

# Analisis Sentimen Ulasan Restoran Franchise di Purwokerto pada Google Maps Menggunakan Algoritma Naive Bayes

Katherin Gracia Sibarani  
Fakultas Rekayasa Industri  
Universitas Telkom  
Purwokerto, Indonesia  
katherinsibarani@student.telkomuniversity.ac.id

Sena Wijayanto  
Fakultas Rekayasa Industri  
Universitas Telkom  
Purwokerto, Indonesia  
senawijayanto@telkomuniversity.ac.id

**Abstrak**—Penelitian ini bertujuan untuk menganalisis sentimen ulasan pelanggan terhadap restoran franchise di Purwokerto yang tersedia di Google Maps dengan menerapkan algoritma Naive Bayes. Restoran yang menjadi objek penelitian meliputi KFC, Rocket Chicken, McDonald's, Almaz Fried Chicken dan Richeese Factory. Data ulasan dikumpulkan melalui teknik web scraping dan kemudian diproses menggunakan text preprocessing seperti tokenisasi, stemming, dan pembobotan kata dengan TF-IDF. Algoritma Naive Bayes digunakan untuk mengklasifikasikan sentimen ulasan ke dalam kategori positif, negatif, dan netral. Evaluasi model dilakukan menggunakan metrik akurasi, presisi, recall, dan f1-score. Berdasarkan hasil analisis, model memperoleh akurasi sebesar 89%, dengan nilai f1-score tertinggi pada sentimen positif sebesar 0.91, diikuti oleh netral 0.89, dan negatif 0.88. Hasil confusion matrix juga menunjukkan bahwa sebagian besar prediksi sesuai dengan label sebenarnya, meskipun terdapat sejumlah kesalahan klasifikasi antar kelas. Hal ini menunjukkan bahwa model tersebut terbukti cukup andal dalam membedakan ketiga jenis sentimen dan dapat dimanfaatkan sebagai alat bantu analisis sentimen yang efektif terhadap restoran franchise di Purwokerto.

**Kata kunci**— Analisis sentimen, Algoritma Naive Bayes, Restoran Franchise, Google Maps, Ulasan pelanggan

## I. PENDAHULUAN

Kemajuan teknologi informasi dan komunikasi di era digital saat ini telah secara signifikan memengaruhi transformasi dalam dunia bisnis, terutama di industri kuliner. Kehadiran platform online seperti Google Maps telah memberikan ruang bagi konsumen untuk membagikan ulasan, pengalaman atau opini mereka. Ulasan yang diberikan oleh pengguna menjadi acuan penting bagi konsumen dalam mempertimbangkan keputusan mereka saat memilih suatu produk atau layanan, termasuk dalam memilih restoran. Ulasan ini sering kali mengandung sentimen dan emosi yang dapat dianalisis untuk mendapatkan wawasan lebih dalam mengenai persepsi konsumen terhadap suatu restoran[1].

Bisnis restoran juga termasuk ke dalam sektor bisnis franchise karena dapat dijalankan melalui model franchising.

Pada bisnis franchise, perusahaan induk memberikan akses terhadap brand, produk dan pengetahuan bisnis yang dimiliki melalui lisensi, kepada perusahaan yang membeli lisensi tersebut[2]. Di Indonesia, beberapa bisnis waralaba atau franchise mengalami perkembangan pesat.

Bisnis restoran di Indonesia, termasuk di Purwokerto, banyak yang mengadopsi model franchise. Model ini memungkinkan perusahaan induk memberikan lisensi kepada mitra untuk menggunakan merk, produk dan sistem bisnis yang telah teruji. Persaingan di sektor ini sangat ketat, sehingga pemilik restoran perlu memahami beberapa aspek yang turut memengaruhi keputusan konsumen antara lain lokasi usaha, cita rasa makanan, serta kualitas pelayanan yang diberikan[3]. Namun, banyak pelaku usaha belum memanfaatkan ulasan pelanggan secara optimal sebagai dasar evaluasi dan pengambilan keputusan strategis.

Penelitian ini akan fokus pada analisis sentimen ulasan restoran franchise *fast food* ayam di Purwokerto, yaitu Restoran KFC, Rocket Chicken, McDonald's, Almaz Fried Chicken dan Richeese Factory dengan alasan kelimanya memiliki ulasan terbanyak di Google Maps. Masing-masing restoran ini Memiliki ciri khas serta jenis penawaran yang beragam. Oleh karena itu, penting untuk mengkaji sentimen yang terdapat dalam ulasan-ulasan tersebut. Dengan menganalisis ulasan yang ada, kita dapat mengkategorikan opini atau ulasan pengguna menjadi pendapat positif, negatif maupun netral[4].

Analisis sentimen adalah teknik untuk memahami persepsi publik terhadap berbagai topik dan digunakan untuk mengekstrak, memahami dan menganalisis opini subjektif dari data teks[5]. Naive Bayes merupakan teknik klasifikasi yang didasarkan pada pendekatan probabilistik yang bersifat sederhana, yang bekerja dengan menghitung peluang dari kombinasi nilai dan frekuensi data yang tersedia dalam suatu dataset[6]. Walaupun tergolong metode yang sederhana, Naive Bayes merupakan algoritma yang telah terbukti memberikan kinerja yang efektif dalam beragam penerapan klasifikasi teks, salah satunya dalam analisis sentimen. Dalam penerapan pada ulasan restoran, algoritma ini dapat dimanfaatkan guna mengelompokkan ulasan berdasarkan kategori sentimen, seperti sentimen positif, negatif, maupun

netral. yang dapat membantu melihat kepuasan pelanggan terhadap restoran tersebut[7].

Penelitian yang dilakukan oleh Ghina berjudul “Analisis Sentimen Ulasan Pelanggan Mie Gacoan Menggunakan Metode Naive Bayes Classifier” membuktikan bahwa algoritma Naive Bayes mampu memberikan performa yang cukup optimal, dengan tingkat akurasi mencapai 88,06%, recall sebesar 77,78%, serta presisi sebesar 69,23%. Temuan dalam penelitian ini berpotensi memberikan dampak positif terhadap peningkatan mutu produk dan layanan pada restoran[8]. Penelitian sebelumnya juga yang dilakukan oleh Angga dengan judul “Implementasi Algoritma Naive Bayes Terhadap Review Aplikasi KFCKU” menunjukkan hasil penelitian variasi tingkat akurasi sebesar 85%, presisi 82%, recall sebesar 90% dan F1 Score 90%. Metode ini menunjukkan performa yang unggul dalam mengelompokkan sentimen konsumen ke dalam klasifikasi positif, negatif, dan netral, disertai dengan tingkat akurasi yang memuaskan[4].

