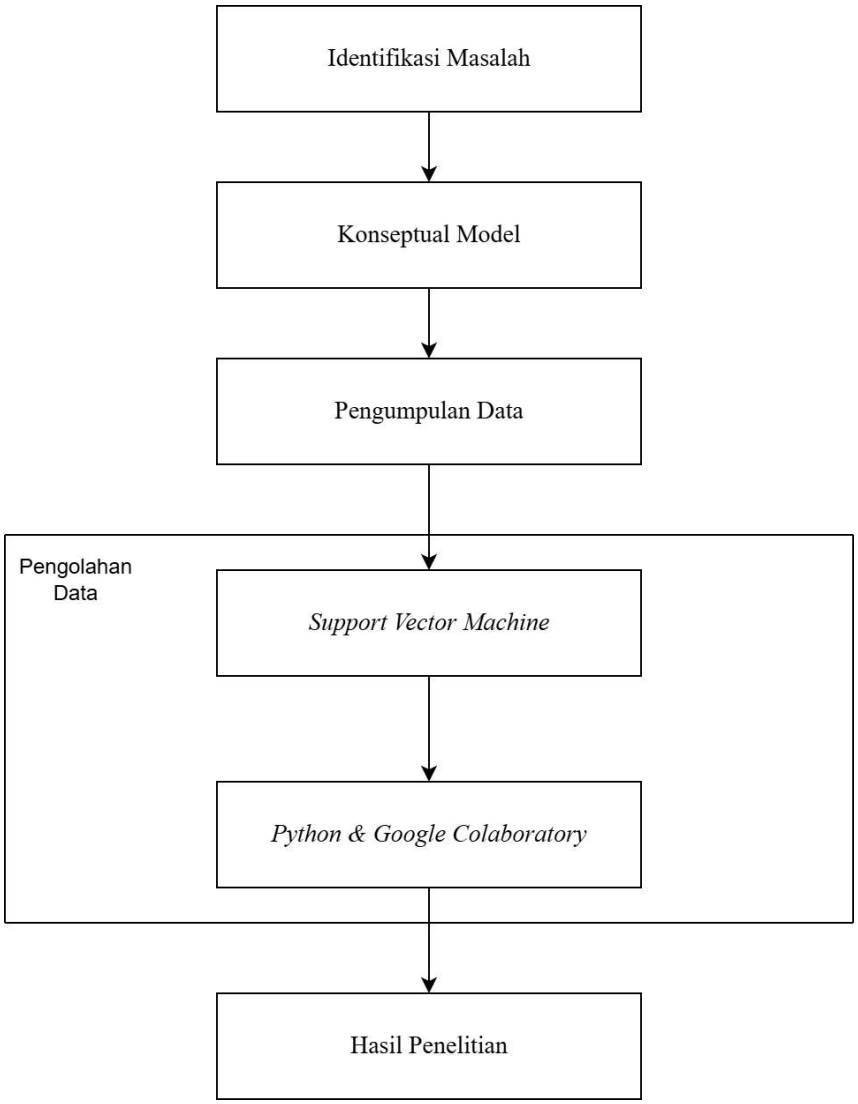
BAB III

METODE PENELITIAN

3.1 Desain Penelitian

Desain penelitian merupakan suatu rancangan sistematis yang dibuat untuk mengarahkan jalannya proses penelitian. Tujuan utama dari penyusunan desain ini adalah untuk membantu peneliti dalam merancang dan menjalankan penelitian secara terstruktur. Setiap tahapan dalam penelitian dirancang agar dapat menghasilkan temuan yang dapat dipahami secara konseptual oleh semua pihak yang berkepentingan, serta mendukung pengembangan model penelitian yang sesuai dan akurat. Berdasarkan uraian mengenai desain penelitian tersebut, penjelasan masing-masing tahapan dalam proses penelitian dapat dipaparkan sebagai berikut:



Gambar 3. 1 Desain Penelitian
Sumber: (Data Penelitian., 2025)

Keterangan:

1. Identifikasi Masalah

   Pada tahap ini, peneliti mengidentifikasi permasalahan utama yang dihadapi oleh pengguna layanan ojek online Grab, khususnya berdasarkan komentar yang diberikan di Google *Play Store*. Fokus dari identifikasi ini adalah untuk memahami sentimen publik, baik positif maupun negatif, terhadap berbagai

aspek layanan Grab seperti keandalan aplikasi, kualitas pengemudi, tarif, dan sistem pembayaran. Proses ini menjadi dasar untuk menyusun solusi berbasis analisis data yang akurat.

