# BAB 3
# METODOLOGI PENELITIAN

## 3.1 Metodologi Penelitian

Pada bagian ini dijabarkan tahap-tahap yang hendak dilakukan dalam menyusun dan mengerjakan penelitian dengan judul ”Implementasi Algoritma Logistic Regression Untuk Klasifikasi Teks Pada Judul Penelitian Dosen Universitas Multimedia Nusantara”. Tahapan penelitian akan dijabarkan dan dijelaskan secara detail mulai dari awal (studi literatur) hingga akhir selesai (dokumentasi). Berikut metodologi yang digunakan:

1. Studi Literatur
   Pada tahap awal ini, peneliti melakukan studi literatur untuk mengumpulkan informasi-informasi yang berkaitan pada penelitian ini. Literatur yang digunakan adalah berupa literatur yang berhubungan dengan klasifikasi teks, tingkat akurasi presisi, *recall*, *f1-score*, *logisitic regression*, dan beberapa penelitian sebelumnya yang berkaitan dengan penelitian ini.

2. Pengumpulan data
   Pada tahapan ini, peneliti melakukan pengumpulan data melalui memilih *dataset* sebagai data *training* dan melakukan wawancara terhadap salah satu staf Lembaga Penelitian dan Pengabdian Masyarakat UMN (LPPM UMN) mengenai implementasi algoritma *logistic regression* dan mengenai *Sustainable Development Goals* (SDG) dari Perserikatan Bangsa-Bangsa (PBB). Selain itu, terdapat juga *dataset* yang berisi sekumpulan data-data mengenai deskripsi singkat ataupun judul yang terkait dengan SDG dari PBB dari sebuah situs bernama Hugging Face [21].

3. Perancangan Model
   Pada tahapan ini, peneliti melakukan perancangan terhadap model klasifikasi teks yang akan dibangun pada tahapan implementasi model. Perancangan model yang akan dibuat berupa *flowchart*.

4. Implementasi Model
   Dari hasil perancangan model yang telah dibuat sebelumnya, model akan dibangun agar dengan utuh agar dapat digunakan sesuai dengan rancangan model sebelumnya.

5. Pengujian dan Evaluasi Model

   Setelah model dibangun, terdapat pengujian model yang dilakukan agar mendapatkan masukkan dan saran terhadap model yang telah dibangun. Selain itu, model juga akan dievaluasi menggunakan metrik evaluasi yang terdiri dari hasil perhitungan akurasi, presisi, *recall*, dan *f1-score* agar dapat mengetahui hasil penerapan algoritma *logistic regression* dalam klasifikasi teks terhadap judul penelitian milik dosen-dosen Universitas Multimedia Nusantara.

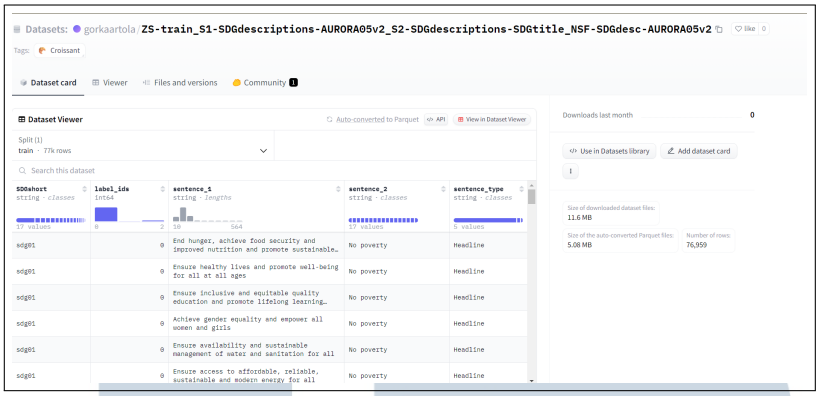
6. Penulisan Laporan

   Setelah tahapan sebelumnya dilakukan, terdapat juga tahapan penulisan dokumentasi dari hasil sistem dan penelitian ini dalam bentuk laporan.

