

BAB III

METODOLOGI PENELITIAN

3.1 Jenis dan Pendekatan Penelitian

Penelitian ini merupakan penelitian kuantitatif dengan pendekatan *text mining* untuk menganalisis sentimen masyarakat terhadap Program Makan Bergizi Gratis (MBG) pada media sosial X. Penelitian kuantitatif dipilih karena berfokus pada pengolahan data dalam bentuk numerik yang selanjutnya dianalisis menggunakan metode statistik dan algoritma *machine learning* untuk memperoleh informasi mengenai kecenderungan sentimen masyarakat.

Metode yang digunakan dalam penelitian ini adalah SEMMA (Sample, Explore, Modify, Model, Assess) yang dikembangkan oleh SAS Institute. Metode SEMMA dipilih karena memiliki tahapan yang sistematis dalam proses pengolahan data, mulai dari pengumpulan data, eksplorasi data, persiapan data, pembangunan model, hingga evaluasi model. Setiap tahapan pada metode SEMMA saling berkaitan sehingga mampu menghasilkan model klasifikasi sentimen yang lebih terstruktur dan mudah dievaluasi.

3.2 Lokasi dan Waktu Penelitian

Penelitian ini dilaksanakan secara *online* dengan memanfaatkan media sosial X sebagai sumber data penelitian. Seluruh tahapan penelitian dilakukan tanpa terikat pada lokasi fisik tertentu karena proses pengumpulan data, pengolahan data, pembangunan *Model*, hingga implementasi aplikasi dilakukan menggunakan perangkat lunak berbasis komputasi daring.

Penelitian dilaksanakan selama periode Juni hingga Juli 2026. Pengumpulan data dilakukan pada bulan Juni 2026, sedangkan proses pengolahan data, pembangunan *Model* klasifikasi, evaluasi *Model*, serta implementasi aplikasi analisis sentimen dilaksanakan selama Juni hingga Juli 2026 menggunakan *Google Colaboratory*, Visual Studio Code, dan *framework Streamlit*.

3.3 Objek dan Subjek Penelitian

Objek penelitian ini adalah sentimen masyarakat terhadap Program Makan Bergizi Gratis (MBG) yang diperoleh dari data teks berupa *tweet* pada media sosial X. Sentimen yang dianalisis merupakan opini masyarakat yang berkaitan dengan pelaksanaan Program Makan Bergizi Gratis, baik berupa dukungan, kritik, maupun tanggapan terhadap berbagai isu yang berkembang selama implementasi program.

Subjek dalam penelitian ini adalah pengguna media sosial X yang memberikan opini mengenai Program Makan Bergizi Gratis melalui *tweet* yang dipublikasikan pada akun masing-masing. Data yang digunakan merupakan data publik sehingga tidak melibatkan interaksi secara langsung dengan responden.

3.4 Alat dan Bahan Penelitian

Dalam pelaksanaan penelitian ini digunakan beberapa perangkat keras (*hardware*), perangkat lunak (*software*), *dataset*, serta pustaka (*library*) *Python* yang mendukung proses pengumpulan data, pengolahan data, pembangunan model, hingga implementasi sistem. Adapun alat dan bahan yang digunakan adalah sebagai berikut.

a. *Hardware*

Perangkat keras yang digunakan dalam penelitian ini memiliki spesifikasi sebagai berikut:

Laptop Asus : AMD Ryzen 3 3250U with RadeonDraphics (4 CPUs), ~2.6Ghz

RAM : 8 GB.

b. *Software*

Perangkat lunak yang digunakan dalam penelitian ini disajikan pada Tabel 3.1. berikut.

Tabel 3. 1 Perangkat Lunak yang Digunakan

No	Perangkat Lunak/Library	Fungsi
1	Windows 11	Sistem operasi
2	<i>Google Colab</i>	Pengolahan data dan pembangunan model machine learning

No	Perangkat Lunak/Library	Fungsi
3	<i>Visual Studio Code</i>	Penulisan <i>script</i> proses <i>scraping</i> dan Pengembangan aplikasi <i>Streamlit</i>
4	<i>Python</i>	Bahasa pemrograman
5	<i>Apify</i>	Pengambilan data dari media sosial X
6	<i>Streamlit</i>	Implementasi aplikasi analisis sentimen

c. Komponen Model Analisis Sentimen

Selain perangkat keras dan perangkat lunak, penelitian ini menggunakan beberapa komponen yang dihasilkan selama proses pengolahan dan pemodelan data. Komponen tersebut digunakan untuk mendukung proses klasifikasi sentimen serta implementasi model pada aplikasi pendukung. Komponen yang digunakan meliputi:

1. *Dataset* hasil *scraping* media sosial X yang telah melalui proses *filtering*.
2. *Dataset* hasil *text preprocessing* dan pelabelan sentimen yang digunakan sebagai data untuk proses pemodelan.
3. TF-IDF Vectorizer yang digunakan untuk mengubah data teks menjadi representasi numerik sebelum proses klasifikasi.
4. Model *Multinomial Naïve Bayes* hasil pelatihan.
5. Model *Support Vector Machine* (SVM) hasil pelatihan dan optimasi parameter menggunakan *GridSearchCV*.
6. Berkas model (.pkl) yang digunakan oleh aplikasi *Streamlit* untuk melakukan prediksi terhadap data baru.

Seluruh komponen tersebut dihasilkan melalui tahapan pengolahan dan pemodelan data, kemudian digunakan dalam aplikasi pendukung untuk menampilkan hasil analisis, evaluasi model, dan melakukan prediksi sentimen.

3.5 Teknik Pengumpulan Data

a. *Scraping Data*

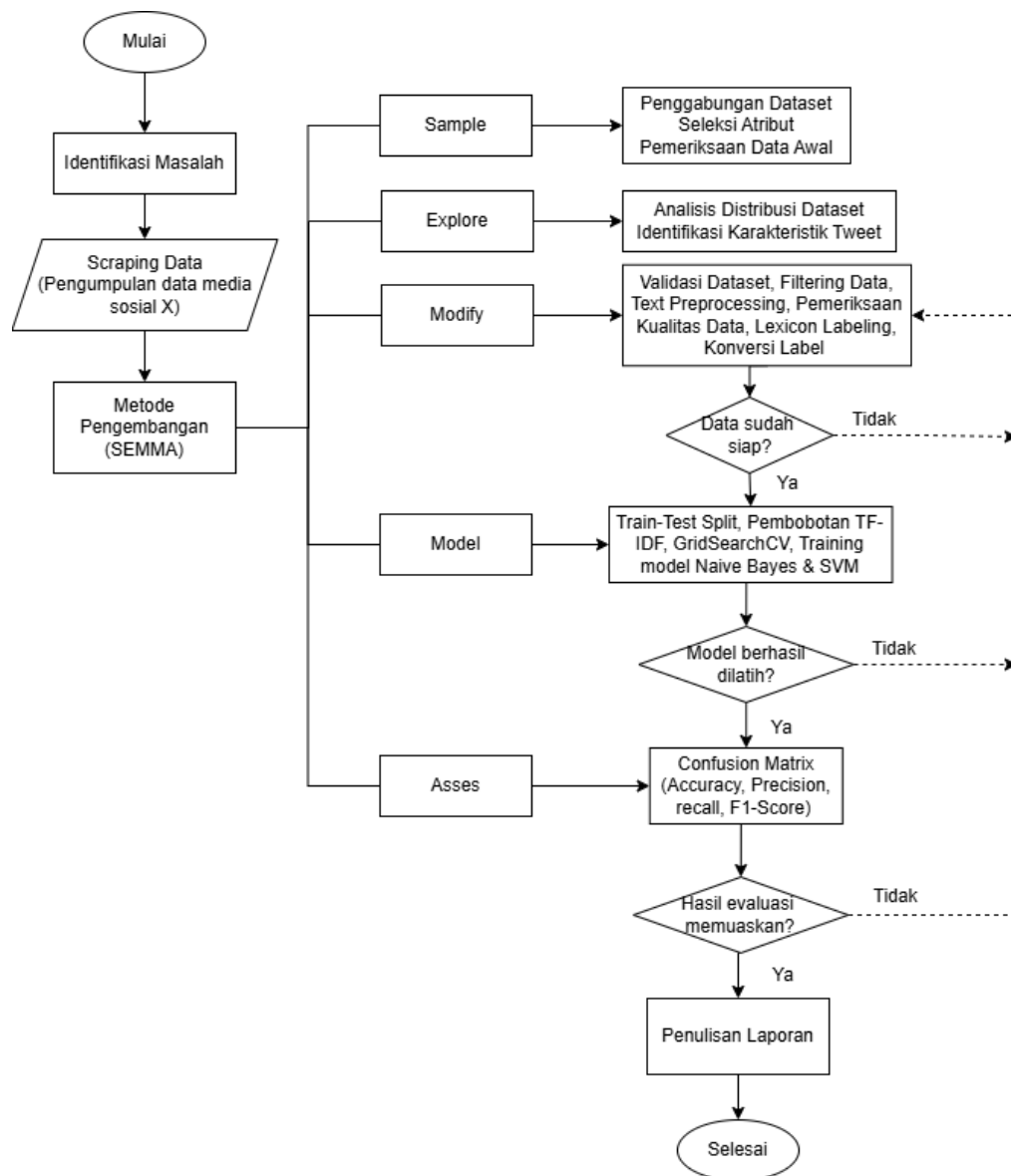
Data penelitian diperoleh melalui proses *web scraping* pada media sosial X dengan memanfaatkan layanan *Apify*. Teknik ini dipilih karena mampu melakukan pengambilan data secara otomatis berdasarkan kata kunci tertentu sehingga proses akuisisi data dapat dilakukan secara lebih efisien. Data yang diperoleh berupa *tweet* publik yang memiliki keterkaitan dengan topik Program Makan Bergizi Gratis (MBG) dan digunakan sebagai *dataset* awal sebelum memasuki tahap pengolahan data.

b. Studi Literatur

Studi literatur dilakukan sebagai tahap pengumpulan referensi untuk memperkuat dasar teori dan mendukung penentuan metode penelitian. Referensi yang digunakan berasal dari buku, jurnal ilmiah, prosiding, skripsi, tesis, serta dokumen relevan yang membahas analisis sentimen, *text mining*, NLP, *Machine learning*, algoritma *Naïve Bayes*, SVM, TF-IDF, dan metodologi SEMMA. Selain itu, studi literatur digunakan untuk mengidentifikasi penelitian terdahulu serta menentukan posisi dan kontribusi penelitian (*State of the Art*).

3.6 Prosedur Penelitian

Prosedur penelitian menunjukkan rangkaian aktivitas yang dilakukan mulai dari proses akuisisi data hingga evaluasi *Model* klasifikasi sentimen. Penelitian ini menerapkan metodologi SEMMA (*Sample, Explore, Modify, Model, Assess*) sebagai kerangka kerja pengolahan data karena mampu memberikan tahapan yang terstruktur dalam proses analisis dan pembangunan *Model*.



