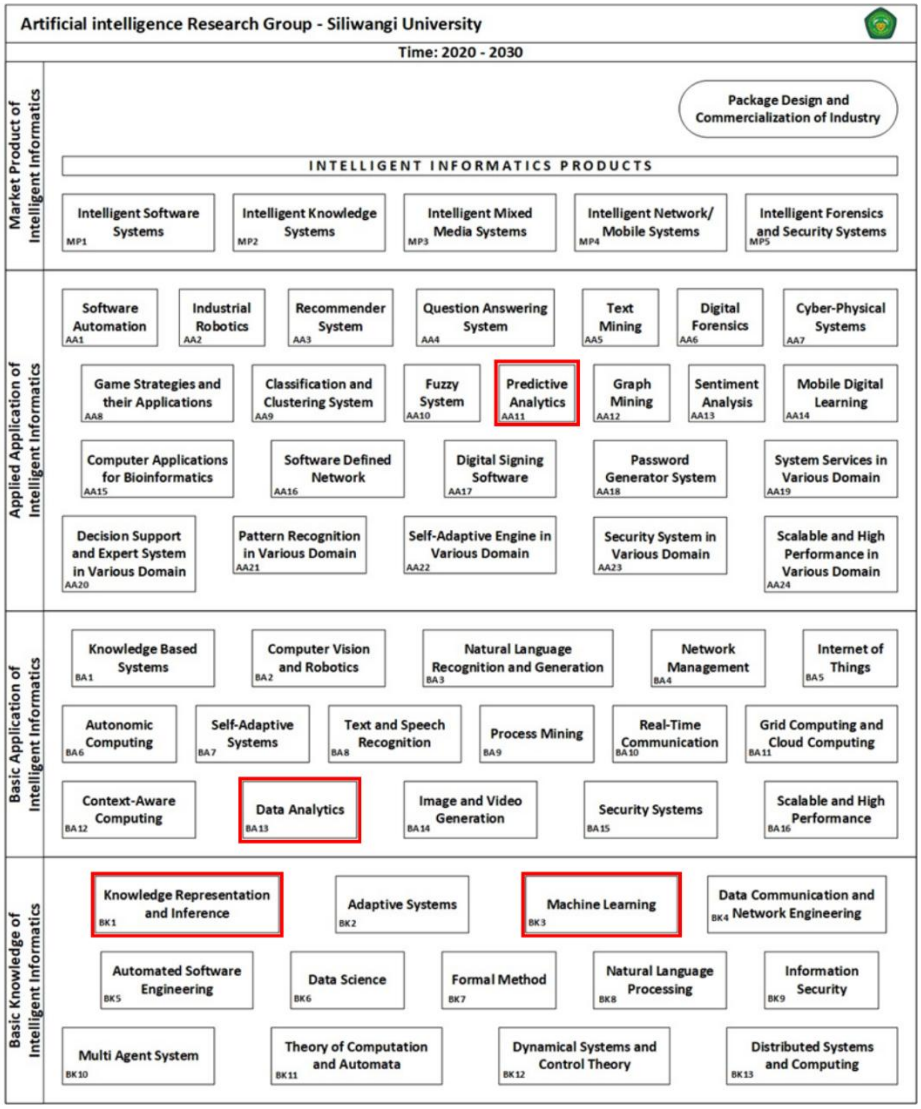
BAB III

METODOLOGI PENELITIAN

3.1 Peta Jalan (*Road Map*) Penelitian

Penelitian ini mengacu pada *Roadmap Artificial Intelligence Research Group* Universitas Siliwangi tahun 2020-2030 yang dapat dilihat pada Gambar 3. 1



Gambar 3. 1 Peta Jalan Penelitian (AIS Universitas Siliwangi, 2020)

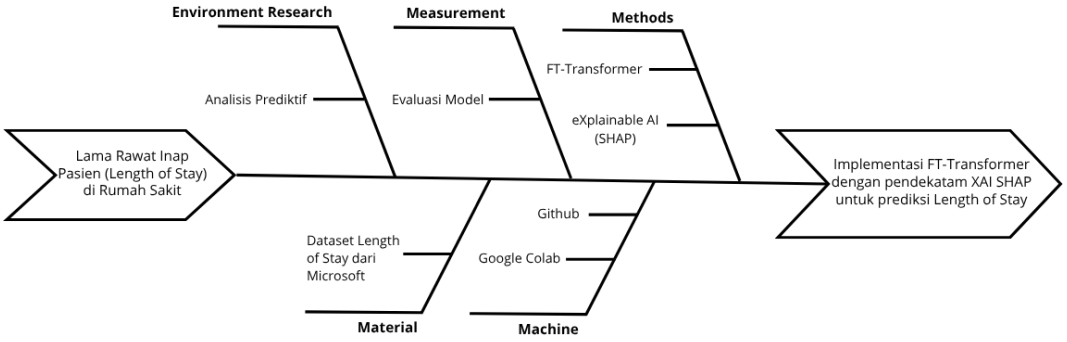
Gambar 3. 1 menyajikan *roadmap* penelitian, dengan ruang lingkup penelitian yang diusulkan ditandai menggunakan kotak berwarna merah. Berdasarkan *roadmap* tersebut, penelitian ini mencakup bidang *predictive analytics*, *machine learning*, *data analytics*, serta *knowledge representation and inference* yang saling terintegrasi untuk membangun sistem prediksi yang tidak hanya memiliki akurasi tinggi, tetapi juga mampu memberikan hasil yang dapat diinterpretasikan.