## 3.2 Pengumpulan Data

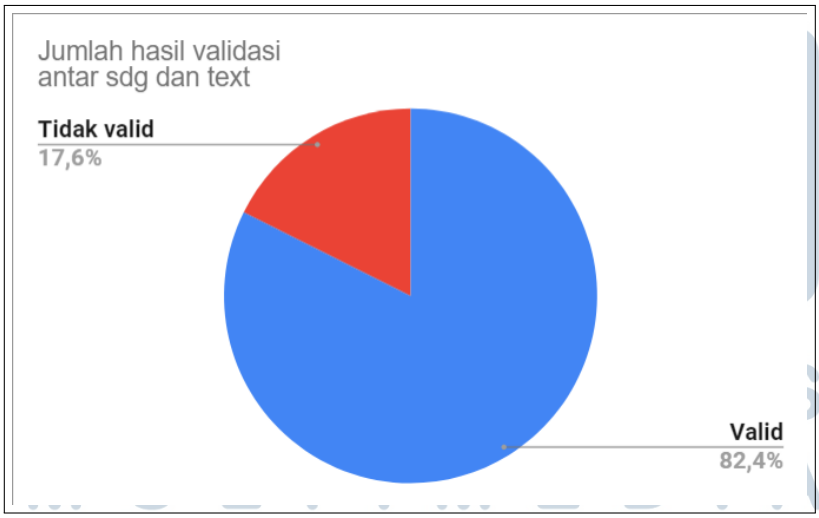
Pada tahapan ini, terdapat proses pengumpulan data yang dilakukan yaitu wawancara dan pengambilan *dataset* dari *website* bernama Hugging Face. Hugging Face merupakan sebuah perusahaan yang menyediakan *open-source software* untuk mengembangkan alat dan sumber daya untuk *natural language processing* (NLP) [22]. Hugging Face sendiri juga menyediakan platform untuk saling berbagi model, demo aplikasi *machine learning* dan juga *dataset*. Wawancara untuk penelitian ini dilakukan sebanyak dua kali dengan salah satu pihak dari Lembaga Penelitian dan Pengabdian Masyarakat Universitas Multimedia Nusantata (LPPM UMN) yaitu Bapak Ifan Bagus Haryanto, S.Si selaku staf *Research Center Officer* dari LPPM UMN. Wawancara dilakukan pada tanggal 24 Januari 2024 dan 17 Mei 2024 yang dilakukan secara *online* dengan menggunakan aplikasi Zoom. Wawancara pertama dilakukan untuk mengidentifikasi masalah yang dibutuhkan untuk penelitian ini. Sedangkan wawancara kedua dilakukan untuk mempresentasikan hasil dari pembangunan model yang telah dilakukan.

*Dataset* yang digunakan pada penelitian ini untuk melakukan pembangunan model adalah *dataset* mengenai judul artikel, jurnal, atau deskripsi singkat yang terkait dan telah dikategorikan ke salah satu dari 17 tujuan *Sustainable Development Goals* (SDG) dari Perserikatan Bangsa-Bangsa (PBB). Dataset ini berjumlahkan 76,959 data judul artikel, jurnal, atau deskripsi singkat. Tampilan *dataset* dari *website* Hugging Face dapat dilihat pada gambar 3.1 *dataset* dari Hugging Facae.




Gambar 3.1. *Dataset* dari Hugging Face

*Dataset* yang digunakan ini tidak dijelaskan mengenai jumlah penggunaan dari *dataset* tersebut ataupun mengenai hasil pengelompokkannya yang sudah valid atau belum. Maka dari itu, perlu dilakukannya pemeriksaan manual dengan cara mengambil 5 data per setiap ke 17 tujuan SDG sehingga jumlah data yang akan diperiksa ulang adalah sebanyak 85 data. Validasi atau pemeriksaan yang dilakukan adalah dengan cara mencocokkan data teks ke tujuan SDG yang telah ditentukan. Setelah itu hasil dari pemeriksaan akan divisualisasikan menjadi sebuah grafik lingkaran. Grafik hasil dari pemeriksaan *dataset* dapat dilihat pada gambar 3.2 validasi *dataset*.