Berdasarkan identifikasi gap tersebut, penelitian ini akan membahas mengenai analisis sentimen ulasan pelanggan terhadap lima restoran franchise *fast food* ayam di Purwokerto. Pemanfaatan algoritma Naive Bayes dalam analisis sentimen dimanfaatkan untuk memperoleh pemahaman dan informasi penting terkait pandangan, opini, dan respons emosional masyarakat terhadap suatu merek, produk, layanan, maupun isu tertentu. Penelitian ini akan mengidentifikasi sentimen yang terkandung dalam ulasan masing-masing restoran sehingga dapat memberikan rekomendasi yang spesifik dan aplikatif bagi restoran. Dengan demikian, temuan dari penelitian ini diharapkan dapat memberikan kontribusi dalam menggambarkan berbagai perspektif pelanggan serta menjadi acuan dalam upaya peningkatan mutu layanan pada restoran franchise di daerah Purwokerto. Serta dapat diperoleh gambaran yang jelas bagaimana pelanggan menilai kelima restoran franchise tersebut melalui ulasan di Google Maps.

## II. KAJIAN TEORI

### A. Google Maps

Google Maps merupakan layanan peta daring yang diberikan secara cuma-cuma oleh Google. Layanan ini dapat diakses melalui browser maupun perangkat mobile, dan memungkinkan pengguna mencari rute perjalanan, menemukan lokasi bisnis di sekitar, serta memperoleh berbagai informasi lainnya seputar tempat atau layanan yang dibutuhkan[9].

### B. Analisis Sentimen

Analisis sentimen adalah pendekatan yang dimanfaatkan untuk mengenali atau mendeteksi persepsi publik terhadap sebuah isu dengan cara mengekstrak, memahami, serta menilai opini subjektif yang terdapat dalam data berbentuk teks[5].

### C. Algoritma Naive Bayes

Naive Bayes merupakan pendekatan klasifikasi yang didasarkan pada Teorema Bayes, dengan anggapan bahwa setiap fitur dalam data bersifat independen satu sama lain. Metode ini menentukan peluang tiap kelas dengan melihat seberapa sering kata muncul dalam data[6]. Rumus 1 menyajikan bentuk umum dari persamaan Naive Bayes[10].

$$P(B|A) = \frac{P(B|A) \times P(A)}{P(B)} \quad (2-1)$$

Keterangan:

P (A|B): Probabilitas bersyarat A yang diberikan oleh B

P (B|A): Probabilitas bersyarat B yang diberikan oleh A

P (A): Probabilitas kejadian A

P (B): Probabilitas kejadian B

### D. Ulasan Pelanggan

Ulasan pelanggan merupakan tanggapan yang diberikan konsumen setelah mencoba suatu produk atau layanan. Ulasan ini biasanya memuat sentimen dan emosi yang bisa dianalisis untuk memperoleh pemahaman lebih mendalam tentang pandangan konsumen terhadap produk atau layanan tersebut[1].

### E. *Bisnis Franchise*

Bisnis franchise merupakan suatu bentuk usaha di mana pemilik merek memberikan izin kepada pihak lain untuk memanfaatkan merek tersebut dalam kegiatan usahanya., produk, serta sistem bisnis yang telah terbukti sukses[2].

### F. Klasifikasi

Klasifikasi adalah salah satu bentuk struktur data yang memanfaatkan teknik analisis dasar guna membagi data ke dalam kategori tertentu berdasarkan informasi yang telah diketahui sebelumnya[10]. Selain itu klasifikasi adalah proses pembelajaran di mana sebuah fungsi dilatih untuk mengaitkan setiap himpunan objek dengan salah satu label kelas yang telah ditentukan sebelumnya[11].

### G. Pengumpulan Data

Pengumpulan data adalah langkah awal dalam mengumpulkan dokumen yang nantinya akan digunakan sebagai masukan untuk sistem[12]. Pengumpulan data dapat dilakukan secara otomatis dengan menginstall Instant Data Scraper, hasil data yang dikumpulkan dapat disimpan dengan format csv[13].

### H. Pembersihan Data (*Cleaning*)

Pembersihan data merupakan tahapan untuk melakukan pembersihan terhadap karakter non-huruf agar data yang akan diproses menjadi lebih bersih dan minim gangguan yang dapat mempengaruhi akurasi hasil klasifikasi[14].

### I. *Preprocessing*

*Preprocessing* adalah proses persiapan data yang akan diolah. Tahap ini untuk memastikan bahwa data yang digunakan pada saat analisis adalah data yang konsisten dan siap untuk masuk ketahap pemodelan[15]. Proses implementasi pada *preprocessing* melibatkan beberapa langkah penting sebagai berikut:

#### 1. *Case Folding*

*Case folding* adalah proses menyeragamkan seluruh huruf dalam dokumen menjadi huruf kecil (*lowercase*) guna menyamakan format penulisan karakter[4].

#### 2. Normalisasi

Normalisasi dilakukan untuk memastikan konsistensi pada teks yang akan dianalisis dengan mencakup beberapa tahapan khusus, seperti mengembangkan kalimat pendek menjadi susunan kalimat yang lebih lengkap dan sesuai dengan kaidah kebahasaan yang benar[15].

### 3. Stopword Removal

*Stopword removal* merupakan kata-kata yang dipilih dari hasil tokenisasi berdasarkan tingkat relevansinya untuk kemudian digunakan dalam proses selanjutnya untuk merepresentasikan isi dokumen secara menyeluruh[16].

### 4. Tokenisasi

Tokenisasi adalah proses memecah rangkaian kata dalam sebuah kalimat, paragraf, atau dokumen menjadi token, yaitu potongan kata atau istilah tunggal yang berdiri sendiri[16].

### 5. Stemming

*Stemming* merupakan proses yang bertujuan untuk mengonversi kata berimbuhan ke dalam bentuk dasar atau bentuk asalnya. Pada proses ini, awalan dan akhiran akan dihapus sehingga diperoleh bentuk dasar dari kata tersebut[17].

### 6. Slangword

*Slangword* atau bahasa gaul adalah variasi bahasa yang dipakai secara tidak formal oleh kelompok sosial tertentu. Kata atau frasa yang digunakan biasanya memiliki arti khusus yang bisa berbeda dari makna dalam bahasa resmi[18].

### 7. Lemmatizing

*Lemmatizing* adalah tahapan dalam pemrosesan bahasa alami (NLP) yang mengubah kata ke bentuk dasarnya sesuai dengan entri dalam kamus (lemma), dengan memperhatikan makna dan konteks kalimat[19].

### J. Labelling

*Labelling* merupakan proses menentukan apakah suatu ulasan dalam dataset termasuk ke dalam kategori positif atau negatif. Proses pelabelan dapat dilakukan secara manual maupun otomatis. Tujuan utamanya adalah untuk memperoleh pemahaman terhadap data serta mengidentifikasi sentimen yang terkandung di dalamnya[10].

