

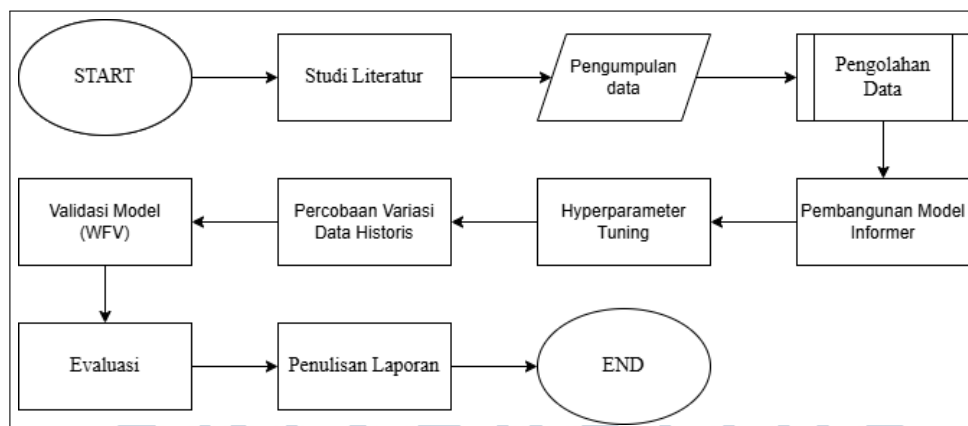
BAB 3

METODOLOGI PENELITIAN

Pada bab ini menjelaskan secara sistematis alur penelitian yang dilakukan untuk mencapai tujuan penelitian. Tahapan yang dibahas mencakup pengumpulan dan pengolahan data, pembangunan model informer, hypeparameter tuning, percobaan variasi data historis, validasi model menggunakan walk-forward validation, dan evaluasi model menggunakan metrik RMSE dan MAPE.

3.1 Alur Penelitian

Penelitian diawali dengan studi pustaka terkait penerapan informer dan prediksi emas, dilanjutkan pengumpulan data, kemudian pengolahan data yang sudah diperoleh, lalu membangun model informer, kemudian dilakukan hyperparameter tuning untuk meningkatkan performa, dilanjutkan percobaan untuk variasi data historis seperti 10-30 tahun terakhir, kemudian proses validasi dan evaluasi menggunakan Walk-FORward Validation dan metrik evaluasi RMSE dan MAPE, dan terakhir penulisan laporan. Gambaran umum dari alur penelitian seperti pada gambar 3.1.



Gambar 3.1. Gambar alur penelitian

3.2 Studi Literatur

Studi literatur dilakukan untuk menelaah berbagai penelitian terdahulu yang berkaitan dengan prediksi harga emas atau metode prediksi time-series berbasis

deep-learning. Hasil kajian ini digunakan sebagai dasar dalam menentukan pendekatan metodologis, pemilihan arsitektur model Informer, serta strategi evaluasi model yang sesuai.

3.3 Pengumpulan Data

Penelitian ini menggunakan data yang bersumber dari investing.com. Data emas yang dipilih dengan kode XAU/USD yang merupakan representasi harga 1 troy ounce emas dalam satuan dolar AS (1 troy ounce = 31,1035 gram). Data dikumpulkan dengan cara mengunduh langsung pada website investing.com, data yang didapatkan memiliki format csv. Kemudian melalui aplikasi Microsoft Excel, data dipisahkan antar kolom agar mudah dilihat. Data yang telah didapatkan berisi pergerakan harga emas harian selama 30 tahun terakhir, dimulai dari tanggal 1 Januari 1995 hingga 31 Desember 2025 dengan total 8068 data. Pergerakan harga emas yang direkam terjadi dalam satu hari dan lima hari seminggu (senin-jumat). Pengecualian pada akhir pekan didasarkan karena perdagangan pasar emas global yang tutup pada jumat malam dan dibuka kembali pada minggu malam. Penutupan pada akhir pekan digunakan proses penyelesaian dan aturan jam oprasional standar dari pelaku institusional utama seperti bank-bank besar dan bursa. Berikut adalah tabel sample data harga emas yang telah diperoleh.

