# BAB III

# METODOLOGI PENELITIAN

## III.1 Jenis dan Sumber Data

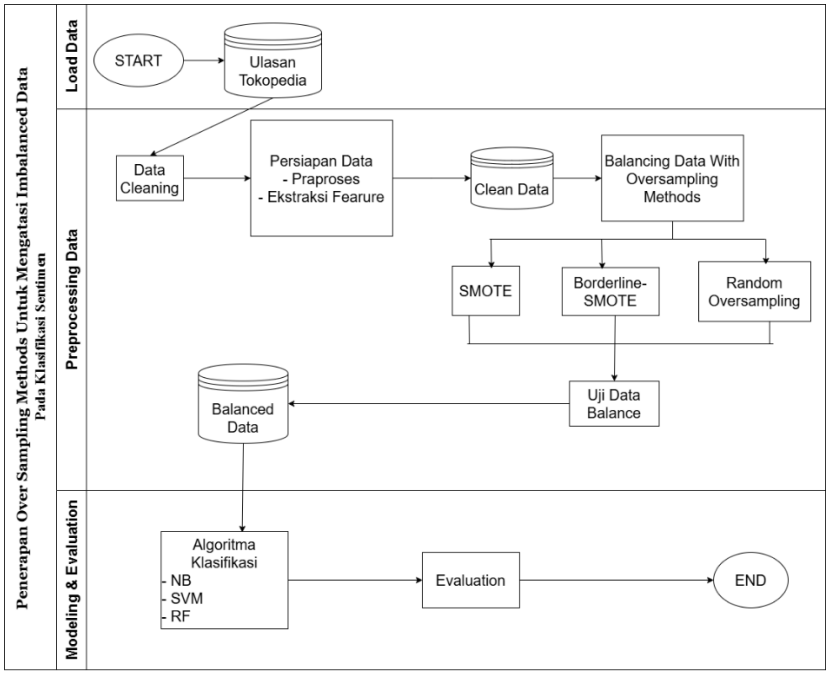
Jenis data yang digunakan dalam penelitian ini adalah data primer, yaitu data yang diperoleh secara langsung dari sumber aslinya (Faisal dkk., 2020). Data ulasan Tokopedia dikumpulkan dari situs Google Play Store dengan menggunakan teknik *web scraping* yang diimplementasikan melalui pustaka Python bernama *google-play-scraper*. Total data yang berhasil dikumpulkan sebanyak 10.000 ulasan berbahasa Indonesia. Jumlah 10.000 ulasan berada dalam skala yang cukup besar untuk machine learning, namun masih dalam batas wajar untuk diolah dengan sumber daya komputasi yang terbatas, tanpa memerlukan infrastruktur komputasi tinggi (Gonjari dkk., 2024). Menjaga efisiensi dalam proses preprocessing, training, dan evaluasi model. Informasi lebih lanjut mengenai tipe data yang dikumpulkan dapat dilihat pada Tabel 3.1.

Tabel 3.1 Tipe Data

| **Kolom** | **Tipe Data** |
|---|---|
| *Username* | *Object* |
| *Score* | *Integer* |
| *At* | *Datetime* |
| *Content* | *Object* |

## III.2 Tahapan Penelitian

Proses penelitian dimulai dengan pengumpulan dataset, yang kemudian dilanjutkan dengan tahap preprocessing untuk membersihkan data dari informasi yang tidak relevan serta menerapkan teknik ekstraksi fitur. Selanjutnya, diterapkan metode *oversampling* guna mengatasi permasalahan ketidakseimbangan kelas pada data (*Imbalanced* data). Setelah data seimbang, dilakukan proses pembagian data (*data splitting*) dengan proporsi 80% untuk data latih (*training*) dan 20% untuk data uji (*testing*). Tahap berikutnya adalah penerapan algoritma *machine learning* untuk mengklasifikasikan ulasan pengguna ke dalam kategori positif dan negatif. Evaluasi performa model dilakukan dengan menggunakan sejumlah metrik, yaitu *Accuracy*, *Recall*, *Precision*, *F1-Score*, dan AUC-ROC. Adapun alur lengkap dari proses penelitian yang telah dijelaskan di atas dapat dilihat pada Gambar 3.1 mengenai alur penelitian.