### K. TF-IDF

TF-IDF (*Term Frequency-Inverse Document Frequency*) merupakan tahap melakukan pembobotan kata (*term*) dalam dokumen[10]. Algoritma seperti TF-IDF menghitung TF dan IDF untuk setiap dokumen dalam korpus sesuai dengan kriteria yang telah ditentukan. Sederhananya untuk menghitung jumlah kata yang muncul dalam dokumen[20]. Rumus yang sering digunakan[21]:

$$Tf(w) = \frac{\text{frekuensi muncul kata } w \text{ di dokumen } d}{\text{Total kata dokumen } d} \quad (2-2)$$

$$IDF(t) = \log\left(\frac{D}{dft}\right) \quad (2-3)$$

$$Wd.t = tf_{d,t} \times IDF_{d,t} \quad (2-4)$$

Keterangan:

TF (*Term Frequency*): Frekuensi kemunculan kata pada dokumen dibagi dengan total jumlah kata dalam dokumen

D = Total semua dokumen

dft = banyak dokumen yang mengandung term

Wd.t: bobot dari term(kata) t pada dokumen d

d = dokumen ke-d

t = kata ke-t dari data kunci

### L. Wordcloud

*Wordcloud* merupakan metode visualisasi data teks yang menyajikan kata-kata dalam bentuk grafis, di mana ukuran masing-masing kata mencerminkan frekuensi kemunculannya dalam data. Semakin sering sebuah kata muncul, maka kata

itu akan ditampilkan dengan ukuran yang lebih besar dalam *wordcloud*[22].

### M. Lexicon

*Lexicon* adalah metode analisis sentimen yang memanfaatkan pendekatan berbasis kamus. Dengan metode ini, kamus *lexicon* mampu memberikan bobot pada setiap kata sesuai dengan daftar yang tersedia, sehingga proses klasifikasi kalimat secara otomatis menjadi lebih mudah[23].

### N. Confusion Matrix

*Confusion matrix* merupakan tabel yang menyajikan total data uji berdasarkan output hasil klasifikasi yang telah dilakukan secara tepat maupun yang mengalami kesalahan klasifikasi[24]. Rumus-rumus utama *confusion matrix*:

TABEL 1  
(CONFUSION MATRIX)

Class	Prediksi Positif	Prediksi Negatif	Prediksi Netral
Positif	TP	FN	FN
Negatif	FP	TN	FN
Netral	FP	FP	TN

Keterangan:

TP (*True Positive*): jumlah data yang sebenarnya positif dan berhasil diidentifikasi sebagai positif

TN (*True Negative*): jumlah data yang sebenarnya negatif dan berhasil diidentifikasi sebagai negatif

FP (*False Positive*): Jumlah data yang sebenarnya negatif tapi diidentifikasi sebagai positif

FN (*False Negative*): Jumlah data yang sebenarnya positif tapi diidentifikasi sebagai negatif.

Rumus-rumus utama confusion matrix:

#### 1. Akurasi

Akurasi yaitu visualisasi keakuratan model dalam mengelompokkan dengan benar[25]. Dihitung dengan rumus:

$$\text{Akurasi} = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \quad (3-5)$$

#### 2. Presisi

Presisi yaitu visualisasi dari persentase keakuratan hasil perkiraan oleh metode yang digunakan[25]. Dihitung dengan rumus:

$$\text{Presisi} = \frac{TP}{TP + FP} \times 100 \quad (3-6)$$

#### 3. Recall

Recall yaitu visualisasi kesesuaian metode dalam mencari ulang sebuah informasi[25]. Dihitung dengan rumus:

$$\text{Recall} = \frac{TP}{TP + FN} \times 100 \quad (3-7)$$

#### 4. F1- Score

F1- Score adalah rata-rata harmonis antara presisi dan recall[25]. Dihitung dengan rumus:

$$F1 - \text{Score} = 2 \times \frac{\text{Presisi} \times \text{Recall}}{\text{Presisi} + \text{Recall}} \quad (3-8)$$

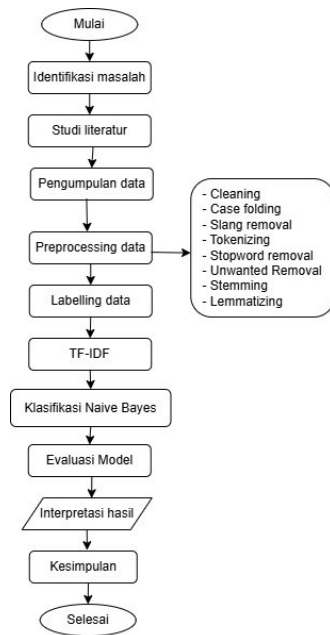
### O. Oversampling

Pada saat terjadi ketidakseimbangan kelas (*imbalanced data*) dapat dilakukan teknik sampling seperti *oversampling* yaitu menambah jumlah data pada kelas minoritas hingga jumlahnya sama atau hampir sama dengan kelas mayoritas, yaitu dengan mengandakan sampel dari kelas minoritas secara acak sampai rasio antar kelas sesuai target[26].

## III. METODE

Dalam penelitian ini, metodologi yang digunakan untuk menganalisis sentimen ulasan lima restoran franchise di

Purwokerto menggunakan algoritma Naive Bayes. Alur penelitian dirancang secara sistematis seperti ditunjukkan pada Gambar 1



GAMBAR 1  
(DIAGRAM ALIR PENELITIAN)

#### A. Identifikasi Masalah

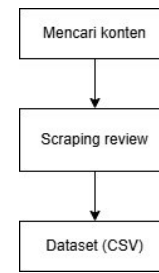
Pada tahap pertama, peneliti melakukan identifikasi dan merumuskan permasalahan pada penelitian. Dengan melihat ulasan di Google Maps yang diberikan pelanggan, Ulasan yang diberikan oleh pelanggan bervariasi dalam hal sentimen (positif, negatif, netral). Meskipun Google Maps menyediakan banyak ulasan, tidak semua ulasan mencerminkan pengalaman yang objektif. Beberapa ulasan dipengaruhi oleh faktor eksternal seperti, kondisi cuaca atau situasi pada saat kunjungan, yang dapat mempengaruhi hasil analisis sentimen.

#### B. Studi Literatur

Tahap kedua melakukan studi literatur, salah satu langkah penting dalam penelitian yang bertujuan untuk mengumpulkan dan mempelajari sumber tertulis yang relevan dengan topik penelitian untuk memperkuat landasan teori mengenai penelitian yang akan dilakukan. Adapun sumber yang digunakan yaitu Google scholar dan Connected paper dengan rentang waktu 5 tahun terakhir.

#### C. Pengumpulan Data

Tahap ketiga ini peneliti melakukan pengumpulan data yang diambil dari platform online Google Maps melalui web scraping dengan menambahkan extension instant data scraper. Penelitian ini akan fokus pada 5 restoran franchise *fast food* ayam di Purwokerto, yaitu: KFC, Rocket Chicken, McDonald's, Almaz Fried Chicken dan Richeese Factory. Pada penelitian ini tahapan pengumpulan data meliputi, mencari konten pada platform Google Maps, melakukan Scraping review melalui web scarping dengan extension instant data scraper kemudian mengubahnya menjadi CSV (*Comma Separated Values*).



