

# PENINGKATAN AKURASI ANALISIS SENTIMEN PADA APLIKASI LOKLOK DENGAN METODE NAÏVE BAYES

Shazifa Azhari<sup>1</sup>, Nining Rahaningsih<sup>2</sup>, Raditya Danar Dana<sup>3</sup>, Mulyawan<sup>4</sup>

<sup>1,2,3,4</sup> Sekolah Tinggi Manajemen Informatika Dan Komputer (STMIK) IKMI Cirebon; JL Perjuangan No. 10B, Karyamula, Kec. Kesambi, Kota Cirebon, Jawa Barat 45135; Telp. (0231) 490480

Received: 18 Desember 2024

Accepted: 14 Januari 2025

Published: 20 Januari 2025

## Keywords:

Analisis Sentimen, Naïve Bayes, TF-IDF, Web Scraping, Preprocessing Data, Multinomial Naïve Bayes, Klasifikasi Sentimen, Google Play Store

## Correspondent Email:

[azharishazifa@gmail.com](mailto:azharishazifa@gmail.com)

**Abstrak.** Penelitian ini berfokus pada analisis sentimen pengguna aplikasi Loklok dengan memanfaatkan ulasan yang tersedia di Google Play Store. Metode Naïve Bayes diterapkan untuk mengklasifikasikan sentimen ke dalam kategori positif, negatif, dan netral, dengan memanfaatkan teknik TF-IDF sebagai pembobot fitur guna meningkatkan akurasi model. Dataset yang digunakan terdiri dari 1.000 ulasan yang dikumpulkan melalui teknik web scraping, kemudian diproses menggunakan langkah prapemrosesan seperti tokenisasi, penghapusan kata umum (stopword), dan stemming.

**Abstract.** *This study focuses on the analysis of Loklok application user sentiment by utilizing reviews available on the Google Play Store. The Naïve Bayes method is applied to classify sentiment into positive, negative, and neutral categories, by utilizing the TF-IDF technique as a feature weighting to improve model accuracy. The dataset used consists of 1,000 reviews collected through web scraping techniques, then processed using preprocessing steps such as tokenization, removal of common words (stopwords), and stemming.*

## 1. PENDAHULUAN

Kemajuan pesat dalam bidang Informatika telah memberikan dampak signifikan pada berbagai aspek kehidupan, termasuk teknologi, bisnis, pendidikan, dan lain-lain. Di era digital saat ini, aplikasi mobile menjadi alat penting yang mempengaruhi cara orang berinteraksi, bekerja, dan belajar. Inovasi dalam teknologi informasi memungkinkan pengumpulan dan analisis data dalam skala besar, memberikan wawasan berharga untuk berbagai keperluan. Salah satu contohnya adalah ulasan pengguna di platform seperti Google Playstore, yang mencerminkan pengalaman dan kepuasan mereka terhadap aplikasi tertentu.

Penerapan metode analisis sentimen pada aplikasi Loklok masih menghadapi sejumlah tantangan. Salah satu tantangan utama adalah ketidakseimbangan data antara kategori sentimen positif dan negatif yang sering kali mempengaruhi akurasi model. Tantangan ini relevan karena data yang tidak seimbang dapat menyebabkan model lebih cenderung memprediksi sentimen yang lebih dominan, sehingga hasil analisis tidak mencerminkan persepsi pengguna secara akurat.

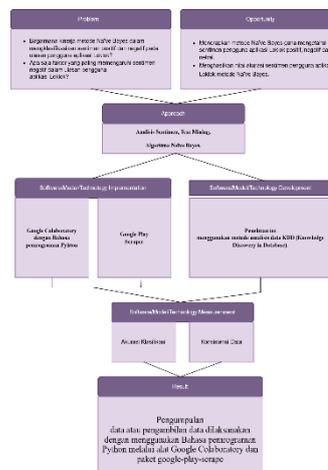
Selain itu, teknik preprocessing yang baik sangat dibutuhkan agar data dapat disaring dan disusun dalam format yang optimal untuk analisis sentimen, terutama pada aplikasi dengan variasi ulasan yang kompleks dan

beragam. Loklok, sebagai aplikasi berbagi konten, sangat bergantung pada feedback pengguna untuk mengevaluasi dan mengembangkan fitur yang sesuai dengan kebutuhan. Meskipun Naïve Bayes terbukti mampu mengklasifikasikan sentimen, kesenjangan dalam literatur menunjukkan kurangnya penelitian mendalam mengenai penerapannya pada platform seperti Loklok, yang memiliki karakteristik ulasan berbeda dibanding platform e-commerce atau media sosial. Selain itu, permasalahan dalam mengelola teks dalam berbagai bahasa dan variasi slang, yang umum dalam ulasan aplikasi ini, memerlukan pendekatan khusus agar hasil analisis lebih relevan.

Penelitian oleh [1] menunjukkan bahwa metode Naïve Bayes dapat mencapai akurasi yang cukup baik dalam analisis sentimen aplikasi. Sementara itu, [2] menekankan pentingnya mengevaluasi persepsi pengguna melalui analisis sentimen sebagai dasar pengembangan fitur yang lebih baik. Penelitian oleh [3] juga menunjukkan bahwa teknik seleksi fitur, seperti Information Gain, dapat meningkatkan akurasi metode Naïve Bayes. Keterbatasan pada penelitian terdahulu memberikan peluang untuk mengeksplorasi penggunaan teknik pembobotan, seperti TF-IDF, guna mengatasi ketidakseimbangan data dan meningkatkan akurasi model dalam analisis sentiment.

Penelitian ini bertujuan untuk memahami persepsi pengguna terhadap aplikasi Loklok melalui analisis ulasan yang diberikan di Google Playstore. Penelitian ini berupaya mengidentifikasi sentimen positif, negatif, dan netral dalam ulasan tersebut untuk memberikan wawasan yang lebih mendalam tentang pengalaman dan kepuasan pengguna. Signifikansi penelitian ini terletak pada kontribusinya dalam mengisi kesenjangan pengetahuan mengenai analisis sentimen pada aplikasi tertentu, analisis sentiment merupakan sebuah metode ekstraksi data berbentuk tekstual. Gagasan di balik analisis sentimen adalah mengakses ribuan sumber online di internet, mengumpulkan pernyataan mengenai ulasan merek atau produk, dan menganalisisnya menggunakan metode analisis teks untuk memahami sentimen yang terkandung di dalamnya [4]

### Kerangka Berfikir



## 2. TINJAUAN PUSTAKA

Metode literature review yang digunakan dalam penelitian ini adalah dengan cara mereview pada artikel-artikel yang terkait dengan Peningkatan Model Analisis Sentimen Terhadap Aplikasi Loklok Menggunakan Metode Naïve Bayes. Langkah-langkah dari literature review meliputi 4 tahapan , yaitu (1) formulasi permasalahan, (2) pencarian literature, (3) evaluasi data, (4) analisis dan interpretasi [5].

Setelah keempat tahapan tersebut dilakukan, proses selanjutnya adalah pelaksanaan literature review. Adapun teknik yang dilakukan dalam literature review yaitu mencai kesamaan (compare), mencari ketidaksamaan (contrast), memberikan pandangan (criticize), membandingkan (synthesize), dan meringkas (summary) [5].