Gambar 3. 1 Alur Penelitian

### III.2.1 Pengambilan Data

Tahapan pertama mengambil data ulasan Tokopedia berasal dari situs *Google Play Store*, diambil dengan metode *web scraping* yang dilakukan dengan menggunakan *library* dari python yaitu `google-play-scraper`. Tokopedia memiliki basis pengguna yang sangat besar di Indonesia, sehingga data ulasan dari platform ini sangat relevan untuk menganalisis sentimen yang terkait dengan pasar lokal, kebutuhan, dan kebiasaan konsumen Indonesia. *Scraping* data dilakukan pada tanggal 10 Januari 2025. Data yang diambil untuk penelitian ini berada pada rentang waktu 13 September 2018 sampai 10 Januari 2025. Jumlah data yang terkumpul pada rentang waktu tersebut sebanyak 10.000 data ulasan berbahasa Indonesia.

### III.2.2 Preprocessing Data

Tahapan ini merupakan proses pengolahan data yang bertujuan untuk menyusun teks menjadi format yang lebih terstruktur dan siap diolah lebih lanjut. Proses ini dilakukan dengan memanfaatkan NLTK (*Natural Language Toolkit*), yaitu sebuah kumpulan pustaka dan perangkat lunak yang dirancang khusus untuk mendukung pemrosesan bahasa alami (*Natural Language Processing*). Beberapa langkah yang dilakukan dalam tahap ini meliputi *Case Folding*, *Tokenizing*, dan *spelling correction*. Perlu dicatat bahwa tidak terdapat aturan baku mengenai urutan maupun jenis teknik preprocessing yang harus diterapkan, karena hal tersebut sangat bergantung pada karakteristik data yang digunakan serta keluaran (output) yang diharapkan dari proses analisis.

### III.2.3 *Feature extraction*

Tahap *filtering* dengan *stopword removal* ini penting untuk membersihkan data teks dan meningkatkan kualitas analisis sentimen. Membuang *stopword*, kita dapat fokus pada kata-kata yang lebih informatif dan relevan untuk menentukan sentimen dalam teks. *Stopword removal* diimplementasikan dengan baik menggunakan library nltk dan juga mempertimbangkan penambahan *custom stopword* yang spesifik untuk dataset, setelah itu dilakukan *Stemming* teknik pemrosesan bahasa alami yang menurunkan infleksi kata ke bentuk akarnya, sehingga membantu dalam pra pemrosesan teks, kata, dan dokumen untuk normalisasi teks, sebagai bagian dari tahap feature engineering, penelitian ini tidak hanya melakukan konversi teks menjadi representasi numerik (TF-IDF), tetapi juga menerapkan visualisasi dan analisis linguistik berupa Wordcloud dan N-gram.

