

## BAB 3 METODOLOGI PENELITIAN

### 3.1 Metodologi Penelitian

Pengerjaan penelitian "Analisis Sentimen Lirik Lagu untuk Prediksi Popularitas Lagu dengan Algoritma BERT" dilaksanakan secara bertahap. Tahap-tahap pengerjaan penelitian ini dijabarkan sebagai berikut.

#### 3.1.1 Telaah Literatur

Telaah literatur dilakukan dengan cara mengumpulkan informasi dari penelitian-penelitian terdahulu yang mendukung penelitian ini. Dalam hal ini, literatur yang dikumpulkan dan digunakan merupakan literatur yang berkaitan dengan lirik lagu, analisis sentimen, *preprocessing*, dan algoritma BERT. Selain itu, teori mengenai *accuracy*, *precision*, *recall*, dan *f1-score* juga dipelajari. Literatur yang telah dikumpulkan dipelajari dan dijadikan referensi dalam penelitian ini.

#### 3.1.2 Analisa Kebutuhan

Pada tahap ini, dilakukan analisis mengenai hal-hal yang dibutuhkan untuk menjalankan penelitian ini. Dalam hal ini, kebutuhan perangkat keras dan lunak, hingga data penelitian berkaitan analisis sentimen lirik lagu berbasis algoritma BERT dianalisis. Dengan demikian, penelitian ini dapat dilakukan dan menghasilkan *output* berupa prediksi popularitas lagu, sesuai dengan tujuan awal penelitian.

#### 3.1.3 Pengumpulan Data

Penelitian ini menggunakan data yang didapatkan dari Kaggle, situs web penyedia *dataset* publik [14]. *Dataset* ini mengandung data-data lagu dalam *playlist* Top 50 di berbagai negara, yang tersedia di platform musik Spotify. *Dataset* ini dibentuk menggunakan API Spotify dan API Genius [13]. Data ini akan melalui tahap *preprocessing* agar siap diolah dan diproses lebih lanjut.

### 3.1.4 Perancangan Model

Rancangan proses kerja model analisis sentimen meliputi penentuan data yang dimasukkan ke dalam model, proses perhitungan dan analisis oleh model, serta *output* yang dihasilkan oleh model. Dalam hal ini, dilakukan perancangan untuk implementasi algoritma BERT untuk analisis sentimen lirik. Dengan adanya analisis sentimen ini, popularitas lagu dapat diprediksi. Adapun rancangan yang ada ditampilkan dalam bentuk diagram *flowchart*.

### 3.1.5 Implementasi Model

Model dibuat sesuai dengan spesifikasi model analisis sentimen, yang sudah dirancang sebelumnya. Model analisis sentimen yang sudah dibuat akan diterapkan pada data yang tersedia. Data lagu akan diproses, dilatih, dan juga diuji menggunakan model yang telah dibuat.

### 3.1.6 Pengujian Model dan Evaluasi

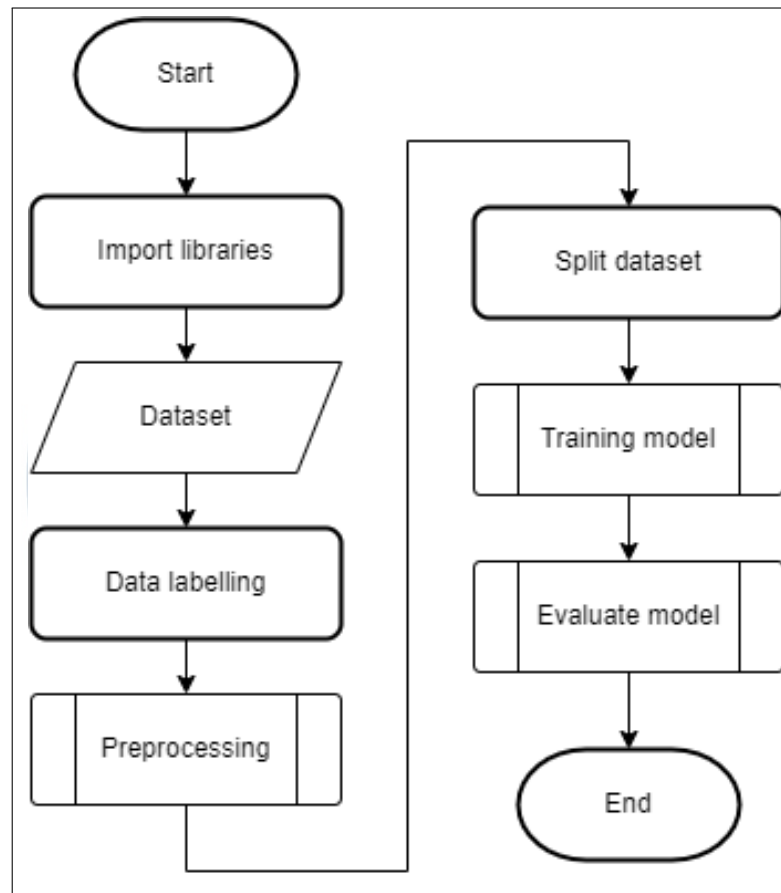
Sistem model yang telah dibuat diuji dan dievaluasi fungsionalitasnya. Model dievaluasi menggunakan metrik evaluasi. Dari hasil perhitungan *accuracy*, *precision*, *recall*, dan *f1-score*, performa model yang telah dibuat dapat dievaluasi. Dengan demikian, dapat dievaluasi hasil penerapan algoritma BERT pada analisis sentimen lirik lagu, untuk keperluan prediksi popularitas lagu.