Penelitian pertama dilakukan oleh [5] Kemajuan teknologi, terutama dalam aplikasi perbankan mobile, telah memudahkan nasabah untuk melakukan transaksi tanpa perlu mengunjungi bank. Ulasan pengguna sangat berharga bagi pengembang untuk mengetahui keluhan dan meningkatkan performa aplikasi. Penelitian ini bertujuan untuk menganalisis sentimen ulasan pengguna aplikasi Brimo di Google Play dengan menggunakan teknik web scraping untuk mengumpulkan data dari Agustus 2022 hingga Januari 2023.

Penelitian kedua adalah sebuah jurnal [6] Penelitian ini mengevaluasi sentimen masyarakat terhadap kinerja DPR melalui Twitter. Tahapan analisis mencakup pengumpulan data, preprocessing data, dan

klasifikasi menggunakan Naive Bayes Classifier. Dari 1546 tweet yang dianalisis, terdapat 95 tweet positif (dengan polaritas 75%), 693 tweet netral (dengan polaritas 79%), dan 758 tweet negatif (dengan polaritas 82%), dengan tingkat akurasi sebesar 80% berdasarkan pengujian pada 20% data.

Penelitian ketiga adalah sebuah jurnal [7] Penelitian ini menggunakan metode naïve bayes classifier dan confusion untuk mengklasifikasikan sentimen pengguna Twitter, sebagai media sosial populer, memiliki banyak cuitan yang mengandung opini positif, negatif, dan netral. Diperlukan sistem analisis sentimen menggunakan Naïve Bayes Classifier (NBC) dan teknik crawling. Studi kasus ini menggunakan 8 data cuitan, terdiri dari 5 data latih dan 3 data uji, yang diproses melalui tahap preprocessing dan klasifikasi, serta dianalisis dengan confusion matrix.

Penelitian keempat adalah sebuah jurnal [8] Penelitian ini membandingkan efektivitas metode klasifikasi SVM dan Naïve Bayes dalam mengklasifikasikan ulasan teks berbahasa Indonesia di Google Play Store. Naïve Bayes dikenal karena prosesnya yang sederhana dan akurasi tinggi dalam klasifikasi teks. Akurasi diukur menggunakan metode K-fold cross validation dengan K=3, dan hasilnya disajikan dalam tabel confusion matrix. Temuan menunjukkan bahwa SVM Classifier memiliki akurasi lebih tinggi (81,46%) dibandingkan Naïve Bayes Classifier (75,41%).

Penelitian kelima adalah sebuah jurnal [9] E-gov adalah inovasi digital dari pemerintah yang bertujuan menciptakan proses bisnis yang lebih efisien dan efektif untuk memenuhi kebutuhan masyarakat. Salah satu contoh inovasi ini dalam bidang pertanahan adalah aplikasi Sentuh Tanahku, yang tersedia di Google Play dan mendapatkan banyak ulasan dari pengguna. Ulasan tersebut mempengaruhi penggunaan dan pengembangan aplikasi. Dengan banyaknya ulasan yang ada, pengolahan data secara manual menjadi sulit.

Penelitian keenam menurut jurnal dari [10] Penelitian ini mengevaluasi opini tentang vaksinasi COVID-19 di Indonesia dengan menerapkan algoritma Naïve Bayes Classifier pada 3780 tweet terkait vaksinasi. Analisis mengungkapkan bahwa sebagian besar tweet menunjukkan sikap positif (60,3%), sementara tweet netral (34,4%) lebih banyak dibandingkan

dengan tweet yang menolak (5,4%). Akurasi analisis mencapai 93%.

Penelitian ketujuh menurut jurnal dari [11] Penelitian ini memanfaatkan perkembangan Twitter untuk mengumpulkan kritik dan saran mengenai layanan BMKG Nasional, dengan menggunakan metode klasifikasi Naïve Bayes Classifier (NBC). Sistem yang dikembangkan memproses data Twitter untuk mengklasifikasikan kalimat sebagai opini positif, netral, atau negatif, menggunakan Python 3.74. Analisis dilakukan pada tingkat kalimat komentar melalui text mining dan klasifikasi tweet ke dalam tiga kategori: positif, negatif, dan netral. Metode Naïve Bayes mencapai akurasi klasifikasi sebesar 69,97%.

Penelitian kedelapan menurut jurnal dari [12] Penelitian ini bertujuan untuk menganalisis sentimen publik terhadap feminisme berdasarkan opini yang tersebar di media sosial. Dengan menggunakan data dari tagar yang terkait dengan feminisme, penelitian ini mengumpulkan 600 data yang dikategorikan sebagai opini positif, negatif, atau netral, dan kemudian dianalisis menggunakan metode Naïve Bayes. Hasilnya menunjukkan performa yang baik dengan nilai recall sebesar 86%, precision 89%, F1-Score 86%, dan akurasi 87%. Penelitian ini mengindikasikan bahwa metode Naïve Bayes efektif untuk menganalisis sentimen terhadap opini tentang feminisme.

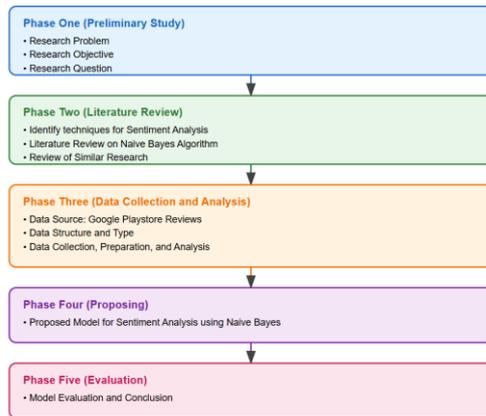
### 3. METODE PENELITIAN

#### 3.1. Metode Penelitian

Penelitian ini menerapkan metode deskriptif kuantitatif dengan pendekatan eksperimen yang digabungkan dengan analisis sentimen menggunakan algoritma Naïve Bayes. Metode deskriptif kuantitatif dipilih karena kemampuannya untuk menggambarkan fenomena atau karakteristik data secara sistematis, faktual, dan akurat, sementara pendekatan eksperimen memungkinkan pengujian serta evaluasi model [9]

Pada tahap pengumpulan data, ulasan pengguna diambil dari Google Playstore menggunakan teknik web scraping otomatis, dibersihkan dari karakter khusus, emoji, dan teks yang tidak relevan, lalu diberi label secara manual menjadi sentimen positif, negatif, dan

netral. Proses preprocessing meliputi standarisasi teks (case folding), tokenisasi, penghapusan kata-kata umum (stopword), dan stemming dengan algoritma Nazief-Adriani yang efektif untuk Bahasa Indonesia



Gambar 3. 1 Tahapan Metode Penelitian

### 3.2. Sumber Data

Penelitian ini menggunakan data sekunder yang diperoleh dari ulasan pengguna aplikasi Loklok di Google Play Store, yang dikumpulkan menggunakan teknik web scraping melalui platform Google Colab. Pemilihan data sekunder ini didasarkan pada beberapa alasan utama: pertama, ulasan di Google Play Store dapat diakses dengan mudah dan dalam jumlah besar tanpa memerlukan pengumpulan data primer seperti survei atau wawancara, yang menghemat waktu dan biaya (Wen et al., 2020).