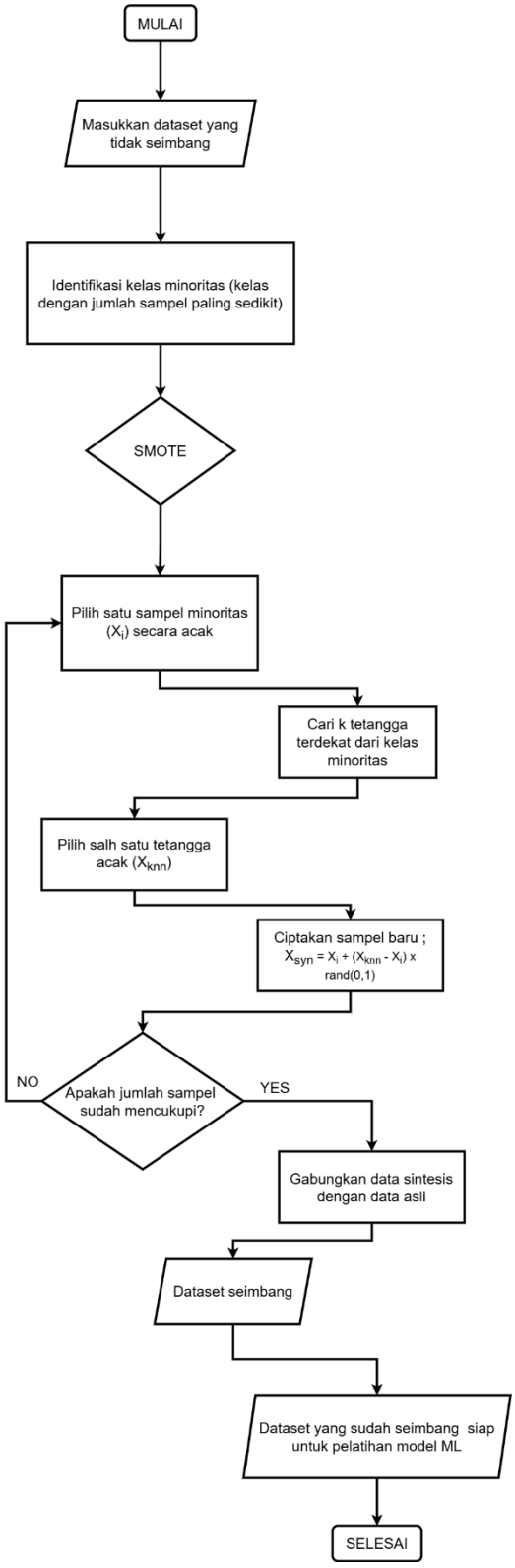
Tujuan dari tahapan ini adalah untuk memvalidasi hasil preprocessing serta memahami konteks dan struktur linguistik dalam data sentimen.

### III.2.4 Balancing Data with *Oversampling methods*

Proses ekstraksi fitur dilakukan menggunakan metode TF-IDF (*Term Frequency–Inverse Document Frequency*) dari pustaka `scikit-learn`. Teknik ini digunakan untuk mengubah teks menjadi representasi numerik berdasarkan frekuensi relatif suatu kata dalam dokumen dan keseluruhan korpus. Parameter tahap ini, seluruh teks dalam atribut `text_string` diubah menjadi vektor numerik menggunakan konfigurasi default dari `TfidfVectorizer()`, tanpa batasan jumlah fitur `(max_features)` maupun pengaturan khusus seperti n-gram. Model mempertimbangkan hanya unigram dengan semua fitur unik yang muncul dalam data. Setelah fitur diekstraksi, data dibagi menjadi dua bagian menggunakan fungsi `traint_split()` dari pustaka `sklearn.model_selection` dengan rasio 80:20, yaitu 80% untuk pelatihan dan 20% untuk pengujian. Parameter `Random_state = 42` digunakan untuk memastikan eksperimen bersifat reproducible. Teknik pembagian yang digunakan adalah *hold-out*, Data pelatihan asli kemudian disalin ke dalam variabel baru `(x_original dan y_original)` sebelum dilakukan proses *oversampling*, selanjutnya akan diterapkan beberapa teknik *oversampling* seperti SMOTE, *Borderline-SMOTE*, dan *Random Oversampling* untuk menyeimbangkan hasil data.

