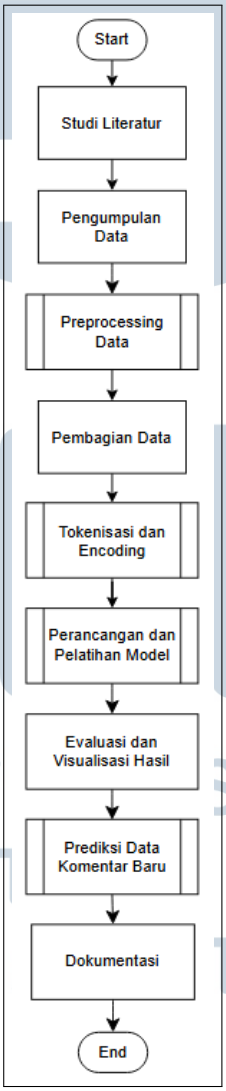
BAB 3
METODOLOGI PENELITIAN

3.1 Tahapan Penelitian

Bab ini membahas metodologi yang digunakan dalam penelitian ini, mulai dari tahap awal hingga tahap akhir. Metodologi penelitian disusun untuk memastikan bahwa proses analisis sentimen terhadap strategi pemasaran kolaborasi NCT DREAM dan TosTos dapat dilakukan secara sistematis dan akurat. Gambar 3.1 adalah *flowchart* yang berisi gambaran keseluruhan dari tahap penelitian yang dilakukan.



Gambar 3.1. Alur penelitian

1. Studi Literatur

   Tahap ini dilakukan dengan mengumpulkan informasi dari penelitian-penelitian pendahulu yang membahas mengenai *Korean Wave (Hallyu)* dalam strategi pemasaran, analisis sentimen, tahap *preprocessing* data, BERT, dan *confusion matrix*. Studi ini bertujuan untuk membangun dasar teori yang kuat dalam merancang metode analisis, termasuk pada pemilihan data, teknik pemrosesan data, serta model *Machine Learning* yang diterapkan.

2. Pengumpulan Data

   Penelitian ini menggunakan data komentar dari platform media sosial Instagram pada akun resmi tostosid yang menjadi pihak penyelenggara kampanye kolaborasi antara TosTos dan NCT DREAM. Komentar yang dikumpulkan berasal dari beberapa postingan yang berkaitan langsung dengan kegiatan promosi, undian, maupun *fanmeeting* dalam rangka kolaborasi tersebut. Pengambilan data dilakukan menggunakan metode *web scraping* dengan bantuan *tools* Apify dan PhantomBuster, sehingga hanya perlu mencari postingan yang membahas tentang kolaborasi TosTos dengan NCT DREAM, lalu salin tautan postingan tersebut dan masukkan ke Apify atau PhantomBuster. Data yang dikumpulkan mencakup rentang waktu dari 1 Maret 2024 hingga 28 Maret 2025 dengan total 6.830 data. Semua data akan dikumpulkan, digabungkan, dan disimpan dalam *file* dengan format .csv untuk diproses lagi ke tahap selanjutnya [11].

3. *Preprocessing* Data

   Dari data yang sudah berhasil dikumpulkan, sebanyak total 1.800 data akan diberi label sentimen secara manual dengan kategori positif, negatif, dan netral. Untuk memastikan kualitas dan validitas dari data ini, proses pelabelan sentimen dilakukan secara manual oleh dua *annotator*. Tingkat kesepakatan antar-*annotator* kemudian divalidasi dengan koefisien Cohen's Kappa. Perhitungan tersebut menghasilkan nilai Cohen's Kappa sebesar 0.9560, yang menunjukkan tingkat kesepakatan pada kategori 'Hampir Sempurna' [25]. Validasi ini menegaskan bahwa *dataset* yang digunakan untuk proses pelatihan, validasi, dan pengujian model memiliki keandalan yang sangat tinggi.