### 3.3. Populasi dan Sampel

Populasi dalam konteks penelitian ini mencakup seluruh pengguna aplikasi Loklok yang memberikan ulasan di Google Play Store. Data yang digunakan diambil secara keseluruhan dari platform tersebut, sehingga tidak melibatkan proses sampling.

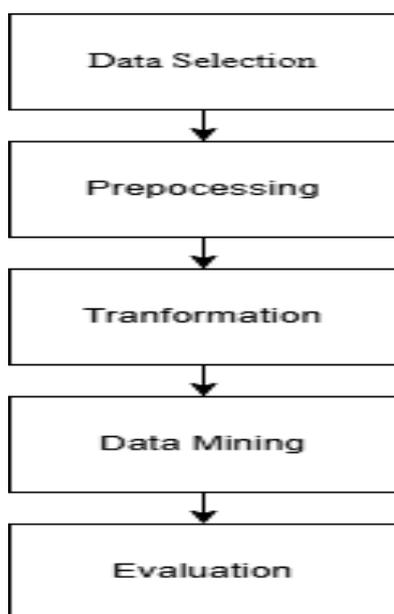
Penelitian ini tidak menerapkan teknik sampling tertentu karena menggunakan data sekunder yang tersedia secara langsung di Google Play Store. Semua ulasan yang dapat diakses dari platform tersebut digunakan sepenuhnya tanpa melakukan pemilihan sampel atau representasi tertentu, sehingga memberikan gambaran lengkap tentang opini pengguna.

### 3.4. Teknik Pengumpulan Data

Dalam penelitian ini, pengumpulan data dilakukan menggunakan Bahasa Pemrograman Python dan Google Colaboratory, dengan memanfaatkan paket google-play-scraper untuk teknik web scraping. Proses scraping ini memungkinkan peneliti untuk mengumpulkan data penting seperti username, konten ulasan, rating, dan tanggal ulasan dari aplikasi Loklok yang ada di Google Play Store. Teknik ini sangat efisien karena memungkinkan pengambilan data dalam jumlah besar secara otomatis dan terstruktur, menghemat waktu serta biaya penelitian (Liu et al., 2020). Setelah data terkumpul, langkah selanjutnya adalah melakukan pembersihan data (data cleaning), di mana elemen-elemen yang tidak relevan seperti karakter khusus, emoji, dan HTML dihapus untuk memastikan data siap dianalisis.

### 3.5. Teknik Analisis Data

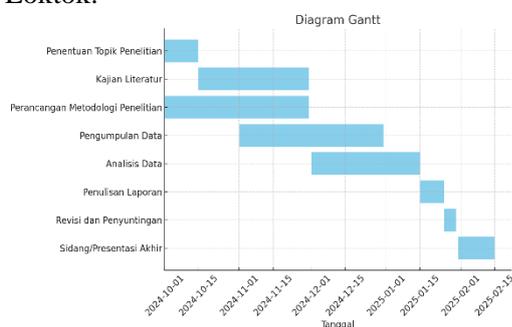
Knowledge Discovery in Databases (KDD) melibatkan proses analisis data yang terstruktur dan terencana untuk memperoleh wawasan berharga. Tahap awal dimulai dengan pemahaman yang jelas mengenai tujuan dan domain penelitian, diikuti oleh pemrosesan data untuk menangani nilai yang hilang atau anomali dalam dataset. Kemudian, dilakukan Exploratory Data Analysis (EDA) melalui analisis deskriptif dan visualisasi guna memahami karakteristik data.



Gambar 3. 5.1 Tahapan KDD

### 3.6. Jadwal Penelitian

Berikut adalah jadwal penelitian yang direncanakan untuk analisis sentimen aplikasi Loktok:



#### Batas Waktu Akhir:

Seluruh proses penelitian direncanakan untuk diselesaikan pada akhir Februari 2025, bertepatan dengan jadwal pelaksanaan sidang atau presentasi akhir. Jadwal ini memberikan peneliti waktu yang memadai untuk menyelesaikan revisi laporan, menyiapkan materi presentasi, dan memastikan seluruh hasil penelitian dapat disampaikan secara sistematis dan terstruktur. Dengan adanya batas waktu yang jelas, setiap tahap penelitian diharapkan dapat diselesaikan dengan efisien dan tepat waktu, sehingga penelitian menghasilkan output yang berkualitas dan dapat dipertanggungjawabkan secara akademis.

## HASIL DAN PEMBAHASAN

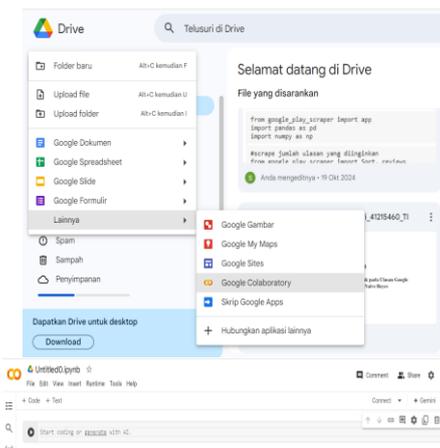
### 4.1. Hasil Penelitian

Hasil penelitian adalah kesimpulan atau temuan yang diperoleh dari suatu studi atau penyelidikan ilmiah yang dilakukan oleh peneliti. Ini mencakup data, informasi, dan analisis yang dihasilkan dari proses penelitian yang dijalankan secara sistematis dan metodologis, berdasarkan fakta, dengan tujuan memecahkan suatu masalah atau menguji hipotesis. Melalui pemikiran kritis dalam mengelola dan menganalisis objek penelitian, terbentuklah prinsip-prinsip umum. Tujuan dari hasil penelitian adalah untuk memberikan kontribusi pada bidang pengetahuan terkait, memperdalam pemahaman tentang fenomena yang diteliti, serta menjadi dasar bagi pengembangan kebijakan, praktik, atau teknologi baru. Keberhasilan hasil penelitian diukur melalui validitas data yang digunakan. Dalam penelitian ini, data dikumpulkan menggunakan bahasa pemrograman Python melalui Google Colaboratory dengan bantuan paket google-play-scraper.[13]

#### 4.1.1 Pengumpulan data atau Data selection

##### a) Data Collection

Pada tahap ini, pengumpulan data dilakukan dengan menggunakan bahasa pemrograman Python melalui platform Google Colaboratory dan paket google-play-scraper. Langkah pertama dimulai dengan membuka Google Drive, lalu masuk ke menu di pojok kiri atas dan klik *New > More*. Selanjutnya, pilih opsi Google Colaboratory. Setelah tampilan *Colab* terbuka, ubah nama proyek di pojok kiri atas. Kemudian, lanjutkan ke tahap pembuatan proyek seperti yang terlihat pada gambar di bawah.



Selanjutnya, di bagian Google Colab, peneliti menggunakan kode from `google_play_scraper` import `review`, `sort` dan memilih data yang paling baru dengan mengatur `sort=sort.NEWST`.



Selanjutnya, peneliti membuat skrip dengan mengetikkan `!pip install google-play-scraper sastrawi`, kemudian menjalankan program tersebut dengan mengklik tombol `display` atau menggunakan pintasan `Ctrl + Enter`. `Google-play-scraper` adalah sebuah paket yang digunakan untuk melakukan scraping data, sedangkan `Sastrawi` merupakan pustaka Python yang berfungsi untuk melakukan stemming teks dalam bahasa Indonesia. Hal ini bisa dilihat pada gambar di bawah ini.