GAMBAR 2  
(DIAGRAM ALIR SCRAPING DATA)

#### 1. Mencari konten

Pada saat mencari konten, peneliti mengidentifikasi dan mengumpulkan ulasan dari pelanggan yang telah mengunjungi restoran franchise di Purwokerto. Ulasan ini mencakup opini, pengalaman dan rating yang diberikan

#### 2. Scraping review

Peneliti menggunakan teknik web *scraping* untuk proses otomatisasi pengambilan data ulasan dari situs web yaitu Google Maps.

#### 3. Dataset

Tahapan ini melakukan penggabungan beberapa file menjadi satu file.

#### D. Pre-processing Data

*Preprocessing* data merupakan tahap penting yang bertujuan untuk mempersiapkan data mentah agar siap untuk dianalisis. Tahap - tahap *pre-processing* data yaitu *cleaning*, *case folding*, *slang removal*, *tokenizing*, *stopword removal*, *unwanted stopword*, *stemming*, dan *lemmatizing*.

#### E. Labelling Data

Pelabelan data adalah proses pemberian label atau kategori pada data mentah, di mana data tersebut diklasifikasikan ke dalam tiga jenis sentimen: positif, negatif, dan netral. Dalam penelitian ini, proses pelabelan dilakukan secara manual berdasarkan analisis terhadap makna kata dalam teks. Setiap kalimat ditinjau dan ditentukan kategorinya dengan mempertimbangkan nilai sentimen dari kata-kata yang muncul.

#### F. Representasi Fitur

Tahapan ini menjelaskan representasi data dengan mengkonversi data mentah ke dalam bentuk numerik sehingga dapat diproses oleh algoritma machine learning dengan menggunakan TF-IDF[27].

#### G. Analisis sentimen klasifikasi dengan Naive Bayes

Pada tahap ini peneliti membuat sebuah sistem untuk menganalisa dari data yang telah dikumpulkan. Pada penelitian ini Naive Bayes digunakan untuk melakukan klasifikasi guna menentukan sentimen (positif, negatif atau netral).

#### H. Evaluasi Model

Evaluasi dilakukan untuk mengetahui seberapa baik model yang dibangun mampu mengklasifikasikan sentimen ulasan ke dalam tiga kategori, yaitu positif, negatif, dan netral. Hasil klasifikasi Naive Bayes dievaluasi menggunakan beberapa metrik yang umum digunakan dalam machine learning, yaitu akurasi, presisi, recall, dan F1-score.

#### I. Interpretasi Hasil

Pada penelitian ini interpretasi hasil bertujuan untuk memahami dan menjelaskan hasil yang diperoleh dari model

klasifikasi, dalam penelitian ini menggunakan algoritma Naive Bayes.

J. Kesimpulan

Tahapan ini dilakukan untuk membuat rangkuman hasil penelitian dan memberikan pandangan tentang implikasi dari temuan penelitian ini.

IV. HASIL DAN PEMBAHASAN

A. Pengumpulan Data

Pada tahap ini, data ulasan pelanggan dikumpulkan dari lima restoran franchise di Purwokerto yaitu KFC, Rocket Chicken, McDonald's, Almaz Fried Chicken, dan Richeese Factory melalui teknik web scraping menggunakan Instant Data Scraper di Google Maps. Ulasan disimpan dalam format CSV untuk memudahkan pengolahan. Data ini merupakan data primer yang valid dan diperoleh langsung dari sumber tanpa modifikasi. Berikut tampilan data mentah hasil pengumpulan pada gambar 3.

	A	B
1	Nama	Ulasan
2	boanerges andri	Selalu langganan disini...pelayanan sgt ramah,...
3	Ibnu Salimi	Meeting nyaman & makan enak, ya di sini. Sejak...
4	Akhmad Nurul Imam	Oke lah

GAMBAR 3 (HASIL SCRAPING DATA)

Setelah pengumpulan, dilakukan analisis awal menggunakan Google colab sebagai media yang digunakan untuk pengekseskuan syntax.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import re
import csv

from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import classification_report, confusion_matrix
from sklearn.utils import resample
from collections import Counter
from wordcloud import WordCloud
```

GAMBAR 4 (PROSES IMPORT LIBRARY)

Pada gambar 4, merupakan library yang dibutuhkan untuk memudahkan proses analisis data, khususnya data teks, mulai dari tahap membaca data, pembersihan, pemrosesan hingga visualisasi hasil analisis.

B. Perancangan Model

Model dirancang untuk mengelompokkan sentimen ulasan pelanggan lima restoran cepat saji di Purwokerto menggunakan algoritma Multinomial Naive Bayes. Data diproses melalui tahapan pra-pemrosesan teks, lalu dibagi menjadi data latih dan uji untuk dilatih dalam mengklasifikasikan sentimen menjadi positif, negatif, atau netral, yang selanjutnya dievaluasi menggunakan metrik akurasi, presisi, recall, dan F1-score.

C. Pengembangan Model

Setelah tahap perancangan selesai, langkah berikutnya adalah membangun model analisis sentimen berdasarkan rancangan sebelumnya. Pengembangan model dilakukan menggunakan bahasa pemrograman Python dengan memanfaatkan platform Google Colaboratory. Adapun proses yang dilakukan meliputi:

1. Pre-processing

Pre-processing data teks secara menyeluruh untuk menjadikan kalimat menjadi data yang bisa diproses dan mendapatkan hasil akurasi.

a. Cleaning

```
def cleaning(teks):
    # Ubah ke huruf kecil
    teks = teks.lower()

    # Hapus emotikon dan karakter non-ASCII
    teks = teks.encode('ascii', 'ignore').decode('ascii')

    # Opsional: Hapus URL
    teks = re.sub(r'http\S+|www\S+|https\S+', '', teks, flags=re.MULTILINE)

    # Opsional: Hapus mention dan hashtag
    teks = re.sub(r'\@\w+|\#','', teks)

    # Hapus angka dan tanda baca (opsional tergantung kebutuhan)
    teks = re.sub(r'[A-Za-z\s]', '', teks)

    # Hapus spasi ganda
    teks = re.sub(r'\s+', ' ', teks).strip()

    return teks

df['cleaned_ulasan'] = df['ulasan'].apply(cleaning)
df.head(5)
```

GAMBAR 5 (CLEANING)

Gambar 5 menunjukkan fungsi `cleaning(teks)` yang digunakan untuk membersihkan data teks agar lebih rapi dan siap untuk dianalisis. Proses pembersihan dilakukan dengan beberapa tahapan, yaitu: mengubah seluruh teks menjadi huruf kecil (lowercase), menghapus emotikon atau karakter non-ASCII, menghapus URL, mention, dan hashtag. Selain itu, fungsi ini juga menghilangkan angka dan tanda baca (bergantung pada kebutuhan), serta menghapus spasi ganda untuk memastikan format teks rapi.

b. Slangword Removal