Tabel 3.1. Tabel Sample Data Harga Emas

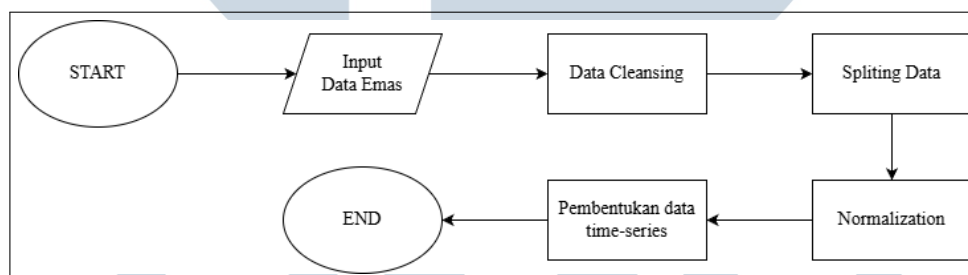
Date	Price	Open	High	Low	Vol.	Change%
01/02/1995	383.15	383.15	383.15	383.15		0.08%
01/03/1995	379.70	383.25	379.70	379.70		-0.90%
01/04/1995	374.55	379.75	374.55	374.55		-1.36%
...
12/25/2025	4495.35	4487.10	4500.35	4483.59		0.35%
12/24/2025	4479.64	4488.81	4525.71	4448.27		-0.20%
12/29/2025	4332.08	4540.70	4550.10	4303.75		-4.42%

Dataset ini terdiri dari 7 kolom seperti Date, Price, Open, High, Low, Vol., dan Change. Kolom "Date" menunjukkan tanggal data harga emas terjadi. Kolom "Price" menunjukkan harga penutup/terakhir sebelum pasar ditutup di hari tersebut. Kolom "Open" menunjukkan harga emas awal saat pasar dibuka di hari tersebut. Kolom "High" menunjukkan harga emas tertinggi yang terjadi selama satu hari

tersebut. Kolom "Low" menunjukkan harga emas terendah yang terjadi selama satu hari tersebut. Kolom "Vol." menunjukkan volume transaksi yang terjadi di hari tersebut. Kolom "Change" menunjukkan persentase perubahan harga emas dibandingkan hari sebelumnya. Kolom yang digunakan didalam penelitian ini hanya kolom "Date" yang dijadikan sebagai index data dan "Price" yang dianggap paling mewakili harga emas dalam satu hari tersebut.

3.4 Pengolahan Data

Pengolahan data bertujuan untuk mempersiapkan data yang telah diperoleh menjadi sesuai dengan kriteria data yang diinginkan sebelum tahap pemodelan. Pada tahap ini akan memastikan penggunaan format data yang digunakan sesuai, seperti format Datetime untuk kolom Date, dan format Float untuk kolom Price. Gambar 3.2 merupakan flowchart dari tahap pengolahan data, yang menampilkan alur dari proses pengolahan data.



Gambar 3.2. Flowchart tahap pengolahan data

Proses pertama dalam penyiapan data ialah data cleansing yang dilakukan untuk menghapus data kosong, Null, dan duplikat. Pengecekan data terlebih dahulu dilakukan untuk memastikan ketersediaan data yang perlu dibersihkan. Pengecekan menghasilkan tidak ada data Null, NaN, ataupun duplikat, yang mana dapat dikatakan data sudah bersih dan siap digunakan pada tahap selanjutnya. Jika setelah pengecekan terdapat data Null atau NaN, maka dilakukan modifikasi data menjadi 0 agar model bisa membedakan data tersebut tidak digunakan, dan jika ditemukan data duplikat maka tergantung dari kolom yang duplikat, jika kolom Date dan Price memiliki nilai duplikat maka salah satu data akan dihapus, jika kolom hanya kolom Date yang duplikat akan diambil nilai rata-rata. Setelah pengecekan data, dilanjutkan pembagian data untuk proses pelatihan dan pengujian dengan rasio 80:20. Kemudian, setiap data latih dan data ujin dinormalisasi menggunakan MinMaxScaler sehingga skala data yang terbentuk