   Setelah proses pelabelan manual selesai, seluruh 1.800 data komentar akan memasuki tahapan pra-pemrosesan data (*data preprocessing*) seperti menghapus baris-baris komentar yang kosong, menghapus URL yang tidak

relevan, mengubah semua teks menjadi huruf kecil (*case folding*), dan mengganti emoji dengan deskripsi kata yang sesuai (sedih, cinta, suka). Dalam penelitian ini, bahasa utama yang digunakan adalah Bahasa Indonesia. Terhadap komentar yang menggunakan bahasa tidak baku atau *slang*, seperti singkatan (bgt, yg), model diatasi dengan tahapan normalisasi teks menggunakan kamus singkatan yang telah disiapkan. Jika terdapat kata-kata dari bahasa lain seperti Bahasa Korea (contohnya: 'Gomawo'), kata-kata tersebut tidak dihapus melainkan dianggap sebagai bagian dari konteks sentimen, yang dapat memengaruhi klasifikasi model, karena sering kali mengandung makna positif atau negatif.

Selain itu, pada tahapan ini proses *label encoding* juga dilakukan untuk mengubah label sentimen kategorikal menjadi format numerik. Kategori '*negative*', '*neutral*', dan '*positive*' masing-masing di-*encode* menjadi nilai 0, 1, dan 2, untuk memudahkan proses pelatihan model.

4. Pembagian Data

   Setelah data selesai dipra-proses, *dataset* yang sudah bersih kemudian akan dibagi menjadi tiga bagian: data *training*, *validation*, dan *testing*. Guna memastikan efektivitas pelatihan dan objektivitas evaluasi, penelitian ini membandingkan performa model menggunakan dua skenario rasio pembagian data. Pada kedua skenario, diterapkan teknik *stratified splitting* agar distribusi setiap kelas sentimen (negatif, netral, positif) seimbang. Dengan demikian, model dapat belajar secara optimal tanpa bias kelas tertentu, dan hasil evaluasi menjadi lebih representatif. Dua skenario rasio pembagian data tersebut adalah sebagai berikut:

   (a) Rasio 70:15:15: Sebanyak 70% data digunakan untuk melatih model, 15% untuk validasi selama proses pelatihan, dan 15% sisanya untuk menguji performa akhir model.

   (b) Rasio 80:10:10: Sebanyak 80% data dialokasikan untuk pelatihan, sementara 10% untuk validasi dan 10% untuk pengujian.

5. Tokenisasi dan *Encoding*

   Setelah pembagian *dataset*, setiap set data (*training*, *validation*, dan *testing*) akan melewati proses tokenisasi dan *encoding*. Tahapan ini merupakan langkah penting untuk mengubah teks menjadi representasi numerik yang dapat dipahami oleh model. Berbeda dengan tokenisasi sederhana, proses

ini menggunakan *tokenizer* khusus dari model IndoBERT yang telah dilatih sebelumnya. Dalam proses ini, *tokenizer* secara spesifik bertugas untuk:

(a) Memecah teks menjadi token-token yang relevan untuk model (*subword tokenization*).

(b) Menambahkan token-token khusus, seperti [CLS] di awal kalimat dan [SEP] di akhir.

(c) Mengubah token menjadi ID numerik (*encoding*) yang unik.

(d) Menyesuaikan panjang setiap kalimat agar seragam melalui *truncation* (memotong) atau *padding* (menambahkan token kosong) hingga mencapai panjang `max\_length` yang ditentukan.

6. Perancangan dan Pelatihan Model

   Pada tahap ini, dilakukan perancangan model dan proses pelatihan (*training*) menggunakan model IndoBERT. Tujuan dari pelatihan ini adalah agar model mampu mempelajari pola dan konteks dalam komentar pengguna media sosial terkait strategi pemasaran kolaborasi antara sebuah *brand* lokal dengan idola Korea Selatan, NCT DREAM, serta mampu mengklasifikasikan sentimen komentar tersebut secara otomatis. Model yang digunakan adalah `indobenchmark/indobert-base-p1`, yaitu varian IndoBERT-base dengan 12 lapisan *transformer*. Proses pelatihan dilakukan dengan konfigurasi *batch size* sebesar 16 dan jumlah *epoch* sebanyak 4. Selain itu, diterapkan metode `EarlyStoppingCallback` untuk menghentikan pelatihan secara otomatis apabila tidak terdapat peningkatan performa model pada data validasi selama dua *epoch* berturut-turut.

