

BAB III

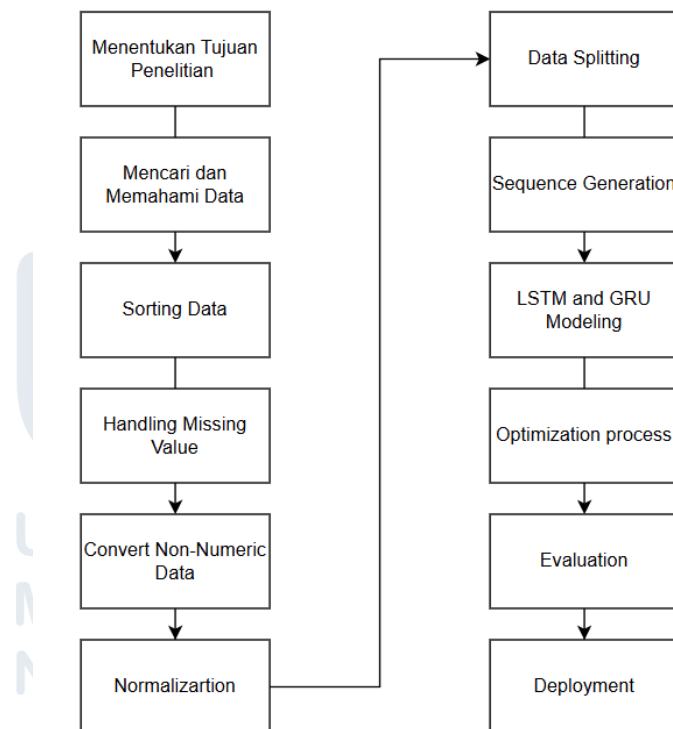
METODOLOGI PENELITIAN

3.1 Gambaran Umum Objek Penelitian

Objek penelitian ini adalah harga histori dari harga bitcoin yang akan dilakukan analisa untuk melakukan prediksi harga bitcoin menggunakan algoritma LSTM, dan GRU. Dataset yang digunakan merupakan harga bitcoin sebelumnya yang mencakup kolom harga pembukaan, harga penutupan, harga tertinggi, harga terendah, tanggal, volume perdagangan, dan persentase.

3.2 Metode Alur Penelitian

Metode alur penelitian ini berdasarkan metode CRISP-DM yang terbagi menjadi beberapa alur. Berikut metode alur penelitian sebagai berikut:



Gambar 3.1 Alur Penelitian

Penelitian ini diawali dengan penentuan tujuan penelitian yang bertujuan untuk mengoptimalkan serta membandingkan kinerja model *Long Short-Term Memory* (LSTM) dan *Gated Recurrent Unit* (GRU) dalam memprediksi pergerakan harga Bitcoin. Pada tahap ini, fokus penelitian ditetapkan secara jelas agar seluruh proses analisis dan pemodelan berjalan sesuai dengan permasalahan yang ingin diselesaikan, yaitu bagaimana memperoleh model prediksi yang paling optimal pada data harga Bitcoin yang memiliki karakteristik volatilitas tinggi.

Setelah tujuan penelitian ditentukan, tahap selanjutnya adalah mencari dan memahami data yang akan digunakan. Data harga Bitcoin dikumpulkan dari sumber yang relevan dan dianalisis untuk mengetahui struktur, atribut, serta karakteristiknya sebagai data deret waktu. Pemahaman terhadap data ini penting untuk memastikan bahwa metode pengolahan dan pemodelan yang diterapkan sesuai dengan sifat data yang digunakan. Selanjutnya, dilakukan proses *sorting data* untuk memastikan seluruh data tersusun secara benar berdasarkan urutan waktu, sehingga hubungan temporal antar data tetap terjaga dan dapat dimanfaatkan secara optimal oleh model prediksi.