2. Konseptual Model

   Tahapan ini merupakan proses perancangan model konseptual penelitian yang digunakan sebagai acuan dalam pelaksanaan penelitian. Peneliti menentukan sumber data, yaitu komentar pengguna aplikasi Grab di *Google Play,* dan juga menetapkan tahapan pra-pemrosesan data seperti pembersihan teks dan tokenisasi. Selain itu, ditentukan juga teknik ekstraksi fitur (TF-IDF), pemilihan algoritma klasifikasi yaitu *Support Vector Machine* (SVM), serta metode evaluasi performa model seperti akurasi, *precision*, dan *recall*.

3. Pengumpulan Data

   Data dikumpulkan melalui proses *web scraping* komentar dari pengguna aplikasi Grab di Google *Play Store* menggunakan *tools* atau *script* berbasis *Python*. Data yang dikumpulkan berupa teks ulasan, rating bintang, dan waktu komentar. Selain itu, peneliti juga melakukan studi literatur untuk memahami teori-teori yang berkaitan dengan analisis sentimen, metode klasifikasi SVM, dan karakteristik layanan Grab sebagai objek penelitian.

4. *Support Vector Machine*

   Metode utama yang digunakan dalam penelitian ini adalah algoritma *Support Vector Machine* (SVM), yang dipilih karena kemampuannya dalam menangani data teks yang bersifat tidak linear dan kompleks. SVM akan digunakan untuk mengklasifikasikan komentar pengguna menjadi dua kelas

sentimen, yaitu positif dan negatif. Keunggulan SVM dalam memproses data berdimensi tinggi menjadikannya metode yang efektif untuk analisis sentimen berbasis teks ulasan pengguna.

5. *Python* & *Google Colaboratory*

   Proses penggelohan data dilakukan dengan bantuan bahasa pemrograman Python dan dijalankan melalui platform *Google Colaboratory*. *Tools* ini dipilih karena mendukung komputasi berbasis *cloud*, serta kompatibel dengan berbagai pustaka seperti *Scikit-learn*, *Pandas*, *NLTK*, dan lainnya. Seluruh tahapan *preprocessing*, ekstraksi fitur, pelatihan model SVM, dan evaluasi performa dilakukan melalui notebook interaktif yang dapat direproduksi dan dibagikan.

6. Hasil Penelitian

   Hasil dari penelitian ini berupa klasifikasi sentimen dari ulasan pengguna terhadap layanan Grab, yang terbagi menjadi dua kategori utama, yaitu sentimen positif dan sentimen negatif. Analisis ini tidak hanya memberikan gambaran umum tentang persepsi pelanggan, tetapi juga mengungkap aspek-aspek spesifik yang paling sering dikomentari, seperti kualitas layanan pengemudi, stabilitas aplikasi, sistem pembayaran, dan kebijakan tarif. Selain itu, sebagai bagian dari implementasi praktis, peneliti juga membangun aplikasi sederhana berbasis Streamlit yang berfungsi untuk menampilkan hasil analisis secara interaktif. Aplikasi ini memungkinkan pengguna untuk mengunggah data ulasan, melihat hasil klasifikasi sentimen secara real-time, dan mengeksplorasi kata kunci dominan dari setiap kategori sentimen.

Dengan demikian, temuan dari penelitian ini tidak hanya bersifat akademis, tetapi juga aplikatif dan dapat dimanfaatkan langsung oleh pihak Grab atau pengembang lain untuk meningkatkan kualitas layanan berdasarkan suara pelanggan secara *data-driven*.