7. Evaluasi dan Visualisasi Hasil

   Setelah proses perancangan dan pelatihan model selesai, tahap selanjutnya adalah melakukan evaluasi untuk mengukur performa model. Evaluasi ini dilakukan menggunakan metrik seperti *accuracy*, *precision*, *recall*, dan *F1-score*. Metrik *F1-score* menjadi fokus utama karena mencerminkan keseimbangan antara *precision* dan *recall*, yang sangat penting untuk dievaluasi saat distribusi kelas sentimen tidak seimbang.

   Hasil evaluasi disajikan dalam bentuk visualisasi untuk memudahkan analisis. Grafik *Training Los*s dan *Validation Loss* digunakan untuk memantau proses pembelajaran model dan mendeteksi kemungkinan terjadinya *overfitting*.

Selain itu, *confusion matrix* juga ditampilkan untuk menunjukkan jumlah prediksi yang benar dan salah pada masing-masing kelas sentimen, sehingga memudahkan analisis terhadap kekuatan dan kelemahan model.

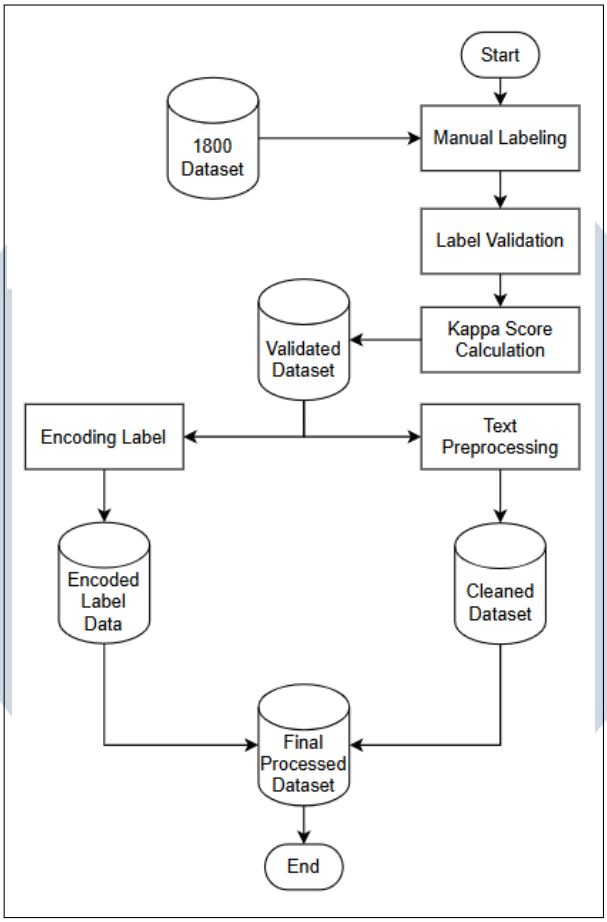
Model terbaik selama pelatihan, yang dipilih berdasarkan performa pada data validasi, akan digunakan untuk evaluasi akhir pada data pengujian. Evaluasi akhir ini berfungsi untuk memberikan gambaran performa model yang paling objektif terhadap data yang tidak pernah dilihat sebelumnya.

8. Prediksi Data Komentar Baru
   Setelah melalui proses pelatihan dan evaluasi, model diuji menggunakan data baru yang belum pernah dilibatkan sebelumnya. Tujuan dari tahap ini adalah untuk menilai kemampuan generalisasi model dalam mengklasifikasikan sentimen terhadap komentar publik yang sebelumnya tidak dikenal oleh model. Pengujian ini penting untuk membuktikan apakah model mampu bekerja secara efektif di luar data latih, serta memiliki potensi untuk diaplikasikan dalam analisis sentimen nyata, khususnya dalam mengevaluasi persepsi publik terhadap strategi pemasaran kolaboratif.

9. Dokumentasi
   Tahap ini merupakan tahapan terakhir untuk menuliskan semua teori, proses perancangan, implementasi, hingga hasil dari penelitian yang telah dilakukan ke dalam bentuk laporan.

3.2 Preprocessing Data

Gambar 3.2 merupakan *flowchart* langkah-langkah pra-pemrosesan data sebelum digunakan dalam pelatihan model analisis sentimen berbasis *deep learning*.