Gambar 3. 1 Flowchart Prosedur Penelitian.

Berdasarkan Gambar 3.1, pelaksanaan penelitian terdiri atas lima tahapan utama. Tahap *Sample* dilakukan untuk mempersiapkan *dataset* awal yang diperoleh dari media sosial X melalui proses penggabungan data, pemilihan atribut yang relevan, serta pemeriksaan kondisi awal *dataset*.

Tahap *Explore* digunakan untuk menganalisis karakteristik data melalui pemeriksaan distribusi *dataset*, pola kemunculan kata, serta kondisi data sebelum memasuki tahap pengolahan lebih lanjut. Hasil eksplorasi tersebut menjadi dasar dalam menentukan strategi pemrosesan data pada tahap berikutnya.

Tahap *Modify* mencakup serangkaian proses penyesuaian data, meliputi validasi *dataset*, *filtering*, *text preprocessing*, pemeriksaan kualitas hasil prapemrosesan, pelabelan sentimen berbasis *lexicon*, serta transformasi label ke bentuk numerik agar dapat digunakan sebagai masukan algoritma klasifikasi.

Tahap *Model* dilakukan dengan membangun *Model* klasifikasi menggunakan algoritma *Multinomial Naïve Bayes* dan *Support Vector Machine* (SVM). Proses pada tahap ini meliputi pembagian data menggunakan rasio 80:20, pembobotan fitur dengan TF-IDF, optimasi *hyperparameter* menggunakan *GridSearchCV*, serta pelatihan *Model* berdasarkan konfigurasi parameter terbaik.

Tahap akhir yaitu *Assess* dilakukan untuk mengevaluasi dan membandingkan performa kedua algoritma menggunakan *Confusion matrix*, *Accuracy*, *Precision*, *Recall*, dan *F1-score*. Hasil evaluasi digunakan sebagai dasar pemilihan *Model* terbaik yang kemudian diimplementasikan pada *Dashboard* berbasis *Streamlit* untuk menampilkan hasil analisis serta melakukan pengujian prediksi sentimen terhadap data baru.

3.7 Pengolahan dan Analisis Data

Penelitian ini menerapkan Metode SEMMA sebagai kerangka kerja dalam proses pengolahan dan analisis data. Setiap tahapan SEMMA, yang meliputi *Sample*, *Explore*, *Modify*, *Model*, dan *Assess*, dijelaskan sebagai berikut.

3.7.1 *Sample*

Pada penelitian ini, tahap *Sample* meliputi penggabungan dua *dataset* hasil *Scraping* (format JSON dan CSV), seleksi atribut yang digunakan, serta pemeriksaan kualitas data awal, sebagaimana diuraikan pada subbab berikut.

a. Penggabungan *Dataset*

Data penelitian diperoleh melalui dua kali proses *Scraping* yang menghasilkan dua *dataset* dengan format berbeda, yaitu JSON dan CSV. Kedua

dataset tersebut kemudian digabungkan menggunakan *library* Pandas sehingga membentuk satu *dataset* yang memiliki struktur data yang seragam.

Proses penggabungan dilakukan untuk mempermudah pengelolaan data pada tahapan selanjutnya, sehingga seluruh data dapat diproses dalam satu *dataset* yang terintegrasi.

b. Seleksi Atribut

Tahap seleksi atribut dilakukan setelah seluruh *dataset* hasil penggabungan tersedia. Proses ini bertujuan menentukan variabel yang akan digunakan dalam analisis sehingga data yang diproses hanya mencakup informasi yang memiliki keterkaitan dengan tujuan penelitian.

Pada penelitian ini, atribut *text* dipilih sebagai atribut utama karena berisi isi opini masyarakat mengenai Program Makan Bergizi Gratis (MBG). Atribut lain seperti informasi pengguna, jumlah pengikut, jumlah *likes*, jumlah *retweet*, waktu unggahan, serta metadata tambahan tidak digunakan karena tidak memiliki pengaruh langsung terhadap proses klasifikasi sentimen.

c. Pemeriksaan Missing Value

Sebelum memasuki tahap pengolahan teks, *dataset* diperiksa untuk mengidentifikasi keberadaan data kosong (*missing value*) pada atribut yang digunakan. Pemeriksaan ini dilakukan untuk menjaga kualitas data agar seluruh *tweet* yang masuk ke tahap berikutnya memiliki informasi teks yang dapat diproses. Data yang tidak memiliki isi pada atribut *text* atau mengandung nilai kosong akan dieliminasi karena tidak dapat memberikan informasi yang diperlukan dalam proses analisis sentimen.

d. Pemeriksaan Duplicate Raw Text

Tahap berikutnya adalah pemeriksaan data duplikat berdasarkan atribut *text*. Pemeriksaan ini dilakukan untuk mengetahui jumlah *tweet* yang memiliki isi identik sebelum dilakukan proses *Filtering* maupun *text preprocessing*.

Pada tahap ini data tidak langsung dihapus, melainkan hanya dilakukan identifikasi terhadap jumlah data duplikat sebagai bagian dari analisis kualitas *dataset*. Informasi tersebut digunakan sebagai dasar dalam proses pengolahan data pada tahap berikutnya, khususnya setelah seluruh proses *text preprocessing* selesai dilakukan.

3.7.2 *Explore*

Tahap *Explore* dilakukan untuk memahami karakteristik awal *dataset* yang telah diperoleh pada tahap *Sample* sebelum memasuki proses pengolahan lebih lanjut. Pada tahap ini dilakukan pemeriksaan terhadap kondisi data, pola kemunculan informasi, serta karakteristik *tweet* yang berkaitan dengan Program Makan Bergizi Gratis (MBG). Hasil eksplorasi digunakan sebagai dasar dalam menentukan metode pembersihan dan penyesuaian data pada tahap berikutnya.

Eksplorasi *dataset* pada penelitian ini dilakukan melalui analisis statistik sederhana dan visualisasi data. Proses tersebut mencakup pengamatan jumlah data, karakteristik teks *tweet*, distribusi data, serta perubahan jumlah *dataset* setelah dilakukan proses *filtering*. Informasi yang diperoleh digunakan untuk memastikan bahwa *dataset* yang digunakan telah sesuai dengan kebutuhan analisis sentimen.

a. Analisis Distribusi *Dataset*

Tahap awal eksplorasi dilakukan dengan menghitung jumlah data yang diperoleh setelah proses penggabungan *dataset*. Analisis ini bertujuan untuk mengetahui ukuran *dataset* yang akan digunakan pada proses pengolahan data serta memastikan bahwa seluruh data berhasil digabungkan dengan baik.

Selain menghitung jumlah data, dilakukan pula visualisasi perubahan jumlah *dataset* setelah setiap tahapan *Filtering*. Visualisasi tersebut digunakan untuk menunjukkan pengaruh masing-masing proses *Filtering* terhadap jumlah data yang dipertahankan hingga diperoleh *dataset* akhir yang siap digunakan pada proses analisis sentimen.

b. Identifikasi Karakteristik *Tweet*

Tahap eksplorasi juga dilakukan dengan mengamati karakteristik *tweet* yang terdapat pada *dataset*. Hasil pengamatan menunjukkan bahwa tidak seluruh *tweet* yang berhasil dikumpulkan merepresentasikan opini masyarakat secara langsung. Sebagian data merupakan unggahan dari akun media, informasi resmi pemerintah, konten berita, kampanye, maupun *tweet* yang memiliki pola penulisan berulang.

Selain itu, ditemukan pula *tweet* yang menggunakan bahasa selain Bahasa Indonesia, konten yang bersifat template, meme, serta unggahan yang lebih berorientasi pada penyebaran informasi dibandingkan penyampaian opini. Karakteristik tersebut berpotensi memengaruhi proses pelabelan sentimen apabila tidak dilakukan proses *Filtering* terlebih dahulu.

Berdasarkan hasil identifikasi tersebut, penelitian ini menerapkan beberapa tahapan *Filtering* untuk memperoleh *dataset* yang lebih representatif terhadap opini masyarakat mengenai Program Makan Bergizi Gratis.

c. Visualisasi Data Eksplorasi

Hasil eksplorasi *dataset* disajikan melalui visualisasi data untuk memberikan gambaran mengenai kondisi dan karakteristik data yang diperoleh. Visualisasi yang digunakan mencakup perubahan jumlah *dataset* pada setiap tahapan *filtering* serta frekuensi kemunculan kata pada kumpulan *tweet*.

Penyajian visualisasi tersebut digunakan untuk melihat pola awal pada data, mengevaluasi pengaruh proses *filtering* terhadap jumlah data, serta mengidentifikasi kata-kata yang sering muncul sebelum memasuki tahap *text preprocessing*.

3.7.3 *Modify*

Pada penelitian ini, proses *Modify* meliputi validasi *dataset*, *Filtering* data, prapemrosesan teks (*text preprocessing*), pemeriksaan kualitas data pascapemrosesan, pelabelan sentimen berbasis leksikon (*lexicon-based labeling*), serta konversi label sentimen ke dalam bentuk numerik. Seluruh tahapan

tersebut dilakukan secara berurutan agar kualitas data yang digunakan pada proses pembelajaran mesin (*Machine learning*) dapat ditingkatkan, sebagaimana diuraikan pada subbab berikut.

a. Validasi *Dataset*

Validasi *dataset* dilakukan untuk meninjau kondisi awal data hasil *web scraping* sebelum memasuki tahap *filtering* dan *text preprocessing*. Tahap ini mencakup pemeriksaan terhadap kelengkapan atribut, kesesuaian isi *tweet* dengan topik penelitian, serta karakteristik awal data yang diperoleh agar *dataset* yang digunakan sesuai dengan kebutuhan analisis sentimen.

Validasi dilakukan melalui pemeriksaan struktur *dataset*, tipe data setiap atribut, keberadaan *missing value*, serta identifikasi data duplikat berdasarkan atribut *text*. Pemeriksaan terhadap data duplikat pada tahap ini hanya bertujuan untuk mengetahui kualitas *dataset* awal dan belum dilakukan proses penghapusan data.

Penghapusan data duplikat baru dilakukan setelah seluruh tahapan prapemrosesan teks selesai sehingga kesamaan data dapat diidentifikasi berdasarkan hasil prapemrosesan, bukan berdasarkan teks mentah.

Tahap validasi ini menjadi dasar dalam menentukan proses pengolahan data selanjutnya sehingga setiap tahapan dapat dilakukan secara lebih sistematis dan menghasilkan *dataset* yang memiliki kualitas lebih baik

b. *Filtering* Data

Setelah *dataset* melalui proses validasi, tahap berikutnya adalah *filtering* data untuk menyeleksi *tweet* yang sesuai dengan fokus penelitian. Tahap ini dilakukan dengan menghapus data yang tidak memiliki keterkaitan dengan Program Makan Bergizi Gratis (MBG) sehingga *dataset* yang digunakan lebih merepresentasikan opini masyarakat.

