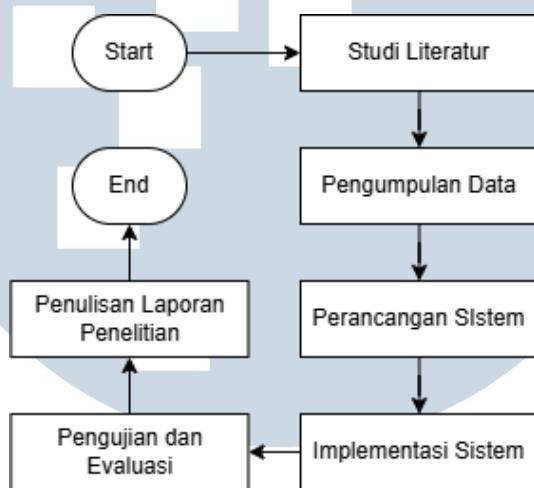


## BAB 3

### METODOLOGI PENELITIAN

#### 3.1 Alur Penelitian

Dalam penelitian ini, dilakukan implementasi algoritma BERT untuk mendeteksi sarkasme pada headline berita. Pada gambar 3.1 tersedia flowchart yang menjelaskan tahap-tahap penelitian



Gambar 3.1. Flowchart Alur Penelitian

#### 3.2 Studi Literatur

Tujuan dilakukannya studi literatur adalah agar dapat memahami topik secara lebih dalam, literatur yang digunakan sebagai bahan studi antara lain adalah artikel atau jurnal yang berkaitan dengan topik penelitian. Jurnal atau artikel yang digunakan sebagai referensi merupakan jurnal atau artikel nasional dan internasional, yang diterbitkan dari rentang tahun 2020 sampai dengan 2025 yang mencakup pembahasan permasalahan sarkasme dan deteksi sarkasme, NLP, penggunaan sarkasme pada headline berita, dan BERT.

#### 3.3 Pengumpulan Data

Dataset yang digunakan dalam penelitian ini adalah dataset yang ada pada kaggle, yaitu data headline berita yang sudah di label mengandung sarkasme atau

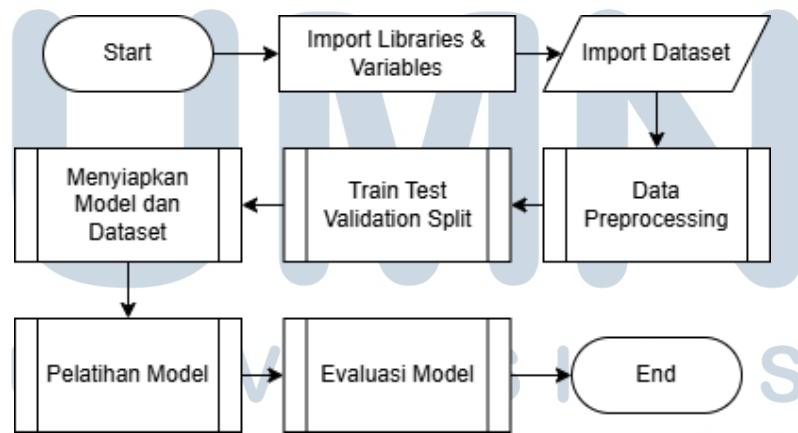
tidak. Headline berita yang bersifat sarkastik diambil dari *website* TheOnion yang adalah *website* khusus berita satir yang mengandung sarkasme, dan headline berita normal dikumpulkan dari *website* HuffPost [10].

### 3.4 Perancangan Sistem

Tahap perancangan sistem dilakukan untuk memperoleh pemahaman yang lebih mendalam terhadap penelitian yang akan dilaksanakan melalui penelusuran dan pengkajian artikel serta jurnal yang relevan dengan topik penelitian. Sumber rujukan yang digunakan terdiri dari jurnal dan artikel nasional maupun internasional yang diterbitkan dalam rentang tahun 2020 hingga 2025, dengan cakupan pembahasan mengenai sarkasme dan deteksi sarkasme, penggunaan sarkasme pada headline berita, metode BERT, serta proses text preprocessing.

