

# Analisis Sentimen Terhadap Kualitas Pelayanan Aplikasi In-Drive Menggunakan Metode Naive Bayes Classifier

Muhammad Sidiq Bagus Prakoso<sup>1</sup>, Isa Faqihuddin Hanif<sup>2,\*</sup>

<sup>1</sup> Fakultas Teknologi Industri dan Informatika, Teknik Informatika, Universitas Muhammadiyah Prof. Dr. Hamka, Jakarta, Indonesia

<sup>2</sup> Fakultas Teknologi Industri dan Informatika, Sistem Teknologi Informasi, Universitas Muhammadiyah Prof. Dr. Hamka, Jakarta, Indonesia

Email: <sup>1</sup>sdqbs05@email.com, <sup>2,\*</sup>isa@uhamka.ac.id

Email Penulis Korespondensi: isa@uhamka.ac.id

Submitted: 11/01/2025; Accepted: 15/03/2025; Published: 16/03/2025

**Abstrak**—Penelitian ini menganalisis sentimen pengguna terhadap kualitas layanan aplikasi *In-Drive* menggunakan metode *Naive Bayes Classifier*. Sebanyak 15.000 ulasan dari Google Play Store dikumpulkan menggunakan teknik *web scraping* dari hasil analisis sentimen terhadap data tersebut, ditemukan 9.665 sentimen negatif dan 5.335 sentimen positif. Data tersebut melalui tahap *pre-processing* termasuk *cleaning*, *case folding*, *stopword removal*, *tokenizing*, dan *stemming*. Algoritma Naive Bayes digunakan untuk mengklasifikasikan ulasan menjadi sentimen positif dan negatif. Evaluasi menggunakan matriks kebingungan menghasilkan akurasi 76,56%, presisi 78,26%, *recall* 87,69%, dan skor F1 82,71%. Hasil ini menunjukkan bahwa sebagian besar ulasan bersifat negatif. Penelitian ini diharapkan dapat membantu pengembang aplikasi *In-Drive* memahami pengalaman pengguna dan meningkatkan kualitas layanan berdasarkan ulasan yang tersedia secara otomatis.

**Kata Kunci:** Analisis Sentimen; Naive Bayes; Web Scraping; Aplikasi In-Drive; Matrix Confusion

**Abstract**—This research analyzes user sentiment towards the service quality of the In-Drive application using the Naive Bayes Classifier method. A total of 15,000 reviews from the Google Play Store were collected using web scraping techniques from the results of sentiment analysis of the data, 9,665 negative sentiments and 5,335 positive sentiments were found. The data went through a pre-processing stage including cleaning, case folding, stopword removal, tokenizing, and stemming. Naive Bayes algorithm was used to classify the reviews into positive and negative sentiments. Evaluation using the confusion matrix resulted in 76.56% accuracy, 78.26% precision, 87.69% recall, and 82.71% F1 score. These results indicate that most reviews are negative. This research is expected to help In-Drive app developers understand user experience and improve service quality based on automatically available reviews.

**Keywords:** Sentiment Analysis; Naive Bayes; Web Scraping; In-Drive Application; Confusion Matrix

## 1. PENDAHULUAN

Peradaban manusia berkembang pesat di era modern, dan sejak revolusi industri 4.0, manusia harus lebih mudah menerima dan fleksibel terhadap kemajuan teknologi[1]. Semakin populernya transportasi online melalui media internet merupakan salah satu konsekuensi dari kemajuan teknologi[2]. Orang-orang memanfaatkan teknologi untuk membantu tugas-tugas harian mereka. Misalnya, mereka menggunakan program online untuk membeli makanan dan mengatur transportasi[3]. Terdapat persaingan ketat karena banyaknya penyedia transportasi internet yang menyediakan berbagai layanan. Aplikasi In-Drive adalah salah satu opsi transportasi online yang digunakan oleh masyarakat Indonesia[4]. Melalui Google Play Store, pengguna ponsel pintar Android dapat mengunduh aplikasi In-Drive dan produk digital lainnya, seperti game, film, e-book, dan banyak lagi, secara gratis atau berbayar. Konsumen potensial dapat dibujuk untuk menggunakan suatu aplikasi oleh ulasan pengguna di Google Play. Namun, pemrosesan manual menjadi sulit ketika ada begitu banyak ulasan yang tersedia. Oleh karena itu, diperlukan cara untuk menilai kecenderungan pengguna secara otomatis terhadap aplikasi baik yang menguntungkan maupun yang tidak menguntungkan[5].

Analisis sentimen, atau yang juga dikenal sebagai opinion mining, adalah proses pengolahan data tekstual secara otomatis untuk menentukan apakah opini yang terkandung dalam sebuah kalimat bersifat positif atau negatif[6]. Analisis ini mencoba mengkategorikan polaritas perasaan orang dan mengidentifikasi pemikiran dan sentimen yang mereka ungkapkan sebelum mengotomatisasi prosesnya. Sejak awal tahun 2000-an, analisis sentimen telah muncul sebagai salah satu bidang studi yang paling aktif dalam pemrosesan bahasa alami, dan bisnis ini juga berkembang dengan cepat. Analisis sentimen sekarang digunakan di berbagai bidang, termasuk riset pasar, perawatan kesehatan, bisnis, pemilihan umum, analisis produk, dan lain-lain[7].

