

SENTIMEN ANALISIS EVALUASI PENGGUNA APLIKASI ORBIT TELKOMSEL PADA ULASAN PLAYSTORE MENGGUNAKAN ALGORITMA NAÏVE BAYES

Vika Febri Lestari*¹⁾, Primandani Arsi²⁾, Pungkas Subarkah³⁾

1. Teknik Informatika, Ilmu Komputer, Universitas Amikom Purwokerto, Indonesia
2. Teknik Informatika, Ilmu Komputer, Universitas Amikom Purwokerto, Indonesia
3. Teknik Informatika, Ilmu Komputer, Universitas Amikom Purwokerto, Indonesia

Article Info

Kata Kunci: Analisis Sentimen; Orbit
Telkomsel; Naïve Bayes Classifie

Keywords: Orbit Telkomsel; Naïve Bayes;
Sentiment Analysis

Article history:

Received 19 September 2024

Revised 10 Oktober 2024

Accepted 7 November 2024

Available online 1 December 2024

DOI :

<https://doi.org/10.29100/jipi.v9i4.5538>

* Corresponding author.

Vika Febri Lestari

E-mail address:

vikafebri0@gmail.com

ABSTRAK

Dalam era digital, aplikasi mobile, khususnya aplikasi internet rumah, memainkan peran penting dalam mendukung aktivitas harian pengguna. Studi ini bertujuan untuk melakukan analisis sentimen terhadap aplikasi Telkomsel Orbit, layanan internet rumah serba digital yang menyediakan koneksi melalui modem WiFi dengan jaringan 4G LTE. Metodologi penelitian melibatkan proses scraping data, preprocessing teks, dan penerapan algoritma klasifikasi Naïve Bayes untuk menganalisis ulasan dan peringkat yang diberikan pengguna di Play Store. Telkomsel Orbit menawarkan fitur unggulan seperti pengaturan Wi-Fi melalui aplikasi MyOrbit, WiFi tamu, dan penyesuaian filter website. Hasil penelitian menunjukkan bahwa penerapan metode Naïve Bayes menghasilkan tingkat ketepatan mencapai 79%, dengan presisi sebesar 80% dan recall 79%. Temuan ini menunjukkan bahwa mayoritas ulasan yang diberikan oleh pengguna cenderung positif terhadap aplikasi Telkomsel Orbit. Hal ini mengindikasikan bahwa layanan tersebut dinilai secara positif oleh pengguna, dengan fitur-fiturnya yang dianggap bermanfaat dan kualitas layanannya yang memuaskan. Penelitian ini memberikan wawasan yang berharga bagi Telkomsel Orbit untuk memahami umpan balik pengguna dan terus meningkatkan pengalaman pengguna mereka.

ABSTRACT

In the digital era, mobile applications, especially home internet applications, play a crucial role in supporting users' daily activities. This study aims to conduct sentiment analysis on the Telkomsel Orbit application, a comprehensive digital home internet service that provides connectivity through WiFi modems with 4G LTE networks. The research methodology involves data scraping, text preprocessing, and the implementation of the Naïve Bayes classification algorithm to analyze user reviews and ratings on the Play Store platform. Telkomsel Orbit offers prominent features such as WiFi management through the MyOrbit application, guest WiFi, and customizable website filtering. The study found that using the Naïve Bayes approach resulted in 79% accuracy, 80% precision, and 79% recall. These findings indicate that the majority of user feedback are positive towards the Telkomsel Orbit application, suggesting that the service is positively evaluated by users due to its perceived beneficial features and satisfactory service quality. This study provides valuable insights for Telkomsel Orbit to understand user feedback and continuously enhance their user experience.

I. PENDAHULUAN

PADA era globalisasi saat ini ketersediaan dan keterbukaan akses informasi menjadi kebutuhan dalam menunjang aktivitas manusia. seseorang dapat dengan mudah berkomunikasi dan memperoleh pengetahuan, pendidikan, serta lapangan pekerjaan.[1] Analisis sentimen merupakan suatu metode yang memanfaatkan Natural Language Processing (NLP) untuk secara otomatis menggali sikap, opini, perspektif, dan emosi yang terdapat dalam teks, audio, kicauan, dan berbagai sumber data lainnya.[2] Dari awal tahun 2000, analisis sentimen telah menjadi salah satu bidang penelitian yang paling dinamis dalam pengolahan bahasa alami.[3] Menilai secara berkelanjutan kualitas penyampaian layanan kepada pengguna sangat penting, karena peringkat dan jumlah ulasan tidak

