BAB 3
METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Penelitian dengan judul ”Implementasi Algoritma BERT untuk Klasifikasi Topik Penelitian di Universitas Multimedia Nusantara” dilakukan secara bertahap. Tahapan ini Memberikan gambaran tentang obyek penelitian, analisis semua permasalahan yang ada, dan pendekatan yang digunakan dalam penelitian. Pada subbab ini juga dilaporkan secara detail rancangan terhadap penelitian yang dilakukan, baik perancangan secara umum dari sistem yang dibangun, maupun perancangan yang lebih spesifik. Berikut metodologi yang digunakan:

1. Studi Literatur
   Penelitian yang dilakukan ini disertai dengan adanya studi literatur. Studi literatur yang dilakukan adalah dengan mengumpulkan informasi berdasarkan penelitian yang telah dilakukan sebelumnya. Pada penelitian ini literatur yang digunakan merupakan literatur-literatur yang berhubungan dengan *multilabel text classification*, *preprocessing*, *precision, recall, f1-score* dan *accuracy* serta algoritma yang digunakan yaitu, BERT. Literatur-literatur yang digunakan telah dikumpulkan dan menjadi referensi pada penelitian ini.

2. Pengumpulan Data
   Pada metode pengumpulan data, penulis melakukan studi literatur, observasi dan wawancara. Studi Literatur yang dilakukan pada penelitian kali ini yaitu, melakukan pendekatan penelitian yang bertumpu pada jurnal ilmiah, buku, dan artikel ilmiah, yang memiliki relevansi dengan pertanyaan penelitian yang telah dirumuskan pada rumusan masalah pada penelitian ini. Sedangkan pada Observasi dan wawancara dilakukan secara langsung terhadap tempat dilakukannya penelitian dan melakukan wawancara dengan narasumber dari tempat penelitian tersebut. Penelitian ini juga menggunakan dataset yang didapatkan dari *website* Huggingface, dimana pada dataset ini terdapat pembagian dari kategori UN SDG serta judul dari artikel yang akan digunakan pada tahapan *preprocessing* dan *training model*.

3. Perancangan Model

Tahapan ini merupakan tahapan dimana peneliti melakukan perancangan tentang sistem yang akan dibuat berupa perancangan *flowchart* yang akan digunakan pada tahapan implementasi.

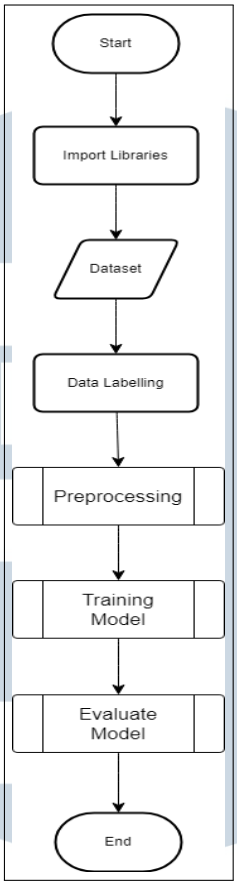
4. Implementasi Model
   Tahapan ini dilakukan untuk menyusun dari rancangan sebelumnya menjadi suatu sistem berfungsi sesuai dengan apa yang telah dirancang sebelumnya. Pada tahapan ini akan dilakukan pembuatan model yang dimulai dengan *load dataset* yang digunakan untuk memanggil dataset yang akan digunakan. Pada penelitian ini dataset yang digunakan ialah dataset dari *website* Huggingface. Kemudian dilanjutkan dengan *data labelling*, melakukan tahapan *preprocessing*, *training model* dan *evaluate model*

5. Pengujian Model dan Evaluasi
   Pada tahapan ini sistem akan melakukan tahapan pengujian yang telah di rancang pada tahapan sebelumnya. Pada tahapan ini terdapat beberapa skenario yang akan dipakai dimana, tujuan menggunakan berbagai skenario ini adalah untuk mencari model terbaik yang nantinya akan digunakan pada akhirnya. Setelah melakukan pengujian, evaluasi akan dilakukan untuk merangkup apa saja hal yang telah dilakukan selama proses pengujian berlangsung.

6. Dokumentasi
   Tahapan ini merupakan tahap akhir yaitu, melakukan dokumentasi melalui penulisan laporan dimana dokumentasi berisikan langkah-langkah penulisan laporan dan pembuatan sistem secara terstruktur.