Tantangan yang dihadapi penelitian ini adalah bagaimana menginterpretasikan analisis sentimen yang telah dilakukan. Kami juga ingin mengetahui bagaimana analisis tersebut dapat membantu meningkatkan kualitas aplikasi atau layanan, dan pengembang aplikasi In-Drive akan merasa sangat terbantu dengan adanya ulasan ini. Pembuat aplikasi In-Drive dapat belajar tentang kekuatan dan kelemahan aplikasi dari ulasan ini. Ulasan ini juga dapat digunakan oleh pengembang untuk membuat perbaikan baru untuk aplikasi. Namun, memilah-milah ratusan ribu ulasan itu perlu. Hal ini sulit dan membutuhkan sistem yang dapat membantu mengatur ulasan[8]. Kerumitan bahasa asli, perbedaan gaya penulisan, dan konteks tertentu dari ulasan memberikan tantangan tersendiri. Menemukan emosi yang tepat sangat penting pada saat konsumen semakin bergantung pada ulasan sebagai sumber informasi utama

mereka saat memilih aplikasi. Untuk menjadi dasar bagi terciptanya teknik analisis sentimen yang lebih canggih di bidang informatika, penelitian ini menyarankan penerapan Algoritma Naive Bayes sebagai solusi yang efisien[9].

Untuk klasifikasi sentimen, Naïve Bayes dipilih. Thomas Bayes, seorang ilmuwan Inggris, menetapkan Teorema Bayes, yang dimodifikasi dalam teknik ini[10]. Dalam hal kategorisasi teks, Naïve Bayes adalah metode yang mudah dan efektif. Dalam analisis sentimen, metode Naïve Bayes telah terbukti bermanfaat[11]. Prosedur klasifikasi teks dari metode klasifikasi Naive Bayes dibagi menjadi dua tahap. Langkah pertama adalah analisis dokumen sampel, termasuk pemilihan kosakata-yaitu, bahasa yang dapat dimasukkan ke dalam koleksi dokumen sampel-dan langkah kedua adalah perhitungan probabilitas untuk setiap kategori dengan menggunakan dokumen sampel. Langkah kedua adalah klasifikasi, yang memberikan nilai kategori dokumen tergantung pada bahasa yang digunakan dalam teks yang diklasifikasikan[12].

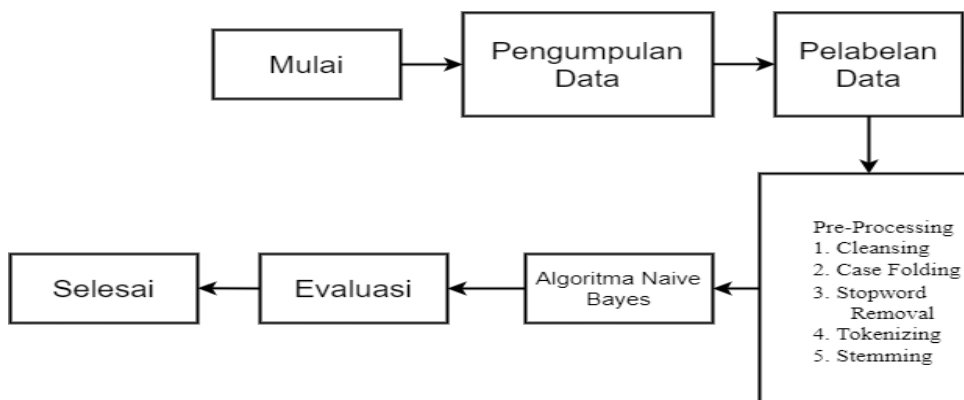
Penelitian sebelumnya menunjukkan bahwa salah satu pendekatan algoritma naïve bayes dapat digunakan untuk mengklasifikasikan analisis sentimen secara tepat. Tujuan dari penelitian “Analisis Sentimen Ulasan Aplikasi Lita Di Play Store Menggunakan Algoritma Naive Bayes” adalah untuk menerapkan metode Naive Bayes dalam menganalisis sentimen pengguna terhadap aplikasi Lita. Melalui Google Play Store, pengguna ponsel pintar Android dapat mengunduh aplikasi dan produk digital lainnya, seperti game, film, e-book, dan banyak lagi, baik secara gratis maupun berbayar. Konsumen potensial dapat dibujuk untuk menggunakan suatu aplikasi oleh ulasan pengguna di Google Play. Namun, pemrosesan manual menjadi sulit ketika ada begitu banyak ulasan yang dapat diakses. Oleh karena itu, diperlukan cara untuk menilai kecenderungan pengguna secara otomatis terhadap aplikasi-apakah menguntungkan atau tidak menguntungkan[13].

Selain itu, analisis sentimen dapat digunakan untuk memeriksa opini publik tentang feminisme. Sebagai contoh, penelitian Widya Wahyuni berjudul “Analisis Sentimen terhadap Opini Feminisme Menggunakan Metode Naive Bayes” akan memberikan gambaran tentang seberapa tinggi atau rendahnya feminisme yang berkembang di Indonesia. Dengan akurasi 87% ketika diterapkan pada 600 sampel, pendekatan Naïve Bayes dapat dianggap berkinerja baik dalam hal analisis sentimen[14]. Penelitian sebelumnya tentang analisis sentimen dalam aplikasi perjalanan dan transportasi sebagian besar berfokus pada pemeriksaan peringkat atau metrik umum lainnya. Penelitian tentang analisis sentimen dalam aplikasi perjalanan, seperti In-Drive masih kurang, terutama bagi mereka yang menggunakan pendekatan Naïve Bayes untuk mendapatkan pemahaman yang lebih dalam tentang ulasan.