selalu mencerminkan keberhasilan dalam memenuhi tuntutan atau mengatasi masalah pengguna.[2] Dari beberapa artikel referensi yang telah peneliti telah, metode Naïve Bayes menonjol dengan akurasi yang mencolok dan waktu komputasi yang lebih efisien dalam menganalisis sentimen jika dibandingkan dengan metode klasifikasi alternatif.[4] Sehingga dalam penelitian kali ini akan menggunakan metode atau algoritma *Naïve Bayes*.

Telkomsel Orbit merupakan penyedia layanan internet rumah serba digital di Indonesia, memanfaatkan modem WiFi sebagai perangkatnya. Mereka menawarkan layanan internet rumah berkualitas tinggi yang merata di seluruh Indonesia, termasuk juga layanan internet portabel dengan harga terjangkau yang mudah diakses oleh masyarakat (sumber: <https://www.myorbit.id>). Telkomsel Orbit menggunakan modem Wi-Fi dan menyajikan jaringan internet berkualitas 4G LTE. Beberapa fitur unggulannya melibatkan pengaturan Wi-Fi melalui aplikasi My-Orbit, penyediaan WiFi tamu, kemampuan untuk mengatur jadwal pemakaian, statistik penggunaan, serta penyesuaian filter website.[5] PT. Telkomsel Tbk merupakan salah satu perusahaan telekomunikasi terbesar di Indonesia. Telkomsel memiliki jaringan terluas yaitu 98% dari total wilayah yang ada di seluruh Indonesia dan merupakan perusahaan operator selular yang mempunyai pelanggan terbanyak di Indonesia.[6]

Platform Play Store pada sistem operasi Android menyediakan beragam aplikasi yang dikembangkan oleh para pengembang dan dapat digunakan oleh pengguna sesuai kebutuhan mereka. Meskipun tersedia berbagai aplikasi dengan fungsi dan manfaat yang beragam, banyaknya opsi dapat menjadi tantangan bagi pengguna yang ingin memilih aplikasi yang paling sesuai dengan kebutuhan mereka. Oleh karena itu, analisis aplikasi di Play Store berdasarkan rating dan jenisnya dapat membantu pengguna dalam membuat pilihan yang tepat sesuai dengan popularitas pada waktu tertentu. Selain memberikan manfaat bagi pengguna, analisis ini juga dapat membantu pengembangan aplikasi untuk meningkatkan kualitas dan popularitas aplikasi yang mereka kembangkan.[7]

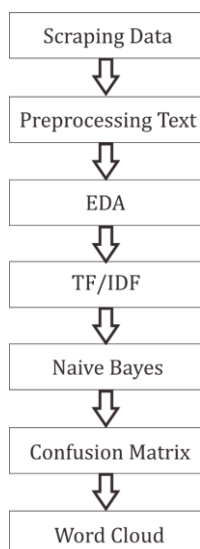
Metode Klasifikasi Naïve Bayes adalah suatu pendekatan statistik yang digunakan untuk memperkirakan probabilitas keanggotaan suatu kelas. Multinomial Naïve Bayes, juga dikenal sebagai Naïve Bayes, adalah suatu teknik klasifikasi yang menggunakan prinsip probabilitas dan statistic, dimana metode ini adalah suatu bentuk penyederhanaan dari algoritma Bayes yang efektif dalam mengklasifikasikan teks atau dokumen.[8] Klasifikasi Bayesian ini bergantung pada teorema Bayes, yang memiliki kemampuan klasifikasi yang sebanding dengan decision tree dan jaringan saraf. Terbukti bahwa klasifikasi Bayesian menunjukkan akurasi dan kecepatan tinggi ketika diterapkan pada basis data dengan volume besar. Pendekatan statistik Bayes merupakan cara untuk melakukan inferensi induktif pada masalah klasifikasi.[9] Dalam penggunaan Naïve Bayes untuk klasifikasi, kategori suatu dokumen akan ditetapkan berdasarkan kemunculan fitur atau kata-kata di dalam dokumen tersebut, [8] ada juga klasifikasi dokumen yang digunakan untuk melakukan filtrasi spam (spam filtering).[10] Salah satu teknik penyaringan email yang paling umum digunakan adalah metode Bayes, yang juga dikenal sebagai penyaringan Bayesian. Teknik spam filtering ini menggunakan teorema probabilitas, yakni teorema Bayes, untuk memproyeksikan probabilitas kejadian di masa mendatang berdasarkan informasi yang ada di masa lampau.[8] Dari penelitian terdahulu yang menggunakan metode Naïve Bayes untuk aplikasi orbit telkomsel yaitu penelitian Nizam (2023) pada ulasan orbit telkomsel mencapai akurasi 70.21%.[9] Kemudian contoh penelitian yang selanjutnya peneliti Budi dengan mendapatkan nilai akurasi 81.58%.[10]