Proses *filtering* dilakukan berdasarkan karakteristik data yang ditemukan pada tahap eksplorasi. Penyaringan mencakup penghapusan data yang tidak relevan, data bersifat informatif, serta data lain yang berpotensi menjadi *noise* dan

memengaruhi proses klasifikasi sentimen. Tahapan *filtering* yang diterapkan pada penelitian ini terdiri dari beberapa proses sebagai berikut.

1. Menghapus data duplikat berdasarkan teks mentah (*raw text*) sehingga setiap opini hanya direpresentasikan satu kali sejak awal proses.
2. Menghapus *tweet* yang tidak berbahasa Indonesia.
3. Menghapus *tweet* yang berasal dari akun media atau portal berita.
4. Menghapus *tweet* yang hanya berisi penyampaian berita tanpa mengandung opini pengguna.
5. Menghapus *template campaign* yang dipublikasikan secara berulang oleh banyak akun.
6. Menghapus *tweet* yang hanya berupa *meme*, gambar, atau konten promosi tanpa opini yang jelas.
7. Menghapus *tweet* yang berasal dari *hashtag campaign* yang bersifat terkoordinasi.
8. Menghapus *tweet* yang mengandung frasa mobilisasi, seperti ajakan untuk menyebarkan informasi atau mengikuti suatu kegiatan.

c. Prapemrosesan Teks (*Text preprocessing*)

Tahap *text preprocessing* merupakan proses penyiapan data teks sebelum dilakukan proses pelabelan sentimen dan pembentukan *Model* klasifikasi. Data yang berasal dari media sosial X memiliki karakteristik tidak terstruktur karena mengandung berbagai variasi penulisan, seperti penggunaan singkatan, simbol, ekspresi berlebihan, serta elemen tambahan yang tidak berkaitan langsung dengan makna sentimen. Oleh karena itu, diperlukan proses pembersihan dan penyesuaian teks agar data memiliki bentuk yang lebih konsisten dan dapat diproses oleh algoritma klasifikasi.

Pada penelitian ini, tahapan *text preprocessing* dilakukan melalui beberapa proses berikut:

1. Cleaning

Tahap *cleaning* dilakukan untuk menghilangkan elemen yang tidak diperlukan dalam proses analisis, seperti URL, *mention*, hashtag, angka,

emoji, tanda baca, simbol, serta karakter khusus lainnya. Proses ini bertujuan mengurangi keberadaan informasi yang tidak memberikan kontribusi terhadap klasifikasi sentimen.

2. Case Folding

Case folding merupakan proses penyamaan format huruf dengan mengubah seluruh karakter teks menjadi huruf kecil (*lowercase*). Tahap ini dilakukan agar kata dengan bentuk penulisan berbeda, seperti "MBG" dan "mbg", dapat dikenali sebagai fitur yang sama oleh *Model*.

3. Remove repeated character

Tahap ini digunakan untuk menangani penulisan karakter yang berulang akibat ekspresi pengguna pada media sosial. Karakter berlebih disederhanakan menjadi bentuk kata yang normal sehingga variasi kata dapat dikurangi. Contohnya, kata "baguuuusss" diubah menjadi "bagus" dan "kereeeennn" menjadi "keren".

4. Tokenizing

Tokenizing merupakan proses memisahkan teks menjadi unit-unit kata (*token*) agar setiap kata dapat diproses secara terpisah pada tahap analisis berikutnya.

5. Normalization

Tahap normalisasi dilakukan untuk mengubah kata tidak baku, singkatan, dan istilah informal yang sering digunakan pada media sosial menjadi bentuk kata yang lebih sesuai. Proses ini menggunakan kamus normalisasi bahasa Indonesia yang diperoleh dari repositori GitHub serta penambahan kamus manual yang disesuaikan dengan karakteristik data penelitian.

6. Stopword Removal

Tahap *stopword removal* dilakukan dengan menghapus kata-kata yang memiliki tingkat informasi rendah terhadap proses klasifikasi. Dengan demikian, fitur yang dipertahankan lebih berfokus pada kata yang memiliki kontribusi terhadap penentuan sentimen.

7. Stemming

Stemming merupakan proses mengubah kata berimbuhan menjadi bentuk

dasarnya. Sebagai contoh, kata "membantu", "dibantu", dan "bantuan" akan dikonversi menjadi kata dasar "bantu". Pada penelitian ini, proses *stemming* dilakukan menggunakan pustaka Sastrawi dengan tambahan metode *dictionary mapping* untuk meningkatkan kesesuaian hasil *stemming* terhadap *dataset* teks media sosial berukuran besar.

d. Pemeriksaan Kualitas Data Pascaprapemrosesan

Setelah seluruh tahapan *text preprocessing* selesai dilakukan, penelitian ini menerapkan serangkaian pemeriksaan kualitas terhadap hasil prapemrosesan sebelum data digunakan pada tahap pelabelan sentimen. Tahap ini dilakukan untuk memastikan data yang digunakan memiliki kualitas, konsistensi, serta mampu menggambarkan opini masyarakat dengan baik.

Tahap pertama dilakukan dengan menghapus data yang menjadi kosong (*empty text*) setelah proses *cleaning*. Kondisi tersebut umumnya terjadi pada *tweet* yang hanya berisi URL, *mention*, simbol, atau karakter lain sehingga tidak lagi memiliki informasi tekstual yang dapat dianalisis.

Selanjutnya dilakukan penghapusan data duplikat berdasarkan atribut *hasil_text preprocessing*. Berbeda dengan pemeriksaan duplikat pada tahap awal yang hanya bertujuan untuk mengidentifikasi keberadaan data duplikat berdasarkan *raw text*, pada tahap ini penghapusan dilakukan terhadap hasil prapemrosesan. Dengan demikian, *tweet* yang memiliki bentuk penulisan berbeda tetapi menghasilkan representasi teks yang sama setelah proses normalisasi dan *stemming* dapat diidentifikasi sebagai data duplikat.

Tahap berikutnya adalah pemeriksaan panjang teks (*quality check*) untuk mengidentifikasi dokumen yang memiliki jumlah kata terlalu sedikit sehingga kurang mampu merepresentasikan suatu opini. Dokumen yang tidak memenuhi batas minimal panjang teks dieliminasi karena dinilai tidak memberikan informasi yang cukup bagi proses analisis sentimen.

Setelah itu dilakukan penyaringan pola kampanye (*final campaign pattern Filtering*) menggunakan sekumpulan frasa yang umum ditemukan pada konten promosi atau narasi kampanye. Proses ini dilakukan setelah tahap prapemrosesan

agar pola kalimat yang memiliki makna serupa dapat dikenali secara lebih konsisten setelah melalui proses normalisasi.

Tahap terakhir adalah *Filtering* data *tweet* yang memiliki jumlah kata kurang dari tiga setelah seluruh proses pemeriksaan kualitas selesai dilakukan. Langkah ini bertujuan memastikan setiap dokumen yang digunakan tetap memiliki informasi yang cukup untuk mendukung proses pelabelan sentimen. Rangkaian pemeriksaan kualitas tersebut menghasilkan *dataset* yang lebih bersih dan konsisten sehingga layak digunakan sebagai masukan pada proses *lexicon-based labeling*.

e. Pelabelan Sentimen (*Lexicon-based labeling*)

Dataset yang telah melalui tahap pemeriksaan kualitas data kemudian diberi label sentimen sebagai persiapan sebelum memasuki proses pembentukan *Model* klasifikasi. Proses pelabelan pada penelitian ini menggunakan pendekatan *lexicon-based labeling*, yaitu metode pemberian kelas sentimen berdasarkan kecocokan kata pada kamus sentimen yang telah memiliki nilai polaritas tertentu.

Pelabelan dilakukan dengan memanfaatkan *Indonesian Sentiment Lexicon* (InSet) sebagai sumber kosakata utama yang terdiri dari kata positif dan negatif beserta bobot polaritasnya. Setiap kata pada hasil *text preprocessing* dicocokkan dengan daftar kata dalam leksikon, kemudian nilai polaritas dari kata-kata yang ditemukan dihitung untuk memperoleh skor sentimen pada masing-masing *tweet*.

Untuk menyesuaikan proses pelabelan dengan karakteristik topik Program Makan Bergizi Gratis (MBG), penelitian ini menambahkan *manual lexicon* yang dikembangkan berdasarkan pola kata dan istilah yang sering muncul pada *dataset*. Penambahan kosakata tersebut dilakukan untuk memperluas cakupan kata sentimen yang belum tercakup pada leksikon utama sehingga hasil pelabelan lebih sesuai dengan konteks pembahasan penelitian.

Selain perhitungan berdasarkan leksikon, proses pelabelan diperkuat melalui penerapan aturan berbasis konteks (*rule-based enhancement*). Aturan yang diterapkan mencakup penanganan kata negasi (*negation handling*), kata penguat (*intensifier*), kata pertentangan (*contrast words*), frasa sentimen (*sentiment phrase*),

serta aturan khusus berdasarkan karakteristik opini masyarakat terhadap Program Makan Bergizi Gratis (MBG).

Hasil perhitungan polaritas kemudian digunakan untuk menentukan kategori sentimen setiap *tweet* berdasarkan nilai skor akhir. Kriteria pemberian label sentimen pada penelitian ini ditampilkan pada Tabel 3.2 berikut.

Tabel 3. 2 Skor Sentimen

Rentang Skor	Label Sentimen
Skor > 0	Positif
Skor = 0	Netral
Skor < 0	Negatif

Dataset yang telah memiliki label sentimen selanjutnya digunakan sebagai data masukan pada tahap *Model* untuk proses pelatihan dan pengujian algoritma klasifikasi.

f. Konversi Label Sentimen

Setelah proses pelabelan selesai, *dataset* memiliki tiga kategori sentimen, yaitu positif, netral, dan negatif. Namun, pada tahap pembangunan *Model* penelitian ini menerapkan klasifikasi biner dengan hanya mempertahankan kelas positif dan negatif. Kelas netral tidak digunakan dalam proses pemodelan karena penelitian berfokus pada identifikasi kecenderungan dukungan atau penolakan masyarakat terhadap Program Makan Bergizi Gratis (MBG).

Tahap berikutnya adalah melakukan transformasi label sentimen ke dalam bentuk numerik agar dapat diproses oleh algoritma *Machine learning*. Pada proses ini, sentimen positif diberi representasi nilai 1, sedangkan sentimen negatif diberi nilai 0 sebagai variabel target (*target variable*) dalam proses klasifikasi.

