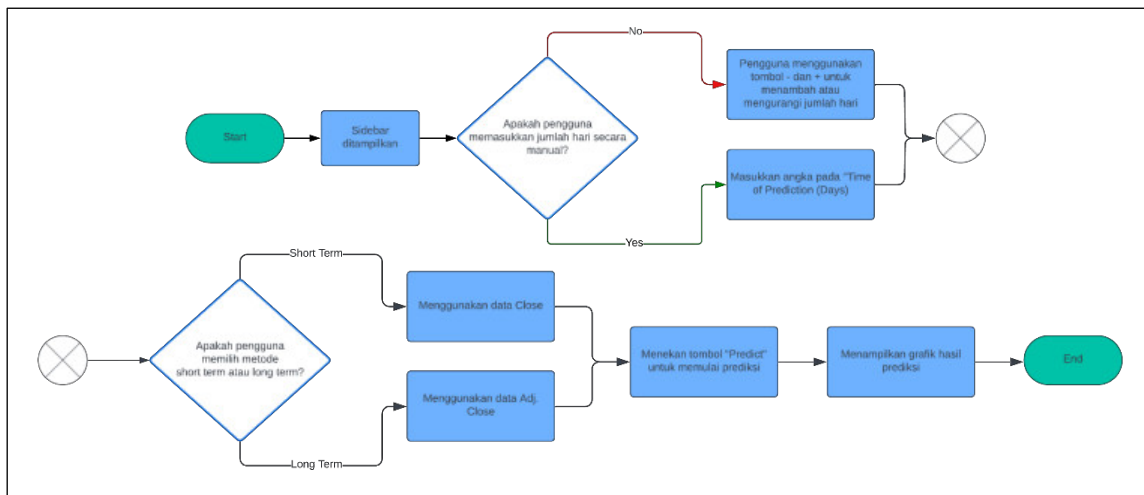




Gambar 2.3 Diagram use case arsitektur sistem

Berikut *activity diagram* untuk setiap *use case* oleh *user*.

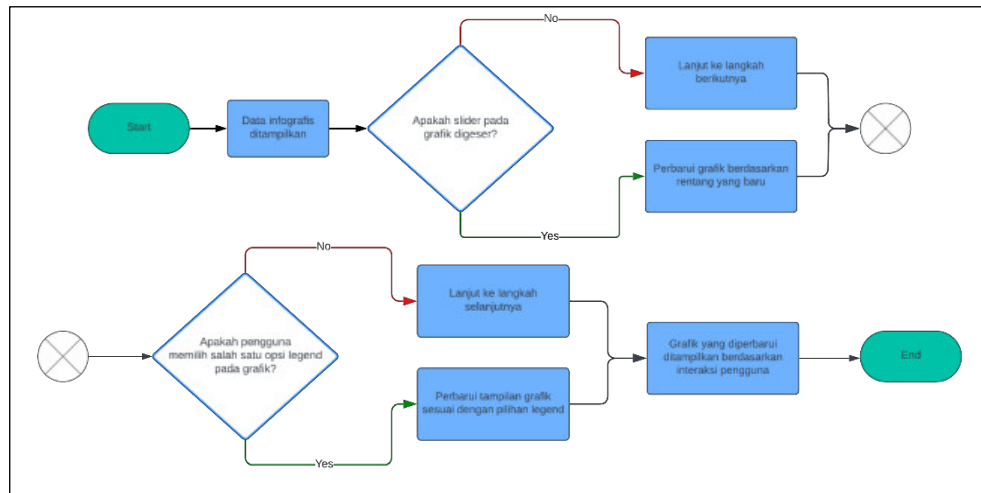
a. Menentukan Waktu Prediksi dan Jenis Prediksi



Gambar 2.4 Activity diagram untuk menentukan waktu prediksi dan jenis prediksi



b. Eksplorasi Infografis



Gambar 2.5 Activity diagram untuk eksplorasi infografis

2.7 Instrumen Penelitian

Adapun instrumen penelitian yang digunakan dalam melakukan penelitian ini dari sisi perangkat keras berupa sebuah laptop dengan spesifikasi adalah sebagai berikut:

- Model: ROG Strix G531GU
- Processor: Intel Core i7-9750
- CPU: 2.60 GHz (12 CPUs)
- Memory: 16384 RAM

Adapun instrumen penelitian yang digunakan dalam melakukan penelitian ini dari sisi perangkat lunak adalah sebagai berikut:

- Sistem operasi: Windows 11 Pro Insider Preview 64-bit (Version 22H2, Build 25352.1)
- Browser: Mozilla Firefox Version 113.0.1
- Menjalankan kode: Google Colaboratory

Adapun instrumen penelitian yang digunakan dalam melakukan penelitian ini dari sisi ML dilakukan dengan menggunakan bahasa pemrograman Python, dengan *library* seperti NumPy, Pandas, Scikit-learn, TensorFlow, Keras, dan Matplotlib. Kemudian untuk membangun aplikasi web dilakukan dengan menggunakan bantuan *framework* Streamlit.