### 3.1.7 Dokumentasi

Setiap proses yang dilakukan pada penelitian ini dicatat dan didokumentasikan dalam bentuk laporan penelitian. Dengan demikian, seluruh proses penelitian berikut dengan hasil dari penelitian ini dapat dipertanggungjawabkan.

## 3.2 Perancangan Model

Pada penelitian ini, model BERT dirancang untuk dapat memprediksi popularitas lagu berdasarkan analisis sentimen lirik lagu. Pada tahap perancangan model, alur kerja dari model dibuat dalam bentuk *flowchart*. Berikut merupakan *flowchart* utama dari perancangan model pada penelitian ini.



Gambar 3.1. *Flowchart* utama

### 3.2.1 Import Libraries

Pada tahap ini, dilakukan pengimporan *library-library* yang dibutuhkan untuk pemodelan. *Library-library* yang dibutuhkan untuk *preprocessing*, pembuatan model, dan evaluasi performa model diimpor terlebih dahulu agar dapat digunakan pada proses pemodelan.

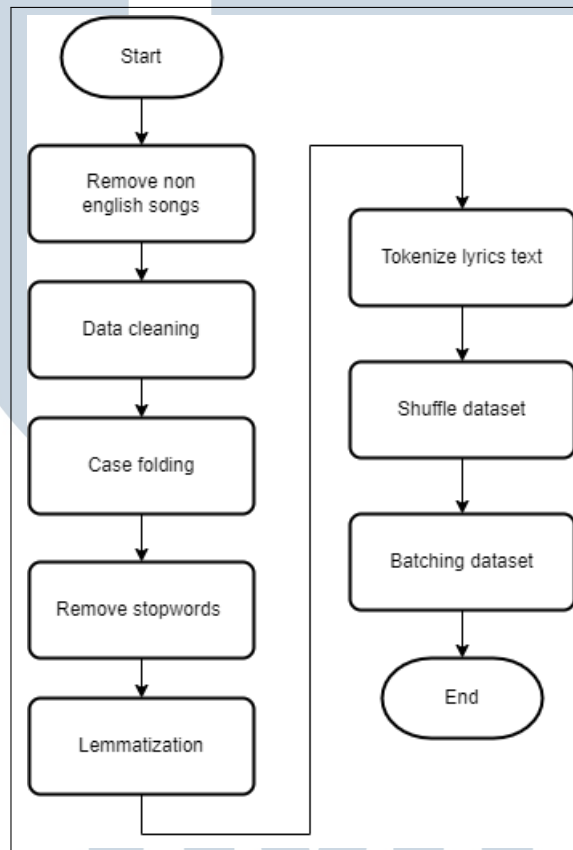
### 3.2.2 Data Labelling

Tahap *data labelling* adalah tahap klasifikasi data, sehingga data memiliki label-label tertentu. Pada tahap *data labelling*, lagu diklasifikasikan berdasarkan indeks popularitasnya. Adapun data popularitas lagu disediakan dalam bentuk indeks popularitas lagu, dengan nilai 0-100. Lagu dengan indeks popularitas menjadi 0-33 diberi label *less popular* (kurang populer), lagu dengan indeks popularitas 34-66 diberi label *quite popular* (cukup terkenal), dan lagu dengan

indeks popularitas 67-100 diberi label *very popular* (sangat terkenal).

### 3.2.3 Preprocessing

Tahap *preprocessing* merupakan tahap pengolahan data, agar data lebih terstruktur dan siap digunakan sebagai masukan dalam pembuatan model.