```
slang_dictionary = pd.read_csv('https://raw.githubusercontent.com/riakus/data-science-portfolio/master/trafic/wordcloud/colloquial_indonesian_english.csv')
slang_dict = pd.Series(slang_dictionary['formal'], index=slang_dictionary['slang']).to_dict()

def slangwords(teks):
    for word in teks.split():
        if word in slang_dict.keys():
            teks = teks.replace(word, slang_dict[word])
    return teks

df['ulasan_slang'] = df['cleaned_ulasan'].apply(slangwords)
df.head(5)
```

GAMBAR 6 (SLANGWORD REMOVAL)

Gambar 4-6 menunjukkan proses slangword, yaitu mengganti kata tidak baku atau bahasa gaul menjadi bentuk baku. Proses ini menggunakan kamus slang yang diperoleh dari sumber daring dan diubah menjadi kamus Python yang memetakan kata slang ke bentuk bakunya.

c. Tokenizing

```
def split_word(teks):
    return teks.split()

df['ulasan_token'] = df['cleaned_ulasan'].apply(split_word)
df.head(5)
```

GAMBAR 7 (TOKENIZING)

Gambar 4-7 menunjukkan proses tokenizing yang dilakukan dengan membagi sebuah kalimat atau teks menjadi bagian-bagian kata yang lebih kecil. Hasil akhir dari proses ini adalah sebuah list yang berisi kata-kata individual dari teks yang telah dibersihkan sebelumnya.

a. Stopword Removal

```

from nlp_id.stopword import Stopword

def remove_stopwords_from_text(text):
    stopword = Stopword()
    cleaned_string = stopword.remove_stopword(text)
    return cleaned_string

df['ulasan_stopword'] = df['ulasan_slang'].apply(remove_stopwords_from_text)
df.head(5)

```

GAMBAR 8 (STOPWORD REMOVAL)

Gambar 8 menunjukkan proses stopwords removal dengan menghilangkan kata-kata yang dinilai tidak memberikan pengaruh besar terhadap hasil analisis. Menggunakan `nlp\_id` dan `StopWord()` untuk mengakses daftar kata umum dalam bahasa Indonesia yang dianggap kurang relevan.

b. Unwanted Removal

```

unwanted_words = ['sy', 'karna', 'gue', 'pun', 'nya', 'yg', 'gw', 'krna']

def RemoveUnwantedwords(teks):
    hasil = []
    for kata in teks.split():
        if kata not in unwanted_words:
            hasil.append(kata)
    return ' '.join(hasil)

df['ulasan_unwanted'] = df['ulasan_stopword'].apply(RemoveUnwantedwords)
df.head(5)

```

GAMBAR 9 (UNWANTED REMOVAL)

Gambar 9 menunjukkan proses unwanted removal, yaitu tahap penyaringan kata-kata yang dianggap tidak penting atau tidak relevan dalam analisis, seperti kata informal atau tidak baku (misalnya: "sy", "karna", "gue", "pun", "nya", "yg", dan "gw").

c. Stemming

```

from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
factory = StemmerFactory()
stemmer = factory.create_stemmer()

def apply_stemming(text):
    hasil = []
    for kata in text.split():
        hasil.append(stemmer.stem(kata))
    return ' '.join(hasil)

df['ulasan_stemmed'] = df['ulasan_unwanted'].apply(apply_stemming)

```

GAMBAR 10 (STEMMING)

Gambar 10 menunjukkan proses stemming yang berfungsi untuk mengubah kata menjadi bentuk dasarnya. Proses ini dilakukan dengan menginstal library Sastrawi dan hasilnya disimpan dalam kolom baru bernama `ulasan\_stemmed`.

d. Lemmatizing

```

from nlp_id.lemmatizer import Lemmatizer
lemmatizer = Lemmatizer()

def apply_lemmatization(text):
    return lemmatizer.lemmatize(text)

df['ulasan_lemmatized'] = df['ulasan_stemmed'].apply(apply_lemmatization)
df.head(5)

```

GAMBAR 11 (LEMMATIZING)

Gambar 11 menunjukkan proses lemmatizing yang sama seperti stemming, dengan mengubah kata menjadi bentuk dasarnya. Namun, lemmatizing menghasilkan bentuk dasar yang lebih akurat karena mempertimbangkan struktur dan makna kata sesuai dengan kaidah bahasa Indonesia.

e. Text Function

```

def preprocess_text(df):
    # Buat list kosong untuk menyimpan hasil preprocessing
    processed_texts = []
    for i, row in df.iterrows():
        full_text = row['ulasan']
        teks_bersih = cleaning(full_text)
        teks_bersih = remove_stopwords_from_text(teks_bersih)
        teks_bersih = RemoveUnwantedwords(teks_bersih)
        teks_bersih = stemmer.stem(teks_bersih)
        processed_texts.append(teks_bersih)
        print('Progres :', round(i / len(df) * 100, 2), "%")

    # Tambahkan kolom hasil preprocessing
    df['full_text_processed'] = processed_texts
    return df

df=preprocess_text(df)
df.head()
print("Columns after preprocessing:", df.columns)

```

GAMBAR 12 (TEXT FUNCTION)

Gambar 12 menunjukkan proses text function, dengan menggunakan fungsi `preprocess\_text(df)` untuk menjalankan seluruh tahapan preprocessing secara terstruktur yang melalui beberapa tahap, mulai dari cleaning, stopwords removal, unwanted removal, stemming, hingga kata lemmatizing. Hasil akhir dari setiap baris teks kemudian disimpan dalam kolom baru bernama `full\_text\_processed`.

2. Labelling (Polarity score)

a. Sentimen Positif

```

pd.set_option('display_max_colwidth', 3000)
positive_review = df[df['sentimen'] == 'positif']
positive_review = positive_review[['ulasan', 'full_text_processed', 'sentimen']]
positive_review.index = 1

df['ulasan_positif'] = positive_review[['ulasan', 'full_text_processed', 'sentimen']]
df.head()

```

GAMBAR 13 (FILTER SENTIMEN POSITIF)

Gambar 13 menunjukkan proses seleksi data ulasan dengan sentimen positif. Pada proses ini, data difilter berdasarkan nilai pada kolom `sentimen` yang bernilai positif.

b. Sentimen Netral

```

pd.set_option('display_max_colwidth', 3000)
neutral_review = df[df['sentimen'] == 'netral']
neutral_review = neutral_review[['ulasan', 'full_text_processed', 'sentimen']]
neutral_review.index = 1

df['ulasan_netral'] = neutral_review[['ulasan', 'full_text_processed', 'sentimen']]
df.head()

```

GAMBAR 14 (FILTER SENTIMEN NETRAL)

Gambar 14 menunjukkan proses seleksi data ulasan dengan sentimen netral. Pada proses ini, data difilter

berdasarkan nilai pada kolom `sentimen` yang bernilai netral.

c. Sentimen Negatif