Penelitian ini termasuk dalam bidang *predictive analysis* karena berfokus dalam pembuatan model untuk memprediksi lama rawat inap atau LoS berdasarkan data historis. Pendekatan ini memanfaatkan pola klinis masa lalu pasien untuk menghasilkan estimasi LoS di masa depan. Dari sisi metodologi, penelitian ini menerapkan pendekatan *machine learning* menggunakan model *deep learning* FT-*Transformer*, yang dirancang untuk memodelkan hubungan kompleks dan nonlinier pada data tabular. Kemampuan tersebut diharapkan dapat meningkatkan performa prediksi model.

Penelitian ini dirancang untuk mencapai tujuan yang telah ditetapkan pada bidang yang dipilih dalam *roadmap* penelitian. Setelah proses akuisisi data, dataset melalui tahapan prapemrosesan yang meliputi transformasi dan eksplorasi data untuk memastikan kualitas data sebelum digunakan dalam proses pemodelan. Tahapan tersebut menjadi bagian dari proses *data analytics* dalam menghasilkan data yang siap dianalisis.

Penelitian ini mengintegrasikan aspek *Knowledge Representation and Inference* melalui penerapan e*Xplainable Artificial Intelligence* (XAI), yaitu SHAP (*SHapley Additive exPlanations*), untuk menginterpretasikan hasil prediksi model. Melalui SHAP, kontribusi masing-masing fitur terhadap hasil prediksi dapat diidentifikasi secara kuantitatif, sehingga model yang bersifat *black box* menjadi lebih transparan. Interpretasi ini memungkinkan ekstraksi pengetahuan berupa faktor-faktor yang berkontribusi terhadap prediksi lama rawat inap pasien. Informasi tersebut tidak hanya meningkatkan pemahaman terhadap mekanisme pengambilan keputusan model, tetapi juga memberikan masukan yang dapat dimanfaatkan sebagai pendukung pengambilan keputusan di bidang kesehatan..

Tujuan penelitian tersebut dapat direpresentasikan menggunakan *Fishbone* Diagram. *Fishbone* diagram atau Ishikawa diagram adalah salah satu alat analisis yang digunakan dengan pendekatan sebab-akibat untuk mengidentifikasi dan mengelompokkan penyebab utama suatu masalah (Kumah dkk., 2024). Secara umum, diagram ini disusun menggunakan kerangka 5M+1E yang mencakup aspek *Man*, *Machine*, *Method*, *Material*, *Measurement*, dan *Environment*. Pendekatan ini sering digunakan untuk memastikan seluruh faktor potensial dapat dianalisis secara komprehensif. Namun, pada penelitian ini digunakan pendekatan 4M+1E dengan mengecualikan aspek *Man*. Hal ini disebabkan oleh penelitian yang berfokus pada pengolahan data sekunder dan pengembangan model berbasis komputasi. Adapun faktor-faktor yang digunakan meliputi *Machine* (mesin/peralatan), *Method* (metode atau prosedur kerja), *Material* (bahan baku), *Measurement* (pengukuran atau sistem kontrol), dan *Environment* (lingkungan kerja) (Ciecińska, 2023).



Gambar 3. 2 *Fishbone* Diagram

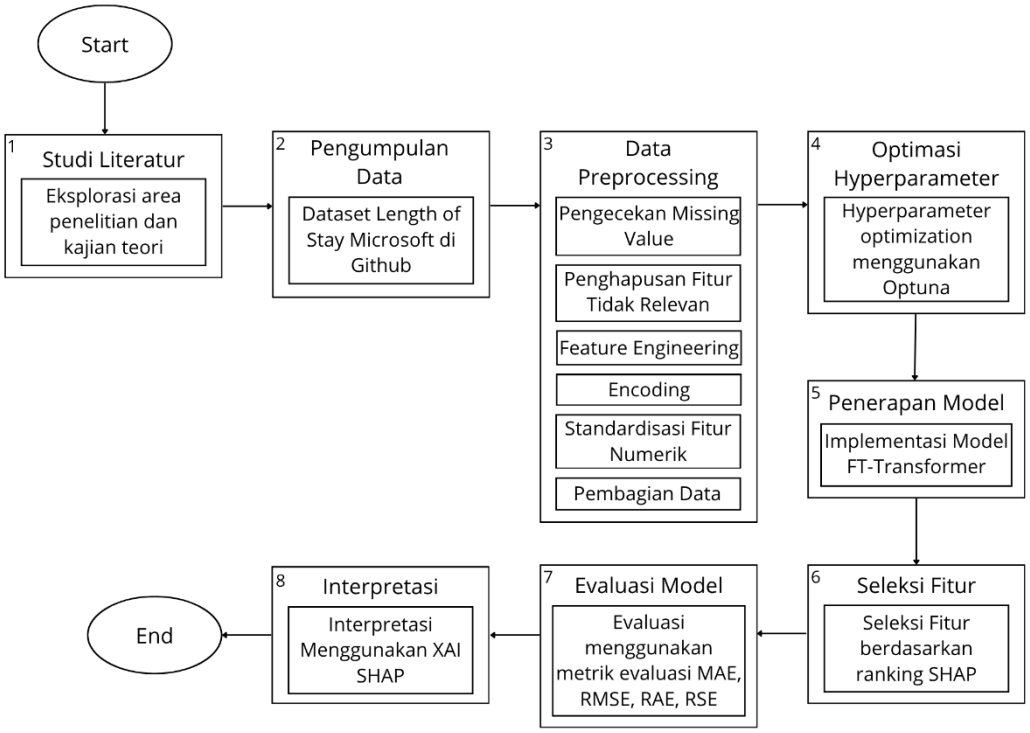
Gambar 3. 2 merupakan *fishbone* diagram pada penelitian ini. *Fishbone* diagram tersebut digunakan untuk mengidentifikasi faktor-faktor utama yang berkaitan dengan proses pengembangan dan implementasi model prediksi lama rawat inap pasien di rumah sakit. Diagram ini membantu memetakan aspek-aspek penting dalam penelitian berdasarkan pendekatan 4M+1E, sehingga keterkaitan antara data, metode, alat, evaluasi, dan konteks penelitian dapat dipahami secara sistematis. Bagian ekor dalam *fishbone* mewakili objek penelitian, yaitu lama rawat inap pasien di rumah sakit, sedangkan bagian kepala pada *fishbone* diagram merupakan tujuan penelitian, yaitu penerapan model *deep learning* FT-*Transformer* dengan pendekatan XAI SHAP untuk menghasilkan prediksi lama rawat inap pasien yang dapat diinterpretasikan. Pada *fishbone* diagram terdapat bagian tulang dan dapat diuraikan sebagai berikut:

a. *Environment Research*, merupakan konteks penelitian yang berfokus pada pentingnya prediksi lama rawat inap pasien sebagai bagian dari analisis prediktif di bidang kesehatan, yang bertujuan untuk meningkatkan manajemen rumah sakit.

b. *Material* merepresentasikan data yang menjadi sumber utama dalam penelitian ini, yaitu dataset *Length of Stay* yang diperoleh dari Microsoft, yang terdiri fitur numerik dan kategorikal terkait kondisi pasien.
c. *Measurement*, merupakan indikator evaluasi yang digunakan untuk menilai performa model prediksi, yaitu MAE, RMSE, RAE, dan RSE.
d. *Machine* mengacu pada alat yang digunakan dalam pelaksanaan penelitian, seperti GitHub yang digunakan untuk akuisisi data, dan Google Colab untuk lingkungan komputasi.
e. *Methods*, merupakan pendekatan yang digunakan pada penelitian ini, yaitu model *deep learning* berbasis *transformer* (FT-*Transformer*) untuk data tabular dan pemanfaatan model XAI menggunakan SHAP untuk interpretasi model prediksi.

3.2 Tahapan Penelitian

Tahapan penelitian ini diawali dengan tahap studi literatur, dilanjutkan dengan pengumpulan data, kemudian data *preprocessing*, optimasi *hyperparameter*, penerapan model, seleksi fitur, evaluasi model, dan interpretasi menggunakan XAI SHAP. Alur penelitian ini diadaptasi dari tahapan yang digunakan oleh Peng & Gao (2025) dengan modifikasi pada tahap seleksi fitur yang mengadopsi pendekatan seleksi fitur berbasis nilai SHAP dari Fa dkk., (2025) yang dapat dilihat pada Gambar 3. 3



Gambar 3. 3 Tahapan Penelitian

3.2.1 Studi Literatur

Tahap studi literatur dilakukan dengan mengeksplorasi informasi dari berbagai sumber referensi, seperti artikel ilmiah, situs *web* resmi, dan laporan penelitian terkait prediksi LoS, *deep learning*, XAI untuk melihat perkembangan penelitian terkait topik tersebut. Studi literatur juga bertujuan untuk mencari celah pada penelitian (*research gap*) yang akan dijadikan *research problem* yang kemudian memunculkan *research question* dan *research object*, hal ini menentukan arah penelitian dan tujuan penelitian yang akan dilakukan.

3.2.2 Pengumpulan Data

Penelitian ini menggunakan dataset sekunder yang bersumber dari Microsoft yang diperoleh dari repositori resmi GitHub

(https://github.com/microsoft/r-server-hospital-length-of-stay). Data tersebut dipublikasikan untuk keperluan riset dan memiliki 100.000 data pasien yang dirawat di rumah sakit dengan berisi 28 fitur yang mempengaruhi LoS pasien.

3.2.3 *Data Preprocessing*

Data preprocessing merupakan tahapan penting dalam pemrosesan data sebelum dilakukan pemodelan menggunakan *deep learning* yang bertujuan untuk meningkatkan kualitas data sesuai kebutuhan algoritma yang digunakan sehingga model dapat belajar dengan pola yang optimal.

Menurut Koukaras & Tjortjis (2025) *preprocessing* data membantu memperbaiki kualitas data, menurunkan *noise*, serta meminimalkan bias yang dapat mempengaruhi hasil prediksi model. Pada penelitian ini, tahap *data preprocessing* dilakukan secara sistematis untuk mempersiapkan dataset agar sesuai dengan kebutuhan model FT-*Transformer*, yaitu sebuah arsitektur *Transformer* yang telah terbukti efektif pada data tabular melalui representasi *embedding* untuk setiap fitur.

3.2.3.1 Pengecekan *Missing Value*

Missing Value mengacu pada keadaan di mana atribut dalam dataset tidak memiliki nilai atau memiliki nilai nol (NaN/NA). Keberadaan *missing value* dapat menghambat proses pelatihan model karena sebagian algoritma *machine learning*, termasuk model berbasis *neural network* tidak dapat memproses nilai nol secara langsung (Nijman dkk., 2022). Oleh karena itu, tujuan dari tahap ini adalah untuk memastikan bahwa seluruh data yang digunakan dalam pemodelan merupakan data yang lengkap dan siap untuk diproses.

3.2.3.2 Penghapusan Kolom yang Tidak Relevan

Penghapusan kolom yang tidak relevan merupakan bagian dari seleksi fitur, yaitu proses pemilihan atribut yang berkontribusi terhadap variabel target dan menghilangkan atribut yang tidak memberikan informasi penting. Tujuan dari tahapan ini adalah untuk mengurangi kompleksitas model, dimensi data, dan meminimalkan *noise* yang dapat mempengaruhi performa prediksi (Y. Chen dkk., 2025). Dalam konteks FT-*Transformer*, setiap fitur dianggap sebagai token yang akan dimasukkan ke dalam *embedding*. Oleh karena itu, penggunaan fitur yang tidak relevan dapat meningkatkan kompleksitas representasi masukan dan menambah informasi yang tidak diperlukan bagi model. Seleksi fitur yang tepat dapat membantu mengurangi *noise* serta meningkatkan efisiensi dan kemampuan generalisasi model (Max Kuhn & Kjell Johnson, 2026).