Gambar 3.2. Flowchart preprocessing

Pertama, lirik lagu akan diidentifikasi bahasanya menggunakan *library* Langdetect. Setelah diketahui bahasa dari lagu tersebut, data lagu yang tidak terdeteksi berbahasa Inggris akan dihapus. Setelah data lagu tersaring sesuai dengan bahasanya, data lirik lagu akan melewati tahap *data cleaning*, *case folding*, *stopwords removal*, dan *lemmatization*. Pada proses *data cleaning*, semua karakter non alfabet pada lirik lagu, seperti simbol dan angka, akan dihapus. Kemudian, semua huruf pada lirik lagu diubah menjadi huruf kecil. Kata-kata yang tidak memiliki makna penting (*stopwords*) juga dihapus dari lirik lagu, menggunakan *library* NLTK (*Natural Language Toolkit*). Ketika suatu kata pada lirik terdeteksi

pada daftar *stopwords*, kata tersebut ditandai dan dibuang. Setiap kata pada lirik lagu juga diubah bentuknya ke bentuk dasar dari kata tersebut, dengan menggunakan *library* NLTK.

Setelah itu, lirik ditokenisasi dengan menggunakan *function* AutoTokenizer dari *library* HuggingFace Transformers. Dengan demikian, teks dapat ditokenisasi sehingga sesuai dengan ketentuan *input* model BERT. Kemudian, sebuah objek *dataset* baru dibuat menggunakan *library* Tensorflow, dengan isi *input* model BERT. *Dataset* ini di-*shuffle* agar persebaran data lagu lebih merata. Setelah itu, *dataset* di-*batching*, sehingga lebih mudah dan lebih cepat diolah pada pembuatan model.

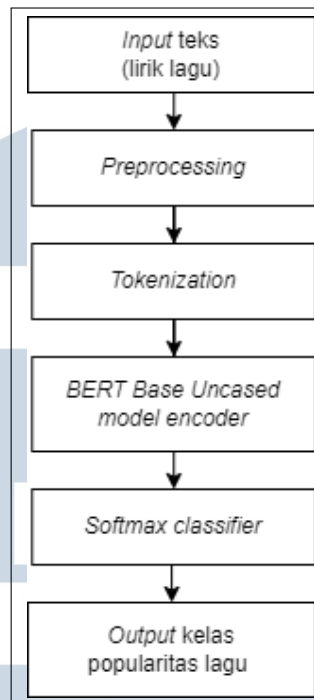
### 3.2.4 Split Dataset

Pada tahap ini, *dataset* dibagi menjadi dua, yaitu *dataset training* dan *dataset testing*. *Dataset training* merupakan *dataset* yang dipakai untuk melatih algoritma dalam pembuatan model, sedang *dataset testing* dipakai untuk menguji performa model yang sudah dibuat. Pembagian *dataset* dengan rasio 80:20 telah dilakukan di banyak penelitian terdahulu, untuk analisis sentimen berbasis BERT. Dengan rasio perbandingan 80:20, performa model dapat mencapai performa yang optimal [25][26]. Dengan demikian, pada penelitian ini *dataset* dibagi menjadi 80% *dataset training* dan 20% *dataset testing*.

### 3.2.5 Training Model

Suatu model BERT dikembangkan untuk keperluan analisis sentimen. Berdasarkan hasil analisis sentimen pada lirik lagu, prediksi popularitas lagu dilakukan. Berikut merupakan diagram rancangan model BERT yang dikembangkan.