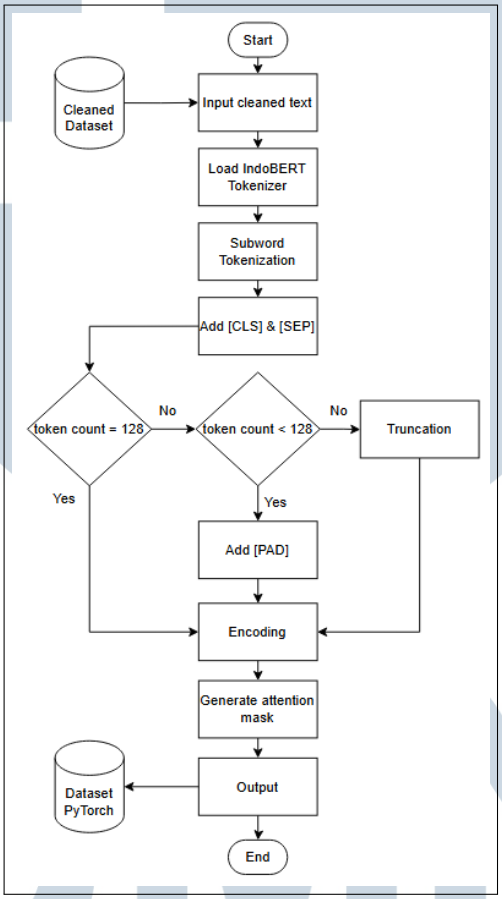
Gambar 3.2. *Flowchart preprocessing* data

Langkah awal yang dilakukan adalah memuat data mentah berupa 1.800 komentar Instagram yang telah dipisahkan dari total data komentar yang diperoleh dari proses *web scraping*. Data mentah ini kemudian melalui proses pelabelan manual oleh dua *annotator* yang mengklasifikasikan setiap komentar ke dalam tiga kategori sentimen, yaitu negatif, netral, dan positif. Untuk memastikan kualitas dan keandalan data berlabel tersebut, dilakukan validasi antar-*annotator* menggunakan koefisien Cohen's Kappa.

Setelah proses pelabelan dan validasi selesai, *dataset* memasuki tahapan pra-pemrosesan data (*text preprocessing*) untuk membersihkan dan menstandarisasi teks. Proses ini mencakup pembersihan teks (menghapus URL, spasi berlebih, dan data kosong), normalisasi (mengubah huruf kapital menjadi huruf kecil, mengganti emoji dengan deskripsi kata, dan mengoreksi kata-kata tidak baku), serta konversi label sentimen dari bentuk teks (*negative*, *neutral*, *positive*) menjadi representasi numerik (0, 1, 2).

3.3 Tokenisasi dan Encoding

Sebelum data teks dapat diproses oleh model IndoBERT, diperlukan tahapan tokenisasi dan *encoding* untuk mengubah teks menjadi format numerik yang dapat dipahami oleh model. Alur proses tokenisasi dan *encoding* ini digambarkan dalam Gambar 3.3.



Gambar 3.3. *Flowchart* tokenisasi dan *encoding*

Tahap ini diawali dengan memuat data komentar yang telah melalui proses pembersihan dan normalisasi pada tahap *preprocessing*. Selanjutnya, *tokenizer* khusus IndoBERT dimuat menggunakan pustaka *transformers* dari Hugging Face. *Tokenizer* ini telah dilatih menggunakan korpus bahasa Indonesia, sehingga mampu menangani karakteristik bahasa Indonesia dengan baik. Langkah pertama adalah tokenisasi, yaitu proses memecah kalimat menjadi unit-unit yang lebih kecil disebut *subword tokens*. Hal ini dilakukan karena dalam model BERT, kata-kata tidak selalu di token-kan secara utuh, terutama jika kata tersebut tidak ada dalam *vocabulary* model. Misalnya, kata tidak dikenal seperti ”tostos” bisa dipecah menjadi ”to” dan

”##stos”. Pemecahan ini memungkinkan model tetap dapat memahami konteks meskipun tidak mengenal kata secara utuh.