Dataset dengan label numerik tersebut kemudian digunakan pada tahap pembagian data menjadi data latih (*training data*) dan data uji (*testing data*). Data hasil pembagian selanjutnya diproses melalui ekstraksi fitur menggunakan TF-IDF dan digunakan untuk melatih serta mengevaluasi *Model Multinomial Naïve Bayes* dan *Support Vector Machine* (SVM).

3.7.4 *Model*

Tahap *Model* merupakan tahapan penerapan algoritma klasifikasi terhadap *dataset* yang telah dipersiapkan pada tahap *Modify*. Pada tahap ini dilakukan beberapa proses, yaitu pembagian *dataset* menjadi data latih dan data uji, pembentukan representasi fitur menggunakan TF-IDF, pelatihan *Model* klasifikasi, serta optimasi parameter untuk memperoleh konfigurasi *Model* dengan kinerja terbaik.

Penelitian ini menerapkan dua algoritma klasifikasi, yaitu *Multinomial Naïve Bayes* dan *Support Vector Machine* (SVM). Kedua algoritma tersebut dipilih karena sesuai untuk pengolahan data teks yang memiliki jumlah fitur tinggi setelah proses ekstraksi TF-IDF. Penerapan kedua metode ini juga memungkinkan dilakukan analisis perbandingan performa berdasarkan hasil evaluasi klasifikasi sentimen.

a. Pembagian Data (Train-Test Split)

Dataset yang telah melalui tahap persiapan data kemudian dibagi menjadi dua bagian, yaitu data latih (*training data*) dan data uji (*testing data*) sebelum memasuki proses pembentukan *Model*. Pembagian ini dilakukan untuk memisahkan data yang digunakan dalam proses pembelajaran algoritma dengan data yang digunakan untuk mengukur kemampuan *Model* dalam melakukan prediksi terhadap data baru.

Penelitian ini menerapkan metode *train-test split* dengan perbandingan 80:20, yaitu sebesar 80% *dataset* digunakan sebagai data latih dan 20% lainnya digunakan sebagai data uji. Komposisi tersebut dipilih agar *Model* memperoleh jumlah data yang memadai untuk mempelajari pola sentimen, sementara data uji tetap tersedia dalam jumlah cukup untuk melakukan evaluasi performa.

Proses pembagian data juga menerapkan teknik *stratified sampling* untuk mempertahankan proporsi kelas sentimen positif dan negatif pada masing-masing kelompok data. Dengan pendekatan tersebut, distribusi label pada data latih dan data uji tetap mencerminkan karakteristik *dataset* secara keseluruhan.

b. Pembobotan Kata Menggunakan TF-IDF

Dataset yang telah terbagi menjadi data latih dan data uji kemudian diproses pada tahap ekstraksi fitur menggunakan metode *Term frequency–Inverse document frequency* (TF-IDF). Tahap ini dilakukan untuk mengonversi data teks hasil *preprocessing* ke dalam bentuk vektor numerik yang dapat digunakan oleh algoritma klasifikasi.

TF-IDF memberikan bobot pada setiap kata berdasarkan tingkat kemunculannya dalam suatu dokumen serta tingkat keunikan kata tersebut pada keseluruhan *dataset*. Perhitungan nilai TF-IDF dilakukan melalui perkalian antara nilai *Term frequency* (TF) dan *Inverse document frequency* (IDF) dengan persamaan berikut (lihat persamaan 2.10).

$$TF\text{-}IDF(t, d) = TF(t, d) \cdot IDF(t) \dots \dots \dots (3.1)$$

dengan:

TF(t,d) merupakan frekuensi kemunculan kata *t* pada dokumen *d*.

IDF(t) merupakan nilai kebalikan frekuensi kemunculan kata *t* pada seluruh dokumen.

Pada penelitian ini, proses ekstraksi fitur menggunakan konfigurasi `max_features = 5000`, sehingga hanya 5000 fitur dengan nilai bobot tertinggi yang digunakan sebagai representasi teks. Hasil transformasi TF-IDF berupa matriks fitur numerik yang kemudian menjadi masukan bagi algoritma *Multinomial Naïve Bayes* dan *Support Vector Machine* (SVM) dalam proses pelatihan dan klasifikasi sentimen.

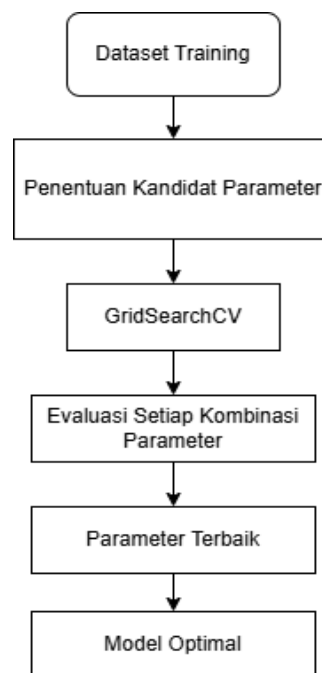
c. Optimasi *Hyperparameter* Menggunakan *Grid Search*

Sebelum tahap pelatihan dan evaluasi *Model*, dilakukan optimasi *hyperparameter* untuk memperoleh konfigurasi algoritma dengan performa klasifikasi terbaik. Proses optimasi diterapkan menggunakan metode *Grid Search Cross-validation* (*GridSearchCV*) dari *library* Scikit-Learn dengan menguji beberapa kombinasi nilai parameter yang telah ditentukan.

Pada penelitian ini, optimasi dilakukan terhadap algoritma *Multinomial Naïve Bayes* dan *Support Vector Machine* (SVM). *GridSearchCV* melakukan

pencarian kombinasi parameter secara sistematis, kemudian mengevaluasi setiap kombinasi menggunakan metode *cross-validation* untuk menentukan konfigurasi dengan nilai performa terbaik.

Parameter hasil optimasi yang diperoleh selanjutnya digunakan pada tahap pelatihan *Model* klasifikasi. Penerapan optimasi ini dilakukan untuk mendapatkan *Model* yang lebih stabil serta meningkatkan kemampuan generalisasi terhadap data uji.



Gambar 3. 2 Alur optimasi Hyperparameter

1. Optimasi *Hyperparameter Multinomial Naïve Bayes*

Pada algoritma *Multinomial Naïve Bayes*, optimasi dilakukan terhadap parameter alpha. Parameter alpha merupakan nilai *smoothing* yang digunakan untuk mengatasi kemungkinan munculnya nilai probabilitas nol (*zero probability*) pada proses klasifikasi.

Nilai alpha yang terlalu kecil dapat menyebabkan *Model* kurang mampu menangani kata yang belum pernah muncul pada data latih, sedangkan nilai yang terlalu besar dapat menyebabkan *Model* menjadi terlalu umum dalam melakukan prediksi.

Proses pencarian parameter terbaik dilakukan dengan menguji beberapa nilai alpha menggunakan *GridSearchCV*. Kombinasi parameter dengan nilai evaluasi tertinggi dipilih sebagai konfigurasi terbaik untuk membangun *Model* akhir.

2. Optimasi *Hyperparameter Support Vector Machine* (SVM)

Optimasi pada algoritma *Support Vector Machine* (SVM) dilakukan menggunakan *GridSearchCV* dengan fokus pada pencarian nilai *hyperparameter C* (*regularization parameter*). Parameter tersebut dipilih karena memiliki pengaruh terhadap kemampuan *Model* dalam menentukan batas pemisah antar kelas sentimen.

Pada penelitian ini, beberapa nilai C diuji untuk memperoleh konfigurasi *Model* yang memberikan hasil klasifikasi terbaik. Setiap kombinasi parameter dievaluasi menggunakan proses *cross-validation*, kemudian nilai dengan performa tertinggi dipilih sebagai konfigurasi akhir yang digunakan pada tahap pelatihan dan pengujian *Model SVM*.

d. Pelatihan *Model* Klasifikasi

Parameter terbaik yang diperoleh dari proses optimasi *hyperparameter* selanjutnya diterapkan pada pembentukan *Model* klasifikasi akhir. Pada tahap ini, algoritma *Multinomial Naïve Bayes* dan *Support Vector Machine* (SVM) dilatih menggunakan data latih yang telah direpresentasikan dalam bentuk fitur TF-IDF. *Model* hasil pelatihan kemudian digunakan untuk melakukan klasifikasi terhadap data uji untuk mengetahui kemampuan *Model* dalam mengenali pola sentimen pada data yang belum pernah digunakan selama proses pembelajaran.

Hasil prediksi dari kedua algoritma selanjutnya dianalisis pada tahap *Assess* menggunakan metrik evaluasi berupa *Confusion matrix*, *Accuracy*, *Precision*, *Recall*, dan *F1-score* untuk menentukan *Model* dengan performa klasifikasi terbaik.

3.7.5 *Assess*

Tahap *Assess* merupakan tahap evaluasi terhadap *Model* klasifikasi yang telah dibangun pada tahap *Model*. Evaluasi dilakukan untuk mengetahui kemampuan *Multinomial Naïve Bayes* dan *Support Vector Machine* (SVM) dalam mengklasifikasikan sentimen pada data uji.

Performa kedua algoritma dibandingkan menggunakan beberapa metrik evaluasi, yaitu *Accuracy*, *Precision*, *Recall*, *F1-score*, serta *Confusion matrix*. Hasil evaluasi tersebut digunakan sebagai dasar dalam menentukan *Model* dengan kemampuan klasifikasi terbaik.

a. Evaluasi Menggunakan *Confusion matrix*

Evaluasi menggunakan *Confusion matrix* dilakukan untuk melihat kesesuaian antara hasil prediksi *Model* dengan label aktual pada data uji. Matriks ini digunakan untuk mengidentifikasi jumlah prediksi yang benar maupun kesalahan klasifikasi pada masing-masing kelas sentimen.

Pada penelitian ini, *Confusion matrix* digunakan untuk menganalisis kemampuan *Model* dalam membedakan sentimen positif dan negatif terhadap Program Makan Bergizi Gratis (MBG). Komponen evaluasi yang digunakan meliputi *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN) sebagaimana telah dijelaskan pada Tabel 2.1.

b. Evaluasi Menggunakan *Accuracy*

Nilai *Accuracy* digunakan untuk mengetahui tingkat keberhasilan *Model* dalam melakukan klasifikasi terhadap seluruh data uji. Metrik ini menghitung perbandingan antara jumlah prediksi yang sesuai dengan label aktual terhadap keseluruhan data yang diuji.

Pada penelitian ini, *Accuracy* digunakan sebagai salah satu indikator awal untuk membandingkan kemampuan *Multinomial Naïve Bayes* dan *Support Vector Machine* (SVM) dalam mengklasifikasikan sentimen masyarakat terhadap Program Makan Bergizi Gratis (MBG).