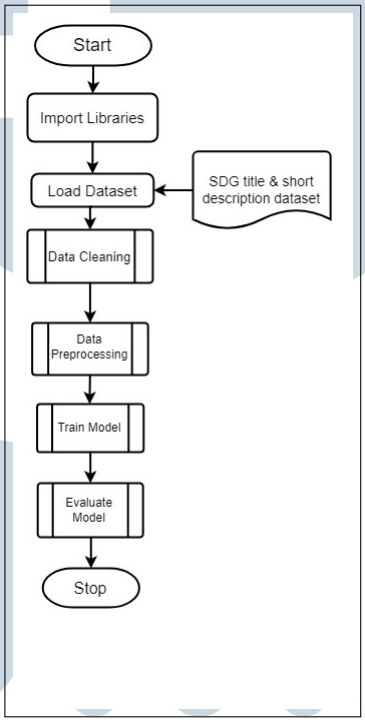

Gambar 3.2. Validasi *Dataset*

Pada grafik tersebut, ditunjukkan bahwa terdapat hasil pengelompokkan yang tidak valid sebanyak 17.6% atau sebanyak 15 data dari total 85 data yang dikumpulkan. Akan tetapi, sebagian besar dari data yang dikumpulkan

merupakan hasil pengelompkkan yang sudah valid dengan persentase sebanyak 82.5% atau sebanyak 70 data dari 85 data yang telah dikumpulkan. Dari hasil ini, dapat disimpulkan bahwa *dataset* yang digunakan pada penelitian ini memiliki tingkat kevalidan yang tinggi sehingga masih dapat digunakan untuk melakukan pembangunan model pada penelitian ini.

## 3.3 Perancangan Model

Pada tahapan ini, terdapat *flowchart* yang dibuat untuk melakukan perancangan model dalam melakukan klasifikasi teks menggunakan algoritma *logistic regression*.



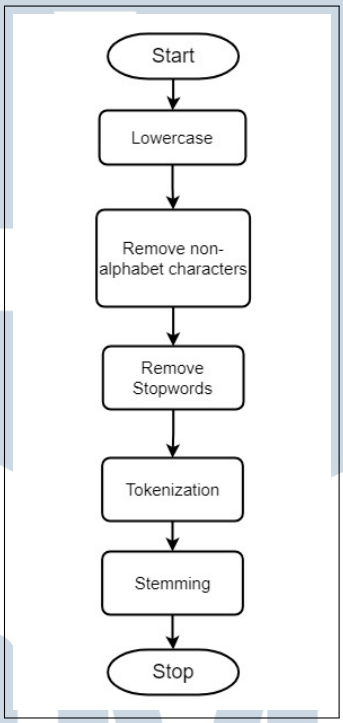

Gambar 3.3. *Flowchart* Utama

*Flowchart* utama merupakan gambaran alur secara keseluruhan pada model yang akan dirancang untuk melakukan klasifikasi teks. Dimulai dari model akan melakukan impor *libraries* yang dibutuhkan untuk pembangunan model ini. Kemudian, model akan membaca dataset yang berisikan data-data judul dan deskripsi singkat yang telah dikategorikan ke 17 label *Sustainable Development Goals* (SDG). Setelah itu, data-data dari *dataset* akan dibersihkan dalam proses *data cleaning*. Kemudian, dataset yang telah dibersihkan akan di-*preprocessing* menggunakan *library countvectorizer*. Setelah dataset di-*preprocessing*, model

akan melakukan *training*. Terakhir setelah model di*training*, model akan dievaluasi atau dihitung tingkat akurasi, presisi, *recall*, dan *f1-score*.

1. Data Cleaning

   Proses *data cleaning* merupakan proses di mana *dataset* akan dibersihkan terlebih dahulu sebelum digunakan dalam melatih dan menguji model *logistic regression*.