Pada penelitian ini dari 28 fitur, 4 fitur dihapus karena tidak relevan untuk proses pelatihan model yaitu ’eid’, *’vdate’*, *’discharged’*, *’lengthofstay’*. Penghapusan fitur tersebut bertujuan untuk memastikan hanya fitur relevan dan tidak menyebabkan bias yang digunakan dalam pemodelan. Semua fitur tersebut akan memengaruhi hasil dari model prediksi karena berisi data aktual tentang berapa lama LoS setiap pasien. Dalam *machine learning*, pemilihan fitur yang tepat sangat penting karena fitur yang tidak relevan atau yang bersifat bocoran informasi (data *leakage*) dapat menurunkan kemampuan generalisasi model (Sendak dkk., 2020).

3.2.3.3 *Feature Engineering*

Pada tahap ini dilakukan pembuatan fitur turunan bernama *number_of_issues* yang mengagregasi sebelas indikator kondisi kesehatan biner pasien. Kesebelas indikator tersebut meliputi '*dialysisrenalendstage', 'asthma', 'irondef', 'pneum', 'substancedependence',' psychologicaldisordermajor', 'depress', 'psychother', 'fibrosisandother', 'malnutrition', dan 'hem*o'. Fitur ini dibuat dengan menjumlahkan seluruh nilai dari indikator biner tersebut untuk setiap baris data, sehingga menghasilkan skor kumulatif yang merepresentasikan beban komorbiditas pasien.

Pembuatan fitur ini didasarkan pada bukti empiris bahwa jumlah kondisi kronis atau multimorbiditas merupakan prediktor yang signifikan terhadap lama rawat inap (*length of stay*). Stahl-Toyota dkk. (2025) menunjukkan bahwa peningkatan jumlah penyakit penyerta berkorelasi positif dengan durasi lama rawat inap pasien. Implementasi fitur ini memungkinkan model untuk menangkap efek interaksi dari berbagai kondisi kesehatan secara lebih ringkas dibandingkan jika kesebelas indikator digunakan secara terpisah.

3.2.3.4 *Encoding*

Arsitektur FT-*Transformer* menggunakan mekanisme *Feature Tokenizer* yang memperlakukan setiap atribut sebagai sebuah token sebelum diproses oleh blok *Transformer*. Berbeda dengan pendekatan *One-Hot Encoding* yang mengubah setiap kategori menjadi sejumlah fitur biner, fitur kategorikal direpresentasikan sebagai indeks kategori pada tahap *preprocessing* dan kemudian dipetakan menjadi representasi *embedding* oleh *Feature Tokenizer*. Pendekatan ini mempertahankan

setiap atribut sebagai satu fitur, sehingga hubungan antar kategori dapat dipelajari secara langsung melalui proses pembelajaran *embedding* (Gorishniy dkk., 2021).

Label *encoding* mengubah setiap kategori menjadi integer indeks, kemudian *embedding layer* memetakan setiap indeks ke vektor berdimensi seragam (*d_token*) yang dipelajari selama proses pelatihan. Dengan demikian, model dapat mempelajari representasi setiap kategori dalam ruang vektor tanpa perlu mengubah satu atribut menjadi beberapa fitur biner (Arat, 2022).

Pada penelitian ini, fitur kategorikal terdiri atas fitur nominal dan fitur ordinal yang diproses sesuai karakteristik masing-masing. Fitur nominal, yaitu *gender* dan *facid*, diproses menggunakan *OrdinalEncoder* dari *scikit-learn*. *Encoder* ini mengubah setiap kategori unik menjadi indeks integer berurutan yang kemudian digunakan sebagai masukan bagi *CategoricalTokenizer* pada arsitektur FT-*Transformer*. Selanjutnya, setiap indeks dipetakan ke dalam vektor embedding berdimensi *d_token* yang dipelajari secara otomatis selama proses pelatihan sehingga model dapat mempelajari representasi setiap kategori tanpa memerlukan *one-hot encoding* (Gorishniy dkk., 2021).

Berbeda dengan fitur nominal, variabel *rcount* diperlakukan sebagai fitur numerik ordinal karena merepresentasikan jumlah kunjungan ulang pasien dengan tingkatan kategori 0, 1, 2, 3, 4, dan 5+. Variabel ini memiliki hubungan urutan (*ordinal relationship*) yang perlu dipertahankan sehingga tidak diproses menggunakan *OrdinalEncoder* maupun *embedding*. Sebagai gantinya, dilakukan pemetaan manual menggunakan kamus nilai yang mengubah kategori 0, 1, 2, 3, 4,

dan 5+ menjadi nilai 0, 1, 2, 3, 4, dan 5. Pemetaan tersebut bertujuan mempertahankan informasi urutan antar kategori sehingga variabel *rcount* dapat diperlakukan sebagai fitur numerik. Setelah proses pemetaan selesai, variabel ini mengikuti tahapan standardisasi bersama fitur numerik lainnya sebelum digunakan sebagai masukan ke model FT-*Transformer*.

3.2.3.5 Standardisasi Fitur Numerik

Proses standardisasi adalah transformasi data numerik untuk distribusi dengan standar deviasi satu dan rata-rata (*mean*) mendekati nol. Standardisasi bertujuan menyeragamkan skala antarfitur sehingga perbedaan rentang nilai tidak menyebabkan suatu fitur memberikan kontribusi yang lebih dominan selama proses pembelajaran model (Yang dkk., 2025).