#### 3.4.1 Alur Utama Penelitian

Gambar 3.2 menunjukkan alur utama penelitian deteksi sarkasme yang dirancang. Proses dimulai dengan melakukan *import library* dan pendefinisian variabel, dilanjutkan dengan pemuatan dataset, tahap prapemrosesan data, pembagian data, serta persiapan model BERT untuk proses pelatihan (training) dan evaluasi model.

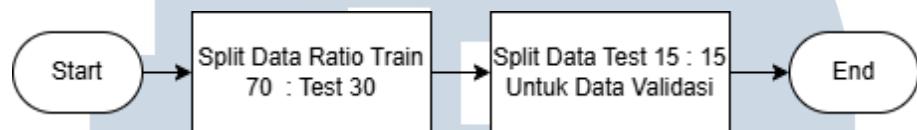


Gambar 3.2. Alur Utama Penelitian

#### 3.4.2 Pembagian Data

Dataset dibagi menjadi tiga bagian, yaitu data *training*, *validation*, dan *testing*. Proses pembagian data dilakukan menggunakan *library* scikit-learn.

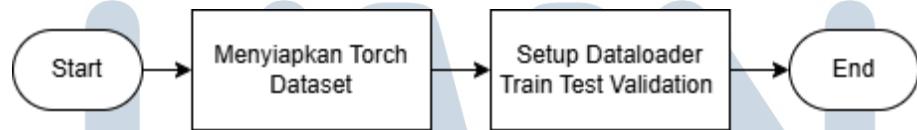
Tahapan pembagian dimulai dengan memisahkan data menjadi data *training* dan data sementara. Selanjutnya, data sementara tersebut dibagi kembali menjadi data *validation* dan data *testing*. Pembagian data *training* dan *testing* menggunakan rasio 70:30, dengan tujuan menyeimbangkan kebutuhan data yang cukup untuk proses pelatihan agar model dapat mempelajari parameter dengan baik, serta data yang memadai untuk pengujian agar estimasi performa model tidak terlalu bervariasi [21]. Gambar 3.3 menampilkan flowchart proses pembagian data tersebut.



Gambar 3.3. Pembagian Data

### 3.4.3 Alur Persiapan Model dan Dataset

Gambar 3.4 menggambarkan tahapan persiapan model dan dataset. Dataset perlu diproses terlebih dahulu ke dalam format PyTorch agar data teks dapat dikembalikan serta hasil prediksi dapat divalidasi. Tahapan ini meliputi pembuatan *torch* dataset untuk memasukkan dataset ke dalam PyTorch, serta pembuatan dataloader untuk data *training*, *validation*, dan *testing*.