Persamaan *Accuracy* mengacu pada Persamaan 2.11 pada Bab II, yaitu:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \dots\dots\dots (3.2)$$

c. Evaluasi Menggunakan *Precision*

Precision digunakan untuk melihat tingkat ketepatan hasil prediksi yang diberikan oleh *Model* pada suatu kelas sentimen. Metrik ini menunjukkan proporsi prediksi yang benar dibandingkan seluruh data yang diprediksi sebagai kelas tertentu.

Pada penelitian ini, nilai *Precision* digunakan untuk mengetahui tingkat keakuratan prediksi sentimen positif maupun negatif yang dihasilkan oleh masing-masing algoritma.

Persamaan *Precision* mengacu pada Persamaan 2.12 pada Bab II, yaitu:

$$Precision = \frac{TP}{TP+FP} \dots\dots\dots (3.3)$$

d. Evaluasi Menggunakan *Recall*

Recall *Recall* digunakan untuk mengukur kemampuan *Model* dalam menemukan data yang sebenarnya termasuk ke dalam suatu kelas sentimen. Metrik ini menunjukkan seberapa baik *Model* mengenali seluruh anggota dari kelas tertentu.

Dalam penelitian ini, nilai *Recall* digunakan untuk mengetahui kemampuan algoritma dalam mengenali opini masyarakat yang termasuk ke dalam kategori positif maupun negatif.

Persamaan *Recall* mengacu pada Persamaan 2.13 pada Bab II, yaitu:

$$Recall = \frac{TP}{TP+FN} \dots\dots\dots (3.4)$$

e. Evaluasi Menggunakan *F1-score*

F1-score digunakan sebagai metrik gabungan yang mempertimbangkan keseimbangan antara nilai *Precision* dan *Recall*. Penggunaan metrik ini diperlukan karena distribusi jumlah data pada kelas sentimen positif dan negatif tidak sepenuhnya seimbang.

Nilai *F1-score* yang lebih tinggi menunjukkan bahwa *Model* memiliki kemampuan yang lebih baik dalam menghasilkan prediksi yang tepat sekaligus mengenali data pada masing-masing kelas.

Persamaan *F1-score* mengacu pada Persamaan 2.14 pada Bab II, yaitu:

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \dots\dots\dots (3.5)$$

f. *Classification Report*

Selain pengukuran menggunakan metrik utama, evaluasi *Model* juga dilengkapi dengan *Classification Report* dari pustaka Scikit-learn. Hasil tersebut menampilkan nilai *Precision*, *Recall*, *F1-score*, dan *Support* untuk setiap kelas sentimen secara terpisah. Informasi pada *Classification Report* digunakan untuk menganalisis performa masing-masing algoritma secara lebih detail, terutama dalam melihat kemampuan *Model* pada kelas positif dan negatif.

g. Perbandingan Performa *Model*

Tahap perbandingan performa dilakukan setelah kedua algoritma selesai melalui proses pelatihan dan pengujian menggunakan data uji. Pada tahap ini, hasil klasifikasi *Multinomial Naïve Bayes* dan *Support Vector Machine* (SVM) dianalisis untuk mengetahui algoritma yang memberikan kinerja terbaik dalam mengidentifikasi sentimen masyarakat terhadap Program Makan Bergizi Gratis (MBG). Perbandingan *Model* dilakukan berdasarkan hasil pengukuran *Accuracy*, *Precision*, *Recall*, *F1-score*, serta analisis *Confusion matrix*. Hasil evaluasi tersebut menjadi dasar dalam menentukan *Model* terpilih yang selanjutnya digunakan pada tahap implementasi.

h. Validasi Kinerja *Model*

Setelah proses evaluasi menggunakan *confusion matrix* serta metrik *accuracy*, *Precision*, *Recall*, dan *F1-score*, dilakukan validasi terhadap kinerja model untuk memberikan interpretasi terhadap tingkat akurasi yang diperoleh. Validasi ini bertujuan untuk mengetahui kategori kualitas model klasifikasi berdasarkan nilai *accuracy* hasil pengujian.

Penilaian kategori kinerja model mengacu pada klasifikasi tingkat akurasi menurut Gorunescu, yang membagi nilai *accuracy* ke dalam beberapa kategori, mulai dari sangat rendah hingga sangat baik. Dengan adanya klasifikasi tersebut, hasil evaluasi model tidak hanya disajikan dalam bentuk nilai numerik, tetapi juga dapat diinterpretasikan berdasarkan kategori tingkat performanya.

Nilai *accuracy* dari model *Multinomial Naïve Bayes* dan *Support Vector Machine* (SVM) selanjutnya dibandingkan dengan kategori tersebut untuk menentukan tingkat kinerja masing-masing model.

Tabel 3. 3 Kategori Kinerja Model Berdasarkan Nilai *Accuracy*

Nilai <i>Accuracy</i>	Kategori
0,90–1,00	Sangat Baik (<i>Excellent</i>)
0,80–0,89	Baik (<i>Good</i>)
0,70–0,79	Cukup (<i>Fair</i>)
0,60–0,69	Kurang (<i>Poor</i>)
0,00–0,59	Sangat Kurang (<i>Failure</i>)

Sumber: (Gorunescu, 2011)

3.7.6 Perancangan Sistem

Perancangan sistem dilakukan untuk menggambarkan struktur, komponen, serta interaksi yang mendukung implementasi hasil penelitian analisis sentimen. Pada penelitian ini, sistem yang dirancang berupa aplikasi pendukung berbasis *Streamlit* yang digunakan untuk menyajikan hasil pengolahan data, visualisasi hasil analisis, evaluasi model, serta melakukan prediksi sentimen terhadap teks baru.

Perancangan sistem melibatkan aktor sebagai pengguna aplikasi dan beberapa komponen pendukung yang digunakan dalam proses implementasi. Dokumentasi perancangan sistem menggunakan *Unified Modeling Language* (UML) yang meliputi *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, *Component Diagram*, dan *Deployment Diagram*. *Use Case Diagram* digunakan untuk menggambarkan interaksi aktor dengan aplikasi, *Activity Diagram* menggambarkan alur aktivitas dalam penggunaan aplikasi, *Sequence Diagram* menggambarkan urutan interaksi antara pengguna dan komponen aplikasi,

Component Diagram menggambarkan hubungan antar komponen yang membangun aplikasi, sedangkan *Deployment Diagram* menggambarkan lingkungan implementasi dan penempatan komponen sistem.

Dengan adanya perancangan tersebut, aplikasi pendukung berbasis *Streamlit* dapat digambarkan secara terstruktur sehingga hubungan antara pengguna, aplikasi, model klasifikasi, dan komponen pendukung lainnya dapat dipahami sebelum aplikasi diimplementasikan.

a. *Component Diagram*

Component Diagram digunakan untuk menggambarkan hubungan antar komponen perangkat lunak yang mendukung proses penelitian, mulai dari pengumpulan data hingga implementasi *Model* analisis sentimen pada *Dashboard*. Diagram ini menunjukkan keterkaitan antar komponen yang memiliki fungsi berbeda dalam membangun satu alur kerja penelitian.

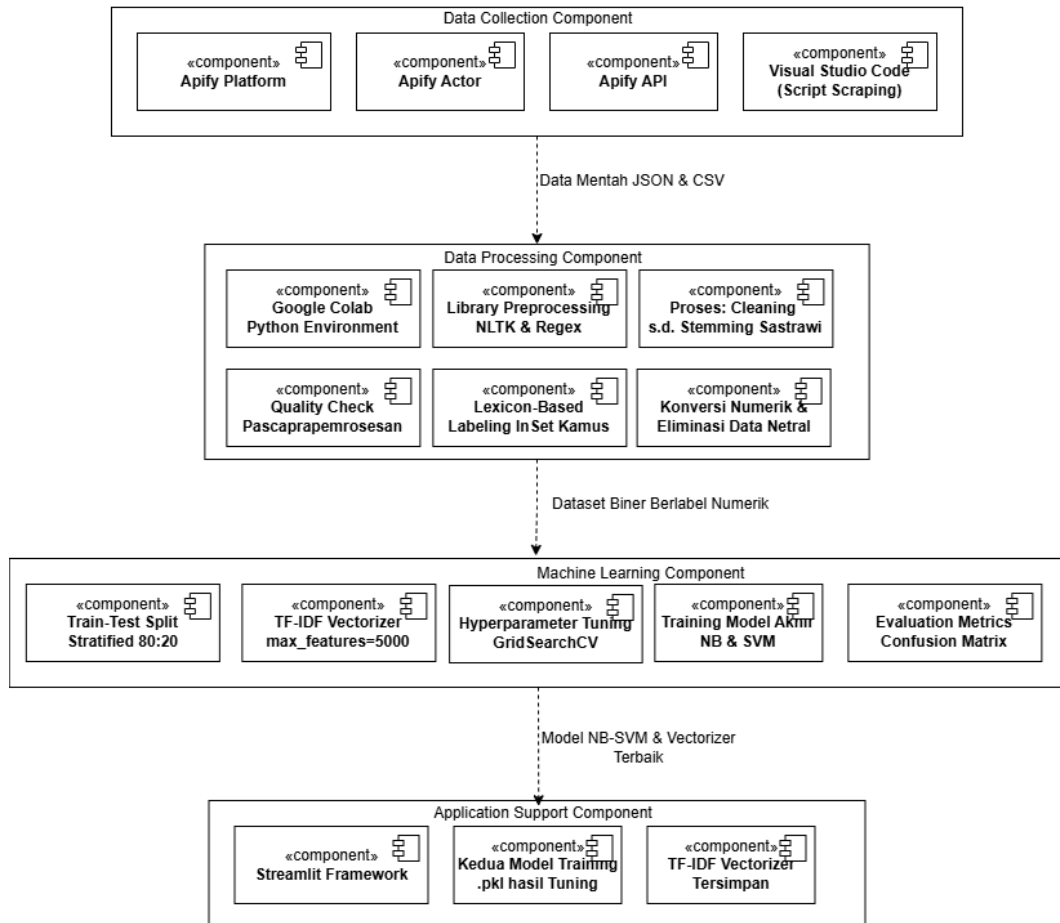
Proses dimulai dari pengumpulan data menggunakan *Python* pada Visual Studio Code yang terhubung dengan *Apify* API untuk memperoleh *tweet* dari media sosial X berdasarkan kata kunci penelitian. Data hasil *scraping* kemudian disimpan dalam format JSON atau CSV sebagai *dataset* awal.

Dataset tersebut selanjutnya diproses melalui *Google Colaboratory* untuk menjalankan tahapan pengolahan data menggunakan metodologi SEMMA, meliputi persiapan data, *text preprocessing*, pelabelan sentimen, ekstraksi fitur TF-IDF, pembangunan *Model Multinomial Naïve Bayes* dan *Support Vector Machine* (SVM), serta evaluasi performa. *Model* terbaik beserta komponen pendukung seperti TF-IDF Vectorizer kemudian disimpan dalam format *.pkl* untuk digunakan pada *Dashboard Streamlit*.