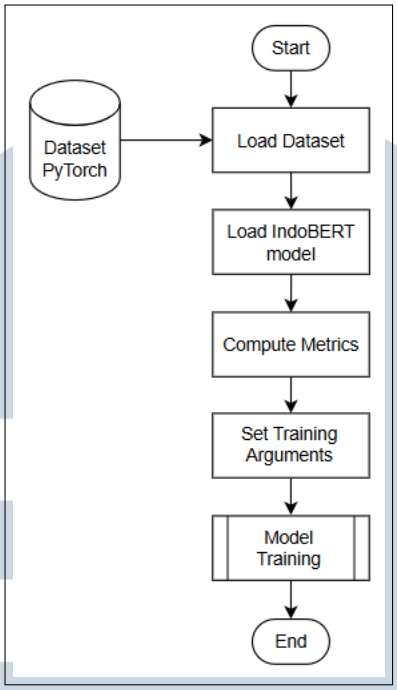
Setelah tokenisasi, ditambahkan token khusus [CLS] di awal kalimat yang akan digunakan oleh model sebagai representasi keseluruhan teks untuk keperluan klasifikasi dan token [SEP] di akhir kalimat untuk menandai batas akhir dari *input* atau segmen teks. Langkah selanjutnya adalah proses *padding* dan *truncation*. Jika jumlah token kurang dari batas panjang maksimum yang telah ditetapkan (128), maka ditambahkan token [PAD] (*padding*) agar seluruh *input* memiliki panjang yang seragam. Jika jumlah token melebihi panjang maksimum, maka token akan dipotong (*truncated*) dari akhir agar sesuai dengan panjang yang ditentukan.

Setelah urutan token sudah seragam panjangnya, dilakukan proses *encoding*, yaitu mengubah seluruh token menjadi ID numerik berdasarkan *vocabulary* dari *tokenizer* IndoBERT. Proses ini berbeda dengan *label encoding* yang hanya mengonversi label sentimen (seperti ”positif”, ”negatif”, ”netral”) menjadi angka. Pada tahap ini, seluruh teks komentar diubah ke bentuk numerik agar dapat diproses oleh model. Selain token ID, juga dihasilkan *attention mask*, yaitu *array biner* yang menunjukkan posisi token penting (1) dan *padding* (0). Ini penting agar model hanya memproses bagian kalimat yang bermakna dan mengabaikan *padding*.

Seluruh keluaran dari proses ini dikemas dalam bentuk *tensors PyTorch* yang dapat digunakan sebagai *input* ke dalam model IndoBERT pada tahap pelatihan (*training*) maupun prediksi. Proses tokenisasi dan *encoding* ini krusial karena memastikan bahwa seluruh teks telah memiliki format dan struktur yang sesuai standar model, menjaga konsistensi *input*, serta mempertahankan informasi penting seperti emoji, *hashtag*, dan tanda baca. Hal ini memungkinkan model menangkap nuansa makna dalam komentar berbahasa Indonesia secara lebih efektif, yang pada akhirnya akan berpengaruh terhadap performa klasifikasi sentimen.

3.4 Perancangan dan Pelatihan Model

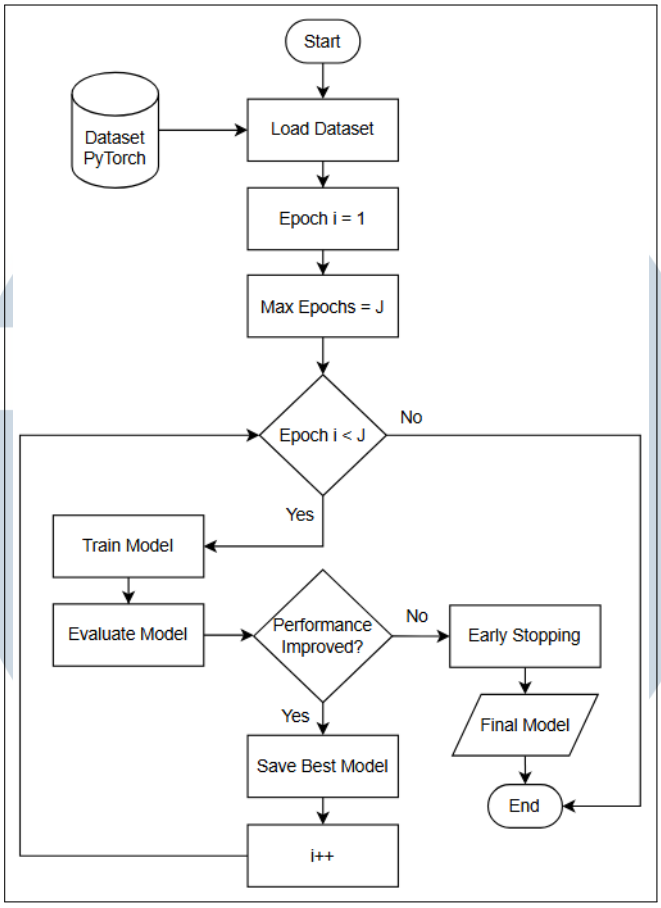
Gambar 3.4 menunjukkan alur proses perancangan dan pelatihan model klasifikasi sentimen menggunakan IndoBERT.