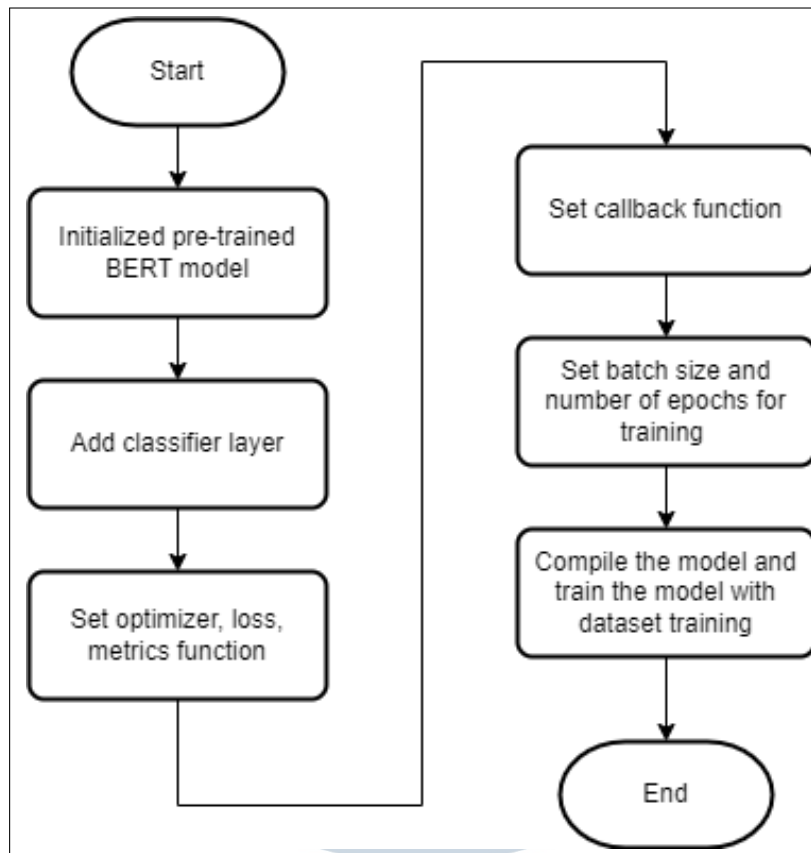
U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A



Gambar 3.3. Rancangan arsitektur model BERT

Pertama, *input* teks berupa lirik lagu, akan melewati tahap *preprocessing*, seperti *data cleaning*, *case folding*, *stopwords removal*, dan *lemmatization*. Setelah itu, *input* teks ditokenisasi. Proses tokenisasi *input* teks meliputi *token embedding*, *segment embedding*, dan *position embedding*. Dengan proses tokenisasi, bentuk *input* teks berubah menjadi vektor, yang berisi representasi token-token *input* teks. *Pre-trained* model BERT jenis *bert-based-uncased* kemudian diimpor. Adapun model *bert-based-uncased* adalah model BERT yang sudah dilatih dengan kata-kata bahasa Inggris non-kapital. Model ini akan menerima token-token representasi *input* teks. Hasil dari proses model BERT ditampung pada *softmax classifier*. Dengan *softmax classifier*, *output* berupa kelas popularitas lagu dapat dihasilkan.

Model BERT kemudian dilatih dengan data *training*. Selain itu, *optimizer*, *loss*, dan *metric function* yang digunakan untuk menghitung akurasi pelatihan model, juga di-*setup*. Kemudian, *batch size* dan jumlah *epoch* pada pelatihan model diatur. Pelatihan model menggunakan metode *callback early stopping* sehingga langsung berhenti ketika tidak ada improvisasi yang signifikan. Progres pelatihan model juga disimpan dengan menggunakan *checkpoint*. Model yang telah dirancang kemudian di-*compile* dan dilatih dengan data *training*.

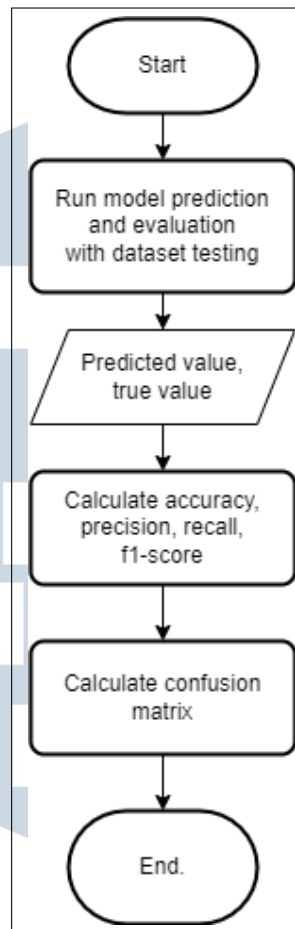


Gambar 3.4. Flowchart training model

### 3.2.6 Evaluate Model

Model yang sudah dilatih, diuji performanya menggunakan *dataset testing*. Model di-*evaluate* dengan *dataset testing*. Proses *prediction* juga dijalankan dengan menggunakan *dataset testing*. Nilai hasil prediksi dari model akan dibandingkan dengan nilai sesungguhnya. Dari hasil prediksi tersebut, *accuracy*, *precision*, *recall*, *f1-score* dari model dapat diketahui. Performa model dalam melakukan prediksi juga dapat dilihat melalui *confusion matrix*.