```

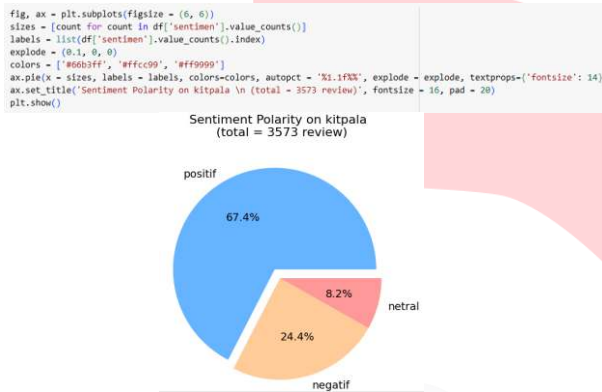
plt.set_option('display.max_colwidth': 200)
negative_review = df[df['sentimen'] == 'negatif']
negative_review = negative_review[['ulasan', 'full_text_processed', 'sentimen']].sort_values(by = 'sentimen', ascending=False, reset_index(drop = True))
negative_review.index = 1
negative_review[:10]

```

GAMBAR 15 (FILTER SENTIMEN NEGATIF)

Gambar 15 menunjukkan proses seleksi data ulasan dengan sentimen negatif. Pada proses ini, data difilter berdasarkan nilai pada kolom `sentimen` yang bernilai negatif.

d. Pie Plot



GAMBAR 16 (PIE PLOT)

Gambar 16 menunjukkan proses pembuatan diagram pai yang digunakan untuk menampilkan distribusi sentimen pada data ulasan, dengan membagi persentase ulasan ke dalam kategori positif, netral, dan negatif berdasarkan jumlah masing-masing sentimen. Dari total 3.573 ulasan, sebanyak 67,4% bersentimen positif, 24,4% negatif, dan 8,2% netral. Visualisasi ini memberikan gambaran umum tentang dominasi sentimen positif dalam dataset.

3. TF-IDF

Setelah pelabelan, tahap selanjutnya yang dilakukan adalah ekstraksi fitur dari data teks menggunakan metode TF-IDF.

```

from sklearn.feature_extraction.text import TfidfVectorizer

vectorizer = TfidfVectorizer()
X_train_tfidf = vectorizer.fit_transform(X_train)
X_test_tfidf = vectorizer.transform(X_test)

print("Ukuran data training (TF-IDF):", X_train_tfidf.shape)
print("Ukuran data testing (TF-IDF):", X_test_tfidf.shape)

```

Ukuran data training (TF-IDF): (5779, 4349)  
Ukuran data testing (TF-IDF): (1445, 4349)

GAMBAR 17 (TF-ID)

Gambar 17 menunjukkan proses TF-IDF yang digunakan untuk mengubah data teks menjadi representasi numerik untuk data training dan testing. Hasilnya menunjukkan bahwa data training memiliki dimensi (5779, 4349) dan data testing berukuran (1445, 4349).

D. Pengujian Modul

1. Data Selection

```

df_positif = df[df['sentimen'] == 'positif']
df_negatif = df[df['sentimen'] == 'negatif']
df_netral = df[df['sentimen'] == 'netral']

```

GAMBAR 18 (DATA SELECTION)

Gambar 18 menunjukkan proses *Data Selection*, yaitu tahap pemisahan data berdasarkan label sentimen pada kolom 'sentimen' ke dalam tiga kategori, yaitu `df\_positif` untuk ulasan positif, `df\_negatif` untuk ulasan negatif, dan `df\_netral` untuk ulasan netral.

2. Target Sample

```

n_samples = max(len(df_positif), len(df_negatif), len(df_netral))
print("Target jumlah sampel per kelas:", n_samples)

```

Target jumlah sampel per kelas: 2408

GAMBAR 19 (TARGET SAMPLE)

Gambar 19 menunjukkan proses penentuan jumlah target sampel untuk setiap kelas sentimen dengan mengambil jumlah terbesar dari ketiga kelas. Berdasarkan hasil yang ditampilkan, jumlah target sampel per kelas adalah 2408.

3. Oversampling (Data Balancing)

```

from sklearn.utils import resample

# Gandakan data minoritas agar seimbang dengan mayoritas
df_positive_balanced = resample(df_positif, replace=True, n_samples=n_samples, random_state=42)
df_negative_balanced = resample(df_negatif, replace=True, n_samples=n_samples, random_state=42)
df_neutral_balanced = resample(df_netral, replace=True, n_samples=n_samples, random_state=42)
df_balanced = pd.concat([df_positive_balanced, df_negative_balanced, df_neutral_balanced])

# Cek distribusi data setelah balancing
print("Distribusi sentimen setelah balancing:")
print(df_balanced['sentimen'].value_counts())

```

Distribusi sentimen setelah balancing:  
sentimen  
positif 2408  
negatif 2408  
netral 2408  
Name: count, dtype: int64

GAMBAR 20 (OVERSAMPLING)

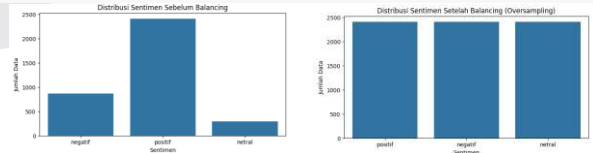
Gambar 20 menunjukkan proses oversampling untuk menyeimbangkan jumlah data pada masing-masing kelas sentimen. Setiap kelas sentimen berhasil diseimbangkan dengan jumlah yang sama yaitu 2408 sampel.

```

# Visualisasi distribusi data sebelum balancing
plt.figure(figsize=(8, 4))
sns.countplot(x='sentimen', data=df)
plt.title('Distribusi Sentimen Sebelum Balancing')
plt.xlabel('Sentimen')
plt.ylabel('Jumlah Data')
plt.show()

# Visualisasi distribusi data setelah balancing
plt.figure(figsize=(8, 4))
sns.countplot(x='sentimen', data=df_balanced)
plt.title('Distribusi Sentimen Setelah Balancing (Oversampling)')
plt.xlabel('Sentimen')
plt.ylabel('Jumlah Data')
plt.show()

```



GAMBAR 21 (VISUALISASI DISTRIBUSI DATA BALANCING)

Gambar 21 menunjukkan proses visualisasi distribusi data sentimen sebelum dan sesudah *balancing* menggunakan grafik batang. Visualisasi ini bertujuan untuk membandingkan jumlah data tiap kelas sentimen dan memastikan hasil *oversampling* berhasil menyamakan distribusinya.

#### 4. Data *Splitting*

```
from sklearn.model_selection import train_test_split

x = df_balanced['ulasan']
y = df_balanced['sentimen']

X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, stratify=y, random_state=42
)

print("Jumlah data training:", len(X_train))
print("Jumlah data testing :", len(X_test))