3.2 Metode Pengumpulan Data

Metode pengumpulan data atau *data collection* merupakan proses sistematis dalam memperoleh informasi yang relevan dengan topik penelitian menggunakan pendekatan ilmiah dan terstruktur. Proses ini dilakukan untuk memastikan bahwa data yang diperoleh valid, dapat dianalisis, dan mampu menjawab rumusan masalah yang telah ditentukan. Pada penelitian ini, teknik pengumpulan data yang digunakan adalah sebagai berikut:

1. Studi Pustaka

   Teknik studi pustaka digunakan oleh peneliti untuk mengumpulkan referensi ilmiah dan wawasan teoretis yang berkaitan dengan topik penelitian, khususnya mengenai analisis sentimen, data mining, algoritma *Support Vector Machine (SVM)*, serta teknik *scraping* data ulasan dari *platform* digital seperti Google *Play Store*. Sumber referensi yang dikaji meliputi jurnal ilmiah nasional dan internasional, buku teks, skripsi, tesis, maupun artikel yang relevan dari situs terpercaya. Studi pustaka ini bertujuan untuk memperkuat dasar teori serta memperoleh pemahaman tentang penelitian sejenis yang telah dilakukan sebelumnya.

2. *Web Scraping* Komentar Aplikasi Grab di Google *Play Store*

Pada penelitian ini, pengumpulan data utama dilakukan melalui teknik *web scraping*, yaitu pengambilan data ulasan pengguna terhadap aplikasi Grab di Google *Play Store* secara otomatis menggunakan skrip Python. Komentar yang dikumpulkan mencakup teks ulasan, rating, dan tanggal unggahan. Teknik ini memungkinkan peneliti untuk mendapatkan data *real-time* dan autentik yang bersumber langsung dari pengguna aktif aplikasi. Data yang diperoleh selanjutnya digunakan sebagai dataset untuk analisis sentimen dalam penelitian ini. *Scraping* dilakukan dengan tetap memperhatikan etika dan kebijakan penggunaan data dari *platform* yang bersangkutan.

3.3 Operasional Variabel

Operasional variabel merupakan bagian penting dalam sebuah penelitian, karena dengan menyusun variabel secara sistematis, peneliti dapat menjelaskan dan mengukur hubungan antar komponen data secara objektif. Dalam penelitian ini, operasional variabel disusun berdasarkan hasil *web scraping* terhadap ulasan pengguna aplikasi Grab di Google *Play Store*. Variabel-variabel tersebut dijelaskan sebagai berikut:

1. Nomor

   Merupakan penomoran urut dari setiap data ulasan yang dikumpulkan. Dalam penelitian ini, sebanyak 1000 ulasan pengguna dijadikan sebagai dataset.

2. *Content*

   Berisi isi atau teks dari ulasan yang diberikan oleh pengguna terhadap layanan aplikasi Grab. Konten ulasan ini dapat berupa pujian, saran, keluhan, maupun

ekspresi kepuasan atau ketidakpuasan pengguna terhadap berbagai fitur dan layanan Grab, seperti performa aplikasi, pengemudi, sistem pemesanan, dan pembayaran.

3. Skor (Bintang)

   Merupakan *rating* atau skor yang diberikan oleh pengguna dalam bentuk bintang (1–5) di Google *Play Store*. Skor ini mencerminkan penilaian pengguna secara kuantitatif terhadap layanan Grab dan menjadi salah satu fitur yang digunakan untuk mendukung klasifikasi sentimen.

4. Label

   Setelah data ulasan dikumpulkan dan melalui tahap pra-pemrosesan, masing-masing ulasan kemudian diberikan *label* sentimen, yaitu positif atau negatif. Proses pelabelan ini mengacu pada korelasi antara isi ulasan dengan skor bintang yang diberikan oleh pengguna. Dalam penelitian ini, penentuan label dilakukan secara otomatis dengan aturan: skor ≤ 3 dikategorikan sebagai negatif, sedangkan skor ≥ 4 dikategorikan sebagai positif. Pemberian label ini menjadi acuan dalam pelatihan dan pengujian model klasifikasi sentimen yang akan dibangun.