diantara 0 dan 1. Kemudian, dilanjutkan pembentukan data time-series dari data hasil normalisasi dengan rentang 65 hari data masa lalu dan 5 hari data masa depan yang akan diprediksi, dimana dari 65 hari data masa lalu, 60 hari digunakan sebagai context yang menggambarkan pergerakan data jangka menengah dan 5 hari untuk menangkap pola jangka pendek, sehingga model tidak hanya memahami gambaran pergerakan harga secara luas tetapi juga dapat memperhatikan pergerakan data terbaru.

3.5 Implementasi Model Informer

Pada penelitian ini proses implementasi model menggunakan library transformers dari HuggingFace. Komponen `InformerConfig` dan `InformerForPrediction` merupakan kelas (class) yang disediakan oleh library tersebut. `InformerConfig` adalah konfigurasi yang berisi hyperparameter yang dibutuhkan dalam membangun model Informer. `InformerForPrediction` digunakan untuk mendeklarasi model informer yang didalamnya menggunakan konfigurasi dari `InformerConfig`. `InformerConfig` memiliki variabel input berupa hyperparameter yang dibutuhkan untuk membangun model. Kedua class tersebut digunakan untuk inisialisasi informer. Daftar hyperparameter yang dibutuhkan oleh `InformerConfig` seperti pada tabel 3.2.

Tabel 3.2. Tabel penjelasan hyperparameter model Informer

No	Nama	Definisi
1	<code>prediction_length</code>	Jumlah data yang ingin diprediksi
2	<code>context_length</code>	Jumlah data masa lalu yang digunakan untuk prediksi
3	<code>input_size</code>	Jumlah fitur pada input
4	<code>d_model</code>	Dimensi ruang fitur internal model
5	<code>encoder_layers</code>	Jumlah lapisan encoder
6	<code>decoder_layers</code>	Jumlah lapisan decoder
7	<code>num_time_features</code>	Jumlah fitur tambahan yang mewakili informasi waktu
8	<code>dropout</code>	Presentase dropout untuk regularisasi
9	<code>attention_type</code>	Jenis mekanisme attention yang digunakan
10	<code>distil</code>	Flag boolean apakah menggunakan mekanisme distilling

11	lags_sequence	Daftar offset waktu yang dipakai sebagai fitur tambahan dari data masa lalu
----	---------------	---

Setelah model informer diinisialisasikan menggunakan hyperparameter yang ditentukan, proses training model dilakukan dengan memasukan beberapa input variabel dari data pelatihan kedalam model. Pemanggil objek model tersebut secara implisit menjalankan proses forward pass untuk menghasilkan prediksi, yang selanjutnya digunakan dalam perhitungan fungsi kerugian (loss) dan pembaruan bobot model melalui algoritma optimasi. Beberapa input variabel yang dibutuhkan dalam pemanggilan objek model untuk proses pelatihan seperti pada tabel 3.3.

Tabel 3.3. Tabel input training model Informer

No	Nama	Definisi
1	past_values	Data masa lalu
2	past_time_features	Fitur tambahan informasi waktu dari data masa lalu
3	past_observed_mask	Penanda untuk data masa lalu yang dapat digunakan (nilai antara 0 atau 1)
4	future_values	Data masa depan
5	future_time_features	Fitur tambahan informasi waktu dari data masa depan

Proses pengujian model dilakukan dengan memanfaatkan fungsi (.generate) pada objek model informer untuk menghasilkan prediksi nilai masa depan. Pada proses ini model tidak memperbaharui bobot dan hanya melakukan prediksi masa depan berdasarkan data historis. Oleh karena itu, pada proses ini tidak dibutuhkan nilai masa depan sebagai input parameter pada fungsi. Input variabel yang dibutuhkan dalam pemanggilan fungsi seperti pada tabel 3.4.