Setelah data ulasan terkumpul, peneliti menyimpan hasilnya dalam format Excel untuk memudahkan proses pelabelan manual. Data tersebut diekspor ke file Excel menggunakan fungsi `df.to_excel`, sehingga file bisa diunduh dari Google Colab dan diberi label secara manual sesuai kategori sentimen yang tepat. Karena jumlah data yang diambil mencapai 1000 ulasan, langkah ini membantu dalam pengelolaan dan analisis sentimen lebih lanjut.

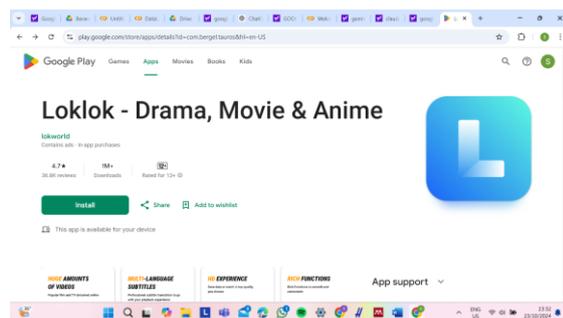


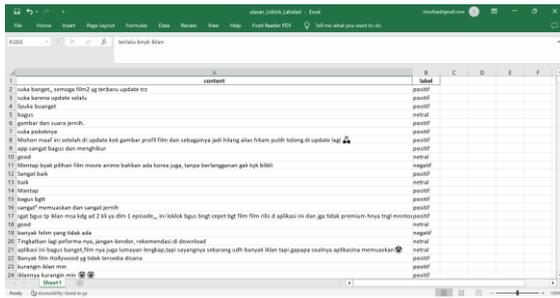
**a) Data Labeling**

Setelah itu, peneliti melanjutkan dengan melakukan scraping data dari situs `Google Play Store`. Peneliti memasuki menu pencarian dan mengetikkan nama aplikasi yang ingin dicari ulasan komentarnya sebagai sampel penelitian. Peneliti membuka menu pencarian dan mengetikkan nama aplikasi yang ingin dicari ulasan komentarnya sebagai sampel penelitian. Dalam penelitian ini, ulasan yang digunakan berasal dari aplikasi `Loklok`.

Setelah data ulasan disimpan dalam format Excel, langkah berikutnya adalah melakukan proses pelabelan secara manual. Pada tahap ini, setiap ulasan diklasifikasikan ke dalam tiga kategori sentimen utama: positif, negatif, atau netral. Proses pelabelan dilakukan secara manual untuk menjamin ketepatan klasifikasi sentimen dari setiap ulasan.

Peneliti membuka file Excel yang sudah diunduh, lalu menambahkan kolom label untuk mengategorikan setiap ulasan sesuai dengan isi teksnya. Gambar berikut menunjukkan contoh data ulasan yang sudah diberi label secara manual.





Gambar ini menampilkan beberapa contoh ulasan beserta label sentimen yang telah diberikan. Label sentimen ini akan digunakan sebagai target dalam pelatihan model klasifikasi, sehingga model dapat mempelajari pola yang terkait dengan masing-masing kategori sentimen.

### 4.1.2 Preprocessing

Tahap preprocessing merupakan salah satu proses krusial dalam penelitian ini, bertujuan untuk mengolah data teks yang telah dikumpulkan agar lebih terstruktur, seragam, dan bersih. Data yang diperoleh dari Google Play Store biasanya memiliki variasi teks yang kompleks, seperti penggunaan karakter khusus, tanda baca, serta kata-kata yang kurang relevan atau tidak bermakna untuk analisis sentimen. Proses ini diperlukan agar teks dapat disederhanakan dan kualitas data meningkat sebelum memasuki tahap pemodelan.

### Mengunggah File yang Telah Diberi Label Manual

Langkah awal dalam proses preprocessing adalah mengunggah file Excel yang berisi data ulasan yang telah dilabeli secara manual. Dalam hal ini, file bernama ulasan\_Loklok\_labeled.xlsx diunggah ke Google Colab dan kemudian dibaca menggunakan pustaka Pandas.



### Definisi Fungsi Prapemrosesan

Setelah file diunggah, setiap tahapan prapemrosesan didefinisikan sebagai fungsi terpisah. Fungsi-fungsi ini mencakup

clean\_text() untuk membersihkan teks, case\_folding() untuk mengubah semua huruf menjadi kecil, tokenize\_text() untuk memecah teks menjadi kata-kata terpisah, normalize\_text() untuk menghapus elemen tidak relevan, remove\_stopwords() untuk menghapus kata-kata umum, dan stemming() untuk mengubah kata-kata menjadi bentuk dasarnya.



#### a). Cleaning

Cleaning merupakan proses untuk menghilangkan karakter khusus, tanda baca, serta simbol-simbol lain yang tidak memiliki kontribusi berarti dalam analisis sentimen. Sebagai contoh, teks seperti "aplikasi bagus, update film jg cepat 👍👍👍" akan melalui tahap ini untuk menghapus emoji dan karakter yang tidak relevan. Hasilnya, teks tersebut disederhanakan menjadi "aplikasi bagus update film jg cepat".

Before Cleaning: aplikasi bagus, update film jg cepat 👍👍👍  
 After Cleaning: aplikasi bagus update film jg cepat

#### b). Case Folding

Case folding adalah proses mengubah seluruh teks menjadi huruf kecil untuk menjaga konsistensi dalam analisis. Langkah ini mencegah kata-kata dengan arti yang sama

dianggap berbeda hanya karena perbedaan dalam penggunaan huruf kapital.

Contoh: Sebelum Case Folding: "aplikasi bagus update film jg cepat "

Setelah Case Folding: "aplikasi bagus update film jg cepat"

Before Case Folding: aplikasi bagus update film jg cepat  
After Case Folding: aplikasi bagus update film jg cepat

#### d). Tokenizing

Tokenizing adalah langkah memecah teks menjadi unit-unit kata yang terpisah. Proses ini memungkinkan setiap kata dalam teks dianalisis secara individual, sehingga mempermudah model untuk memahami makna setiap kata dalam konteksnya.

Sebagai contoh, kalimat "aplikasi bagus update film jg cepat" akan diubah menjadi daftar kata: ['aplikasi', 'bagus', 'update', 'film', 'jg', 'cepat'].

After Case Folding: aplikasi bagus update film jg cepat  
Before Tokenizing: aplikasi bagus update film jg cepat

#### e). Normalization

Normalization adalah proses standarisasi teks dengan menghilangkan elemen-elemen yang tidak relevan, seperti angka atau simbol yang tidak berkontribusi pada analisis. Sebagai contoh, angka-angka yang tidak memiliki makna signifikan akan dihapus dari teks.

Before Normalization: ['aplikasi', 'bagus', 'update', 'film', 'jg', 'cepat']  
After Normalization: ['aplikasi', 'bagus', 'update', 'film', 'jg', 'cepat']

#### f). Stopword Removal

Stopword removal adalah langkah dalam prapemrosesan teks untuk menghapus kata-kata umum (stopword) yang tidak memiliki nilai signifikan dalam analisis, seperti "dan", "di", atau "yang". Proses ini membantu analisis lebih fokus pada kata-kata kunci yang relevan, meningkatkan efisiensi dan akurasi model, terutama dalam analisis sentimen atau klasifikasi teks.