### III.2.4.1 *Synthetic Minority Over-sampling Technique* (SMOTE)

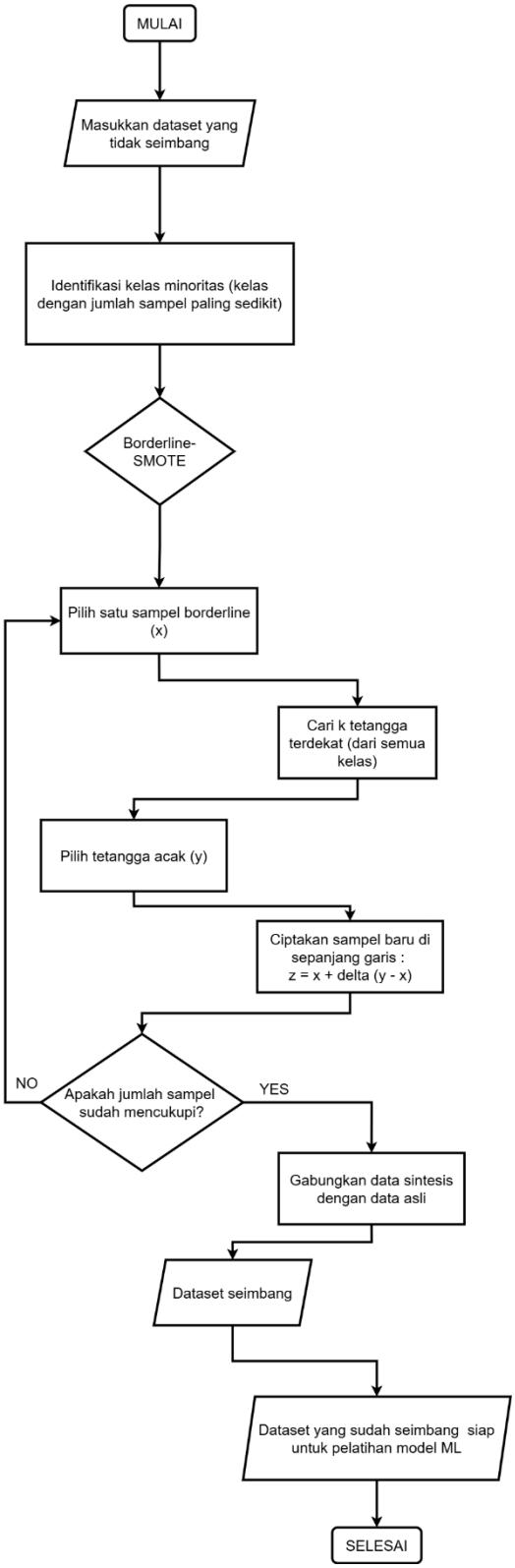
Skenario pertama dalam penelitian ini adalah penerapan metode SMOTE (*Synthetic Minority Over-sampling Technique*) yang bertujuan untuk meningkatkan kinerja algoritma klasifikasi melalui modifikasi dataset yang tidak seimbang. SMOTE bekerja dengan menghasilkan data sintetik baru dari kelas minoritas, sehingga proporsi antara kelas mayoritas dan minoritas menjadi lebih seimbang. Proses pembangkitan data sintetik dilakukan dengan memanfaatkan algoritma *K-Nearest Neighbor*s untuk menentukan titik-titik tetangga terdekat dari kelas minoritas. Parameter `Random_state` digunakan untuk mengontrol proses pengacakan dalam pembangkitan sampel, di mana penggunaan nilai integer yang sama akan menghasilkan hasil *oversampling* yang konsisten. Penting untuk menjamin reproducibility atau keterulangan hasil eksperimen, tahapan pemrosesan dapat dilihat pada gambar 3.2.




Gambar 3.2 Flowchart Oversampling SMOTE

### III.2.4.2 *Borderline-SMOTE*

*Borderline-SMOTE* merupakan varian lanjutan dari algoritma SMOTE yang dikembangkan untuk mengatasi permasalahan tumpang tindih antar kelas, khususnya ketika data kelas mayoritas berada dekat dengan data kelas minoritas dalam proses sintesis. Metode ini secara selektif melakukan *oversampling* terhadap instance kelas minoritas yang berada di sekitar area perbatasan kelas (decision boundary), karena area tersebut cenderung lebih rentan terhadap kesalahan klasifikasi. Metode SMOTE, parameter *Random*_state digunakan untuk memastikan konsistensi hasil dengan mengontrol proses pengacakan, sehingga eksperimen dapat direproduksi secara konsisten, tahapan pemrosesan dapat dilihat pada gambar 3.3.