Pada penelitian ini, beberapa fitur numerik distandardisasi menggunakan metode fit_transform(), fitur numerik seperti '*hemotocrit', 'neutrophils', 'sodium', 'glucose', 'bloodureanitro', 'creatinine', 'bmi', 'pulse', dan 'respiration'* distandarisasi menggunakan *StandarScaler*(). Proses ini dilakukan agar distribusi fitur konsisten, proses ini dilakukan sebelum data dimasukkan pada model. Pada FT-*Transformer*, fitur numerik diproyeksikan menjadi *embedding* sebelum diproses dengan mekanisme *self-attention* (Gorishniy dkk., 2021). Oleh karena itu, proses standardisasi membantu mempercepat konvergensi dan menjaga stabilitas proses optimasi selama pelatihan model.

3.2.3.6 Pembagian Data

Pembagian data adalah proses membagi dataset menjadi data latih dan data uji untuk menilai kemampuan model dalam menggeneralisasi data yang belum pernah dilihat sebelumnya. Tujuan tahap ini adalah untuk menghindari *overfitting* dan memastikan performa model yang diperoleh mencerminkan kemampuan prediksinya pada data baru (Sivakumar dkk., 2024).

Pada penelitian ini, data dibagi menggunakan metode *fixed hold-out split* dengan rasio 80:20, di mana 80% data digunakan sebagai data *trainval* dan 20% sebagai data uji yang dijaga tetap tidak berubah sepanjang proses. Pencarian *hyperparameter* dilakukan menggunakan Optuna dengan satu *split* validasi tetap dari data *trainval* menggunakan rasio 87,5:12,5, sehingga diperoleh 87,5% data pelatihan dan 12,5% data validasi. Setelah *hyperparameter* terbaik ditemukan, model dilatih ulang sebanyak 30 iterasi menggunakan metode *repeated random subsampling* dengan *seed* yang berbeda pada setiap iterasi dan rasio *split* yang sama (87,5:12,5). Performa model pada setiap iterasi dievaluasi menggunakan data uji yang telah dipisahkan sejak awal, sehingga komposisi keseluruhan data adalah 70% data pelatihan, 10% data validasi, dan 20% data uji.

3.2.4 Optimasi *Hyperparamater*

Penentuan parameter model dilakukan melalui *hyperparameter optimization* menggunakan Optuna. Proses ini dilakukan secara iteratif dengan membangun model berdasarkan kombinasi parameter tertentu, melatih model pada data pelatihan, serta mengevaluasi kinerjanya pada data validasi (Akiba dkk., 2019). Parameter yang dioptimasi meliputi dimensi *embedding* (d_token), jumlah

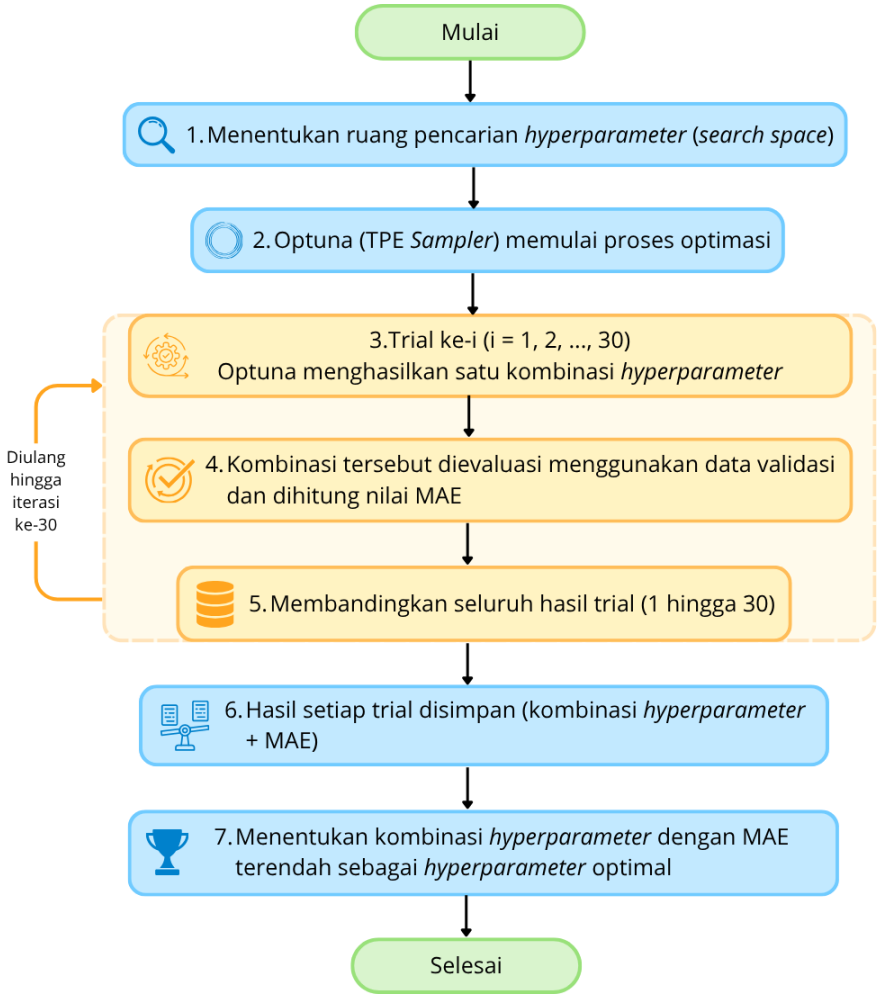
layer encoder (n_layers), jumlah *attention head* (n_*heads*), *dropout*, *learning rate*, *batch size*, faktor dimensi *feed-forward*, dan *weight decay*. Sebelum model dilatih secara menyeluruh menggunakan seluruh data latih, proses optimasi dilakukan untuk meminimalkan nilai kesalahan dengan menggunakan data validasi, sehingga menghasilkan kombinasi parameter yang memiliki kinerja terbaik (Lindauer dkk., 2019).