Tujuan dari penelitian ini adalah untuk menganalisis aplikasi Telkomsel Orbit melalui ulasan dan rating di aplikasi Playstore menggunakan metode atau algoritma *Naïve Bayes*. Diharapkan hasil dari penelitian ini dapat mengevaluasi pengembangan dari aplikasi Telkomsel orbit untuk fitur-fitur yang masih menjadi concern dari pengguna yang sudah diambil garis besarnya seperti menu login pada autentikasi masih banyak terjadi error sehingga pengguna masih banyak yang gagal dalam proses tersebut, kemudian concern aplikasi lemot dan proses transaksi yang masih gagal dari pengguna terkait aplikasi ini menjadi tugas dari development agar memecahkan masalah tersebut sehingga dapat meningkatkan kualitas dari aplikasi Telkomsel Orbit.

II. METODE

A. Kerangka Studi

Dalam rangka penelitian ini, perlu dilakukan serangkaian langkah-langkah untuk mencapai hasil yang terbaik. Langkah-langkah tersebut melibatkan scraping data atau pengumpulan data, preprocessing teks, pembobotan kata dengan menggunakan metode TF-IDF, pengelompokan data dengan algoritma Naïve Bayes, evaluasi data melalui Confusion Matrix, dan visualisasi data dengan memanfaatkan WordCloud, yang dapat ditemukan pada Gambar 1 yang tersedia di bawah ini.[4]



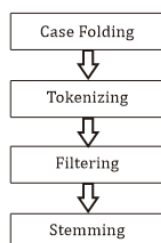
Gambar. 1. Framework Penelitian

B. Teknik Scraping Data

Pada tahap ini, proses pengumpulan data dilakukan dengan memanfaatkan Bahasa pemrograman Python menggunakan platform Google Colaboratory dan library google-play-scraper. Selama pelaksanaan pengambilan data, peneliti mengumpulkan informasi penting seperti nama pengguna, komentar, peringkat, jumlah suka, dan sebagainya, lalu menyimpan hasilnya dalam format file .csv.[11]

C. Preprocessing Text

Dalam langkah ini, kami akan memproses data mentah yang terdiri dari 22.642 baris dan 4 kolom menggunakan empat tahap yaitu: Case Folding, Tokenizing, Filtering, dan Stemming. Rincian proses ini dapat diidentifikasi pada Gambar 2 yang tersedia di bawah ini.[4]



Gambar. 2. Tahap Preprocessing Text

1. Case Folding

Proses untuk menyetarakan teks dengan cara mengkonversi semua menjadi huruf kecil maupun huruf besar sehingga dalam sistem terbaca semua setara.

2. Tokenizing

Proses selanjutnya adalah tokekizing, tahapan ini adalah memecah atau mengelompokan suatu kalimat menjadi beberapa kata.

3. Filtering

Setelah proses tokenizing selanjutnya masuk ke tahap filtering, tahapan ini adalah menghilangkan kata kata yang tidak dibutuhkan dalam proses selanjutnya seperti penggunaan kata penghubung seperti 'dan', 'yang', atau 'ke', serta hanya mempertahankan kata-kata kunci yang bermakna.[12]

4. Stemming

Proses selanjutnya yaitu stemming, dimana pada tahapan ini yaitu menghilang kata yang memiliki imbuhan sehingga kata tersebut menjadi kata dasar.