Gambar 3.4. *Flowchart* perancangan dan pelatihan model

Tahapan perancangan dan pelatihan model dimulai dengan memuat *dataset* yang telah diproses dan dikonversi ke dalam format *tensor PyTorch*. *Dataset* ini kemudian dibagi menjadi tiga bagian, yaitu data *training*, *validation*, dan *testing*. Selanjutnya, model IndoBERT dimuat, dalam hal ini `indobenchmark/indobert-base-p1`, yaitu varian *base* dari IndoBERT yang telah dilatih khusus untuk Bahasa Indonesia. Model ini diadaptasi untuk tugas klasifikasi sentimen dengan tiga label *output*.

Setelah model dimuat, metrik evaluasi didefinisikan untuk mengukur performa. Metrik yang digunakan meliputi *accuracy*, *precision*, *recall*, dan *F1-score* dengan pendekatan rata-rata tertimbang (*weighted average*) agar dapat memperhitungkan distribusi kelas yang tidak seimbang. Seluruh proses pelatihan diatur melalui `TrainingArguments`, yang mencakup berbagai konfigurasi parameter seperti jumlah *epoch*, ukuran *batch*, *learning rate*, dan strategi penyimpanan model terbaik. Berikut adalah gambar dari *flowchart* saat model IndoBERT melakukan *training* data yang dapat dilihat pada Gambar 3.5.



Gambar 3.5. *Flowchart* model *training*

Tahapan ini dimulai dengan memasukkan *dataset PyTorch* yang telah disiapkan. Model kemudian dilatih menggunakan objek *Trainer* dari pustaka Hugging Face dalam sebuah siklus berulang, di mana pada setiap *epoch*, model akan melakukan pelatihan dan evaluasi secara berkala. Pada setiap *epoch*, performa model diukur menggunakan data validasi. Hasilnya kemudian digunakan untuk memastikan apakah terjadi peningkatan performa. Jika performa model meningkat, versi terbaik dari model akan disimpan (*Save Best Model*) dan proses pelatihan berlanjut ke *epoch* berikutnya. Dalam alur ini, dilakukan juga serangkaian eksperimen dengan memvariasikan skenario, termasuk pelatihan dengan *hyperparameter tuning* (menggunakan Optuna) dan pelatihan dengan parameter yang telah ditentukan.
Untuk mengidentifikasi konfigurasi model yang optimal, dilakukan serangkaian eksperimen dengan skenario sebagai berikut:

1. Skenario A: Model dilatih dengan rasio pembagian data 70:15:15, tanpa pembobotan kelas (*class weight*), dan dengan menerapkan *hyperparameter tuning* melalui Optuna.

2. Skenario B: Model dilatih dengan rasio pembagian data 70:15:15 dan tanpa menerapkan pembobotan kelas atau *hyperparameter tuning*. Skenario ini berfungsi sebagai *baseline*.
3. Skenario C: Model dilatih dengan rasio pembagian data 70:15:15, dengan pembobotan kelas, dan dengan menerapkan *hyperparameter tuning*.
4. Skenario D: Model dilatih dengan rasio pembagian data 70:15:15 dan dengan pembobotan kelas, tetapi tanpa *hyperparameter tuning*.
5. Skenario E: Model dilatih dengan rasio pembagian data 70:15:15, tanpa pembobotan kelas dan *hyperparameter tuning*, serta menggunakan *dataset* yang tidak melalui proses validasi Cohen's Kappa untuk perbandingan.
6. Skenario F: Model dilatih dengan rasio pembagian data 80:10:10 dan tanpa menerapkan pembobotan kelas atau *hyperparameter tuning*.