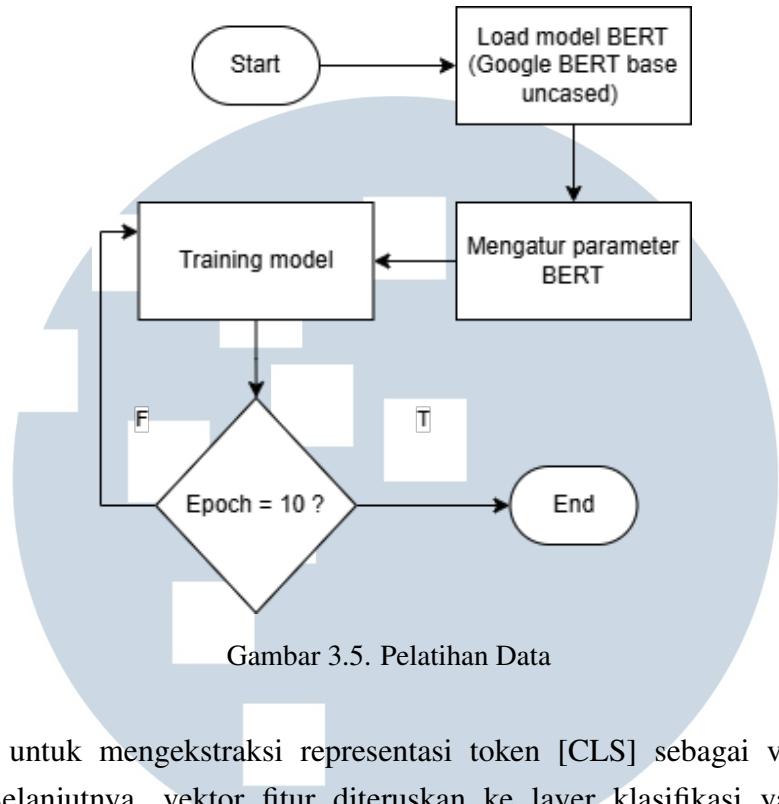


Gambar 3.4. Persiapan Model dan Dataset

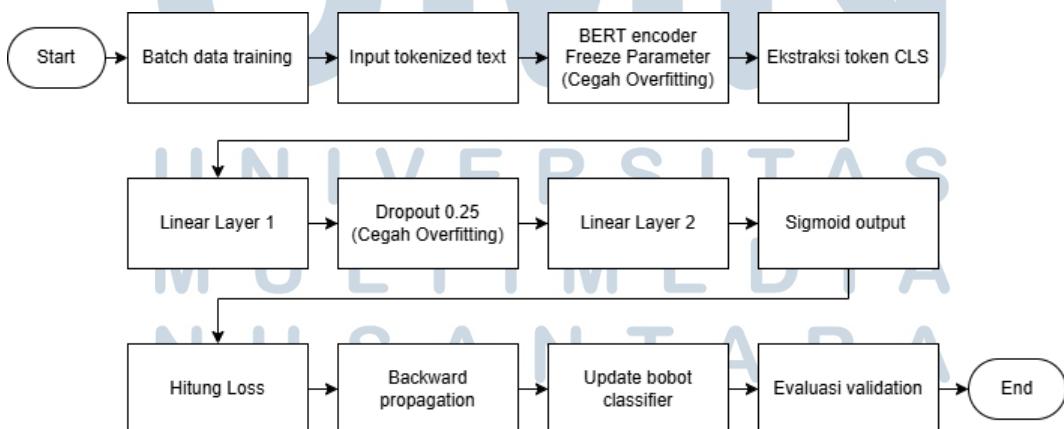
### 3.4.4 Alur Pelatihan Model

Pada tahap ini, data yang telah melalui proses tokenisasi akan digunakan dalam proses *finetuning*. Proses diawali dengan memuat model BERT, kemudian melakukan pengaturan parameter pada model tersebut, dan selanjutnya menjalankan proses *finetuning*. Gambar 3.5 menunjukkan alur persiapan model BERT dan pelatihan data.

Pada setiap *epoch*, proses pelatihan dimulai dengan pengambilan data *training* dalam bentuk *batch* yang telah ditokenisasi menjadi inputids dan attentionmask. Data tersebut diproses melalui *encoder* BERT yang parameternya



dibekukan untuk mengekstraksi representasi token [CLS] sebagai vektor fitur utama. Selanjutnya, vektor fitur diteruskan ke layer klasifikasi yang terdiri dari dua *fully connected layer* dengan mekanisme *dropout* untuk mengurangi resiko *overfitting*, kemudian menghasilkan probabilitas keluaran melalui fungsi aktivasi sigmoid. Nilai probabilitas tersebut digunakan untuk menghitung *loss* menggunakan *Binary Cross Entropy*, dilanjutkan dengan proses *backpropagation* dan pembaruan bobot pada layer klasifikasi. Setelah seluruh batch training diproses, model dievaluasi menggunakan data validasi tanpa pembaruan bobot guna memantau kemampuan generalisasi model pada akhir *epoch*.



Gambar 3.6. Alur tiap epoch

### 3.4.5 Evaluasi Model

Tahap evaluasi dilakukan untuk mengukur kinerja model BERT yang telah disesuaikan menggunakan data *testing*. Data yang telah diberi token kemudian diproses melalui model untuk memperoleh label sentimen hasil prediksi. Selanjutnya, label prediksi dibandingkan dengan label sebenarnya, dan dilakukan perhitungan akurasi pada tiga proses pembelajaran model yaitu akurasi data *training*, akurasi data *validation*, dan akurasi data *testing*.

## 3.5 Pengujian dan Evaluasi

Setelah program deteksi sarkasme berbasis BERT selesai dibuat, dilakukan proses debugging untuk mengidentifikasi permasalahan yang muncul, seperti bug atau error pada program. Setelah program dapat berjalan dengan baik secara keseluruhan, dilakukan beberapa skenario pengujian dan evaluasi untuk memperoleh hasil performa terbaik.