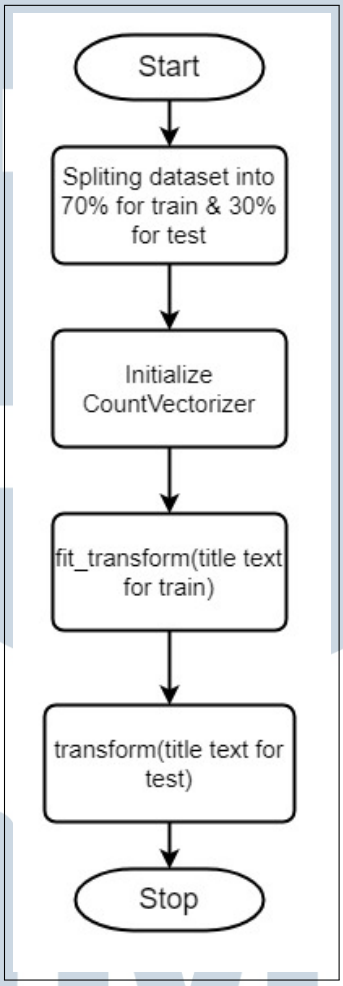

Gambar 3.4. *Flowchart* Data Cleaning

*Flowchart data cleaning* 3.4 menunjukkan alur yang diawali dengan mengubah kata dari setiap data teks menjadi huruf kecil. Setelah diubah menjadi huruf kecil, data teks yang mengandung selain alfabet dan kata-kata yang tidak memiliki makna yang penting dalam sebuah kalimat dibuang. Setelah itu, data teks akan dilakukan tokenisasi yang berarti kalimat dari data teks akan di ubah menjadi satuan kata yang dibentuk menjadi sebuah *array*. Terakhir, data teks akan melakukan *stemming* yang berarti menghapus atau mengurangi kata yang memiliki imbuhan di awal atau akhir kata menjadi kata dasarnya saja.

2. Data Preprocessing

   Setelah *dataset* telah melewati *data cleaning*, *dataset* akan melalui proses

*data preprocessing* atau proses di mana *dataset* akan disiapkan dengan bantuan *library* dari scikit-learn yaitu *count vectorizer* sebelum digunakan untuk dilatih dan diuji dengan model *logistic regression*.

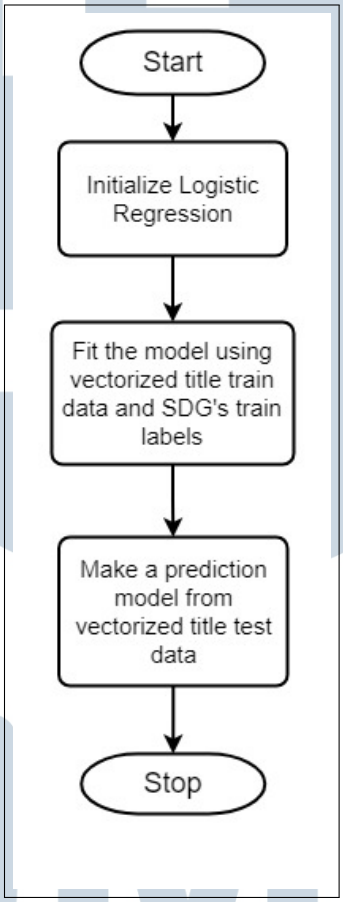



Gambar 3.5. *Flowchart* Data Preprocessing

*Flowchart data preprocessing* pada gambar 3.5 menunjukkan alur yang diawali dengan melakukan pembagian dataset menjadi 70% banding 30% dari dataset utama. 70% dari dataset utama akan digunakan untuk melatih model. Sedangkan 30% dari dataset utama akan digunakan untuk menguji model. Setelah dataset dibagi, *library Count Vectorizer* untuk melakukan data *preprocessing* diinisialisasikan. Setelah diinisialisasikan, data teks judul dari *dataset* yang telah dibagi untuk melatih model diubah menjadi data numerik dengan bantuan sintaks *fit_transform*. Kemudian data teks judul yang telah dibagi untuk menguji model akan diubah menjadi matriks hitungan frekuensi kata dengan bantuan sintaks *transform*.