Tabel 3. 1 Sampel Data yang Diperoleh dari Hasil *Scrapping*

No	Content	Skor	Label
1	banyak driver tolol gak sih, di aplikasinya aturan lalu lintas harus diperketat, yang ngelanggar aturan lalu lintas kalo bisa akunnya langsung di suspend ya, udah sering banget gw hampir kecelakaan gara gara driver yang tiba tiba pelan di jalur	1	Negatif

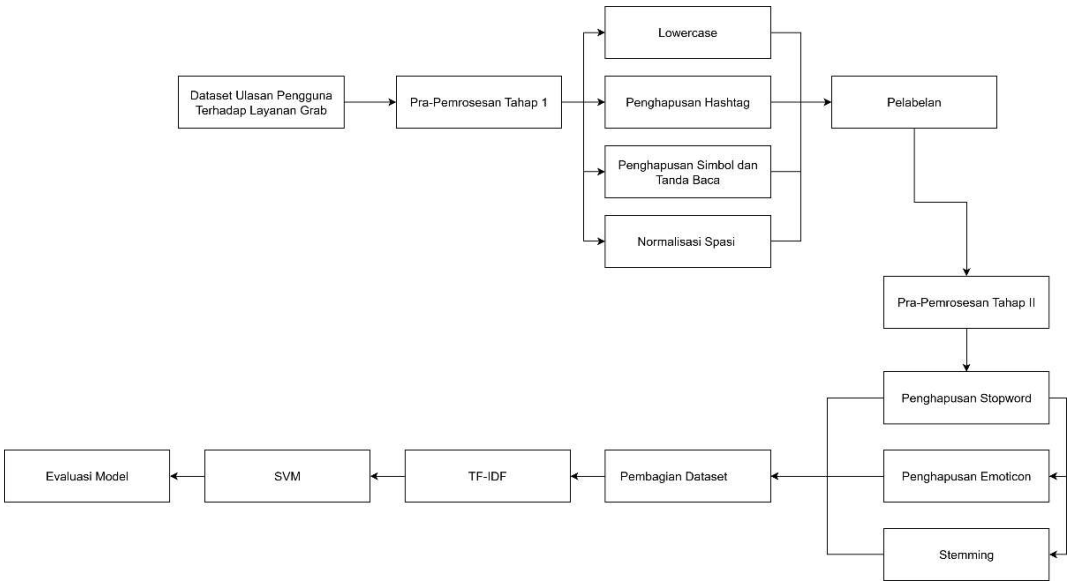
	cepat/kanan untung aja gw masih bisa belok kiri dengan cepat, dan paling banyak itu dari *driver* mau itu Maxim,gojek,grab banyak banget yang main naik-naik trotoar aja, dan lawan arah		
2	setelah pesen grabfood lalu ubah alamat ternyata buat promo nya hilang ya? padahal titik alamat nya hanya beda beberapa langkah. harusnya diberi peringatan terlebih dahulu. kaget tadi tbtb gabisa pake promo lagi mana harga lumayan.	2	Negatif
3	aplikasi ini sekarang tidak bersahabat lagi dengan tunanetra. aplikasi ini tidak kompatibel dengan pembaca layar *TalkBack* sangat berat digunakan dan tidak akses terutama daftar makanan tidak dibacakan Jadi kami tidak bisa melakukan pemesanan harapannya pengembang bisa memperbaiki dan membuat aplikasi ini aksesibel bagi tunanetra yang menggunakan pembaca layar TalkBack	1	Negatif
4	hapus program double order sangat merugikan saya sebagai pemakai apk grab, orderan datangnya telat, hampir 40menit! pesanan baru sampai. Tidak ada pengembalian dana, saya order icecream	1	Negatif

	apa bisa dimakan kalo telat sampe selama itu?		
5	grabfood dari dulu payah terus ya? ga ada upgrade nya sama sekali, suka eror kalo pesen makanan perna 2 jam ga selesai², sampai sekarang blum ada yang berubah malah makin jelekkk, lama² blacklist aja apk nya kalo gini terus	1	Negatif

Sumber: (Data Penelitian, 2025)

3.4 Metode Perancangan

Agar alur kerja algoritma *Support Vector Machine* dalam penelitian ini dapat dipahami dengan baik, penelitian ini mengacu pada kerangka perancangan penelitian berikut ini.



Gambar 3. 2 Kerangka Perancangan
Sumber; (Data Penelitian, 2025)

Berdasarkan kerangka perancangan di atas, maka penelitian ini menerapkan beberapa langkah seperti berikut ini.