3.2 Perancangan Model

Gambar 3.1 merupakan *flowchart* model secara keseluruhan. Pada bagian ini terdapat proses yang dilakukan ketika melakukan perancangan terhadap algoritma BERT dalam melakukan *text classification* pada label SDG. Pada tahapan ini terdapat flowchart dan mockup yang menjadi dasar dari perancangan model ini.



Gambar 3.1. *Flowchart* secara keseluruhan

1. *Import Libraries*

   Import libraries merupakan tahapan untuk melakukan impor terhadap *library-library* yang dibutuhkan selama proses pengerjaan terhadap model berlangsung. *Library* berfungsi untuk mendukung proses pembuatan modelnseperti, *preprocessing, training* dan *evaluate model* menjadi lebih mudah dikarenakan adanya bantuan dari *library* yang digunakan.

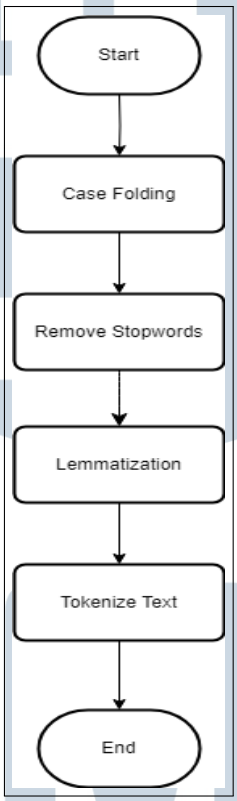
2. *Data Labelling*

   Data labelling merupakan tahapan untuk melakukan label terhadap data-data tertentu. *Data labelling* bertujuan agar membagi data menjadi beberapa kategori berdasarkan karakteristik data. *Data labelling* sendiri dilakukan dengan cara mengimpor dataset kemudian dataset-dataset tersebut dikategorikan berdasarkan csv yang telah ada, sehingga menjadikannya label yang dapat dibaca dan diterapkan pada proses pembuatan model. *Data labelling* pada penelitian ini terbagi menjadi 17 kategori yaitu, 17 kategori UN SDG. Kategori-kategori ini merupakan agenda yang diterima

dan diakui oleh seluruh negara anggota Perserikatan Bangsa-Bangsa (PBB) serta berbagai pihak pemangku kepentingan, termasuk pemerintah, sektor swasta, dan organisasi masyarakat sipil [5] (Lampiran 1, W04).

3. *Preprocessing*

   Gambar 3.2 menunjukkan alur dari tahapan *preprocessing* pada pengerjaan sistem. Tujuan dari tahap *preprocessing* sendiri ialah agar data menjadi lebih rapi dan mudah digunakan ketika sedang mengerjakan pembuatan model.



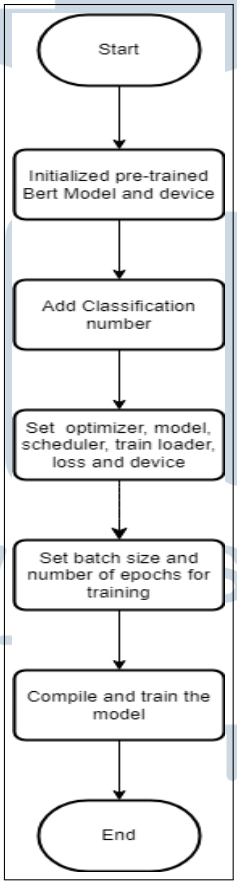
Gambar 3.2. *Flowchart* Modul Preprocessing

Dimulai dengan *case folding* yang merupakan tahapan pengolahan teks yang melibatkan huruf pada teks menjadi huruf besar atau kecil yang bertujuan untuk memastikan konsistensi teks dan mengurangi kompleksitas pada teks. Pada tahapan *remove stopwords* dan *lemmatization* dimana kedua hal ini merupakan bagian dari *library Natural Language Toolkit* (NLTK). Tujuan dari menggunakan kedua *toolkit* ini ialah untuk meningkatkan kualitas pemrosesan dan analisis teks. Dengan *remove stopwords* akan meningkatkan kualitas dari analisa *classification text* dan mengurangi dimensi data. Data yang tergolong pada daftar *toolkit stopwords* ketika berada dalam text

nantinya akan dihapus dari text untuk mengurangi dimensi data. Sedangkan *lemmatization* mengubah beberapa kata yang ada dalam teks menjadi bentuk dasar dari kata tersebut yang berguna untuk meningkatkan konsistensi pada pengulangan kata serta mengurangi variasi bentuk dari kata tersebut. Setelah tahapan tersebut, akan dilanjutkan dengan tokenisasi teks yang menggunakan *function Tokenizer* dan *library transformers*. Tujuan dari adanya tokenisasi sendiri agar model dapat memahami dan memproses teks dengan lebih baik karena dapat mempertimbangkan konteks dari setiap token dengan cara menyesuaikannya dengan input dari model BERT.