Pengalaman konsumen adalah elemen kunci keberhasilan di era digital yang sangat kompetitif. Ulasan buruk yang tidak dijawab dapat menyebabkan turunnya kepercayaan konsumen dan hilangnya pendapatan prospektif. Seiring dengan meningkatnya jumlah evaluasi, In-Drive membutuhkan metode yang otomatis dan efisien untuk menganalisis sentimen pengguna. Tujuan utama analisis sentimen ini adalah untuk membantu In-Drive lebih memahami sikap dan pandangan konsumen tentang produk, layanan, dan fitur yang mereka tawarkan. Hal ini dapat membantu pengambilan keputusan yang lebih baik untuk strategi pemasaran, peningkatan kualitas layanan, dan perbaikan produk. In-Drive juga dapat bereaksi lebih cepat terhadap masalah dan komentar pelanggan dengan mengetahui sudut pandang mereka. Selain itu, amati akurasi, presisi dan recall program In-Drive serta kemampuan pemrosesan data menggunakan metode algoritma Naïve Bayes.

## 2. METODOLOGI PENELITIAN

Tujuan utama dari penelitian ini adalah untuk mengklasifikasikan opini yang berasal dari rating yang telah dikumpulkan. Pendekatan ini menggunakan naive bayes untuk mengklasifikasikan sentimen secara akurat. mulai dari mengumpulkan data ulasan hingga menggunakan Google Colab untuk menyelesaikan proses kategorisasi. Pada Gambar 1, prosedur penelitian ditunjukkan.



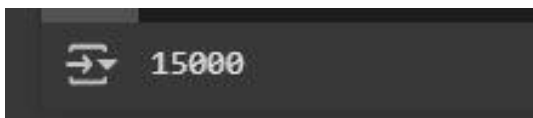
**Gambar 1.** Tahapan Penelitian

Pada tahapan penelitian dalam Gambar 1, mulai dari mengumpulkan data ulasan pengguna aplikasi *In-drive* di *Google Play Store* menggunakan teknik *scrapping* di *Google Colab*. Setelan itu, data akan dilabeli secara otomatis. Kemudian melakukan tahap *processing*, seperti *cleaning*, *case folding*, *stopwoord removal*, *tokenizing* dan *stemming*,

yang setelah itu melakukan pembobotan TF-IDF. Setelah proses selesai, data yang telah di implementasikan menggunakan metode *Naive Bayes*. Tahap akhir dari penelitian ini adalah evaluasi menggunakan *confusion matrix* untuk mengukur akurasi, presisi, recall, dan *F1-Score*.

### 2.1 Pengumpulan Data

Data yang termasuk dalam penelitian ini berasal dari ulasan pengguna aplikasi *In-Drive* di *Google Play*. *Data scraping* digunakan untuk mendapatkan data untuk penelitian ini. Informasi tersebut didasarkan pada evaluasi dari daerah yang dipublikasikan dalam bahasa Indonesia. Metode pengumpulan data ditentukan berdasarkan komentar yang paling relevan menggunakan *Google Colab*. Setelah proses *scraping* selesai, data disimpan dan diformat ke dalam file *CSV* pada tanggal 11 Desember 2024, setelah *scraping* data menggunakan *Google Colab* diikuti [15].

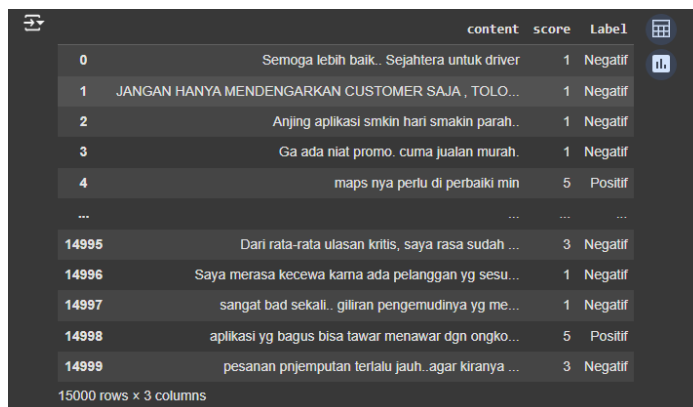


**Gambar 2.** Proses Pengumpulan Data

Pada gambar 2 menjelaskan pengumpulan data menggunakan *scraping* data ulasan sebanyak 15.000 data pada aplikasi *In-Drive* di *Google Play Store*, setelah data ulasan diambil dalam bahasa Indonesia menggunakan *Google Colab*.

### 2.2 Pelabelan Data

Pelabelan data dilakukan pada data yang dikumpulkan untuk penelitian ini akan dipilih dengan mengklasifikasikannya ke dalam kategori positif dan negatif sesuai dengan penilaian yang diberikan pengguna terhadap aplikasi. Penilaian yang mengekspresikan kepuasan, apresiasi, dan kritik yang membantu dan membangun terhadap aplikasi dianggap memiliki sentimen yang baik, sedangkan penilaian yang mengekspresikan sarkasme, kata-kata kotor, atau keluhan terhadap aplikasi dianggap memiliki emosi negatif[16]. Seperti yang ada di gambar 3.



	content	score	Label
0	Semoga lebih baik. Sejahtera untuk driver	1	Negatif
1	JANGAN HANYA MENDENGARKAN CUSTOMER SAJA, TOLO...	1	Negatif
2	Anjing aplikasi smkin hari smakin parah...	1	Negatif
3	Ga ada niat promo. cuma jualan murah.	1	Negatif
4	maps nya perlu di perbaiki min	5	Positif
...	...	...	...
14995	Dari rata-rata ulasan kritis, saya rasa sudah ...	3	Negatif
14996	Saya merasa kecewa karna ada pelanggan yg sesu...	1	Negatif
14997	sangat bad sekali.. giliran pengemudinya yg me...	1	Negatif
14998	aplikasi yg bagus bisa tawar menawar dgn ongko...	5	Positif
14999	pesanan pnjemputan terlalu jauh..agar kiranya ...	3	Negatif