3.4.1 Pra-Pemrosesan Data Tahap I

Setelah 1000 data ulasan pengguna terhadap layanan Grab diperoleh, peneliti melakukan pra-pemrosesan dasar terhadap dataset ulasan pengguna terhadap layanan ojek online Grab yang telah dikumpulkan dalam format CSV. Dataset ini masih mengandung banyak karakter tidak relevan, seperti simbol, tanda baca, dan hashtag yang umum ditemukan dalam ulasan digital.

Proses pra-pemrosesan dilakukan menggunakan bahasa pemrograman *Python* dengan bantuan pustaka pandas untuk membaca dan mengelola dataset, serta re (*regular expression*) untuk membersihkan teks. Pada tahap ini, peneliti menerapkan empat langkah utama pembersihan teks:

1. Mengubah seluruh teks menjadi huruf kecil (*lowercase*) agar format data lebih seragam dan mengindari redudansi kata akibat perbedaan kapitalisasi.
2. Menghapus hashtag menggunakan pola ekspresi reguler #.*?, yang secara otomatis menghapus kata yang diawali simbol pagar.
3. Menghapus simbol dan tanda baca yang tidak diperlukan dalam proses analisis sentimen, seperti !, ?, ., @, %, dan karakter lainnya, menggunakan pola ekspresi reguler yang telah disesuaikan.
4. Merapikan teks dan menghapus spasi berlebih, menggunakan fungsi strip() serta pembersihan *whitespace* untuk memastikan format teks bersih dan konsisten.

Langkah-langkah ini dilakukan secara terstruktur untuk memastikan bahwa data yang digunakan pada tahap analisis selanjutnya telah bersih dan siap diproses lebih lanjut, khususnya untuk tahap pelabelan.

3.4.2 Tahap Pelabelan Berdasarkan Skor Atau Bintang

Setelah data ulasan berhasil melalui proses pra-pemrosesan awal, tahapan berikutnya adalah pelabelan otomatis berdasarkan skor penilaian yang diberikan oleh pengguna terhadap layanan Grab. Pelabelan ini bertujuan untuk mengkategorikan setiap ulasan ke dalam dua kelas sentimen, yaitu *positif* dan *negatif*, yang akan digunakan sebagai target label dalam proses pelatihan model klasifikasi.

Pada penelitian ini, pelabelan dilakukan mengacu pada nilai numerik dari kolom *score*, yang yang menunjukkan tingkat kepuasan pengguna dalam skala 1 sampai 5. Peneliti menggunakan aturan pelabelan sebagai berikut:

1. Ulasan dengan skor ≤ 3 dikategorikan sebagai sentimen negatif, karena nilai ini umumnya mencerminkan ketidakpuasan pengguna.
2. Ulasan dengan skor ≥ 4 dikategorikan sebagai sentimen positif, karena mencerminkan kepuasan terhadap layanan.

Implementasi pelabelan dilakukan menggunakan fungsi *label_sentiment(score)* yang dibuat dalam bahasa pemrograman Python. Fungsi ini diterapkan pada setiap baris data melalui metode *.apply*() pada kolom score. Hasil dari proses ini adalah tambahan kolom label sentimen (sentiment_label) pada

dataset, yang berisi nilai kategorikal “positif” atau “negatif” dan akan digunakan pada tahap pelatihan algoritma klasifikasi selanjutnya.

3.4.3 Pra-Pemrosesan Tahap II

Setelah proses pra-pemrosesan awal dan pelabelan otomatis selesai dilakukan, penelitian ini melanjutkan dengan tahapan pra-pemrosesan lanjutan guna meningkatkan kualitas representasi teks yang akan digunakan dalam pelatihan model analisis sentimen. Tahapan ini bertujuan untuk menyederhanakan struktur kalimat, menghapus elemen yang tidak relevan, serta menormalkan kata-kata agar lebih sesuai dalam proses ekstraksi fitur dan klasifikasi.

Proses pra-pemrosesan lanjutan dilakukan menggunakan bahasa pemrograman *Python,* dengan bantuan pustaka Sastrawi sebagai alat utama dalam proses stemming untuk Bahasa Indonesia, serta pustaka re (regular expression) untuk pembersihan karakter tertentu. Daftar *stopword* tambahan dimuat dari file eksternal dalam format .txt, yang dikurasi berdasarkan referensi penelitian sebelumnya (Pebiana et al., 2023).