Tahap berikutnya adalah penanganan *missing value* yang bertujuan untuk mengatasi nilai yang hilang atau tidak lengkap dalam dataset. Proses ini dilakukan agar data yang digunakan memiliki kualitas yang baik dan tidak menimbulkan bias atau *error* pada saat modeling. Setelah itu, data non-numerik yang terdapat dalam dataset dikonversi ke dalam bentuk numerik agar dapat diproses oleh algoritma deep learning. Tahap ini penting karena model LSTM dan GRU hanya dapat menerima input berupa data numerik. Selanjutnya, dilakukan proses normalisasi data untuk menyamakan skala nilai antar variabel, sehingga perbedaan rentang nilai tidak mempengaruhi proses pembelajaran model dan dapat meningkatkan stabilitas serta kecepatan konvergensi selama pelatihan.

Penelitian dilanjutkan dengan proses *data splitting*. Pada tahap ini, dataset dibagi menjadi data latih dan data uji dengan tujuan untuk melatih model serta menguji

kemampuan generalisasi model terhadap data yang belum pernah dilihat sebelumnya. Data yang telah dibagi kemudian diproses lebih lanjut melalui tahap *sequence generation*, yaitu pembentukan urutan data berdasarkan jendela waktu tertentu agar sesuai dengan kebutuhan input model LSTM dan GRU yang memanfaatkan ketergantungan temporal dalam data deret waktu.

Tahap selanjutnya adalah pemodelan menggunakan algoritma LSTM dan GRU. Pada tahap ini, kedua model dilatih menggunakan data latih untuk mempelajari pola pergerakan harga Bitcoin berdasarkan urutan data yang telah dibentuk sebelumnya. Proses pemodelan dilakukan secara terpisah untuk masing-masing algoritma agar dapat dibandingkan kinerjanya secara objektif. Setelah model terbentuk, dilakukan proses optimisasi untuk memperoleh kombinasi parameter yang menghasilkan performa terbaik. Optimisasi ini bertujuan untuk meningkatkan akurasi prediksi serta mengurangi tingkat kesalahan model dalam memprediksi harga Bitcoin.

Model yang telah melalui proses optimisasi kemudian dievaluasi menggunakan matrik evaluasi yang telah ditentukan. Tahap evaluasi bertujuan untuk mengukur kinerja masing-masing model secara kuantitatif, sehingga dapat diketahui model mana yang memberikan hasil prediksi paling optimal. Hasil evaluasi ini digunakan sebagai dasar dalam membandingkan performa LSTM dan GRU. Tahap akhir dari penelitian ini adalah *deployment*, yaitu penerapan model terbaik sebagai sistem prediksi harga Bitcoin yang dapat digunakan secara praktis dan berpotensi dikembangkan lebih lanjut untuk kebutuhan analisis maupun pengambilan keputusan di masa mendatang.

3.2.2 Sorting

Langkah pertama yang dilakukan adalah dengan mengurutkan data secara ascending berdasarkan tanggal, sehingga struktur data mengikuti urutan kronologis yang benar. Pengurutan ini penting karena model deret waktu sangat bergantung pada alur data yang berurutan

3.2.3 Handling Missing Value

Tahap selanjutnya adalah melakukan pengecekan dan penanganan missing value untuk memastikan tidak ada nilai kosong yang dapat mengganggu proses pelatihan. Baris data yang hilang diperbaiki menggunakan teknik forward fill atau dihapus apabila dianggap tidak relevan.

3.2.4 Data Cleaning / Data Convert

Tahap berikutnya adalah data cleaning, termasuk mengubah tipe data non-numerik menjadi format numerik agar dapat diproses oleh model. Proses ini juga meliputi pengecekan nilai ekstrim atau format yang tidak sesuai.

3.2.5 Feature Engineering

Setelah data bersih, dilakukan feature engineering menggunakan lag feature, yaitu membuat fitur baru berdasarkan nilai harga beberapa hari sebelumnya. Teknik ini digunakan untuk membentuk pola historis yang menjadi dasar bagi model mempelajari hubungan antar waktu.