Contoh: Sebelum Stopword Removal: ['aplikasi', 'bagus', 'update', 'film', 'jg', 'cepat']  
Setelah Stopword Removal: ['aplikasi', 'bagus', 'update', 'film', 'jg', 'cepat'].

Before Stopword Removal: ['aplikasi', 'bagus', 'update', 'film', 'jg', 'cepat']  
After Stopword Removal: ['aplikasi', 'bagus', 'update', 'film', 'jg', 'cepat']

#### g). Stemming

Stemming adalah proses menyederhanakan kata-kata menjadi bentuk dasar atau akarnya. Langkah ini bertujuan untuk mengurangi variasi kata yang memiliki arti serupa tetapi bentuknya berbeda. Sebagai contoh, kata seperti "update" akan diubah menjadi "updat". Setelah melalui semua tahap prapemrosesan, teks "aplikasi bagus update film jg cepat" akan menjadi daftar kata: ['aplikasi', 'bagus', 'update', 'film', 'jg', 'cepat'].

Before Stemming: ['aplikasi', 'bagus', 'update', 'film', 'jg', 'cepat']  
After Stemming: ['aplikasi', 'bagus', 'update', 'film', 'jg', 'cepat']

#### Hasil Akhir dari Preprocessing

Setelah semua tahapan prapemrosesan dilakukan, hasilnya disimpan dalam kolom baru bernama cleaned\_text. Tabel di bawah ini menampilkan perbandingan antara teks asli (content) dengan teks yang telah diproses (cleaned\_text), yang telah disederhanakan dan siap digunakan untuk analisis seperti analisis sentimen atau klasifikasi teks.

content	cleaned_text
suka banget,, semoga film2 yg terbaru update trz	suka banget moga film yg baru update trz
suka karena update selalu	suka update
Syuka buanget	syuka buanget
Bagus	bagus
gambar dan suara jernih.	gambar suara jernih

	content	cleaned_text
0	suka banget,, semoga film2 yg terbaru update trz	suka banget moga film yg baru update trz
1	suka karena update selalu	suka update
2	Syuka buanget	syuka buanget
3	bagus	bagus
4	gambar dan suara jernih.	gambar suara jernih

Tahap preprocessing sangat penting untuk memastikan teks ulasan dalam format yang bersih, ringkas, dan konsisten. Proses ini menyederhanakan data, menghilangkan elemen yang tidak relevan, dan mengurangi noise, sehingga teks menjadi lebih terstruktur dan siap untuk pemodelan. Dengan preprocessing yang baik, model dapat fokus pada informasi penting, meningkatkan akurasi dalam analisis sentimen dan tugas lainnya.

#### 4.1.3 Transformation

Tahap Transformation mengubah teks yang telah diproses menjadi representasi numerik

yang dapat digunakan dalam pemodelan. Dua teknik utama yang digunakan dalam tahap ini adalah TF-IDF dan Information Gain. TF-IDF menilai pentingnya kata dalam dokumen, sementara Information Gain memilih fitur yang relevan untuk analisis. Kedua teknik ini memastikan model fokus pada kata-kata yang memiliki makna penting, meningkatkan akurasi dalam analisis sentimen atau tugas lainnya.

#### a) TF-IDF

Proses ini bertujuan untuk menyiapkan data agar sesuai dengan kebutuhan analisis. TF (*Term Frequency*) mengukur seberapa sering sebuah kata muncul dalam dokumen, dihitung sebagai rasio kemunculan kata terhadap jumlah kata dalam dokumen. Di sisi lain, IDF (*Inverse Document Frequency*) menilai keunikan kata dengan melihat seberapa sering kata tersebut muncul di seluruh koleksi dokumen. Kata yang jarang muncul di seluruh koleksi akan memiliki nilai IDF lebih tinggi. Gabungan dari keduanya, yaitu TF-IDF, memberikan bobot pada setiap kata dalam dokumen. Proses ini dapat dilakukan dengan menggunakan `TfidfTransformer` dan `CountVectorizer`, yang pertama menghitung frekuensi kata dan IDF secara terpisah sebelum menggabungkannya untuk mendapatkan skor TF-IDF. Sementara itu, TF-IDF Vectorizer menggabungkan ketiga langkah tersebut dalam satu proses menggunakan data yang sama. Seperti gambar di bawah ini.

```
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.feature_selection import SelectKBest, chi2
import pandas as pd

# Mengasumsikan kolom label bernama 'label'
label_column_name = 'label'

# transformasi TF-IDF
tfidf = TfidfVectorizer()
X = tfidf.fit_transform(data['cleaned_text']) # Mengubah teks menjadi representasi numerik menggunakan TF-IDF
```

Hasil dari transformasi ini adalah representasi numerik dari teks, di mana setiap kata diberikan bobot tertentu. Bobot ini memungkinkan model untuk mengidentifikasi kata-kata yang lebih sering muncul di satu kelas namun jarang muncul di kelas lain, sehingga kata-kata tersebut menjadi lebih relevan dalam analisis sentimen.

#### b) Information Gain

Information Gain adalah teknik seleksi fitur yang digunakan untuk memilih kata atau fitur yang paling berpengaruh dalam membedakan kelas data. Teknik ini memungkinkan model untuk fokus pada kata-kata yang paling relevan

dengan label sentimen (seperti positif, negatif, atau netral) dan mengurangi gangguan dari fitur yang kurang signifikan. Dalam penelitian ini, Information Gain dihitung menggunakan metode `SelectKBest` dengan fungsi `chi2`, yang memilih 10 fitur teratas berdasarkan kontribusinya terhadap klasifikasi. Fitur-fitur ini dipilih dari hasil transformasi TF-IDF yang telah dilakukan sebelumnya. Berikut adalah kode yang digunakan untuk memilih fitur berdasarkan Information Gain:

```
from sklearn.feature_selection import SelectKBest, chi2

# Pilih fitur berdasarkan Information Gain
selector = SelectKBest(chi2, k=10) # Memilih 10 fitur terbaik berdasarkan nilai chi-square (Information Gain)
X_new = selector.fit_transform(X, data[label_column_name]) # Menggunakan kolom label yang benar
```

Setelah proses ini, daftar fitur yang dipilih berdasarkan Information Gain akan ditampilkan. Fitur-fitur ini mencakup kata-kata yang memiliki pengaruh signifikan dalam membedakan kelas sentimen. Dalam penelitian ini, fitur-fitur terpilih meliputi kata-kata seperti “ajb”, “bagus”, “banyak”, “buka”, “donlod”, “gak”, “iklan”, “jelek”, “minta”, “terbaikkkkkk”.

```
[33] # Mendapatkan nama fitur yang terpilih berdasarkan Information Gain
selected_features = tfidf.get_feature_names_out()[selector.get_support()] # Mendapatkan fitur yang relevan berdasarkan Information Gain
print("Fitur terpilih berdasarkan Information Gain:")
print(selected_features)
```

```
Fitur Terpilih Berdasarkan Information Gain:
['ajb' 'bagus' 'banyak' 'buka' 'donlod' 'gak' 'iklan' 'jelek' 'minta'
 'terbaikkkkkk']
```

Melalui penerapan metode TF-IDF dan Information Gain, data teks yang semula tidak terstruktur berhasil diubah menjadi representasi numerik yang lebih bermakna, dengan penekanan pada kata-kata yang memiliki bobot signifikan. Proses transformasi ini memungkinkan model untuk beroperasi dengan efisiensi lebih tinggi sekaligus meningkatkan tingkat akurasi dalam analisis sentimen pada tahap pemodelan.