3. Train Model

   Setelah *dataset* disiapkan dan telah dibersihkan, maka *dataset* akan melewati proses *train model* dengan model *logistic regression*. Proses ini berguna untuk membangun model *logistic regression* dengan menggunakan *dataset* yang telah disiapkan untuk dapat melakukan klasifikasi teks.

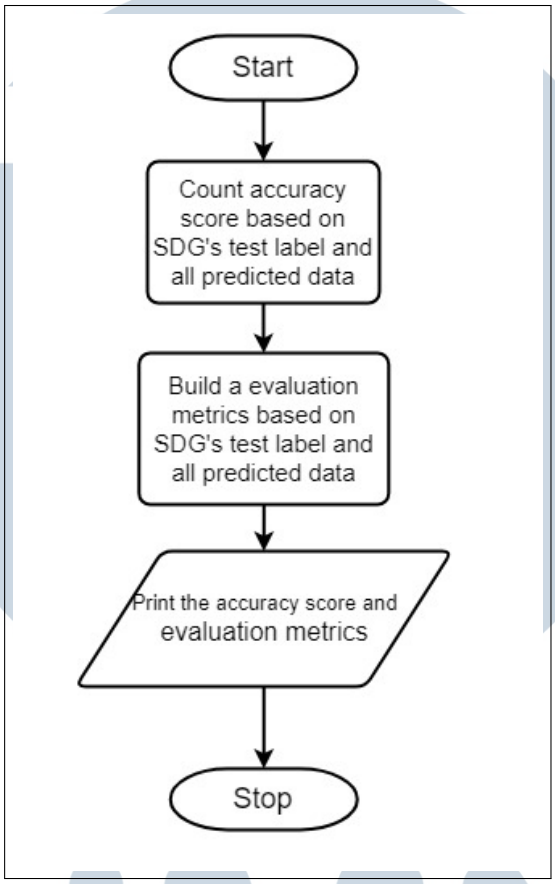



Gambar 3.6. *Flowchart* Train Model

*Flowchart train model* pada gambar 3.6 menunjukkan alur diawali dengan inisialisasi model dengan algoritma *logistic regression* dengan menggunakan parameter regularisasi L1 sebagai penalti dan *Stochastic Average Gradient Augmented* (SAGA) sebagai *solver*. Kemudian, model disesuaikan dengan *dataset* yang telah dibagi untuk *training*. Setelah itu, membuat prediksi pada label SDG untuk sampel pada data teks judul dari *dataset* yang telah dibagi menjadi data uji yang telah di vektorisasi atau yang telah diubah menjadi data numerik pada alur *data preprocessing* sebelumnya.

4. Evaluate Model

Terakhir, model yang telah dilatih pada proses sebelumnya akan dievaluasi untuk mengukur tingkat akurasi dan performa dari model itu sendiri.




Gambar 3.7. *Flowchart* Evaluate Model

*Flowchart evaluate model* pada gambar 3.7 menunjukkan alur yang diawali dari menghitung tingkat akurasi berdasarkan label-label SDG dan semua data teks judul yang telah diprediksi sebelumnya dari *dataset* yang telah dibagi untuk menguji model dengan sintaks ”accuracy_score”. Tingkat akurasi berguna untuk memahami proporsi dari prediksi yang benar yang dihasilkan dari model [5]. Setelah itu, menghitung metrik evaluasi yang juga berdasarkan label-label SDG dan semua data teks judul yang telah diprediksi sebelumnya dari *dataset* yang telah dibagi untuk menguji model dengan sintaks ”classification_report”. Metrik evaluasi ini akan menentukan tingkat presisi, *recall*, dan *f1-score* per masing-masing 17 label SDG. Tingkat presisi adalah pengukuran untuk menentukan seberapa baik model dalam menentukan atau mengidentifikasi data positif secara akurat dari total prediksi yang dinyatakan positif. *Recall* adalah pengukuran yang mengevaluasi

kemampuan model dalam menemukan data positif dari semua data positif yang tersedia [6]. *F1-score* adalah rata-rata dari presisi dan *recall* [5]. Setelah itu hasil skor akurasi dan metrik evaluasi akan ditampilkan sebagai *output*.