3.2.6 Normalization

Data yang sudah diperkaya kemudian dinormalisasi menggunakan MinMaxScaler agar seluruh nilai berada pada rentang 0–1. Normalisasi diperlukan untuk mempercepat proses pembelajaran dan menjaga stabilitas gradien pada model.

3.2.7 Data Spitting

Setelah data dinormalisasi, dilakukan pembagian data menjadi dua bagian, yaitu 80% untuk pelatihan dan 20% untuk pengujian, dengan tetap mempertahankan urutan waktu. Terdapat penelitian yang melakukan evaluasi berbagai rasio pembagian data dari 60:40 hingga 90:10 terhadap deep learning. Hasil eksperimen tersebut menghasilkan bahwa ketika menggunakan pelatihan diatas 70% cenderung memiliki kinerja lebih baik [42].

3.2.8 Sequence Generation

Tahap terakhir adalah sequence generation, di mana data diubah menjadi bentuk deret berlapis (misalnya 60 hari sebelumnya sebagai input untuk memprediksi hari

berikutnya). Proses ini memastikan bahwa model menerima pola urutan waktu yang sesuai dengan karakteristik data time series.

3.2.9 LSTM and GRU Modeling

Tahap selanjutnya adalah melakukan modeling terhadap LSTM dan GRU Model. LSTM dan GRU dalam penelitian ini sama-sama menggunakan arsitektur dua lapisan berulang untuk mempelajari pola temporal pada data harga Bitcoin. Model LSTM terdiri dari dua lapisan LSTM yang diselingi oleh dropout untuk mengurangi *overfitting*, kemudian diakhiri dengan satu lapisan dense sebagai keluaran prediksi. Model ini dilatih menggunakan *optimizer Adam* dan fungsi loss mean squared error dengan parameter yang diperoleh melalui proses optimasi. Sementara itu, model GRU memiliki struktur serupa namun menggunakan unit GRU yang lebih sederhana secara komputasi, dengan dua lapisan GRU, dropout sebagai regularisasi, serta lapisan dense sebagai keluaran. Penggunaan optimizer Adam dan fungsi loss mean squared error juga diterapkan pada model GRU. Kedua arsitektur dirancang untuk menangkap dinamika harga Bitcoin yang bersifat fluktuatif dengan tetap menjaga stabilitas dan efisiensi pembelajaran.

3.2.10 Optimization Process

Proses optimasi hyperparameter pada penelitian ini dilakukan menggunakan Bayesian Optimization agar pemilihan konfigurasi model tidak bergantung pada perkiraan manual peneliti. Pendekatan ini digunakan untuk mencari nilai hyperparameter terbaik pada model LSTM dan GRU, terutama parameter jumlah unit, tingkat dropout, dan *learning rate*. Tahapan optimasi dimulai dari penentuan ruang pencarian (search space), yaitu rentang nilai yang diperbolehkan untuk masing-masing hyperparameter. Rentang ini kemudian dijadikan batas bagi Bayesian Optimization untuk mengeksplorasi kombinasi nilai yang mungkin menghasilkan performa terbaik.

Setelah ruang pencarian ditetapkan, fungsi objektif (*objective function*) disusun untuk masing-masing model. Fungsi ini bertanggung jawab membangun arsitektur LSTM atau GRU sesuai konfigurasi hyperparameter yang diberikan oleh algoritma optimasi. Model yang terbentuk kemudian dilatih secara singkat

menggunakan sebagian data pelatihan yang telah dipisahkan menjadi set pelatihan dan validasi. Pada tahap ini, proses pelatihan tidak dilakukan secara penuh, tetapi hanya cukup lama untuk memungkinkan model menunjukkan kecenderungan performanya. Nilai *validation loss* pada akhir pelatihan menjadi metrik utama yang dievaluasi oleh fungsi objektif. Karena Bayesian Optimization berusaha memaksimalkan nilai fungsi objektif, maka nilai *validation loss* diubah menjadi negatif sebelum dikembalikan ke optimizernya.