D. *Exploratory Data Analysis* (EDA)

Tahapan ini merujuk pada suatu metode dalam statistika dan ilmu data yang digunakan untuk menganalisis serta memahami karakteristik dasar dari kumpulan data tertentu. EDA bertujuan untuk meraih pemahaman awal tentang distribusi, pola, dan hubungan dalam data sebelum melibatkan metode analisis statistik yang lebih mendalam.[13]

E. *Term Frequency Inverse Document Frequency* (TF-IDF)

Metode TF-IDF adalah cara untuk memberi bobot pada hubungan sebuah kata (term) dengan sebuah dokumen. Metode ini menggabungkan dua konsep untuk menghitung bobot, yaitu frekuensi kemunculan suatu kata dalam dokumen tertentu dan frekuensi kebalikan dari dokumen yang mengandung kata tersebut. Frekuensi kemunculan kata dalam dokumen yang disediakan menunjukkan betapa pentingnya kata tersebut dalam dokumen. Frekuensi dokumen yang berisi kata tersebut menunjukkan seberapa umum kata tersebut. Jadi, bobot hubungan antara sebuah kata dan dokumen akan tinggi jika frekuensi kata dalam dokumen juga tinggi dan frekuensi keseluruhan dokumen yang mengandung kata rendah dalam dokumen.[14] Untuk menghitung TF-IDF dapat dirumuskan menggunakan persamaan 1.

$$TF.IDF_{std}(t) = tf_d^t \times \log \frac{N}{df^t} \quad (1)$$

Di mana tf_d^t menunjukkan frekuensi kemunculan istilah t dalam dokumen d , N menunjukkan jumlah total dokumen dalam korpus, dan df^t menunjukkan jumlah dokumen yang mengandung istilah t .[15]

F. Algoritma Naïve Bayes

Langkah berikutnya setelah penerapan TF/IDF adalah tahap klasifikasi model menggunakan algoritma atau metode Naïve Bayes. Algoritma Naïve Bayes dipilih karena dapat membuat klasifikasi yang sederhana namun efektif dalam mengenali berbagai jenis data dengan tingkat akurasi yang tinggi.[16] Penerapan klasifikasi ini pada basis data dengan dataset yang besar dapat memberikan tingkat akurasi dan kecepatan yang tinggi.[7] Karena keunggulan dari metodenya jadi dipilih karena memiliki kemampuan yang bagus untuk memproses 22.642 data dengan cepat dan akurat. Terdapat dua tahapan klasifikasi dokumen ulasan aplikasi Orbit Telkomsel pada penelitian ini. Tahap pertama adalah melibatkan pelatihan dokumen yang sudah terkategori, sementara tahapan kedua melibatkan pengujian dengan mengklasifikasikan dokumen yang belum terkategori. Meskipun sederhana, pengklasifikasi Naive Bayesian sering kali memberikan hasil yang sangat baik dan menjadi pilihan yang populer karena seringkali melampaui metode klasifikasi yang lebih kompleks.[17] Dengan menggunakan metode naive Bayes, pengguna dapat mencapai probabilitas Bayesian dan menghasilkan pernyataan probabilitas. Metode naive Bayes dapat dijabarkan dalam bentuk Persamaan 2.

$$P(H|X) = \frac{P(H|X) P(H)}{P(X)} \quad (2)$$

Dalam formula ini, variabel X mewakili data dokumen dengan kelas yang belum diketahui, sementara X merujuk pada hipotesis yang menentukan kelas spesifik dari data X . $P(H|X)$ mengindikasikan probabilitas hipotesis H berdasarkan kondisi X , yang juga dikenal sebagai probabilitas posteriori. Di sisi lain, $P(H)$ menunjukkan probabilitas hipotesis H sebelum mempertimbangkan data X , yang disebut sebagai probabilitas prior. Kemudian, probabilitas data X jika hipotesis H benar dinyatakan sebagai $P(H|X)$, sedangkan $P(X)$ adalah probabilitas data X secara keseluruhan.[18]

G. Confusion Matrix

Tahapan selanjutnya adalah confusion matrix, di mana kita mengevaluasi performa model klasifikasi dengan menyajikan ringkasan hasil prediksi pada dataset uji. Matriks ini biasanya digunakan untuk tugas klasifikasi yang melibatkan dua atau lebih kelas dan melibatkan empat istilah kritis: Nilai True Positive (TP), True Negative (TN), False Positive (FP), dan False Negative (FN). Setelah memperoleh hasil dari keempat istilah tersebut, kita dapat menghitung akurasi, presisi, dan recall untuk mengevaluasi performa model.[4] Pada tahapan Confusion Matrix ini dapat dihitung menggunakan rumus berikut :

$$accuracy = \frac{TP + TN}{Total} \quad (3)$$

Penjelasan :

TP (True Positive) adalah data dengan prediksi dan nilai aktual yang sama-sama positif.