Ruang pencarian hyperparameter mengacu pada rentang hyperparameter yang direkomendasikan oleh Gorishniy dkk. (2021) sebagai acuan resmi dalam pengembangan FT-*Transformer*. Nilai d_token=192 turut disertakan karena merupakan nilai *default* resmi FT-*Transformer* sebagaimana tercantum pada Tabel 12 dalam *paper* yang sama. Selain itu, beberapa nilai tambahan seperti *n_heads*=4 dan *batch_size*=128 dimasukkan ke dalam ruang pencarian berdasarkan eksperimen pendahuluan untuk menyesuaikan karakteristik *dataset* LoS, sehingga Optuna memiliki ruang eksplorasi yang lebih luas dalam menemukan kombinasi *hyperparameter* terbaik. Rentang nilai *hyperparameter* untuk pencarian dengan optuna disajikan pada Tabel 3. 1.

Tabel 3. 1 Rentang Nilai *Hyperparameter*

Parameter	**Nilai yang Dicoba**
d_token	64, 128, 192, 256
n_heads	4, 8
n_layers	1, 2, 3, 4
feed forward network factor	4/3–8/3 (≈ 1,3333–2,6667)
dropout	0,0 hingga 0,5
learning rate	1×10^{-5}–1×10^{-3} (≈ 0,00001–0,001)
weight decay	1×10^{-6}–1×10^{-3} (≈ 0,000001–0,001)
batch size	128, 256, 512

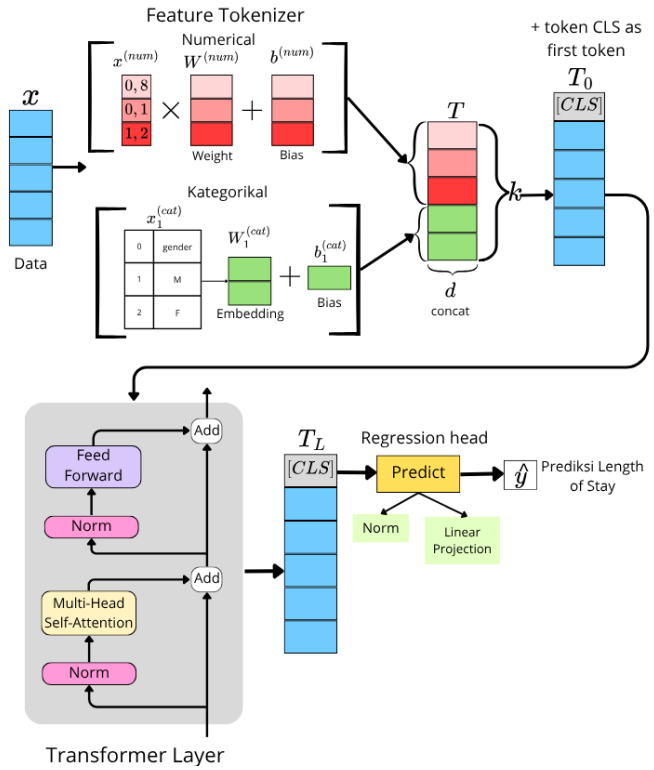
Alur optimasi hyperparameter menggunakan Optuna pada penelitian ini ditunjukkan pada Gambar 3. 4. Proses diawali dengan penentuan ruang pencarian *hyperparameter*, kemudian Optuna menghasilkan kombinasi *hyperparameter* pada setiap *trial* yang selanjutnya dievaluasi menggunakan nilai MAE pada data validasi. Proses tersebut dilakukan secara berulang hingga mencapai jumlah *trial* yang ditentukan, kemudian kombinasi *hyperparameter* dengan nilai MAE terendah dipilih sebagai konfigurasi *hyperparameter* terbaik yang digunakan pada seluruh skenario pemodelan FT-*Transformer*.



Gambar 3. 4 Alur proses optimasi *hyperparameter* menggunakan Optuna

3.2.5 Penerapan Model

Penerapan model pada penelitian ini menggunakan FT-*Transformer* (*Feature Tokenizer Transformer*) untuk prediksi LoS berbasis data tabular. FT-*Transformer* memperlakukan setiap fitur sebagai token dan memodelkan interaksi antar fitur melalui mekanisme *self-attention* (Gorishniy dkk., 2021). Metode ini dipilih karena mampu mengidentifikasi hubungan nonlinier dan interaksi kompleks yang muncul dalam data klinis. Dalam implementasi, model dibangun menggunakan PyTorch dan terdiri dari beberapa tahapan utama, yaitu *feature tokenization*, penambahan token khusus, pemrosesan token menggunakan *Transformer Encoder*, dan *regression head* untuk menghasilkan *output* kontinu (Gorishniy dkk., 2021). Gambar 3. 5 merupakan tahapan penerapan model.



Gambar 3. 5 Alur Penerapan Model (Modifikasi dari (Peng & Gao, 2025))

Pada tahap *feature tokenizer*, setiap fitur numerik dipetakan ke dalam ruang *embedding* tertentu (*d_token*) menjadi vektor berdimensi tertentu. Fitur numerik diproyeksikan ke dalam ruang *embedding* menggunakan transformasi linier sebagaimana dirumuskan pada Persamaan 2. 1, sedangkan fitur kategorikal direpresentasikan menggunakan *embedding layer*. Dengan demikian, setiap fitur diperlakukan sebagai token yang dapat diproses oleh arsitektur *Transformer* (Gorishniy dkk., 2021).