15000 rows x 3 columns

**Gambar 3.** Pelabelan Data

Pada Gambar 3 menjelaskan pelabelan data menggunakan pelabelan *score* 1-3 menunjukkan perasaan negatif, sedangkan *score* 4-5 menunjukkan sentimen yang baik

### 2.3 Pre-processing

Pada tahap ini data yang terkumpul melewati langkah-langkah yang diselesaikan dalam langkah prapemrosesan ini. Prosedur-prosedur ini digunakan untuk membersihkan dan menghapus data dari tanda baca dan simbol yang tidak perlu, termasuk tanda tanya, tanda seru, koma, titik, dan lain-lain. Pada tahap ini, ia juga dapat mengubah semua huruf data menjadi huruf kecil, membagi kalimat menjadi beberapa segmen kata, menghilangkan stopwords, dan mengubah kata-kata yang dilampirkan menjadi kata-kata sederhana[17]. Langkah-langkah pada *pre-processing* yaitu *Cleaning*, *Case Folding*, *Stopword Removal*, *Tokenizing*, dan *Stemming*.

### 2.4 Algoritma Naive Bayes

Teknik klasifikasi yang dikenal sebagai *Naive Bayes Classifier (NBC)* didasarkan pada *Teorema Bayes*. Setiap karakteristik atau variabel dipandang independen dari setiap karakteristik atau variabel lainnya dalam konteks “Naive”. NBC membuat asumsi bahwa keberadaan sebuah fitur dalam sebuah kelas tidak ada hubungannya dengan fitur lain dalam kelas yang sama. Data pelatihan awal diperlukan bagi NBC untuk membuat penilaian di seluruh fase pembelajaran karena NBC adalah algoritma pembelajaran yang diawasi. Berdasarkan input, kemungkinan setiap label kelas ditentukan selama langkah klasifikasi. Untuk data yang disediakan, label kelas yang memiliki probabilitas tertinggi akan digunakan[18].

$$P(X|Y) = \frac{p(Y|X \cdot x)}{p(y)} \tag{1}$$

Rumus untuk Naive Bayes  $P(Y|X)$  mewakili kemungkinan bahwa  $X$  akan benar,  $P(X)$  mewakili kemungkinan bukti penyakit  $X$ ,  $P(Y)$  mewakili kemungkinan nilai  $Y$ , dan  $P(X|Y)$  mewakili probabilitas posterior  $l$ , atau kemungkinan bahwa  $X$  akan benar dengan kondisi  $Y$ .

### 2.5 Evaluasi

Tahap evaluasi yang merupakan langkah terakhir dalam proses penelitian ini, bertujuan untuk menjamin hasil dari model. *Confusion matrix* digunakan dalam penelitian ini untuk menilai kinerja model. *Confusion matrix* terdiri dari empat bagian: FP (False Positive), FN (False Negative), TN (True Negative), dan TP (True Positive). Akurasi, presisi, dan recall harus dihitung untuk menentukan nilai *confusion matrix*[19].

Tabel 1. *Confusion Matrix*

	Predict	
	Positif	Negatif
Positif	True Positif (TP)	False Positif (FN)
Negatif	False Negatif (FN)	True Negatif (TN)

Pada Tabel 1 yaitu *confusion matrix* yaitu *True Positif (TP)*, *True Negatif (TN)*, *False Positif (FP)*, *False Negatif (FN)*, *Recall*, *akurasi*, *presisi*, dan nilai F-1 semuanya dapat dihitung menggunakan informasi dalam *confusion matrix*. Dengan membagi jumlah total sampel dengan jumlah prediksi yang akurat, seseorang dapat menghitung akurasi klasifikasi model. Nilai akurasi yang tinggi menunjukkan lebih sedikit nilai positif palsu, presisi mengukur tingkat kesesuaian antara hasil prediksi model dan data yang diperlukan. Nilai negatif palsu yang rendah ditunjukkan oleh skor *recall* yang baik. Rasio pengamatan positif yang diprediksi secara akurat terhadap semua pengamatan di kelas yang sebenarnya dikenal sebagai *recall*. Skor presisi dan memori digabungkan untuk memberikan satu metrik yang dikenal sebagai rata-rata harmonik tertimbang dari *akurasi* dan *recall*, atau skor F1[20].

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{2}$$

$$Precision = \frac{TP}{TP+FP} \tag{3}$$

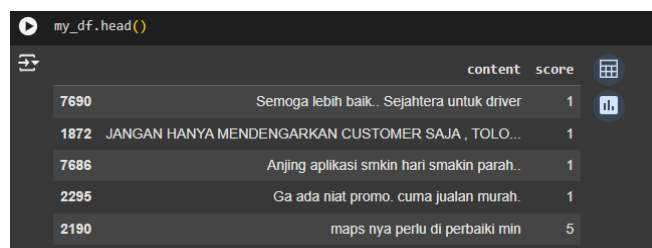
$$Recall = \frac{TP}{TP+FN} \tag{4}$$

## 3. HASIL DAN PEMBAHASAN

Bab ini akan membahas studi dan tujuan penggunaan teknik Naive Bayes untuk memastikan pendapat rata-rata pengguna terhadap aplikasi di platform Play Store. menyadari bagaimana pengguna menilai dan menanggapi pengalaman aplikasi In-Dirve mereka. Dan mengklasifikasikan evaluasi sebagai positif atau negatif adalah tujuan utama dari penelitian ini.