#### 4.1.4 Pemodelan

Pada tahap pemodelan, penelitian ini mengadopsi algoritma Naive Bayes, khususnya dalam dua varian utama, yaitu Multinomial Naive Bayes. Algoritma ini dipilih karena terbukti efektif dalam klasifikasi teks, terutama untuk analisis sentimen, berkat kemampuannya dalam mengolah data berbasis frekuensi kata. Selanjutnya, evaluasi dilakukan terhadap kedua varian model tersebut guna membandingkan kinerjanya pada dataset ulasan yang telah melalui tahap pra-proses.

##### a) Naive Bayes Multinomial

Multinomial Naive Bayes sering digunakan untuk data yang merepresentasikan frekuensi kata, seperti jumlah kemunculan masing-masing kata dalam sebuah dokumen. Algoritma ini sangat efektif untuk data dengan nilai numerik yang menggambarkan frekuensi atau bobot kata, sehingga sering dimanfaatkan dalam berbagai tugas pemrosesan teks, seperti analisis sentimen, pengelompokan dokumen, atau pendeteksian spam.

Algoritma ini beroperasi dengan menghitung probabilitas posterior berdasarkan distribusi multinomial dari fitur-fitur masukan, yang biasanya disajikan dalam bentuk vektor frekuensi kata. Selain itu,

Multinomial Naive Bayes memiliki keunggulan dalam menangani dataset besar secara efisien dan mampu memberikan prediksi akurat dengan cepat, terutama ketika asumsi independensi antar fitur terpenuhi. Algoritma ini juga sangat ideal untuk data dengan perbedaan distribusi frekuensi kata yang signifikan antar kategori, sehingga mempermudah pengelompokan kelas.

```

from sklearn.metrics import accuracy_score, f1_score, precision_score, recall_score, classification_report
from sklearn.feature_extraction.text import TfidfVectorizer # Import TfidfVectorizer
# Assumikan 'label' adalah nama kolom label Anda
label_column_name = 'label'

# 1. Split data into training and testing sets
X = data['cleaned_text'] # Using the preprocessed text data
y = data[label_column_name] # Using the label column
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# 2. Create and train the Multinomial Naive Bayes model
tfidf = TfidfVectorizer() # Initialize TfidfVectorizer
X_train_tfidf = tfidf.fit_transform(X_train) # Fit and transform training data
X_test_tfidf = tfidf.transform(X_test) # Transform testing data

model = MultinomialNB()
model.fit(X_train_tfidf, y_train)

# 3. Make predictions on the test set
y_pred_multinomial = model.predict(X_test_tfidf)

# 4. Evaluate the model
print("Evaluasi Multinomial Naive Bayes")
print("Accuracy:", accuracy_score(y_test, y_pred_multinomial))
print("F1 Score:", f1_score(y_test, y_pred_multinomial, average='weighted', zero_division=1))
print("Precision:", precision_score(y_test, y_pred_multinomial, average='weighted', zero_division=1))
print("Recall:", recall_score(y_test, y_pred_multinomial, average='weighted', zero_division=1))
print("\nClassification Report:\n", classification_report(y_test, y_pred_multinomial, zero_division=1))
    
```

Hasil Evaluasi Model:

**Naive Bayes Multinomial**

Accuracy: 0.835  
 F1 Score: 0.7599182561307901  
 Precision: 0.862225  
 Recall: 0.835

Classification report menilai performa Multinomial Naive Bayes pada setiap kelas (negatif, netral, positif) melalui metrik precision, recall, dan F1-score. Model ini menunjukkan hasil yang baik, terutama pada kelas positif, mencerminkan kemampuan dalam mengidentifikasi kategori dengan akurasi tinggi dan keseimbangan antara presisi serta sensitivitas.

Berdasarkan hasil evaluasi ini, Multinomial Naive Bayes menunjukan

performa yang lebih baik. Model Multinomial menunjukkan nilai accuracy, precision, dan F1-score yang lebih tinggi, yang menunjukkan bahwa model ini lebih optimal untuk analisis sentimen berbasis frekuensi kata pada data ulasan Loklok.

Evaluasi Multinomial Naive Bayes				
Accuracy:	0.835			
F1 Score:	0.7599182561307901			
Precision:	0.862225			
Recall:	0.835			
Classification Report:				
	precision	recall	f1-score	support
negatif	1.00	0.00	0.00	21
netral	1.00	0.00	0.00	12
positif	0.83	1.00	0.91	167
accuracy			0.83	200
macro avg	0.94	0.33	0.30	200
weighted avg	0.86	0.83	0.76	200

**b) Visualisasi WordCloud**

Visualisasi Wordcloud digunakan untuk memberikan gambaran mengenai kata-kata yang paling sering muncul dalam setiap kategori sentimen, yaitu positif, netral, dan negatif. Teknik ini sangat berguna dalam analisis sentimen karena memungkinkan identifikasi pola kata atau istilah yang mendominasi pada setiap sentimen. Dalam visualisasi ini, kata-kata dengan frekuensi atau bobot lebih tinggi akan ditampilkan menggunakan ukuran font yang lebih besar, sementara kata-kata dengan frekuensi lebih rendah akan terlihat dengan ukuran lebih kecil, sehingga pola distribusi kata dapat dengan mudah diamati.

Pada tahap analisis ini, Wordcloud dihasilkan secara terpisah untuk setiap kategori sentimen dengan menggunakan pustaka WordCloud yang tersedia di Python. Visualisasi ini membantu memberikan wawasan mendalam tentang istilah-istilah utama yang memengaruhi klasifikasi sentimen. Dengan kata lain, Wordcloud mempermudah interpretasi data secara visual, menjadikannya alat yang efektif dalam memahami hasil analisis teks. Berikut adalah cuplikan kode yang digunakan untuk menghasilkan Wordcloud untuk masing-masing label sentimen:

```

from wordcloud import WordCloud

# Visualisasi WordCloud untuk masing-masing label (positif, netral, negatif)
for sentiment in ['positif', 'netral', 'negatif']:
    # Changed 'label' to 'label' to match the actual column name in the DataFrame
    # Ensure sentiment is lowercase for comparison as 'label' column values might be lowercase
    sentiment_text = ' '.join(data[data['label'].str.lower() == sentiment.lower()]['cleaned_text'])

    # Check if sentiment_text is empty and skip if so
    if not sentiment_text:
        print(f"Skipping word cloud for sentiment: {sentiment} - No text found.")
        continue

    wordcloud = WordCloud(width=800, height=400, background_color='white').generate(sentiment_text)

    plt.figure(figsize=(10, 5))
    plt.imshow(wordcloud, interpolation='bilinear')
    plt.title(f"WordCloud Sentimen {sentiment.capitalize()}")
    plt.axis('off')
    plt.show()
    
```

Penjelasan Kode:



ulasan berdasarkan hasil prediksi pada data uji. Dalam penelitian ini, Confusion Matrix digunakan sebagai metode utama untuk mengevaluasi akurasi model dengan membandingkan hasil prediksi terhadap data asli.