Selanjutnya, token khusus [CLS] ditempatkan pada awal urutan token sebagai representasi global dari seluruh fitur. Representasi tersebut kemudian diproses oleh *Transformer Encoder* yang terdiri atas beberapa lapisan *Multi-Head Self-Attention* dan *Feed Forward Network* dengan fungsi aktivasi GELU untuk mempelajari hubungan serta interaksi antarfitur secara global. Mekanisme *Multi-Head Self-Attention* menghitung hubungan antar token menggunakan mekanisme *attention* sebagaimana dirumuskan pada Persamaan 2. 2, kemudian hasil dari setiap *attention head* digabungkan melalui mekanisme *Multi-Head Attention* sebagaimana ditunjukkan pada Persamaan 2. 3. Selanjutnya, representasi fitur diproses *oleh Feed Forward Network* untuk meningkatkan kemampuan representasi model sebagaimana dirumuskan pada Persamaan 2. 4. Pada FT-*Transformer*,. Gorishniy dkk. (2021) merekomendasikan penggunaan fungsi aktivasi ReGLU, yang mekanismenya ditunjukkan pada Persamaan 2.5 dan Persamaan 2. 6, sehingga model dapat mempelajari hubungan antarfitur secara lebih efektif.

Representasi akhir diambil dari *embedding* token [CLS] yang telah diproses oleh *Transformer Encoder*. Representasi tersebut selanjutnya diteruskan ke

regression head yang terdiri atas beberapa lapisan linier untuk menghasilkan satu nilai kontinu berupa prediksi LoS sebagaimana dirumuskan pada Persamaan 2. 7 (Gorishniy dkk., 2021).

Dalam proses pelatihan, digunakan *optimizer* AdamW karena mampu menerapkan regularisasi *weight decay* secara terpisah dari mekanisme pembaruan gradien sehingga membantu meningkatkan generalisasi model (Zhou dkk., 2024). Fungsi *loss* yang digunakan adalah Huber *Loss* yang pertama kali diperkenalkan oleh Huber (1964) dan telah diterapkan dalam konteks regresi *deep learning* oleh (Tyralis dkk., 2025) yang lebih tahan terhadap *outlier* daripada *Mean Squared Error* (Yuan & Ren, 2024), sehingga sesuai untuk prediksi LoS yang mungkin memiliki nilai ekstrem. Selain itu, diterapkan *learning rate scheduler* berbasis *cosine annealing with warm restarts* (Loshchilov & Hutter, 2017) untuk membantu stabilitas konvergensi (C. Zhang dkk., 2023), serta *gradient clipping* yang dapat mencegah *exploding gradient* selama pelatihan (Pascanu dkk., 2013; Brenndoerfer, 2025).

Pelatihan model FT-*Transformer* dilakukan menggunakan seluruh fitur yang tersedia tanpa seleksi fitur terlebih dahulu. Proses pelatihan menerapkan metode *repeated random split* sebanyak 30 iterasi untuk memperoleh estimasi performa model yang lebih stabil. Pada setiap iterasi, model dilatih pada data pelatihan menggunakan *hyperparameter* terbaik hasil pencarian Optuna dengan fungsi *loss* Huber dan *optimizer* AdamW. Selama pelatihan, kinerja model dievaluasi menggunakan data validasi setiap lima *epoch* untuk memantau tanda-

tanda *overfitting*. Model dengan nilai MAE terkecil pada data validasi disimpan sebagai model terbaik untuk iterasi tersebut.

3.2.6 Seleksi Fitur

Seleksi fitur dilakukan menggunakan nilai SHAP (*SHapley Additive exPlanations*) yang diperoleh setelah model FT-*Transformer* selesai dilatih. Proses ini bertujuan untuk mengidentifikasi fitur-fitur yang paling berpengaruh terhadap prediksi lama rawat inap, kemudian mengurangi dimensi fitur dengan mempertahankan hanya fitur-fitur terpenting. SHAP *KernelExplainer* digunakan untuk menghitung kontribusi setiap fitur terhadap prediksi model pada data validasi. Nilai SHAP absolut dari setiap fitur dirata-rata untuk masing-masing iterasi, kemudian dirata-rata kembali ke seluruh 30 iterasi sehingga diperoleh stabilitas estimasi pengaruh fitur (Lundberg & Lee, 2017). Berdasarkan nilai rata-rata SHAP absolut tersebut, seluruh fitur diurutkan dari yang paling berpengaruh hingga paling tidak berpengaruh.

Selanjutnya dilakukan evaluasi subset fitur secara bertahap menggunakan beberapa nilai *k*. Pendekatan evaluasi subset fitur diadaptasi dari penelitian (Peng & Gao, 2025) yang mengevaluasi performa model menggunakan penambahan jumlah fitur secara bertahap, yaitu Top-10, Top-20, Top-30, Top-40, dan seluruh fitur. Namun, penelitian ini menggunakan 25 fitur setelah tahap *preprocessing*, sehingga nilai *k* disesuaikan dengan karakteristik data dan kebutuhan desain eksperimen. Oleh karena itu, subset fitur yang dievaluasi terdiri atas Top-5, Top-10, Top-15, Top-20, dan seluruh fitur.

Untuk setiap subset fitur, model FT-*Transformer* dilatih kembali menggunakan prosedur *Repeated Random Split* sebanyak 30 iterasi dengan konfigurasi *hyperparameter* yang identik dengan model terbaik hasil optimasi. Seluruh skenario kemudian dievaluasi menggunakan nilai rata-rata MAE pada data validasi. Subset dengan nilai MAE rata-rata terendah dipilih sebagai subset fitur terbaik.