### 3.1 Pengumpulan Data

Web scraping adalah proses pengumpulan data ulasan menggunakan Google Colab. Tahap web scraping dilakukan pada 11 Desember 2024 dengan mengumpulkan informasi dari 15.000 komentar ulasan dari aplikasi In-Drive di Google Play Store. Proses web scraping adalah sumber dari semua temuan pengumpulan data yang baik maupun yang buruk.

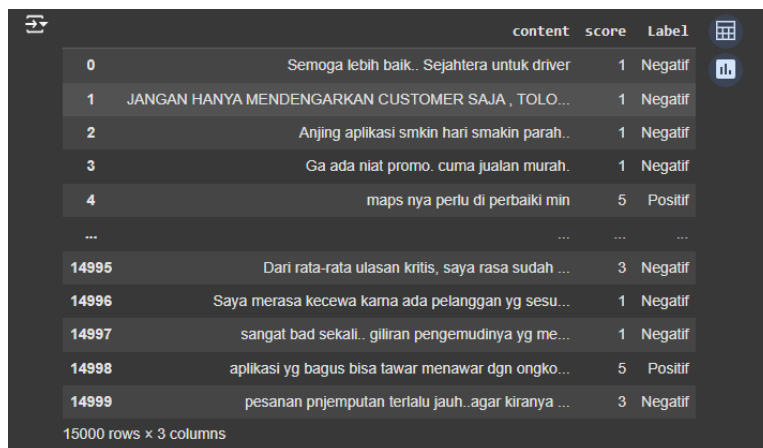


Gambar 4. Pengumpulan Data dari *Google Colab*

Gambar 4 menunjukkan bahwa hingga 15.000 data ulasan dikumpulkan menggunakan *Google Colab*.

### 3.2 Pelabelan Data

Pelabelan digunakan untuk membedakan antara evaluasi produk yang baik dan negatif. Gambar 5 di bawah ini akan menampilkan pelabelan dataset yang digunakan dalam penelitian ini.



	content	score	Label
0	Semoga lebih baik. Sejahtera untuk driver	1	Negatif
1	JANGAN HANYA MENDENGARKAN CUSTOMER SAJA, TOLO...	1	Negatif
2	Anjing aplikasi smkin hari smakin parah..	1	Negatif
3	Ga ada niat promo. cuma jualan murah.	1	Negatif
4	maps nya perlu di perbaiki min	5	Positif
...	...	...	...
14995	Dari rata-rata ulasan kritis, saya rasa sudah ...	3	Negatif
14996	Saya merasa kecewa kama ada pelanggan yg sesu...	1	Negatif
14997	sangat bad sekali.. giliran pengemudinya yg me...	1	Negatif
14998	aplikasi yg bagus bisa tawar menawar dgn ongko...	5	Positif
14999	pesanan pnjemputan terlalu jauh.. agar kiranya ...	3	Negatif

Gambar 5. Pelabelan data ulasan sentimen positif dan negatif pada aplikasi In-Drive

Gambar 5 menunjukkan pelabelan data dari ulasan In-Dirve menggunakan *Google Colab* untuk menghasilkan perasaan positif dan negatif.

### 3.3 Pre-Processing

Persiapan data selesai pada titik ini. Setelah data yang terkumpul dibersihkan untuk menghilangkan duplikasi, pendekatan pretreatment dataset akan digunakan untuk evaluasi aplikasi In-Drive yang menggabungkan data sentimen menggunakan *Google Colab*. *Pre-processing* dataset meliputi *case folding*, *stopword removal*, *tokenizing*, dan *stemming*.

- a. *Case folding* adalah proses mengubah semua huruf dalam teks menjadi huruf kecil. Tujuannya adalah untuk menghilangkan perbedaan antara huruf besar dan huruf kecil. Langkah pertama adalah memuat pustaka *impor re*, modul Python bawaan yang menyediakan kemampuan kerja ekspresi reguler. Prosedur pelipatan huruf ditunjukkan pada Tabel 2 mengilustrasikan pemrosesan dataset dalam *case folding*.

Tabel 2. Hasil *Case Folding*

Sebelum <i>Case Folding</i>	Sesudah <i>Case Folding</i>
Semoga lebih baik Sejahtera untuk driver	'semoga', 'lebih', 'baik', 'sejahtera', 'untuk', 'driver'
JANGAN HANYA MENDENGARKAN CUSTOMER SAJA TOLONG	'jangan', 'hanya', 'mendengarkan', 'customer', 'saja', 'tolong'
Ga ada niat promo cuma jualan murah maps nya perlu di perbaiki min	'ga', 'ada', 'niat', 'promo', 'cuma', 'jualan', 'murah', 'maps', 'nya', 'perlu', 'di', 'perbaiki', 'min'

- b. *Stopword Removal* adalah proses menghilangkan kata-kata umum yang sering digunakan dalam suatu bahasa, seperti “dan”, “atau”, “juga”, “di”, “dapat”, dan sebagainya. Kata-kata ini biasanya tidak memiliki arti khusus dan tidak secara signifikan mengubah pemahaman kita terhadap konteks atau nada dari materi yang kita pelajari. Langkah pertama, mengimpor *nlk.corpus* di *Python*, yang mengimpor modul *nlk.corpus* dari National Language (NLTK). NLTK adalah paket pemrosesan bahasa alami (NLP) Python yang terkenal. Model NLP dapat dilatih dan diuji menggunakan korpus teks ini, yang merupakan kumpulan data teks untuk eksplorasi dan analisis teks. Selain itu, unduh *nlk.download ('stopwords')* untuk mengakses Bahasa Indonesia. Tabel 3 di bawah ini mengilustrasikan hasil dari pra-pemrosesan *Stopword Removal*.