Gambar 3.3 Flowchart Oversampling Borderline-SMOTE

### III.2.4.3 *Random Oversampling*

Teknik *oversampling* terakhir yang digunakan dalam eksperimen ini adalah *Random Oversampling*. Metode ini bekerja dengan meningkatkan jumlah sampel pada kelas minoritas secara acak, yaitu dengan menggandakan data dari kelas tersebut hingga mencapai proporsi yang setara dengan kelas mayoritas. Tujuan utama dari pendekatan ini adalah untuk memperkuat representasi kelas minoritas, sehingga algoritma *machine learning* mampu mempelajari pola dari kedua kelas secara seimbang. Performa model diharapkan dalam mengidentifikasi kelas minoritas dapat meningkat secara signifikan. Teknik SMOTE dan *Borderline-SMOTE*, penggunaan parameter *Random_state* dalam *Random Oversampling* berfungsi untuk memastikan hasil eksperimen dapat direproduksi secara konsisten, tahapan pemrosesan dapat dilihat pada gambar 3.4.


Gambar 3. 4 Flowchart Random Oversampling

### III.2.5 Implementasi & Pengujian

Penelitian ini memanfaatkan algoritma *machine learning* untuk melakukan klasifikasi sentimen pengguna Tokopedia. Tiga algoritma yang diterapkan dalam studi ini meliputi *Random Forest*, *Support Vector Machine* (SVM), dan *Naïve Bayes*. Algoritma *Naïve Bayes* dinilai memiliki kapabilitas yang tinggi dalam menganalisis karakteristik klasifikasi dari interaksi data yang kompleks, serta memiliki ketahanan yang baik terhadap data baru maupun data yang mengandung nilai hilang. *Naïve Bayes* sering dipilih dalam klasifikasi teks karena kesederhanaannya serta efektivitas kinerjanya dalam berbagai studi terdahulu. *Support Vector Machine* (SVM) dikenal unggul dalam menghasilkan tingkat presisi yang tinggi pada tugas klasifikasi teks.

### III.2.6 Uji Balanced & Evaluasi Kinerja

Penelitian ini akan dilakukan analisis uji balanced dari sebelum menerapkan *oversampling methods* dan setelah menerapkan *oversampling methods* lalu akan dilihat dari hasil uji mana yang lebih baik, dapat dilihat juga hasil evaluasi kinerja *machine learning* sebelum penerapan *oversampling methods* dan setelah *oversampling methods* menghasilkan evaluasi kinerja yang baik, dengan menggunakan teknik uji menggunakan *Confusion matrix* merupakan matriks untuk mengukur performa model klasifikasi melalui perhitungan *Accuracy*, *Precision*, *Recall,* dan *F1-Score*. *Accuracy* mengacu pada rasio antara jumlah sampel yang berhasil diklasifikasikan dengan benar dan total seluruh sampel yang diuji. *Precision* menunjukkan seberapa besar proporsi prediksi positif yang benar dari keseluruhan prediksi yang diklasifikasikan sebagai positif. *Recall* mengukur

seberapa besar bagian dari sampel positif yang berhasil terdeteksi secara benar oleh model. *F1-Score* merupakan metrik gabungan yang merepresentasikan keseimbangan antara *Precision* dan *Recall*, dan dihitung sebagai rata-rata harmonis dari keduanya. Pengujian AUC-ROC sangat sesuai untuk menangani permasalahan data yang tidak seimbang, karena AUC mampu memberikan evaluasi menyeluruh terhadap kinerja model dan membantu menentukan model mana yang memiliki performa prediksi lebih unggul secara keseluruhan.