Jumlah data training: 5779
Jumlah data testing : 1445
```

GAMBAR 22  
(DATA SPLITTING)

Gambar 22 menunjukkan proses pembagian data menjadi data latih dan data uji menggunakan fungsi `train\_test\_split` dari pustaka `sklearn`. Proses ini dilakukan dengan rasio 80 untuk data training dan 20 untuk data testing. Hasil dari proses pembagian tersebut, yaitu sebanyak 5779 data digunakan untuk pelatihan dan 1445 data untuk pengujian.

#### 5. Pelatihan Model Naive Bayes

```
from sklearn.naive_bayes import MultinomialNB

model = MultinomialNB()
model.fit(X_train_tfidf, y_train)
y_pred = model.predict(X_test_tfidf)
```

GAMBAR 23  
(PELATIHAN MODEL NAIVE BAYES)

Gambar 23 menunjukkan proses pelatihan dan prediksi model klasifikasi menggunakan algoritma Multinomial Naive Bayes. Model dilatih dengan data TF-IDF ( $X_{train\_tfidf}$ ) dan label ( $y_{train}$ ), lalu digunakan untuk memprediksi data uji ( $X_{test\_tfidf}$ ) dengan hasil disimpan di  $y_{pred}$ .

#### 6. Classification Report

```
from sklearn.metrics import classification_report, confusion_matrix

print("\nClassification Report:")
print(classification_report(y_test, y_pred))
```

Classification Report:				
	precision	recall	f1-score	support
negatif	0.81	0.96	0.88	481
netral	0.92	0.86	0.89	482
positif	0.97	0.85	0.91	482
accuracy			0.89	1445
macro avg	0.90	0.89	0.89	1445
weighted avg	0.90	0.89	0.89	1445

GAMBAR 24  
(CLASSIFICATION REPORT)

Gambar 24 menunjukkan proses evaluasi model menggunakan dua metrik utama yaitu `classification\_report` dan `confusion\_matrix`. Fungsi `classification\_report` memberikan ringkasan performa model dalam bentuk precision, recall, f1-score, dan support untuk tiap kelas. Model memperoleh akurasi sebesar 0.89 atau 89%, dengan skor precision, recall, dan f1-score yang cukup tinggi di semua kelas. Kelas positif memiliki kinerja terbaik dengan f1-score 0.91, diikuti oleh netral 0.89, dan negatif 0.88. Nilai rata-rata (*macro* dan *weighted*) untuk precision, recall, dan f1-score semuanya juga berada pada angka 0.89 yang menunjukkan bahwa model bekerja secara konsisten dan seimbang dalam mengklasifikasikan ketiga kategori sentimen.

#### E. Evaluasi Hasil Pengujian

Evaluasi dilakukan menggunakan dua metode utama, yaitu *Confusion Matrix* untuk mengukur performa klasifikasi sentimen (positif, negatif, netral), serta WordCloud untuk memvisualisasikan kata-kata yang paling sering muncul dalam ulasan, sehingga dapat memberikan gambaran umum mengenai persepsi pelanggan terhadap masing-masing restoran.

##### 1. *Confusion Matrix*

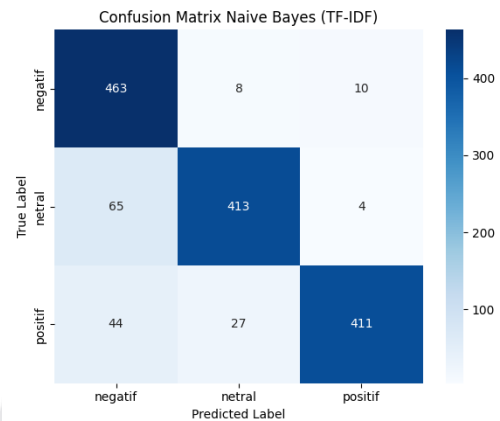
```
# Buat confusion matrix
cm = confusion_matrix(y_test, y_pred, labels=['negatif', 'netral', 'positif'])

# Buat heatmap
plt.figure(figsize=(6, 5))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues',
            xticklabels=['negatif', 'netral', 'positif'],
            yticklabels=['negatif', 'netral', 'positif'])