4. *Training Model*

   Gambar 3.3 menunjukkan alur dari pengembangan model yaitu, *training model*. Pada *training model* terdapat beberapa tahapan yaitu, *pre-trained model, add classification number, set optimizer,model,scheduler,train loader, loss and device, set batch size and number of epochs for training* dan *compile and train the model using dataset*.



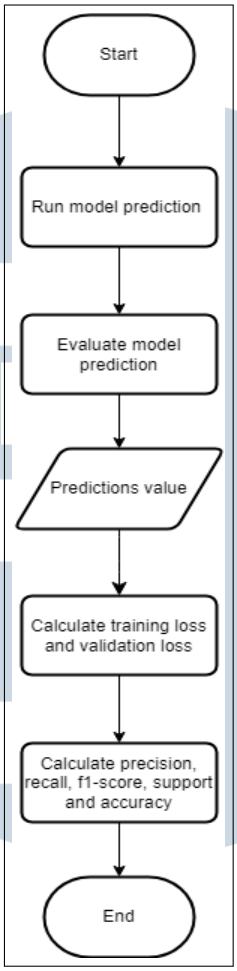
Gambar 3.3. *Flowchart* Modul Training Model

Pada tahapan *pre-trained model* algoritma BERT akan menggunakan tokenisasi pada tahapan *preprocessing* sehingga model BERT dapat melatih dataset sesuai dengan apa yang telah ditokenisasi pada tahapan *preprocessing*. Kemudian dilanjutkan dengan *Add classification number* yang berguna untuk menunjukkan seberapa banyak kategori yang akan digolongkan dalam *training model* kali ini.

Pada *set optimizer,model,scheduler,train loader, loss and device* berguna untuk menghitung akurasi dari model yang sedang dilatih, serta mengoptimalkan proses pelatihan yang sedang dilakukan dan memudahkan pengelolaan serta pemrosesan data training. Pada *set batch size and number of epochs for training* berguna untuk menetapkan *batch size* dan memanfaatkan sumber daya komputasi secara efisien, dimana *batch size* yang terlalu besar atau terlalu kecil dapat mempengaruhi kinerja pelatihan. Dengan menentapakan jumlah *epochs* akan memungkinkan *training model* mengalami pengulangan pada data yang sedang dilatih sehingga mendapatkan pola yang relevan tanpa mengalami overfitting atau underfitting. Dan dilanjutkan dengan *compile and train the model using dataset* yang menjalankan *training model* dengan tahapan-tahapan sebelumnya yang memungkinkan untuk melakukan pelatihan model dengan baik serta memastikan kinerja model benar dan memberikan kinerja yang baik pada data.

5. *Evaluate Model*

   Gambar 3.4 menunjukkan hasil setelah proses *training model* yaitu, *evaluate model*. Pada tahap *evaluate model*, model yang telah dilatih sebelumnya akan diuji dengan dataset *testing* untuk menunjukkan kualitas dari model tersebut. Selanjutnya pada tahapan *calculate training loss and validation loss* berfungsi untuk mengukur seberapa baik model mengikuti data pelatihan, sedangkan loss validasi mengukur kinerja model pada data dengan tujuan agar model dapat beradaptasi dengan data baru. Hasil dari prediksi tersebut akan terbagi menjadi *precision, recall, f1-score* dan akurasi dari model yang telah dibangun.



Gambar 3.4. *Flowchart* Modul Evaluate Model

Tujuan dari *evaluate model* ini adalah untuk menilai kinerja model yang telah dibuat agar dapat dipastikan bahwa model tersebut berfungsi dengan baik dan memberikan hasil yang diharapkan. Hasilnya dapat dilihat dalam bentuk *classification report* yang dapat membantu menentukan seberapa akurat model dalam memprediksi atau mengklasifikasikan data baru. *Classification report* sendiri meliputi nilai *precision, recall, f1-score* dan akurasi pada model.