Model hasil pelatihan selanjutnya diterapkan pada *Dashboard* berbasis *Streamlit* bersama dengan TF-IDF Vectorizer untuk melakukan klasifikasi sentimen terhadap data baru serta menampilkan hasil analisis dalam bentuk visualisasi dan informasi pendukung.

Berdasarkan *Component Diagram*, proses pelatihan dan evaluasi *Model* tidak dilakukan pada *Dashboard*, melainkan telah diselesaikan sebelumnya pada

Google Colaboratory. Dashboard hanya berperan sebagai antarmuka implementasi yang memanfaatkan *Model* terlatih untuk penyajian hasil analisis dan pengujian prediksi sentimen.



Gambar 3. 3 Component Diagram

b. Use Case Diagram

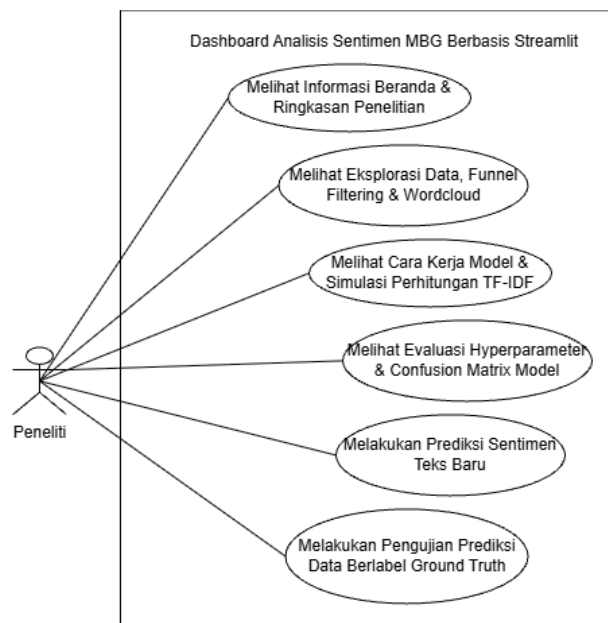
Use Case Diagram pada penelitian ini digunakan untuk menggambarkan interaksi antara aktor dengan aplikasi pendukung berbasis *Streamlit* dalam proses implementasi dan pengujian model analisis sentimen. Diagram ini menunjukkan fungsi-fungsi yang dapat dilakukan oleh aktor melalui aplikasi, serta batasan fungsi aplikasi dalam mendukung hasil penelitian.

Aktor yang berinteraksi dengan aplikasi adalah Peneliti. Peneliti menggunakan aplikasi pendukung berbasis *Streamlit* untuk mengakses hasil pengolahan dan pemodelan yang telah dilakukan sebelumnya. Melalui aplikasi,

Peneliti dapat melihat informasi *dataset*, hasil *filtering*, distribusi sentimen, hasil pembagian data, visualisasi analisis, hasil evaluasi model, serta melakukan pengujian prediksi sentimen.

Pada proses pengujian prediksi, Peneliti dapat memasukkan teks baru ke dalam aplikasi. Teks tersebut selanjutnya diproses menggunakan TF-IDF Vectorizer dan diteruskan ke model klasifikasi yang telah dilatih, yaitu *Multinomial Naïve Bayes* dan *Support Vector Machine* (SVM). Hasil klasifikasi kemudian ditampilkan oleh aplikasi dalam bentuk kategori sentimen.

Dengan demikian, *Use Case Diagram* pada penelitian ini tidak menggambarkan proses pelatihan model sebagai fungsi yang dilakukan langsung oleh aplikasi. Proses pengolahan data, pelabelan sentimen, ekstraksi fitur, optimasi *hyperparameter*, pelatihan, dan evaluasi model telah dilakukan pada tahap penelitian sebelumnya menggunakan *Google Colaboratory*. Aplikasi berbasis *Streamlit* berperan sebagai media implementasi untuk menyajikan hasil penelitian dan melakukan pengujian prediksi menggunakan model yang telah tersedia. *Use Case Diagram* penelitian ini ditunjukkan pada Gambar 3.4.



Gambar 3. 4 Use Case Diagram

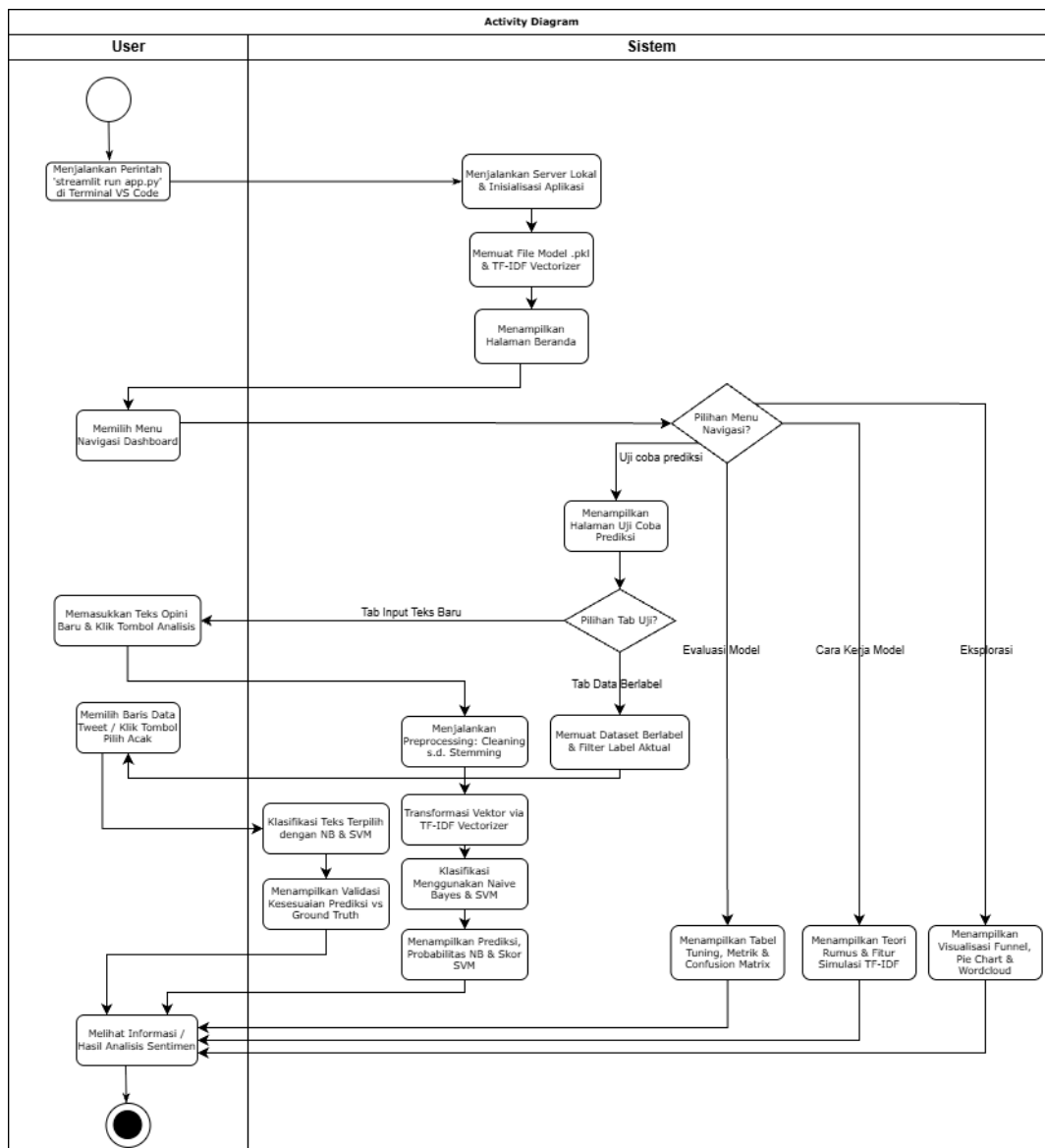
c. *Activity Diagram*

Activity Diagram digunakan untuk menggambarkan alur aktivitas antara Peneliti sebagai aktor dengan aplikasi pendukung berbasis *Streamlit*. Diagram ini menunjukkan urutan aktivitas yang dilakukan mulai dari pengguna mengakses aplikasi, memilih fitur yang tersedia, melihat informasi hasil penelitian, hingga melakukan pengujian prediksi sentimen.

Aktivitas pada aplikasi dimulai ketika Peneliti mengakses Dashboard *Streamlit*. Setelah aplikasi berhasil dijalankan, Peneliti dapat memilih halaman yang tersedia sesuai dengan kebutuhan. Aplikasi kemudian menampilkan informasi atau hasil analisis berdasarkan fitur yang dipilih.

Pada aktivitas pengujian prediksi, Peneliti memasukkan teks opini yang akan dianalisis. Aplikasi menerima input tersebut dan melakukan prapemrosesan teks, kemudian mengubah teks menjadi representasi numerik menggunakan TF-IDF Vectorizer. Hasil transformasi selanjutnya diteruskan ke model *Multinomial Naïve Bayes* dan *Support Vector Machine* (SVM) yang telah disimpan sebelumnya. Model menghasilkan prediksi sentimen yang kemudian ditampilkan kembali pada aplikasi.

Selain menggunakan input teks baru, Peneliti juga dapat melakukan pengujian menggunakan data yang telah memiliki label sebenarnya (*ground truth*) untuk membandingkan hasil prediksi model dengan label aktual. Seluruh aktivitas tersebut dilakukan melalui aplikasi tanpa melakukan proses pelatihan ulang model. Berdasarkan alur tersebut, *Activity Diagram* menggambarkan hubungan aktivitas antara Peneliti dan aplikasi pendukung dalam menggunakan hasil penelitian. *Activity Diagram* aplikasi pendukung ditunjukkan pada Gambar 3.5.