### Confusion Matrix

**Confusion Matrix** adalah alat evaluasi yang sering dipakai dalam klasifikasi untuk menunjukkan jumlah prediksi yang benar dan salah untuk setiap kelas dalam bentuk matriks. Alat ini membantu untuk mengidentifikasi kesalahan yang dilakukan oleh model serta menilai seberapa baik model mengenali masing-masing kategori sentimen. Matriks ini terdiri dari empat kategori hasil prediksi:

True Positive (TP): Prediksi yang benar untuk positif.

True Negative (TN): Prediksi yang benar untuk negatif.

False Positive (FP): Prediksi positif yang salah.

False Negative (FN): Prediksi negatif yang salah.

Confusion Matrix memberikan gambaran yang komprehensif tentang kinerja model dan memungkinkan perhitungan metrik evaluasi lainnya, seperti akurasi, presisi, recall, dan F1-score, yang berguna untuk menilai efektivitas model dalam klasifikasi sentimen. Berikut adalah kode yang digunakan untuk menghasilkan Confusion Matrix menggunakan model Multinomial Naive Bayes:

```
from sklearn.metrics import confusion_matrix
import seaborn as sns
import matplotlib.pyplot as plt
# Confusion Matrix untuk Multinomial Naive Bayes
conf_matrix_multinomial = confusion_matrix(y_test, y_pred_multinomial)
plt.figure(figsize=(6, 4))
sns.heatmap(conf_matrix_multinomial, annot=True, fmt='d', cmap='Blues')
plt.title('Confusion Matrix Multinomial NB')
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.show()
```

**Hasil Confusion Matrix:** Confusion Matrix menunjukkan distribusi hasil prediksi model untuk setiap kategori sentimen (0: negatif, 1: netral, 2: positif). Berikut adalah hasil Confusion Matrix yang diperoleh dari model Multinomial Naive Bayes:

Actual/ Predicted	0 (Negatif)	1 (Netral)	2 (Positif)
0 (Negatif)	0	0	21
1 (Netral)	0	0	12
2 (Positif)	0	0	167

- **Negatif** : Tidak ada ulasan negatif yang berhasil diklasifikasikan dengan benar. Sebanyak **21 ulasan negatif** salah diklasifikasikan sebagai positif.
- **Netral** : Semua ulasan netral salah diklasifikasikan, dengan **12 ulasan netral** diprediksi sebagai positif.
- **Positif** : Model secara eksklusif memprediksi ulasan sebagai positif, dengan **167 ulasan positif** berhasil diprediksi dengan benar.

### Analisis Hasil Evaluasi:

- **Dominasi Prediksi Positif:** Model cenderung mengklasifikasikan seluruh data ke dalam kategori positif, mengindikasikan bahwa model memiliki bias kuat terhadap sentimen positif.
- **Ketidakseimbangan Prediksi:** Tidak ada prediksi yang berhasil untuk kategori negatif maupun netral, yang menunjukkan bahwa model kesulitan mengenali pola dari kedua kategori ini.
- **Kemungkinan Penyebab:**

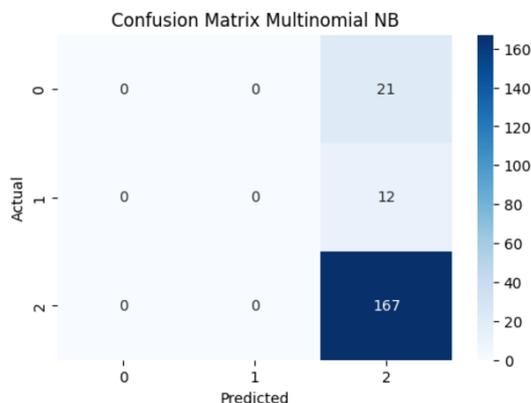
1. **Ketidakseimbangan Data:** Jika jumlah ulasan positif jauh lebih besar dibandingkan dengan kategori lainnya, model dapat terlatih untuk lebih memprioritaskan kategori ini.
2. **Kelemahan Algoritma Multinomial Naive Bayes:** Naive Bayes memiliki asumsi independensi fitur, yang seringkali tidak sesuai dalam data teks dengan pola yang kompleks.

- **Rekomendasi Perbaikan:**

1. Gunakan metode **oversampling** atau **undersampling** untuk menangani ketidakseimbangan data.
2. Gabungkan **algoritma hybrid**, seperti **Support Vector Machine (SVM)** atau **Random Forest**, untuk meningkatkan kemampuan klasifikasi pada kategori minoritas.
3. Lakukan optimasi preprocessing, termasuk

eksplorasi fitur yang lebih informatif, seperti **n-grams** atau **embedding** teks.

4.



Dengan menggunakan Confusion Matrix, penelitian ini memperoleh wawasan yang lebih mendalam mengenai kinerja model dalam klasifikasi sentimen. Confusion Matrix memudahkan identifikasi kelas-kelas yang sering salah diprediksi, memberikan dasar yang kuat untuk perbaikan lebih lanjut guna meningkatkan akurasi model, terutama pada sentimen-sentimen tertentu.

## 4. KESIMPULAN

### 5.1. Kesimpulan

Berdasarkan hasil penelitian yang dilakukan, diperoleh kesimpulan sebagai berikut:

- Peningkatan Performa Model Naïve Bayes untuk Analisis Sentimen**  
Penelitian ini berhasil meningkatkan kinerja model Naïve Bayes dalam analisis sentimen pada aplikasi Loklok. Peningkatan tersebut terukur melalui evaluasi menggunakan metrik akurasi, presisi, recall, dan F1-score, yang menunjukkan hasil lebih baik dibandingkan model baseline. Hal ini membuktikan bahwa pendekatan dan teknik yang digunakan dalam penelitian ini efektif untuk menghasilkan analisis sentimen yang lebih akurat dan optimal.
- Efektivitas Teknik Pemrosesan Data dan Pemilihan Fitur**  
Proses pemrosesan data yang mencakup pembersihan data, tokenisasi, normalisasi, dan stemming, serta penerapan metode pemilihan fitur

seperti TF-IDF, terbukti memberikan kontribusi signifikan terhadap peningkatan performa model Naïve Bayes. Teknik-teknik ini tidak hanya meningkatkan akurasi, tetapi juga mendukung model dalam mengklasifikasikan sentimen secara lebih tepat. Hasil ini menunjukkan pentingnya tahap pemrosesan data yang terstruktur dan seleksi fitur yang relevan dalam pengembangan model analisis sentimen.

- Manfaat Analisis Sentimen untuk Pemahaman Mendalam terhadap Opini Pengguna**

Model Naïve Bayes yang ditingkatkan berhasil menyediakan analisis sentimen yang lebih akurat, sehingga mampu menggambarkan opini pengguna terhadap aplikasi Loklok secara komprehensif. Wawasan yang diperoleh dari analisis ini sangat relevan bagi pengembang aplikasi untuk memahami kebutuhan, persepsi, dan ekspektasi pengguna. Selain itu, hasil ini memberikan informasi strategis yang dapat digunakan untuk meningkatkan kualitas layanan, kenyamanan pengguna, dan pengalaman keseluruhan dalam menggunakan aplikasi. Kesimpulan ini menegaskan bahwa penelitian telah berhasil menjawab permasalahan yang dirumuskan, mencapai tujuan penelitian yang telah ditetapkan, serta memberikan kontribusi nyata terhadap pengembangan metode analisis sentimen dan pemahaman lebih mendalam terhadap persepsi pengguna aplikasi.