Tabel 3. Hasil *Stopword Removal*

Sebelum <i>Stopword Removal</i>	Sesudah <i>Stopword Removal</i>
'semoga', 'lebih', 'baik', 'sejahtera', 'untuk', 'driver'	'semoga', 'sejahtera', 'driver'
'jangan', 'hanya', 'mendengarkan', 'customer', 'saja', 'tolong'	'mendengarkan', 'customer', 'tolong'
'ga', 'ada', 'niat', 'promo', 'cuma', 'jualan', 'murah'	'ga', 'niat', 'promo', 'jualan', 'murah'
'maps', 'nya', 'perlu', 'di', 'perbaiki', 'min'	'maps', 'perbaiki', 'min'

- c. *Tokenizing*, Dengan menggunakan teknik tokenisasi, teks dibagi menjadi unit-unit terpisah yang disebut token. Token dapat berupa kata, frasa, atau bahkan karakter individual. Impor dari *nlk.tokenize* digunakan dalam proses tokenisasi untuk memecah teks menjadi potongan-potongan yang dapat dikelola. Hasil dari *Tokenizing* ditunjukkan pada Tabel 4 di bawah ini.

Tabel 4. Hasil *Tokenizing*

Sebelum <i>Tokenizing</i>	Sesudah <i>Tokenizing</i>
'semoga', 'sejahtera', 'driver'	'semoga', 'sejahtera', 'driver'

'mendengarkan', 'customer', 'tolong'	'mendengarkan', 'customer', 'tolong'
'ga', 'niat', 'promo', 'jualan', 'murah'	'ga', 'niat', 'promo', 'jualan', 'murah'
'maps', 'perbaiki', 'min'	'maps', 'perbaiki', 'min'

d. *Stemming*, mengembalikan kata-kata dalam teks ke bentuk yang paling dasar, atau kata dasar, adalah proses *stemming*. Tujuannya adalah untuk mengurangi frasa yang berbeda ke bentuk dasar yang sama. Kode pemrograman Python, dan pemrosesan *stemming* bahasa Indonesia kita dapat menggunakan library Python Sastrawi pada dataset. Informasi sebelum pemrosesan dan hasil akhir setelah proses *stemming* disajikan pada Tabel 5 berikut.

Tabel 5. Hasil *Stemming*

Sebelum <i>Stemming</i>	Sesudah <i>Stemming</i>
'semoga', 'sejahtera', 'driver'	'semoga', 'sejahtera', 'driver'
'mendengarkan', 'customer', 'tolong'	'dengar', 'customer', 'tolong'
'ga', 'niat', 'promo', 'jualan', 'murah'	'ga', 'niat', 'promo', 'jual', 'murah'
'maps', 'perbaiki', 'min'	'maps', 'perbaiki', 'min'

### 3.4 Naïve Bayes Classifier

Setelah dataset berhasil menyelesaikan langkah *preprocessing*, peneliti menerapkan *splitting data* dan pembobotan TF-IDF.

a. *Splitting data*, Teknik *splitting data* adalah proses membagi data menjadi beberapa subset untuk tujuan pengembangan dan evaluasi model pembelajaran mesin. Proses *splitting data* memisahkan dataset asli menjadi dua atau lebih bagian yang saling terpisah dikenal sebagai *splitting data*. Model dilatih atau dikembangkan menggunakan satu bagian dataset, dan kinerjanya diuji atau dievaluasi menggunakan bagian yang tersisa. Gambar 6 di bawah ini menunjukkan dataset yang akan diuji oleh peneliti.

```
[112] #membagi data menjadi data training dan testing dengan test_size = 0.20 dan random state nya 0
      from sklearn.model_selection import train_test_split
      X_train, X_test, y_train, y_test = train_test_split(data_clean['content'], data_clean['Label'],
                                                    test_size = 0.20,
                                                    random_state = 0)
```

Gambar 6. *Splitting Data*

Peneliti akan mempartisi data uji menjadi 20% dari total data, seperti yang ditunjukkan pada Gambar 6. Dalam hal ini, data pelatihan dan pengujian akan dipisahkan menggunakan `test_size = 0.20` dan `random state = 0`.

b. Pembobotan *Term Frequency-Inverse Document Frequency* (TF-IDF) adalah sebuah metode yang digunakan dalam pemrosesan bahasa alami (NLP) untuk mengatur nilai kata dalam sebuah dokumen atau kumpulan dokumen. Pembobotan TF-IDF ditampilkan pada Gambar 7.

```
(12000,)
(12000,)
(3000,)
(3000,)
```

Gambar 7. Hasil Pembobotan TF-IDF

Pada Gambar 7, menunjukkan pembobotan TF-IDF dengan library *sklearn*. Modul *tfidf\_vectorizer* digunakan oleh *feature\_extraction.txt* untuk menerjemahkan teks ke dalam modul *tfidf*. Variabel *syntax x\_train* dan *x\_test* diisi dengan data teks. *Tfidf vectorizer* akan mengidentifikasi nilai *tfidf* setiap kata atau dokumen dan kemudian menghasilkan vektor matriks yang menunjukkan pemrograman dalam sektor numerik.