TN (True Negative) adalah data di mana prediksi dan nilai aktual keduanya negatif.

$$presisi = \frac{TP}{TP + FP} \quad (4)$$

Penjelasan :

TP (True Positive) adalah data yang menunjukkan prediksi positif dengan hasil aktual juga positif.

FP (False Positive) adalah data yang menunjukkan prediksi positif namun hasil aktualnya negatif.

$$recall = \frac{TP}{TP + FN} \quad (5)$$

Penjelasan :

TP (True Positive) menunjukkan data dengan prediksi positif yang benar dan nilai aktualnya juga positif

FN (False Negative) mengacu pada data dengan prediksi negatif, tetapi nilai aktualnya positif.[19]

H. WordCloud

Tahap berikutnya adalah melakukan evaluasi menggunakan wordcloud, di mana seluruh sentimen ulasan positif dan negatif terhadap aplikasi Orbit Telkomsel dapat diilustrasikan secara visual pada wordcloud.[20]

III. HASIL DAN PEMBAHASAN

A. Hasil Scraping Data

Penelitian ini mengumpulkan data melalui metode scraping dengan memanfaatkan Google Collaboratory. Proses ini melibatkan instalasi google-play-scraper dan penggunaan bahasa pemrograman Python untuk mengumpulkan semua ulasan aplikasi Orbit Telkomsel dari Playstore. Data yang terkumpul mencakup: Username, rating, waktu, dan ulasan, dengan total 55.800 entri. Setelah dilakukan proses pembersihan data, jumlah entri yang tersisa menjadi 22.642, seperti yang ditunjukkan pada gambar di bawah ini.

```
[2] !pip install google-play-scraper

Collecting google-play-scraper
  Downloading google_play_scraper-1.2.4-py3-none-any.whl (28 kB)
  Installing collected packages: google-play-scraper
  Successfully installed google-play-scraper-1.2.4

[3] from google_play_scraper import app

import pandas as pd

import numpy as np

from google_play_scraper import Sort, reviews_all

# Definisikan ID aplikasi dari tautan
app_id = 'com.myorbit'

# Pengaturan lokal
lang = 'id' # Bahasa (default adalah 'en')
country = 'id' # Negara (default adalah 'us')

# Dapatkan ulasan
result = reviews_all(
    app_id,
    sleep_milliseconds=0, # Waktu tidur dalam milidetik (default adalah 0)
    lang=lang,
    country=country,
    sort=Sort.NEWEST, # Urutkan berdasarkan ulasan terbaru (default adalah Sort.MOST_RELEVANT)
    count=100, # Jumlah ulasan yang akan diambil (default adalah 100)
    filter_score_with=None # Filter berdasarkan skor (default adalah tidak ada; 1 hingga 5 untuk skor tertentu)
)
```

Gambar. 3. Process Scraping Data

	userName	score	at	content
0	Agustinus Bua Padang	1	2023-11-29 12:32:40	Semoga kedepannya ada menu pembatasan pengguna...
1	Yulia Lapendala	1	2023-11-29 12:09:11	Tidak dapat mengubah Password, tidak bisa meng...
2	Haerul Haerul	1	2023-11-29 11:10:38	Sy mengisi paket data transaksi sukses kok pak...
3	Timothy Evan	1	2023-11-29 10:51:56	Minta kode verifikasi gak dikirim kirim anj
4	botski obot	5	2023-11-29 10:17:41	bagus banget

Gambar. 4. Hasil Data Mentah

B. Hasil Preprocessing Data

Setelah melakukan Scraping data selanjutnya yaitu mulai preprocessing data dengan menyamaratakan semua format huruf nya, menggunakan 4 tahap yaitu: case folding, filtering, tokenizing, dan stemming.[21]

TABEL I
HASIL CASE FOLDING

Sebelum	Sesudah
Mantaaap.... gak pke ribet...	mantaaap gak pke ribet
Kasih 4 bintang dulu ya	kasih bintang dulu ya
Mantap ! Setelah download aplikasi gak terjadi apa2	mantap setelah download aplikasi gak terjadi apa