Tabel 3.4. Tabel input prediksi model Informer

No	Nama	Definisi
1	past_values	Data masa lalu
2	past_time_features	Fitur tambahan informasi waktu dari data masa lalu

3	past_observed_mask	Penanda untuk data masa lalu yang dapat digunakan (nilai antara 0 atau 1)
5	future_time_features	Fitur tambahan informasi waktu dari data masa depan

3.6 Hyperparameter Tuning

Hyperparameter adalah parameter yang diatur sebelum proses pelatihan model dan tidak mengalami perubahan selama proses pelatihan, melainkan mempengaruhi proses pembelajaran secara keseluruhan. Hyperparameter tuning bertujuan untuk menentukan kombinasi parameter yang terbaik agar menghasilkan kinerja model yang optimal. Hyperparameter tuning dilakukan dengan beberapa kombinasi skenario dari hyperparameter, seperti d_model, encoder layer, decoder layer, dan epoch. Setiap percobaan skenario hanya dilakukan pada satu hyperparameter dan sisa parameter lainnya akan mengikuti nilai default pada tabel 3.5. Setiap percobaan skenario akan diikuti dengan sepuluh kali percobaan dan dihitung nilai kesalahan menggunakan metrik RMSE dan MAPE.

Tabel 3.5. Tabel nilai default hyperparameter

Parameter	Nilai
d_model	128
Encoder Layer	2
Decoder Layer	1
Epoch	60

3.7 Percobaan Variasi Data Historis

Pada tahap ini dilakukan percobaan dengan menggunakan variasi panjang data historis untuk menganalisis pengaruh panjang data terhadap kinerja model prediksi. Variasi panjang data historis yang digunakan, yaitu 10, 15, 20, 25, dan 30 tahun terakhir. Setiap variasi data digunakan secara terpisah sebagai input dalam proses pelatihan dan pengujian model. Proses pelatihan setiap variasi data menggunakan model informer dengan hyperparameter yang sama dan dievaluasi menggunakan metrik RMSE dan MAPE. Variasi data yang dipilih untuk menggambarkan perbedaan kemampuan model dalam menangkap pola temporal

data serta menilai stabilitas kinerja model terhadap perubahan jumlah data.

3.8 Validasi Model

Pada penelitian ini dilakukan validasi model menggunakan skema walk-forward validation (WFV). WFV merupakan metode validasi yang dirancang khusus untuk time-series dengan mempertahankan urutan temporal data. Proses diawali dengan membagi 5000 data sebagai window awal dan pada setiap window akan dibagi menjadi data latih dan data uji dengan rasio 80:20. Kemudian proses pelatihan dan pengujian dilakukan dan dihitung nilai RMSE dan MAPE untuk mengukur kinerja model prediksi. Kemudian window digeser sebanyak 100 data kedepan tanpa memperbesar ukuran window. Penerapan skema WFV membentuk sejumlah iterasi yang menguji kemampuan adaptasi dan generalisasi model terhadap perubahan data, sehingga memungkinkan evaluasi model dilakukan secara realistis.

3.9 Evaluasi

Tahap Evaluasi untuk mengukur performa kinerja model yang telah dibangun dan dilatih pada tahap sebelumnya serta membandingkan kinerja model pada berbagai skenario pengujian. Evaluasi model pada penelitian ini menggunakan metrik evaluasi RMSE dan MAPE, RMSE mengukur seberapa jauh tingkat kesalahan nilai prediksi terhadap nilai aktual dalam skala yang sama, sementara MAPE menghitung rata-rata tingkat kesalahan terhadap nilai aktual dalam bentuk persentase yang informatif dan mudah dipahami. Evaluasi dilakukan pada setiap iterasi dalam skema *walk-forward validation*, dimana hasil prediksi dibandingkan dengan data aktual pada periode pengujian. Nilai RMSE dan MAPE yang rendah menandakan model memiliki kinerja prediksi yang lebih baik.

UNIVERSITAS
MULTIMEDIA
NUSANTARA