Gambar 3. 5 Activity Diagram Aplikasi Pendukung

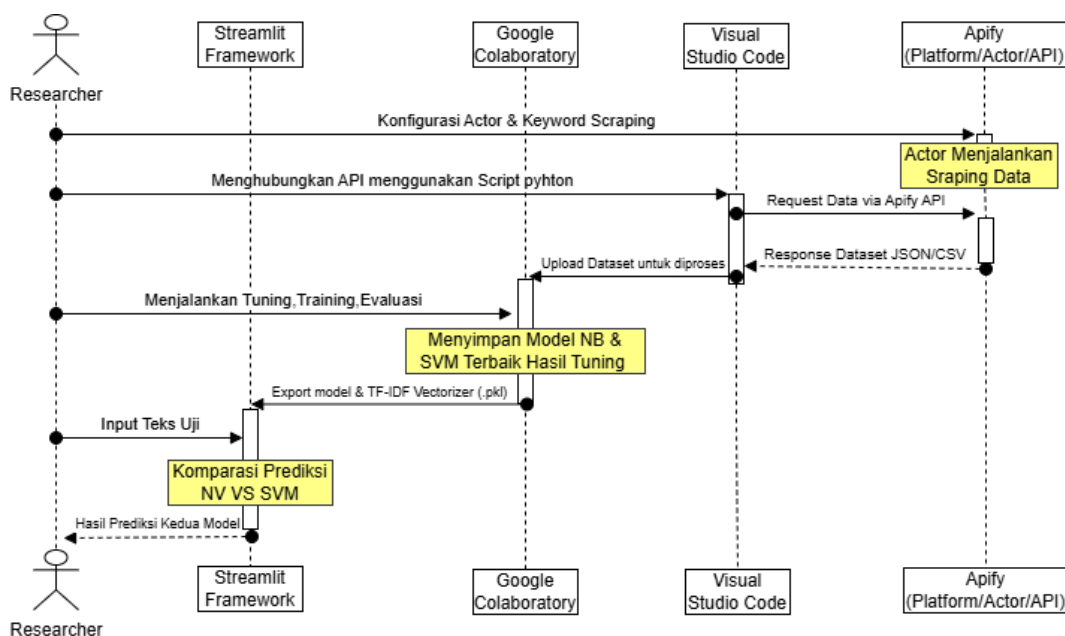
d. Sequence Diagram

Sequence Diagram digunakan untuk menjelaskan alur komunikasi antarobjek dalam proses prediksi sentimen pada *Dashboard Streamlit*. Diagram ini menggambarkan tahapan pertukaran data antara pengguna, aplikasi, *TF-IDF Vectorizer*, dan *Model* klasifikasi hingga menghasilkan keluaran prediksi.

Proses dimulai ketika *Researcher* mengakses *Dashboard* dan memasukkan teks yang akan dianalisis. Input tersebut kemudian diproses oleh *Dashboard* melalui *TF-IDF Vectorizer* untuk mengubah teks menjadi representasi numerik sesuai fitur yang digunakan pada proses pelatihan.

Data hasil transformasi selanjutnya diteruskan ke *Model Multinomial Naïve Bayes* atau *Support Vector Machine (SVM)* untuk dilakukan klasifikasi sentimen. *Model* menghasilkan prediksi berupa kategori sentimen beserta nilai *confidence*, kemudian hasil tersebut dikembalikan ke *Dashboard* untuk ditampilkan kepada pengguna.

Alur tersebut menunjukkan bahwa proses pada *Dashboard* hanya menggunakan *Model* yang telah tersedia tanpa melakukan pelatihan ulang, sehingga fungsi utama sistem berfokus pada implementasi dan pengujian prediksi sentimen.



Gambar 3. 6 Sequence Diagram

e. *Deployment Diagram*

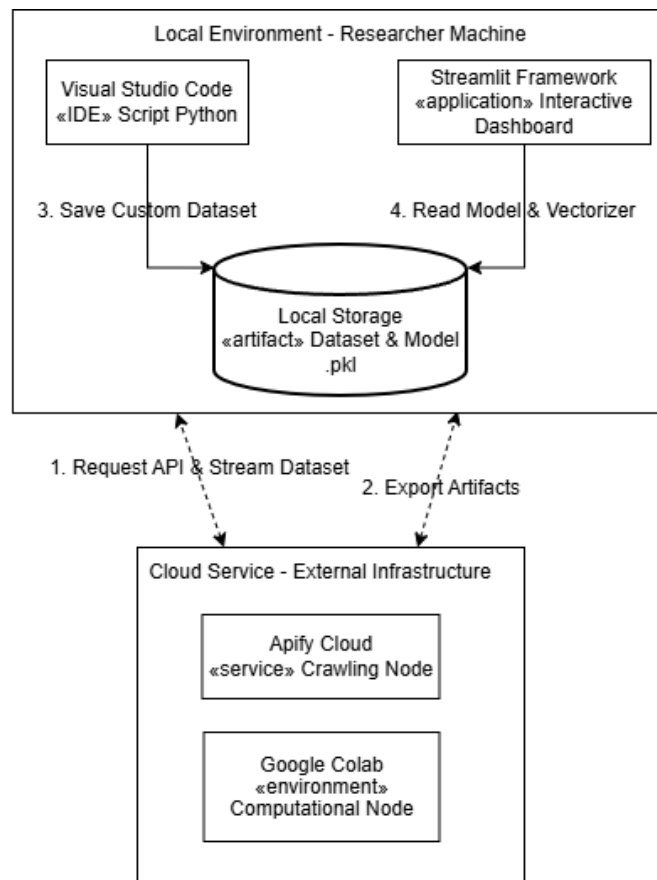
Deployment Diagram digunakan untuk menggambarkan lingkungan implementasi sistem beserta hubungan antarperangkat yang digunakan selama penelitian. Diagram ini menunjukkan bahwa setiap komponen ditempatkan pada lingkungan yang berbeda sesuai dengan fungsi masing-masing.

Komputer lokal (*local machine*) digunakan sebagai lingkungan utama untuk menjalankan Visual Studio Code dan mengembangkan *Dashboard* berbasis *Streamlit*. Pada lingkungan ini, peneliti melakukan proses *Scraping* menggunakan *Python*, mengembangkan antarmuka aplikasi, serta menyimpan *Model* hasil pelatihan dalam format *.pkl*.

Proses pembangunan *Model* dilakukan pada lingkungan *Google Colaboratory* yang menyediakan sumber daya komputasi untuk menjalankan tahapan *SEMMA*, mulai dari pengolahan data hingga evaluasi *Model*. Setelah proses pelatihan selesai, *Model* yang dihasilkan diekspor ke komputer lokal dan digunakan oleh *Dashboard* sebagai komponen utama dalam proses prediksi.

Sementara itu, *Apify* berperan sebagai layanan eksternal yang menyediakan akses pengambilan data dari media sosial X melalui *Application Programming Interface* (API). Layanan tersebut hanya digunakan pada tahap akuisisi data dan tidak terlibat dalam proses klasifikasi maupun penyajian hasil analisis.

Melalui pembagian lingkungan tersebut, *Deployment Diagram* menunjukkan bahwa proses pengumpulan data, pembangunan *Model*, dan implementasi aplikasi dilaksanakan pada platform yang berbeda namun saling terintegrasi sehingga membentuk satu kesatuan sistem pendukung analisis sentimen. *Deployment Diagram* pada Penelitian ini disajikan pada gambar 3.7 berikut.



Gambar 3. 7 Deployment Diagram

3.7.7 Implementasi Aplikasi Pendukung

Implementasi aplikasi pendukung merupakan tahap penerapan *Model* klasifikasi sentimen yang telah melalui proses pelatihan dan evaluasi. Aplikasi dikembangkan menggunakan *framework Streamlit* berbasis *Python* untuk menyediakan antarmuka interaktif dalam menampilkan hasil analisis, informasi *dataset*, evaluasi performa *Model*, serta pengujian prediksi sentimen.

Dashboard yang dibangun memiliki cakupan penggunaan yang terbatas sesuai dengan konteks penelitian, yaitu analisis sentimen masyarakat terhadap Program Makan Bergizi Gratis (MBG) pada media sosial X. Aplikasi ini tidak dirancang sebagai platform analisis sentimen untuk berbagai topik, karena *Model* yang digunakan telah dilatih berdasarkan karakteristik *dataset* dan objek penelitian yang telah ditentukan.

Fungsi utama aplikasi pendukung adalah menghubungkan hasil pemodelan *Machine learning* dengan antarmuka visual sehingga proses implementasi *Model* dapat ditunjukkan secara langsung. Dengan adanya aplikasi ini, peneliti dapat melakukan demonstrasi kemampuan *Model* dalam melakukan klasifikasi sentimen tanpa perlu menjalankan kembali seluruh proses pelatihan *Model*.

Tahapan implementasi aplikasi pendukung meliputi:

1. Perancangan struktur aplikasi.
2. Implementasi halaman aplikasi.
3. Integrasi *Model* klasifikasi.
4. Pengujian fungsi prediksi.

a. Tujuan Implementasi Aplikasi Pendukung

Implementasi aplikasi pendukung bertujuan untuk menyediakan media demonstrasi terhadap *Model* klasifikasi sentimen yang telah dikembangkan sehingga proses prediksi dan evaluasi *Model* dapat ditampilkan secara lebih interaktif.

Tujuan implementasi aplikasi pendukung meliputi:

1. Menampilkan informasi mengenai penelitian, *dataset*, dan tahapan analisis sentimen.
2. Menyajikan hasil *Filtering*, *text preprocessing*, pelabelan sentimen, dan visualisasi data.
3. Menyajikan hasil evaluasi kinerja algoritma *Multinomial Naïve Bayes* dan *Support Vector Machine* (SVM) berdasarkan metrik pengujian yang digunakan.
4. Mengintegrasikan *Model* klasifikasi untuk melakukan proses prediksi sentimen terhadap data baru melalui *Dashboard*.
5. Mempermudah proses demonstrasi hasil penelitian pada saat pengujian maupun presentasi.

Aplikasi ini tidak menggantikan proses utama penelitian. Seluruh tahapan pembentukan *Model*, mulai dari pengumpulan data hingga evaluasi algoritma, tetap dilakukan menggunakan *Google Colaboratory*. *Dashboard* hanya memanfaatkan *Model* yang telah dihasilkan sehingga tidak dapat digunakan untuk menganalisis topik lain tanpa melalui proses pembangunan *Model* kembali.

b. Lingkungan Implementasi

Implementasi aplikasi pendukung dilakukan dengan mengintegrasikan beberapa perangkat lunak yang memiliki fungsi berbeda pada setiap tahapan penelitian. Pembagian lingkungan implementasi ini menunjukkan bahwa proses pembangunan *Model* dan implementasi aplikasi dilakukan secara terpisah.

c. Lingkungan Pengembangan Aplikasi

Lingkungan ini digunakan untuk membangun aplikasi berbasis *Streamlit*.

Tabel 3. 4 Lingkungan Pengembangan Aplikasi

Komponen	Fungsi
Visual Studio Code	Mengembangkan kode program <i>Dashboard</i>
<i>Python</i>	Bahasa pemrograman utama
<i>Streamlit</i>	<i>Framework</i> antarmuka aplikasi
Library <i>Python</i>	Mendukung visualisasi data dan prediksi <i>Model</i>

d. Lingkungan Pembangunan Model

Lingkungan ini digunakan untuk membangun *Model* klasifikasi sentimen.