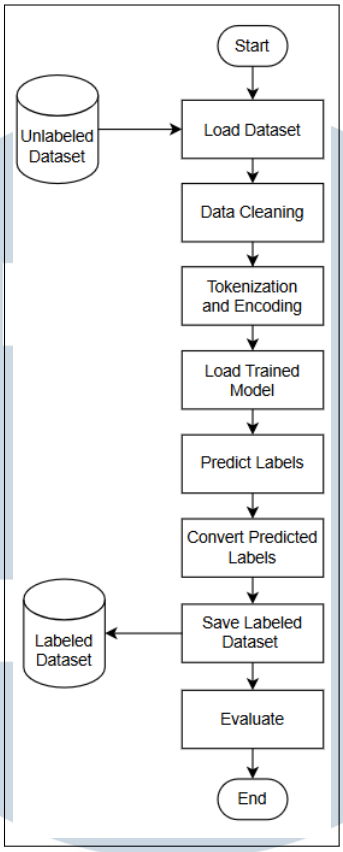
Proses pelatihan akan dihentikan jika salah satu dari dua kondisi berikut terpenuhi:

1. performa model tidak lagi meningkat pada data validasi, yang memicu mekanisme *early stopping*.
2. jumlah *epoch* maksimal yang telah ditentukan tercapai.

Output dari setiap skenario eksperimen (A hingga F) adalah model yang sudah terlatih. Perbandingan dan analisis performa dari masing-masing model ini akan dibahas secara mendalam pada Bab 4, yang menjadi dasar untuk pemilihan model terbaik.

3.5 Prediksi Data Komentar Baru

Tahap ini bertujuan untuk menguji kemampuan generalisasi model dengan menggunakan data komentar yang belum pernah dilihat sebelumnya dan belum memiliki label sentimen (*unlabeled data*). Alur proses prediksi dapat dilihat dalam Gambar 3.6.



Gambar 3.6. *Flowchart* prediksi data komentar baru

Setelah proses pembersihan, dilakukan tokenisasi dan *encoding* menggunakan *tokenizer* IndoBERT yang sama dengan yang digunakan saat pelatihan. Model IndoBERT yang telah dilatih sebelumnya kemudian dimuat kembali dari direktori penyimpanan. Model ini digunakan untuk memprediksi sentimen dari setiap komentar dalam *dataset* tak berlabel. Hasil prediksi awal berupa nilai numerik (0, 1, 2) dikonversi ke label sentimen '*negative*', '*neutral*', dan '*positive*' menggunakan pemetaan label.

Guna menilai kualitas hasil pelabelan otomatis oleh model, dilakukan validasi manual pada sampel acak dari 100 komentar yang telah diprediksi. Komentar-komentar ini diberi label secara manual oleh peneliti dan *annotator* kedua. Untuk memastikan keandalan label manual tersebut, dihitung koefisien Cohen's Kappa, yang menghasilkan nilai 0.842. Nilai ini menunjukkan tingkat kesepakatan yang 'substansial' antara kedua penilai.

Selanjutnya, hasil pelabelan manual yang telah di validasi ini digunakan sebagai 'kunci jawaban' untuk mengukur performa model dalam memprediksi 100 sampel tersebut. Proses evaluasi dilakukan menggunakan berbagai metrik:

1. *Classification Report*, yang menampilkan *accuracy*, *precision*, *recall*, dan *F1-score* untuk setiap kelas sentimen.
2. *Confusion Matrix*, untuk menunjukkan distribusi prediksi benar dan salah pada masing-masing kelas.
3. *Error Rate per Class*, untuk melihat tingkat kesalahan model dalam memprediksi komentar negatif, netral, dan positif

Visualisasi distribusi label hasil prediksi juga ditampilkan menggunakan diagram batang (*countplot*) untuk menunjukkan proporsi kelas sentimen hasil prediksi otomatis. Dari visualisasi dan evaluasi ini, dapat dinilai sejauh mana model dapat mengenali dan mengklasifikasikan sentimen dari komentar publik yang sebelumnya belum diberi label. Selain itu, proses ini juga digunakan untuk menganalisis hasil sentimen publik terhadap kolaborasi produk TosTos dengan NCT DREAM.