### 3.5 Evaluasi

Penilaian model adalah proses mengevaluasi kinerja model pembelajaran mesin menggunakan data uji yang belum pernah digunakan sebelumnya. Kemampuan model untuk meramalkan atau mengkategorikan data yang belum pernah dilihat sebelumnya disebut sebagai penilaian model. Sintaks pemrograman Python dataset pada Gambar 8 berikut.

```

import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix

clf = MultinomialNB()
clf.fit(X_train, y_train)
predicted = clf.predict(X_test)

print("MultinomialNB Accuracy:", accuracy_score(y_test,predicted))
print("MultinomialNB Precision:", precision_score(y_test,predicted, average="binary", pos_label="Negatif"))
print("MultinomialNB Recall:", recall_score(y_test,predicted, average="binary", pos_label="Negatif"))
print("MultinomialNB f1_score:", f1_score(y_test,predicted, average="binary", pos_label="Negatif"))

print(f'confusion_matrix:\n {confusion_matrix(y_test, predicted)}')
print('=====\n')
print(classification_report(y_test, predicted, zero_division=0))

# Load dataset
data_clean = pd.read_csv('hasil_TextPreProcessing_InDrive.csv')
    
```

Gambar 8. Proses Evaluasi Model

*Accuracy\_score* diimpor ke dalam *syntax sklearn.metrics* pada Gambar 8 di atas. Gunakan *accuracy=accuracy\_Score(y\_test, y\_pred)* untuk memverifikasi dan menyimpan skor akurasi pada dataset. Nilai performa model pada gambar di atas dihitung dengan menggunakan beberapa metrik, seperti *accuracy\_score*, *precision\_score*, *recall\_score*, dan *f1\_score*.

```

MultinomialNB Accuracy: 0.7656666666666667
MultinomialNB Precision: 0.7826896230805026
MultinomialNB Recall: 0.8769551616266945
MultinomialNB f1_score: 0.8271453159577083
confusion_matrix:
[[1682  236]
 [ 467  615]]
=====
              precision    recall  f1-score   support

   Negatif      0.78      0.88      0.83      1918
   Positif      0.72      0.57      0.64      1082

 accuracy              0.77      3000
 macro avg      0.75      0.72      0.73      3000
 weighted avg    0.76      0.77      0.76      3000
    
```

Gambar 9. Tampilan Hasil Naive Bayes

Berdasarkan perhitungan *confusion matrix*, akurasi, presisi, recall, dan F1-score dari hasil Naive Bayes dari kategorisasi data ulasan pengguna pada aplikasi In-Drive berturut-turut adalah 76.56%, 78.26%, 87.69%, dan 82.71% pada Gambar 9.

#### 4. KESIMPULAN

Berdasarkan hasil pengumpulan data dengan teknik web scraping pada aplikasi In-Drive yang tersedia di Google Play Store, terdapat sekitar 15.000 titik data yang terkumpul pada tanggal 11 Desember 2024. Setelah dilakukan analisis terhadap 15.000 titik data sentimen, didapatkan hasil 9.665 negatif dan 5.335 positif dari total 15.000 titik data tersebut. Berdasarkan analisis data, dapat disimpulkan bahwa pengguna secara umum tidak menyukai aplikasi In-Drive di Google Play Store. Analisis data sentimen dengan menggunakan metode Naive Bayes memberikan hasil sebagai berikut: akurasi sebesar 76.56%, prediksi sebesar 78.26%, recall sebesar 87.69%, dan F1-Score sebesar 82.71%. Untuk membandingkan hasil penelitian ini dengan evaluasi dan hasil kerja dari penelitian selanjutnya, metode kategorisasi sentimen yang lebih menyeluruh dapat digunakan untuk meningkatkan penelitian tentang aplikasi perjalanan. Hal ini termasuk temuan-temuan yang menjelaskan setiap isu yang diangkat oleh penelitian ini.

#### REFERENCES

[1] J. Khab Sulaiman Dalam, A. Oktavia Praneswara, N. Cahyono, and U. Amikom Yogyakarta, "Analisis Sentimen Ulasan Aplikasi TikTok Shop Seller Center di Google Playstore Menggunakan Algoritma Naive Bayes," *Indonesian Journal of Computer Science Attribution*, vol. 12, no. 6, p. 3925, 2023, doi: <https://doi.org/10.33022/ijcs.v12i6.3473>.