Setelah fungsi objektif siap, Bayesian Optimization diinisialisasi dengan ruang parameter yang telah ditetapkan. Pada setiap iterasi, algoritma memilih kombinasi hyperparameter tertentu berdasarkan *probability model* internalnya. Pendekatan ini membuat proses pencarian lebih efisien dibanding metode brute-force seperti Grid Search, karena Bayesian Optimization memperkirakan area yang paling menjanjikan dalam ruang pencarian dan secara adaptif memfokuskan eksplorasi pada wilayah tersebut. Dengan cara ini, algoritma tidak mencoba semua kemungkinan, tetapi mengarahkan pencarian pada konfigurasi yang memiliki peluang besar memberikan *validation loss* lebih rendah.

Proses optimasi berlangsung selama sejumlah iterasi hingga Bayesian Optimization menemukan kombinasi parameter yang dianggap optimal. Nilai terbaik yang diperoleh kemudian digunakan sebagai konfigurasi final untuk membangun model LSTM dan GRU pada tahap pelatihan penuh. Dengan pendekatan ini, pemilihan hyperparameter menjadi lebih objektif dan tidak bias, serta memberikan dasar yang lebih kuat untuk membandingkan kedua model secara adil.

3.2.11 Evaluation

Tahap selanjutnya merupakan tahap evaluasi untuk mengukur kinerja model LSTM dan GRU yang telah dibuat. Evaluasi pada penelitian ini menggunakan RMSE, MAE, R2 untuk mengukur berapa banyak error yang dihasilkan dari kedua model. Confusion matrix untuk mengukur berapa akurasi model yang dihasilkan dalam memprediksi nilai aktual dari naik dan turunnya harga bitcoin.

3.2.12 Deployment

Deployment dilakukan untuk menampilkan prediksi harga bitcoin yang dapat digunakan secara nyata. Data yang digunakan adalah bentuk realtime, sehingga setiap harinya dapat berubah-ubah sesuai dengan harga nyata. Deployment ini dibuat agar trader atau investor dapat melihat dengan nyata mengenai pandangan prediksi yang dihasilkan dari kedua model tersebut.

3.3 Teknik Pengumpulan Data

Data diambil dari website resmi dari website (investing.com) yang merupakan website yang menyediakan data market dan merupakan website mengenai finansial dan ekonomi. Atribut data yang tertera pada data berisikan kolom date, price, open, price(close), high, low, vol., change %. Data yang dipakai adalah data historis mulai dari harga bitcoin dari tanggal 2 februari 2012 hingga tanggal 20 oktober 2025. Harga bitcoin yang diambil merupakan BTC/USD, dimana harga bitcoin dengan satuan USD. Data tersebut merupakan data harga harian atau *daily*. Dengan data yang banyak dan menyeluruh prediksi diharapkan menjadi lebih akurat dan dapat dibaca oleh algoritma dengan kinerja lebih baik.

3.4 Variable Penelitian

3.4.2 Variable independent

Variabel independen adalah variabel yang berdiri sendiri yang digunakan untuk memprediksi variabel dependen yaitu close. Variabel independen yang terdiri dari:

1. date: merupakan tanggal data harga bitcoin perhari.
2. open: merupakan harga pembukaan disaat hari tersebut.
3. high: merupakan harga tertinggi pada hari tersebut.
4. low: merupakan harga terendah pada hari tersebut.
5. vol.: merupakan berapa volume perdagangan pada hari tersebut.

6. change %: merupakan berapa persentase perubahan harga yang terjadi di hari tersebut.

3.4.3 Variable Dependent

Variabel dependen adalah variabel yang terikat atau variabel target yang dimana yang akan dilakukan prediksi yaitu fitur price/close (harga penutupan). Variabel ini yang nantinya akan diprediksi.