Tabel 3. 5 Lingkungan Pembangunan Model

Komponen	Fungsi
<i>Google Colaboratory</i>	Pengolahan data dan pelatihan <i>Model</i>
TF-IDF Vectorizer	Transformasi teks menjadi fitur numerik
<i>Model Multinomial Naïve Bayes</i>	<i>Model</i> klasifikasi probabilistik

Komponen	Fungsi
<i>Model Support Vector Machine</i>	<i>Model</i> klasifikasi berbasis <i>hyperplane</i>

Model hasil pelatihan beserta komponen pendukungnya disimpan dalam format *.pkl* agar dapat dimuat kembali oleh *Dashboard Streamlit* pada proses implementasi tanpa memerlukan pelatihan *Model* dari awal.

Pemisahan antara lingkungan pembangunan *Model* dan lingkungan implementasi menunjukkan bahwa *Dashboard* tidak memiliki kemampuan untuk membangun *Model* analisis sentimen secara otomatis. *Dashboard* hanya memanfaatkan *Model* yang telah dihasilkan pada proses penelitian. Oleh karena itu, apabila akan digunakan pada topik atau *dataset* yang berbeda, peneliti harus melakukan kembali seluruh tahapan penelitian mulai dari *Scraping* data, pengolahan data, pelabelan sentimen, pelatihan *Model*, hingga menghasilkan *Model* baru yang kemudian diintegrasikan ke dalam aplikasi.

3.7.8 Perancangan Struktur Aplikasi Pendukung

Struktur aplikasi pendukung dirancang sebagai media implementasi hasil penelitian dalam bentuk *Dashboard* interaktif. Setiap halaman disusun mengikuti alur tahapan analisis sentimen berdasarkan metodologi SEMMA, sehingga pengguna dapat melihat proses penelitian mulai dari pengolahan data hingga hasil evaluasi *Model*.

Dashboard yang dikembangkan memiliki cakupan khusus untuk analisis sentimen Program Makan Bergizi Gratis (MBG) pada media sosial X. Seluruh informasi, visualisasi, dan fitur prediksi yang tersedia bersumber dari *dataset* serta *Model* klasifikasi yang telah dibangun pada tahap penelitian.

Struktur aplikasi pendukung terdiri atas beberapa halaman yang saling terhubung sebagaimana ditampilkan pada Tabel 3.6.

Tabel 3. 6 Struktur Halaman Aplikasi Pendukung

No	Halaman	Fungsi
1	Beranda	Menyajikan informasi umum mengenai penelitian, tujuan penelitian, metodologi, ringkasan <i>dataset</i> , serta gambaran umum <i>Dashboard</i> .
2	Eksplorasi Data	Menyajikan hasil pengolahan <i>dataset</i> , meliputi proses <i>filtering</i> , <i>quality data check</i> , distribusi sentimen, pembagian data latih dan data uji, serta karakteristik <i>dataset</i> melalui visualisasi.
3	Cara Kerja <i>Model</i>	Menjelaskan konsep <i>TF-IDF</i> , <i>Multinomial Naïve Bayes</i> , dan <i>Support Vector Machine</i> beserta alur kerja <i>Model</i> menggunakan ilustrasi, rumus, dan simulasi sederhana.
4	Evaluasi <i>Model</i>	Menyajikan hasil optimasi <i>Hyperparameter</i> , metrik evaluasi, <i>Confusion matrix</i> , serta perbandingan performa kedua <i>Model</i> klasifikasi.
5	Uji Coba Prediksi	Menyediakan fitur pengujian klasifikasi sentimen terhadap data baru menggunakan <i>Model Multinomial Naïve Bayes</i> dan <i>Support Vector Machine</i> (SVM).

Dashboard dirancang dengan struktur halaman yang mengikuti alur penelitian, sehingga pengguna dapat memahami tahapan pengolahan data, pembangunan *Model*, hingga proses pengujian prediksi. Penyusunan tersebut menjadikan *Dashboard* berfungsi sebagai media implementasi *Model* sekaligus penyajian informasi mengenai proses analisis sentimen yang dilakukan secara sistematis.

Seluruh data, *Model* klasifikasi, dan *TF-IDF Vectorizer* yang digunakan pada *Dashboard* merupakan hasil penelitian yang telah dibangun sebelumnya menggunakan *Google Colaboratory*. *Dashboard* tidak melakukan proses *Scraping*, pelatihan *Model*, maupun optimasi *Hyperparameter*. Seluruh proses tersebut telah diselesaikan sebelum tahap implementasi aplikasi sehingga *Dashboard* hanya berfungsi sebagai media visualisasi dan demonstrasi terhadap *Model* yang telah dihasilkan.

3.7.9 Integrasi *Model* Klasifikasi

Integrasi *Model* klasifikasi merupakan proses penerapan *Model Machine learning* hasil pelatihan pada *Google Colaboratory* ke dalam aplikasi pendukung berbasis *Streamlit*. Proses ini dilakukan agar *Model* yang telah melalui tahap pelatihan dan evaluasi dapat digunakan untuk melakukan prediksi sentimen tanpa menjalankan kembali proses pembentukan *Model*.

Model yang digunakan merupakan hasil dari tahap *Model* dan *Assess* pada metodologi SEMMA yang telah disimpan dalam format *.pkl*. Komponen yang diintegrasikan ke dalam aplikasi meliputi:

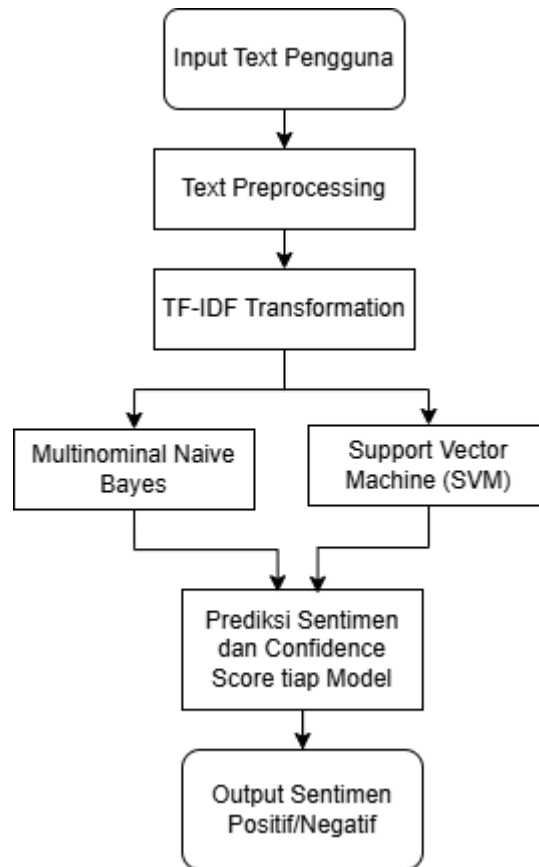
Komponen yang diintegrasikan ke dalam aplikasi meliputi:

- a. *Model* klasifikasi *Multinomial Naïve Bayes*.
- b. *Model* klasifikasi *Support Vector Machine (SVM)*.
- c. *Model TF-IDF Vectorizer*.

Ketika pengguna melakukan prediksi, aplikasi akan memuat komponen *Model* yang diperlukan. Teks masukan dari pengguna kemudian diproses melalui tahapan *text preprocessing* yang sama dengan data pelatihan, selanjutnya dikonversi menjadi representasi numerik menggunakan *TF-IDF Vectorizer*. Hasil transformasi tersebut digunakan sebagai input *Model* klasifikasi untuk menghasilkan kategori sentimen dan nilai *confidence*.

Alur tersebut menunjukkan bahwa proses prediksi pada *Dashboard* mengikuti mekanisme yang sama dengan proses pembentukan *Model*, sehingga data masukan dapat diproses menggunakan representasi fitur dan *Model* klasifikasi yang telah ditetapkan.

Alur integrasi *Model* klasifikasi pada aplikasi pendukung ditunjukkan pada Gambar 3.8.



Gambar 3. 8 Alur Prediksi Dashboard Streamlit

3.6.5 Pengujian Aplikasi Pendukung

Pengujian aplikasi pendukung dilakukan untuk memastikan bahwa seluruh fungsi utama pada *Dashboard* berbasis *Streamlit* dapat berjalan sesuai dengan rancangan. Pengujian ini tidak bertujuan mengevaluasi kualitas perangkat lunak secara menyeluruh, melainkan memastikan bahwa setiap fitur yang dibangun mampu mengimplementasikan *Model* klasifikasi hasil penelitian dengan baik.

Pengujian dilakukan secara fungsional (*functional Testing*) dengan menjalankan setiap halaman aplikasi dan memastikan bahwa proses pemanggilan *Model*, visualisasi data, serta prediksi sentimen dapat berjalan tanpa kesalahan.

Hasil pengujian aplikasi pendukung disajikan pada Tabel 3.7.

Tabel 3. 7 Pengujian Fungsional Aplikasi Pendukung

No	Fitur yang Diuji	Hasil yang Diharapkan	Hasil
1	Menjalankan aplikasi <i>Streamlit</i>	<i>Dashboard</i> berhasil dijalankan melalui browser	Berhasil
2	Menampilkan informasi penelitian	Halaman Beranda dapat ditampilkan	Berhasil
3	Menampilkan eksplorasi data	Visualisasi <i>dataset</i> dan hasil <i>Filtering</i> tampil dengan baik	Berhasil
4	Menampilkan evaluasi <i>Model</i>	Nilai evaluasi dan <i>Confusion matrix</i> dapat ditampilkan	Berhasil
5	Melakukan prediksi sentimen menggunakan <i>Naïve Bayes</i>	Label sentimen dan confidence berhasil ditampilkan	Berhasil
6	Melakukan prediksi sentimen menggunakan <i>Support Vector Machine</i>	Label sentimen dan confidence berhasil ditampilkan	Berhasil

Berdasarkan hasil pengujian fungsional, seluruh fitur utama pada aplikasi dapat dijalankan sesuai dengan fungsi yang dirancang. Dengan demikian, *Dashboard* mampu digunakan sebagai media implementasi untuk menampilkan hasil penelitian dan melakukan demonstrasi prediksi sentimen menggunakan *Model Multinomial Naïve Bayes* maupun *Support Vector Machine*.

3.8 Jadwal Penelitian

Jadwal penelitian disusun sebagai pedoman pelaksanaan seluruh kegiatan penelitian mulai dari penyusunan proposal hingga penyelesaian tugas akhir. Penyusunan jadwal bertujuan agar setiap tahapan penelitian dapat dilaksanakan secara sistematis sesuai dengan waktu yang telah direncanakan. Adapun rincian jadwal penelitian disajikan pada Tabel 3.8 berikut.

Tabel 3. 8 Jadwal Penelitian.

No	Kegiatan	Mar	Apr	Mei	Jun	Jul
1	Penyusunan Proposal					
2	Seminar Proposal					
3	Pengumpulan Data					
4	Pengolahan & Analisis Data					
5	Penyusunan Laporan TA					
6	Seminar Hasil Penelitian					
7	Sidang Tugas Akhir					