plt.xlabel('Predicted Label')
plt.ylabel('True Label')
plt.title('Confusion Matrix Naive Bayes (TF-IDF)')
plt.tight_layout()
plt.show()
```

GAMBAR 25  
(MENAMPILKAN *CONFUSION MATRIX*)

Pada gambar 25 menampilkan *Confusion Matrix* "confusion\_matrix()" dari "sklearn.metrics" digunakan untuk menghitung matriks dengan urutan label ['negatif', 'netral', 'positif'], yang memastikan posisi baris dan kolom sesuai. Kemudian, `sns.heatmap()` dari library Seaborn digunakan untuk membuat *heatmap* dari matriks tersebut.



GAMBAR 26  
(VISUALISASI *CONFUSION MATRIX*)

Gambar 26 merupakan visualisasi *confusion matrix* dari hasil prediksi model Naive Bayes. Matriks ini menunjukkan bahwa model berhasil mengklasifikasikan sebagian besar data dengan benar, misalnya 463 dari 481 data sentimen negatif diprediksi tepat, 413 dari 482 data netral, dan 411 dari 482 data positif. Kesalahan prediksi terjadi antar kelas, seperti 65 data netral diprediksi sebagai negatif dan 44 data positif diprediksi sebagai negatif. Secara keseluruhan, model menunjukkan performa yang baik dalam membedakan ketiga kategori sentimen dengan sedikit kesalahan klasifikasi.



- [4] A. A. Permana, R. Taufiq, dan M. I. Wijaya, "Implementasi Algoritma Naïve Bayes Terhadap Review Aplikasi KFCKU," *JTS : Jurnal Teknik*, vol. 12, no. 2, hlm. 128–137, 2023, doi: 10.31000/jt.v12i2.10646.
- [5] T. Arifqi, N. Suarna, dan W. Prihartono, "Penggunaan Naive Bayes dalam Menganalisis Sentimen Ulasan Aplikasi Mcdonald's di Indonesia," *JATI : Jurnal Mahasiswa Teknik Informatika*, vol. 8, no. 2, hlm. 1949–1956, 2024, doi: 10.36040/jati.v8i2.8740.
- [6] I. A. Mastan dan Y. Toni, "Analisis Sentimen Terhadap Tempat Kuliner Ayam Gedebuk dari Komentar Pengunjung dengan Menggunakan Metode Naïve Bayes Classifier," *JBASE - Journal of Business and Audit Information Systems*, vol. 3, no. 1, hlm. 42–50, 2020, doi: 10.30813/jbase.v3i1.2062.
- [7] V. A. Permadi, "Analisis Sentimen Menggunakan Algoritma Naive Bayes Terhadap Review Restoran di Singapura," *Jurnal Buana Informatika*, vol. 11, no. 2, hlm. 141–151, 2020, doi: <https://doi.org/10.24002/jbi.v11i2.3769>.
- [8] G. Shalihah, R. Kurniawan, dan T. Suprpti, "Analisis Sentimen Ulasan Pelanggan Mie Gacoan Menggunakan Metode Naive Bayes Classifier," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 1, hlm. 593–601, 2024, doi: 10.36040/jati.v8i1.8302.
- [9] B. Bill, "What is Google Maps?," GCFGlobal. [Daring]. Tersedia pada: <https://edu.gcfglobal.org/en/google-maps/what-is-google-maps/1/>
- [10] E. Apriani, F. Oktavianalisti, L. D. H. Monasari, I. Winarni, dan I. F. Hanif, "Analisis Sentimen Penggunaan TikTok Sebagai Media Pembelajaran Menggunakan Algoritma Naïve Bayes Classifier," *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 4, no. 3, hlm. 1160–1168, 2024, doi: 10.57152/malcom.v4i3.1482.
- [11] A. Srirahayu dan L. S. Pribadie, "Review Paper Data Mining Klasifikasi Data Mining," *Jurnal Ilmiah Informatika Global*, vol. 14, no. 1, hlm. 7–12, 2023.
- [12] N. M. A. J. Astari, D. G. H. Divayana, dan G. Indrawan, "Analisis Sentimen Dokumen Twitter Mengenai Dampak Virus Corona Menggunakan Metode Naive Bayes Classifier," *Jurnal Sistem dan Informatika (JSI)*, vol. 15, no. 1, hlm. 27–29, 2020, doi: 10.30864/jsi.v15i1.332.
- [13] M. I. Petiwi, A. Triayudi, dan I. D. Sholihati, "Analisis Sentimen Gofood Berdasarkan Twitter Menggunakan Metode Naïve Bayes dan Support Vector Machine," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 6, no. 1, hlm. 542–550, 2022, doi: 10.30865/mib.v6i1.3530.
- [14] N. Charibaldi, A. Harfiani, dan O. Samuel Simanjuntak, "Comparison of the Effect of Word Normalization on Naïve Bayes Classifier and K-Nearest Neighbor Methods for Sentiment Analysis," *Inform : Jurnal Ilmiah Bidang Teknologi Informasi dan Komunikasi*, vol. 9, no. 1, hlm. 25–31, 2023, doi: 10.25139/inform.v9i1.7111.
- [15] T. Tundo dan D. N. Rachmawati, "Implementasi Algoritma Naive Bayes untuk Analisis Sentimen Terhadap Program Makan Siang Gratis," *Jurnal Indonesia : Manajemen Informatika dan Komunikasi*, vol. 5, no. 3, hlm. 2925–2939, 2024, doi: 10.35870/jimik.v5i3.978.
- [16] D. Darwis, N. Siskawati, dan Z. Abidin, "Penerapan Algoritma Naive Bayes untuk Analisis Sentimen Review Data Twitter BMKG Nasional," *Jurnal Tekno Kompak*, vol. 15, no. 1, hlm. 131, 2021, doi: 10.33365/jtk.v15i1.744.
- [17] S. D. S. Kurniawan dan A. Fauzy, "Penggunaan Naïve Bayes Classifier dalam Analisis Sentimen Ulasan Aplikasi McDonald's: Perspektif Pengguna di Indonesia," *mib*, vol. 8, no. 3, hlm. 1545, Jul 2024, doi: 10.30865/mib.v8i3.7765.
- [18] D. T. Wiyanti, N. P. Setiawati, S. Zakaria, R. S. Soemarwoto, dan I. Indrawardana, "Use of Slang Words in Tik Tok Application by Student," *Biokultur*, vol. 13, no. 1, hlm. 1–15, Jun 2024, doi: 10.20473/bk.v13i1.53986.
- [19] R. Maulid, "Mengenal Lemmatization dalam Machine Learning NLP," DQLab. [Daring]. Tersedia pada: <https://dqlab.id/mengenal-lemmatization-dalam-machine-learning-nlp>
- [20] T. Salsabilla dan D. Alita, "Analisis Sentimen Inses di Social Media menggunakan Algoritma Naïve Bayes," *Jurnal Informatika: Jurnal Pengembangan IT*, vol. 9, no. 3, hlm. 271–280, 2024, doi: 10.30591/jpit.v9i3.6611.
- [21] K. T. Putra, M. A. Hariyadi, dan C. Crysdiyan, "Perbandingan Feature Extraction TF-IDF dan BOW untuk Analisis Sentimen Berbasis SVM," *Jurnal Cahaya Mandalika (JCM)*, vol. 3, no. 2, hlm. 1449–1463, 2023.
- [22] M. G. Pradana, "Penggunaan Fitur Wordcloud dan Document Term Matrix dalam Text Mining," *Jurnal Ilmiah Informatika (JIF)*, vol. 8, no. 1, hlm. 38–43, 2020, doi: 10.33884/jif.v8i01.1838.
- [23] M. Fernanda dan N. Fathoni, "Perbandingan Performa Labeling Lexicon InSet dan VADER pada Analisa Sentimen Rohingya di Aplikasi X dengan SVM," *Jurnal Informatika dan Sains Teknologi*, vol. 1, no. 3, hlm. 62–76, 2024, doi: 10.62951/modem.v1i3.112.
- [24] D. Normawati dan S. A. Prayogi, "Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter," *Jurnal Sains Komputer & Informatika (J-SAKTI)*, vol. 5, no. 2, hlm. 697–711, 2021, doi: 10.30645/j-sakti.v5i2.369.
- [25] F. Amaliah dan I. K. D. Nuryana, "Perbandingan Akurasi Metode Lexicon Based Dan Naive Bayes Classifier Pada Analisis Sentimen Pendapat Masyarakat Terhadap Aplikasi Investasi Pada Media Twitter," *Journal of Informatics and Computer Science (JINACS)*, vol. 3, no. 03, hlm. 384–393, 2022, doi: 10.26740/jinacs.v3n03.p384-393.
- [26] A. Miftahusalam, A. F. Nuraini, A. A. Khoirunisa, dan H. Pratiwi, "Perbandingan Algoritma Random Forest, Naïve Bayes, dan Support Vector Machine Pada Analisis Sentimen Twitter Mengenai Opini Masyarakat Terhadap Penghapusan Tenaga Honorer," *Seminar Nasional Official Statistics*, vol. 2022, no. 1, hlm. 563–572, 2022.
- [27] D. Septiani dan I. Isabela, "Analisis Term Frequency Inverse Document Frequency (TF-IDF) Dalam Temu

Kembali Informasi Pada Dokumen Teks.,” *SINTESIA: Jurnal Sistem dan Teknologi Informasi Indonesia*, vol. 1, no. 2, 2022, [Daring]. Tersedia pada: <https://journal.unj.ac.id/unj/index.php/SINTESIA/article/view/39364>