- [2] R. Wahyudi *et al.*, “Analisis Sentimen pada review Aplikasi Grab di Google Play Store Menggunakan Support Vector Machine,” *JURNAL INFORMATIKA*, vol. 8, no. 2, 2021, doi: doi.org/10.31294/ji.v8i2.9681.
- [3] K. Kevin, M. Enjeli, and A. Wijaya, “Analisis Sentimen Penggunaan Aplikasi Kinemaster Menggunakan Metode Naive Bayes,” *Jurnal Ilmiah Computer Science*, vol. 2, no. 2, pp. 89–98, Jan. 2024, doi: 10.58602/jics.v2i2.24.
- [4] D. Nugraha and D. Gustian, “Analisis Sentimen Penggunaan Aplikasi Transportasi Online Pada Ulasan Google Play Store dengan Metode Naive Bayes Classifier,” *Kesatria*, Vol 5, No 1, 2024, doi: doi.org/10.30645/kesatria.v5i1.341.
- [5] S. N. S. Muslim, F. Nurdiansyah, and A. Y. Rahman, “Perbandingan Algoritma Naive Bayes Dan Knn Dalam Analisis Sentimen Ulasan Pengguna Aplikasi Capcut,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 3S1, Oct. 2024, doi: 10.23960/jitet.v12i3S1.5156.
- [6] M. Khoirul, U. Hayati, and O. Nurdiawan, “Analisis Sentimen Aplikasi Brimo Pada Ulasan Pengguna Di Google Play Menggunakan Algoritma Naive Bayes,” *JATI*, Vol 7, No 1, 2023, doi: doi.org/10.36040/jati.v7i1.6373.
- [7] E. Hasibuan and E. A. Heriyanto, “Analisis Sentimen Pada Ulasan Aplikasi Amazon Shopping Di Google Play Store Menggunakan Naive Bayes Classifier,” *JTS*, vol. 1, no. 3, 2022, doi: doi.org/10.56127/jts.v1i3.434.
- [8] N. Febrinikmah Siharta, M. Muzakki Bhaswara, W. Firmantara, and A. Puspita Sari, “Analisis Sentimen Ulasan Pengguna Aplikasi Disney+ Hotstar Menggunakan Naive Bayes Classifier,” *Journal of Multidisciplinary Inquiry in Science Technology and Educational Research*, vol. 1, no. 4, pp. 1847–1855, 2024, doi: 10.32672/mister.v1i4.2120.
- [9] T. Hartati, R. T. Sohadi, E. Tohidi, and E. Wahyudin, “Penerapan Algoritma Naive Bayes pada Analisis Sentimen Ulasan Aplikasi Whoosh-Kereta Cepat Di Google Play Store,” *Jurnal Informatika dan Rekayasa Perangkat Lunak*, 2024, Vol 6, No 1, 2024, doi: doi.org/10.36499/jinrpl.v6i1.10307.
- [10] M. Yusuf, R. Gaja, I. Maulana, and O. Komarudin, “Analisis Sentimen Opini Pengguna Aplikasi Vidio Pada Ulasan Playstore Menggunakan Algoritma Naive Bayes,” *JATI*, Vol 7 No 4, 2023, doi: doi.org/10.36040/jati.v7i4.7197.
- [11] J. Khatib Sulaiman, D. Atmajaya, A. Febrianti, H. Darwis, “Metode SVM dan Naive Bayes untuk Analisis Sentimen ChatGPT di Twitter,” *Indonesian Journal of Computer Science Attribution*, vol. 12, no. 4, p. 2173, 2023, doi: doi.org/10.33022/ijcs.v12i4.3341.
- [12] A. P. Giovani, A. Ardiansyah, T. Haryanti, L. Kurniawati, and W. Gata, “Analisis Sentimen Aplikasi Ruang Guru Di Twitter Menggunakan Algoritma Klasifikasi,” *Jurnal Teknoinfo*, vol. 14, no. 2, p. 115, Jul. 2020, doi: 10.33365/jti.v14i2.679.
- [13] T. Septiani Gumilar, R. Astuti, and Y. A. Wijaya, “Analisis Sentimen Ulasan Aplikasi Lita Di Play Store Menggunakan Algoritma Naive Bayes,” *JATI*, Vol 8, No 1, 2024, doi: doi.org/10.36040/jati.v8i1.8778.
- [14] W. Wahyuni, “Analisis Sentimen terhadap Opini Feminisme Menggunakan Metode Naive Bayes,” *Jurnal Informatika Ekonomi Bisnis*, pp. 148–153, Sep. 2022, doi: 10.37034/infec.v4i4.162.
- [15] M. R. Hanafi and R. K. R., “Analisis Sentimen pada Ulasan Aplikasi Sirekap di Google Play Menggunakan Algoritma Naive Bayes,” *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 4, no. 4, pp. 1578–1586, Oct. 2024, doi: 10.57152/malcom.v4i4.1693.
- [16] Y. Nurtikasari, Syariful Alam, and Teguh Iman Hermanto, “Analisis Sentimen Opini Masyarakat Terhadap Film Pada Platform Twitter Menggunakan Algoritma Naive Bayes,” *INSOLOGI: Jurnal Sains dan Teknologi*, vol. 1, no. 4, pp. 411–423, Aug. 2022, doi: 10.55123/insologi.v1i4.770.
- [17] F. Matheos Sarimole and K. Kudrat, “Analisis Sentimen Terhadap Aplikasi Satu Sehat Pada Twitter Menggunakan Algoritma Naive Bayes Dan Support Vector Machine,” *Jurnal Sains dan Teknologi*, vol. 5, no. 3, pp. 783–790, doi: 10.55338/saintek.v5i1.2702.
- [18] M. Gamma, A. Hakim, and F. Irwiensyah, “Analisis Sentimen Terhadap Ulasan Pengguna Pada Aplikasi Traveloka Menggunakan Metode Naive Bayes,” *Technology and Science (BITS)*, vol. 6, no. 3, pp. 2024–1448, 2024, doi: 10.47065/bits.v6i3.6119.
- [19] J. Khatib Sulaiman, R. Abdan Syakura, A. Davy Wiranata, U. D. Muhammadiyah HAMKA Jakarta, and K. Kunci Analisis, “Analisis Sentimen Ulasan Kepuasan Pengguna Aplikasi Bsi Mobile Dengan Menggunakan Naive Bayes,” *Indonesian Journal of Computer Science*, vol. 13, no. 3, 2023, doi: doi.org/10.33022/ijcs.v13i3.4063.
- [20] B. A. Maulana, M. J. Fahmi, A. M. Imran, and N. Hidayati, “Analisis Sentimen Terhadap Aplikasi Pluang Menggunakan Algoritma Naive Bayes dan Support Vector Machine (SVM),” *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 4, no. 2, pp. 375–384, Feb. 2024, doi: 10.57152/malcom.v4i2.1206.