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

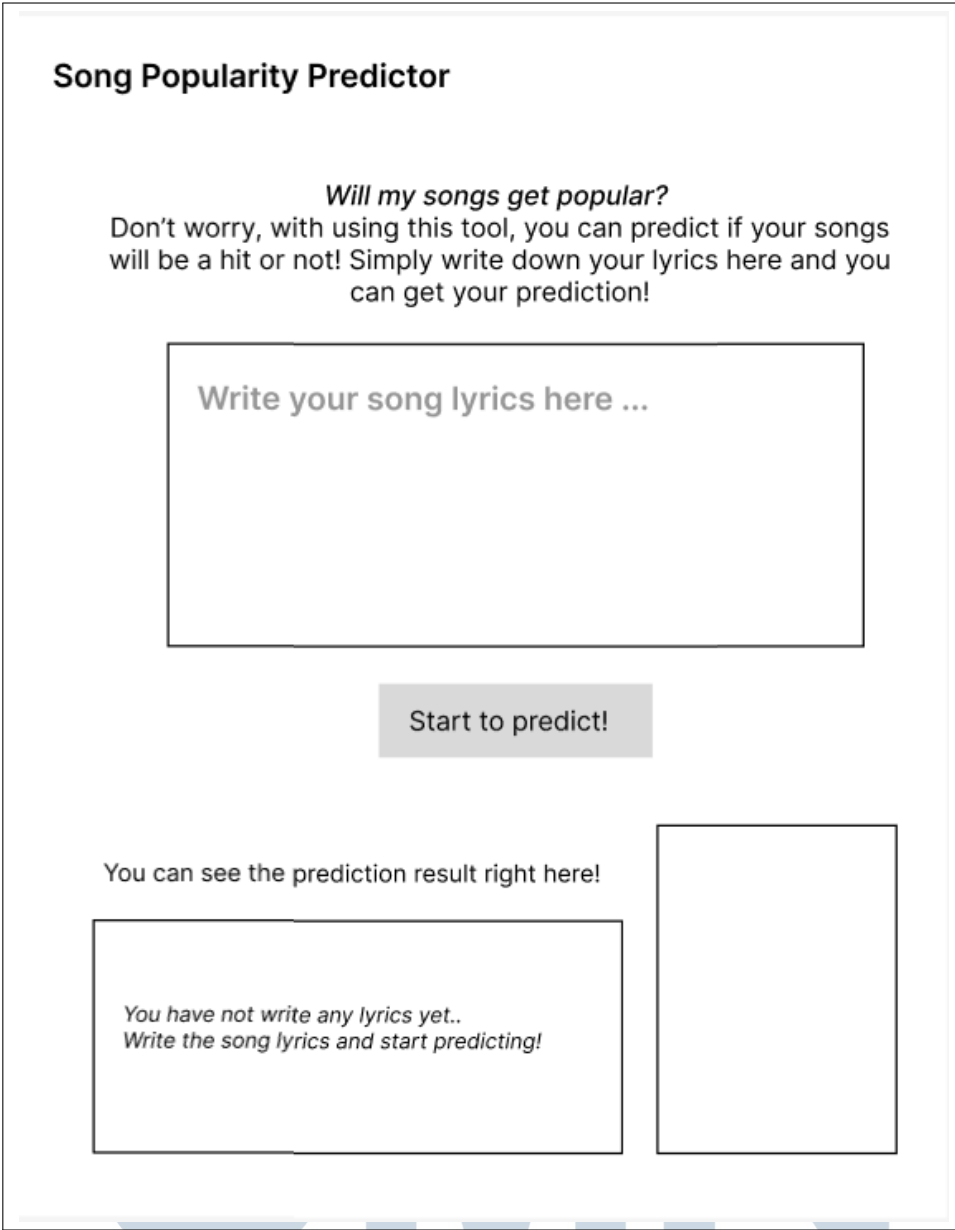


Gambar 3.5. Flowchart evaluate model

### 3.3 Perancangan Desain Antarmuka

Perancangan desain antarmuka (*interface*) dilakukan sehingga bentuk tampilan sistem yang dikembangkan dapat diketahui. Pada penelitian ini, sebuah aplikasi *website* sederhana akan dibuat. Melalui *website* ini, model yang sudah dikembangkan dapat diimplementasikan. Dengan demikian, *user* dapat melakukan prediksi popularitas suatu lagu berdasarkan liriknya, dengan menggunakan *website* tersebut. Berikut merupakan desain tampilan *interface* dari aplikasi hasil implementasi analisis sentimen lirik lagu, dengan algoritma BERT.





Gambar 3.6. Desain tampilan *user interface*

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A