### 5.2. Saran

Sebagai tindak lanjut dari penelitian ini, disarankan beberapa langkah berikut:

- Pengembangan Model Lebih Lanjut**  
Dalam upaya meningkatkan performa analisis sentimen, metode hybrid dapat dieksplorasi dengan menggabungkan Naïve Bayes dengan algoritma lain seperti Support Vector Machine (SVM) atau Deep Learning. Kombinasi ini bertujuan untuk memanfaatkan keunggulan masing-masing metode, seperti kemampuan

SVM dalam menangani data tidak seimbang atau kekuatan Deep Learning dalam mempelajari pola kompleks pada data teks. Pendekatan hybrid ini dapat menghasilkan model yang lebih akurat dan tahan terhadap variasi data, termasuk ulasan dengan konteks yang tidak terstruktur atau bahasa informal.

## 2. Analisis Multibahasa

Aplikasi Loklok memiliki basis pengguna global yang mencakup berbagai bahasa. Oleh karena itu, perluasan analisis dengan dukungan multibahasa pada tahap preprocessing dan pemodelan sangat penting. Hal ini melibatkan penyesuaian teknik tokenisasi, stemming, dan stopword removal agar sesuai dengan berbagai bahasa. Dengan mengintegrasikan kemampuan multibahasa, model akan lebih relevan dalam memahami sentimen pengguna dari berbagai negara, sehingga meningkatkan wawasan yang dapat diambil untuk pengembangan aplikasi.

## 3. Integrasi Feedback Real-Time

Sistem analisis sentimen real-time memungkinkan pengembang untuk secara langsung memantau ulasan pengguna dan mendeteksi perubahan persepsi secara dinamis. Hal ini penting untuk merespons keluhan atau saran dengan cepat, sehingga meningkatkan kepuasan pengguna. Dengan implementasi real-time, pengembang dapat mengotomatisasi pengumpulan data, analisis, dan pelaporan, yang tidak hanya menghemat waktu tetapi juga meningkatkan efisiensi pengambilan keputusan berbasis data.

## 4. Penelitian Lanjutan Penelitian

Penelitian serupa dapat diterapkan pada aplikasi lain yang memiliki karakteristik berbeda untuk mengevaluasi generalisasi model Naïve Bayes dalam berbagai konteks. Misalnya, pada aplikasi e-commerce, perbankan, atau media sosial, di mana pola ulasan dan kebutuhan pengguna berbeda. Pendekatan ini akan memberikan wawasan lebih luas tentang kekuatan dan kelemahan model, serta bagaimana menyesuaikan teknik analisis untuk domain tertentu.

5. Pendalaman Sentimen Negatif Analisis Sentimen negatif sering kali mencerminkan masalah teknis atau aspek aplikasi yang membutuhkan perbaikan. Analisis mendalam terhadap sentimen ini dapat mengidentifikasi solusi spesifik, seperti mengurangi iklan yang mengganggu, meningkatkan stabilitas aplikasi, atau menyempurnakan fitur tertentu.

Pendekatan ini melibatkan identifikasi kata kunci dalam ulasan negatif untuk menentukan area prioritas perbaikan, yang pada akhirnya dapat meningkatkan pengalaman pengguna dan citra aplikasi secara keseluruhan.

## DAFTAR PUSTAKA

- [1] P. Subarkah, S. A. Solikhatin, I. Darmayanti, A. N. Ikhsan, D. U. Hidayah, and R. M. Anjani, "Prediction of Education Level in Population Data Using Naïve Bayes Algorithm," *TIERS Inf. Technol. J.*, vol. 3, no. 2, pp. 69–75, 2022, doi: 10.38043/tiers.v3i2.3865.
- [2] Syahril Dwi Prasetyo, Shofa Shofiah Hilabi, and Fitri Nurapriani, "Analisis Sentimen Relokasi Ibukota Nusantara Menggunakan Algoritma Naïve Bayes dan KNN," *J. KomtekInfo*, vol. 10, pp. 1–7, 2023, doi: 10.35134/komtekinfo.v10i1.330.
- [3] N. Syafitri Kustanto, N. Gusriani, and Firdaniza, "Analisis Sentimen dengan Metode Klasifikasi Naïve Bayes Dan Seleksi Fitur Information Gain (Studi Kasus: Ulasan Aplikasi PeduliLindungi)," *In Search*, vol. 21, no. 02, pp. 134–144, 2022.
- [4] A. Hendra, "Analisis Sentimen Review Halodoc Menggunakan Naïve Bayes Classifier," vol. 6, no. 2, pp. 78–89, 2021.
- [5] M. K. Khoirul Insan, U. Hayati, and O. Nurdiawan, "Analisis Sentimen Aplikasi Brimo Pada Ulasan Pengguna Di Google Play Menggunakan Algoritma Naive Bayes," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 7, no. 1, pp. 478–483, 2023, doi: 10.36040/jati.v7i1.6373.
- [6] D. D. Putri, G. F. Nama, and W. E. Sulistiono, "ANALISIS SENTIMEN KINERJA DEWAN PERWAKILAN RAKYAT ( DPR ) PADA TWITTER MENGGUNAKAN METODE NAIVE BAYES CLASSIFIER," vol. 10, no. 1, pp. 34–40, 2022.
- [7] D. Normawati and S. A. Prayogi, "Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter," *J. Sains Komput. Inform. (J-SAKTI)*, vol. 5, no. 2, pp. 697–711, 2021.
- [8] L. Budi and A. Mude, "Perbandingan Metode Klasifikasi Support Vector Machine dan Naïve Bayes untuk Analisis Sentimen pada Ulasan Tekstual di Google Play Store," vol. 12, no. 2, pp. 154–161, 2020.
- [9] S. Tanahku, "Analisis Sentimen Aplikasi E-Government Pada Google Play Menggunakan Algoritma Naïve Bayes," vol. 9, no. 2, pp. 785–795, 2022.
- [10] W. Yulita, "Analisis Sentimen Terhadap Opini

- Masyarakat Tentang Vaksin Covid-19 Menggunakan Algoritma Naïve Bayes Classifier,” *J. Data Min. dan Sist. Inf.*, vol. 2, no. 2, p. 1, 2021, doi: 10.33365/jdmsi.v2i2.1344.
- [11] D. Darwis, N. Siskawati, and Z. Abidin, “Penerapan Algoritma Naive Bayes Untuk Analisis Sentimen Review Data Twitter Bmkg Nasional,” *J. Tekno Kompak*, vol. 15, no. 1, p. 131, 2021, doi: 10.33365/jtk.v15i1.744.
- [12] W. Wahyuni, “Analisis Sentimen terhadap Opini Feminisme Menggunakan Metode Naive Bayes,” *J. Inform. Ekon. Bisnis*, vol. 4, pp. 148–153, 2022, doi: 10.37034/infeb.v4i4.162.
- [13] G. Darmawan, S. Alam, and M. I. Sulisty, “Analisis Sentimen Berdasarkan Ulasan Pengguna Aplikasi My Pertamina Pada Google Playstore Menggunakan Metode Naïve Bayes,” *STORAGE – J. Ilm. Tek. dan Ilmu Komput.*, vol. 2, no. 3, pp. 100–108, 2023.