Langkah-langkah yang dilakukan dalam tahapan ini adalah sebagai berikut:

1. Penghapusan emotikon dan karakter Unicode non-ASCII, seperti emoji, menggunakan ekspresi reguler re.sub(r'[^\x00-\x7F]+', '', text), untuk mencegah gangguan saat proses vektorisasi.
2. Tokenisasi manual, yaitu memisahkan teks menjadi token (kata) individual berdasarkan spasi antar kata.

3. Penghapusan *stopword*, dilakukan dengan mencocokkan setiap token terhadap daftar *stopword* yang telah dimuat dari file eksternal, untuk menghilangkan kata-kata umum yang tidak memberikan kontribusi berarti terhadap analisis sentimen.
4. *Stemming*, yaitu proses mengembalikan setiap kata ke bentuk dasarnya menggunakan Sastrawi.Stemmer. Proses ini bertujuan menyatukan variasi morfologis dari suatu kata, seperti "membantu", "dibantu", dan "bantuan", menjadi bentuk dasar "bantu".

Tahapan ini diekspektasi menghasilkan data teks yang lebih bersih, terstandardisasi, dan representatif terhadap makna sentimen yang terkandung, sehingga diharapkan dapat meningkatkan performa algoritma *Support Vector Machine* yang digunakan dalam penelitian ini.

3.4.4 Pembagian Dataset

Setelah seluruh tahapan pra-pemrosesan data selesai dilaksanakan, langkah berikutnya dalam penelitian ini adalah melakukan pembagian dataset menjadi dua bagian, yaitu data latih (*training set*) dan data uji (*testing set*). Pembagian ini bertujuan untuk memisahkan data yang digunakan untuk melatih model dengan data yang digunakan untuk menguji performa model terhadap data yang belum pernah dilihat sebelumnya, sebagai bagian dari strategi validasi untuk mencegah *overfitting*.

Proses pembagian dilakukan menggunakan fungsi train_test_split dari pustaka *scikit-learn*, dengan rasio pembagian 80:20. Sebanyak 80% dari total data

digunakan sebagai data latih, sedangkan 20% sisanya digunakan sebagai data uji, rasio ini dipilih karena merupakan praktik umum yang memberikan keseimbangan optimal antara jumlah data yang cukup untuk melatih model secara efektif dan jumlah data uji yang memadai untuk mengevaluasi performa secara objektif. Parameter random_state=42 ditentukan agar hasil pembagian dapat direplikasi secara konsisten pada setiap eksekusi. Sebelum proses pembagian dilakukan, peneliti terlebih dahulu membersihkan dataset dari entri kosong pada kolom content guna menghindari error saat pelatihan model.

Hasil pembagian data menghasilkan dua subset sebagai berikut:

1. Data latih: 80% dari total data (sekitar 800 entri), digunakan untuk membangun dan melatih model klasifikasi sentimen berbasis *Support Vector Machin*e.
2. Data uji: 20% dari total data (sekitar 200 entri), digunakan untuk menguji dan mengevaluasi performa model terhadap data yang tidak termasuk dalam pelatihan.

Kedua subset dataset tersebut kemudian disimpan dalam dua file terpisah dengan nama data_latih.csv dan data_uji.csv, agar memudahkan akses dalam proses pelatihan dan pengujian model secara sistematis di tahap berikutnya.

3.4.5 Pembobotan Menggunakan TF-IDF

Penelitian ini menggunakan metode TF-IDF (*Term Frequency–Inverse Document Frequency*) sebagai teknik pembobotan kata. Metode ini berfungsi untuk menghitung bobot masing-masing kata berdasarkan frekuensi kemunculannya

dalam sebuah dokumen (*Term Frequency*) serta tingkat kekhususan kata tersebut di seluruh kumpulan dokumen (*Inverse Document Frequency*). Dengan demikian, TF-IDF dapat menyoroti kata-kata yang memiliki nilai penting dalam konteks analisis teks. Adapun langkah-langkah menghitung TF-IDF adalah sebagai berikut:

1. Mengahitung TF (*Term frequency*)

$$TF(t) = \frac{jumlah\ kata\ t\ dalam\ dokumen}{jumlah\ kata\ dalam\ dokumen}$$

2. Menghitung IDF (*Inverse Document Frequency*)

$$IDF\ (t) = Log(\frac{Total\ Dokumen}{Dokumen\ yang\ mengandung\ kata\ t})$$

3. Menghitung TF-IDF

$$TF - IDF(t,d) = TF(t,d)x\ IDF(t)$$

3.4.6 *Support Vector Machine*

Algoritma *Support Vector Machine (SVM)* digunakan dalam penelitian ini untuk melakukan klasifikasi biner terhadap data ulasan pengguna layanan Grab, yaitu membedakan antara sentimen positif dan negatif. SVM dipilih karena algoritma ini mampu membentuk hyperplane pemisah yang optimal pada ruang fitur berdimensi tinggi, serta telah terbukti memiliki performa yang baik dalam tugas-tugas klasifikasi teks.

Implementasi SVM dilakukan menggunakan pustaka *scikit-learn*, dengan langkah-langkah sebagai berikut:

1. Inisialisasi Model

Model SVM diinisialisasi menggunakan kernel linear (SVC(kernel='linear')) karena data telah direpresentasikan dalam bentuk TF-IDF yang bersifat linier.

2. Pelatihan Model (*Training*)

   Model dilatih menggunakan data latih (X_train_tfidf dan y_train) yang sebelumnya telah diproses dan diberi label sentimen.

3. Prediksi (*Testing*)

   Model yang telah dilatih digunakan untuk memprediksi label sentimen pada data uji (X_test_tfidf), menghasilkan label prediksi (y_pred).

4. Evaluasi Model

   Kinerja model diukur menggunakan *confusion matrix*, *classification report* (presisi, recall, F1-score), serta *akurasi keseluruhan*. Hasil evaluasi ini menjadi acuan dalam menilai efektivitas model.

5. Penyimpanan Model

   Model SVM yang telah dilatih disimpan ke dalam file .pkl menggunakan pustaka joblib, agar dapat digunakan kembali pada tahap implementasi sistem atau pengujian lanjutan.

3.4.7 Evaluasi Model

Model yang telah dibangun perlu melalui tahap pengujian guna mengetahui sejauh mana tingkat kinerjanya. Dari *confusion matrix*, dapat dihitung sejumlah metrik evaluasi seperti *accuracy*, *recall*, *precision*, *F1-score*, dan *AUC*.

Perhitungan metrik evaluasi tersebut mengacu pada nilai-nilai yang terdapat dalam tabel *confusion matrix*, yaitu: *True Positive (TP)*, *True Negative (TN)*, *False Positive (FP)*, dan *False Negative (FN)*

Accuracy adalah ukuran yang menunjukkan seberapa tepat prediksi model dibandingkan dengan nilai aktual. Nilai ini diperoleh dari perbandingan jumlah prediksi yang benar terhadap seluruh jumlah data. *Precision* merupakan ukuran ketepatan model dalam memprediksi kelas positif. Nilai ini menunjukkan seberapa banyak prediksi positif yang benar dibandingkan dengan seluruh prediksi positif. *Recall* atau sensitivitas menunjukkan kemampuan model dalam mengenali seluruh data yang termasuk dalam kelas positif secara benar. *F1-score* merupakan rata-rata harmonis antara *precision* dan *recall*. Metrik ini berguna ketika diperlukan keseimbangan antara keduanya, terutama pada data yang tidak seimbang. *AUC* adalah ukuran yang menggambarkan performa model klasifikasi secara keseluruhan dengan menggunakan kurva *Receiver Operating Characteristic (ROC)*. Semakin besar nilai *AUC*, maka semakin baik pula kemampuan model dalam membedakan antara kelas positif dan negatif.

3.5 Jadwal Penelitian

Penelitian ini akan dilakukan selama 7 bulan pada tahun 2025 dengan jadwal penelitian yang dapat dilihat pada tabel di bawah ini.

Tabel 3. 2 Jadwal Penelitian

No	Kegiatan	Tahun 2025															
		April				Mei				Juni				Juli			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Pendefinisian dan perumusan masalah	■															
2	Pengumpulan Data		■	■	■	■	■										
3	Pengolahan Data							■	■	■	■	■	■				
4	Pengujian *Software*													■	■		
5	Penyusunan Hasil Analisi															■	■

Sumber: (Data Penelitian, 2025)