3.2.7 Evaluasi Model

Evaluasi model dilakukan untuk mengukur tingkat kinerja model melakukan generalisasi pada data yang tidak dilihat saat pelatihan, dan untuk menilai performa prediksi pada data uji. Pada penelitian ini digunakan beberapa metrik evaluasi seperti MAE, RMSE, RAE, dan RSE (Peng & Gao, 2025) karena setiap metrik mengevaluasi kesalahan prediksi dari perspektif yang berbeda.

Penggunaan beberapa metrik evaluasi tersebut bertujuan untuk memberikan penilaian yang lebih komprehensif terhadap performa model seperti apakah model cenderung memiliki error rata-rata kecil (MAE), atau model sensitif terhadap error besar (RSME), dan seberapa baik kinerjanya dibandingkan dengan pendekatan *baseline* berbasis rata-rata (RAE dan RSE). Kombinasi ini dapat membuat kesimpulan performa model lebih kuat daripada hanya bergantung pada satu metrik (Miller dkk., 2024).

3.2.8 Interpretasi

Tahap interpretasi model dilakukan untuk menjelaskan bagaimana model FT-*Transformer* menghasilkan prediksi LoS serta mengidentifikasi fitur-fitur yang

memberikan kontribusi terbesar terhadap keluaran model. Pada penelitian ini, interpretasi model dilakukan menggunakan SHAP (*SHapley Additive exPlanations*), yaitu metode yang didasarkan pada nilai Shapley dari teori permainan untuk mengalokasikan kontribusi setiap fitur secara aditif terhadap hasil prediksi model (Lundberg & Lee, 2017). Metode ini memungkinkan untuk menjelaskan setiap prediksi sebagai nilai dasar, atau nilai yang diharapkan, ditambah dengan kontribusi masing-masing fitur.

SHAP dipilih karena mampu menjelaskan model kompleks seperti FT-*Transformer* yang berbasis *neural network* dan *self-attention*. SHAP diimplementasikan menggunakan *KernelExplainer*, yang bersifat *model-agnostic* dan bekerja dengan mengestimasi kontribusi fitur berdasarkan perubahan prediksi ketika fitur dihilangkan atau dipertahankan dalam kombinasi tertentu (shap.KernelExplainer - SHAP latest documentation, 2018).

Interpretasi dilakukan dalam tiga tingkat analisis, yaitu global, lokal dan berbasis kelompok resiko. Interpretasi global diperoleh dengan menghitung rata-rata nilai absolut SHAP (*mean* SHAP) untuk setiap fitur. Nilai tersebut merepresentasikan besarnya kontribusi rata-rata suatu fitur terhadap hasil prediksi LoS pada seluruh sampel. Fitur dengan nilai *mean* SHAP tertinggi dianggap memiliki pengaruh paling besar terhadap keputusan model (Ponce-Bobadilla dkk., 2024). Pendekatan ini umum digunakan dalam analisis SHAP untuk menentukan peringkat kepentingan fitur secara keseluruhan. Selain itu, interpretasi lokal dilakukan untuk menjelaskan prediksi pada individu tertentu. Dalam konteks ini, nilai SHAP positif menunjukkan peningkatan prediksi LoS fitur, sedangkan nilai

SHAP negatif menunjukkan penurunan prediksi (Ponce-Bobadilla dkk., 2024). Dengan demikian, alasan khusus mengapa model memprediksi lama rawat inap yang tinggi atau rendah pada pasien tertentu dapat diketahui.

Analisis berbasis kelompok risiko juga dilakukan dengan mengelompokkan pasien berdasarkan fitur *number_of_issues* yang merepresentasikan jumlah kondisi kesehatan. Pasien dikelompokkan menjadi tiga kategori yaitu *low-risk* dengan jumlah isu atau kondisi kesehatan kurang dari atau sama dengan 2, *medium-risk* dengan jumlah isu antara 3 hingga 5, dan *high-risk* dengan jumlah isu lebih dari 5 (Peng & Gao, 2025). Nilai SHAP setiap fitur dianalisis pada masing-masing kelompok untuk mengidentifikasi apakah faktor-faktor yang mempengaruhi prediksi berbeda antartingkat keparahan pasien (Ponce-Bobadilla dkk., 2024).

Pendekatan interpretasi bertingkat ini memungkinkan penelitian untuk tidak hanya mengetahui fitur apa yang penting secara global, tetapi juga memahami bagaimana fitur tersebut berperan pada tingkat individu dan pada subkelompok pasien yang berbeda (Mermer dkk., 2026).

3.2.9 Ancaman terhadap Validitas

Pengujian validitas dilakukan untuk meninjau dan mengevaluasi keseluruhan proses penelitian yang telah dilakukan, mulai dari pengumpulan data, *preprocessing*, optimasi *hyperparameter*, hingga implementasi model FT-*Transformer*. Evaluasi dilakukan untuk memverifikasi bahwa hasil eksperimen bersifat akurat, konsisten, dan dapat dipertanggungjawabkan berdasarkan kaidah ilmiah.

Pendekatan yang digunakan dalam proses validasi penelitian ini adalah ancaman terhadap validitas (*Threats to Validity)*. *Threats to Validity* merupakan faktor-faktor yang berpotensi mempengaruhi validitas hasil penelitian maupun kesimpulan yang dihasilkan. Pada penelitian prediksi *Length of Stay* menggunakan FT-*Transformer*, ancaman validitas dapat berasal dari beberapa aspek seperti *internal validity*, *construct validity*, *conclusion validity*, dan *external validity*. Oleh karena itu, identifikasi terhadap ancaman validitas dilakukan agar hasil eksperimen dan evaluasi model dapat dianalisis secara lebih objektif dan komprehensif (Lago dkk., 2024).