Tabel 1 ini menunjukkan hasil dari case folding untuk text kapital diubah menjadi menjadi huruf kecil semua bertujuan untuk menyamakan hasil teks. Dalam proses ini, juga ada penghapusan tanda baca atau karakter-karakter tertentu yang tidak diperlukan dari kata-kata.[22]

TABEL II
HASIL TOKENIZING

Sebelum	Sesudah
mantaaap gak pke ribet	[mantaaap,gak,pke,ribet]
kasih bintang dulu ya	[kasih,bintang,dulu,ya]
mantap setelah download aplikasi gak terjadi apa	[mantap,setelah,download,aplikasi,gak,terjadi,apa]

Selanjutnya yaitu Tabel 2 ini bertujuan untuk mengurai atau memisahkan suatu kalimat menjadi bagian-bagian yang disusun berdasarkan spasi.[22]

TABEL III
HASIL FILTERING

Sebelum	Sesudah
Mantaaap.... gak pke ribet...	mantaaap gak pke ribet
Kasih 4 bintang dulu ya	kasih bintang dulu ya
Mantap ! Setelah download aplikasi gak terjadi apa2	mantap setelah download aplikasi gak terjadi apa

Kemudian Tabel 3 yaitu menyaring kata yang tidak diperlukan lagi dalam tahap selanjutnya.

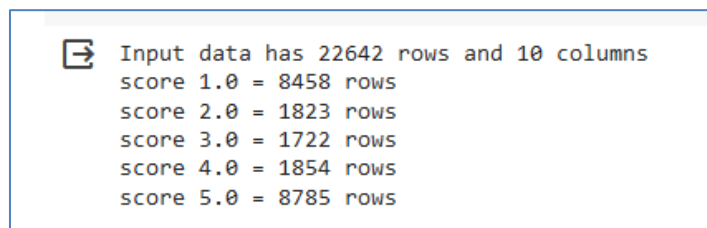
TABEL IV
HASIL STEMMING

Sebelum	Sesudah
mantaaap gak pke ribet	mantaaap gak pke ribet
kasih bintang dulu ya	kasih bintang ya
mantap setelah download aplikasi gak terjadi apa	mantap download aplikasi gak

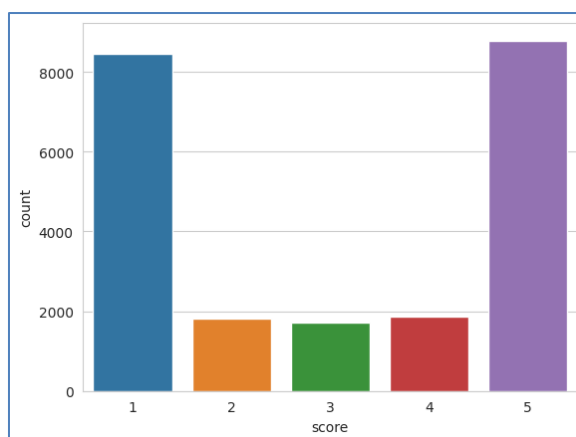
Dan terakhir yaitu Tabel 4 pada proses ini bertujuan untuk menghapus afiks, preposisi, kata ganti, dan kata yang dianggap tidak penting menurut KBBI. [22]

C. Hasil EDA

Langkah-langkah ini diperlukan untuk memastikan sistem dapat mendeteksi pola, memverifikasi asumsi-asumsi dasar, mengidentifikasi anomali, serta menguji hipotesis dari data yang ada guna meningkatkan akurasi.[23] Berikut adalah visualisasi dari hasil metode Analisis Data Eksploratif (EDA).



Gambar. 5. Representasi data dari hasil identifikasi mengenai jumlah entri dan penilaian aplikasi Orbit Telkomsel



Gambar. 6. Visualisasi data yang memperlihatkan hasil identifikasi terhadap jumlah data dan penilaian (rating) aplikasi Orbit Telkomsel

Dapat dijelaskan menurut Gambar. 6. Bahwa penilaian score/ bintang 1 dan bintang 5 hampir sama meskipun selisih 327, yang dimana masih dinilai lebih unggul bintang 5. Untuk penilaian ini sudah menjadi tugas pengembang aplikasi untuk memecahkan masalah yang disubmit pengguna di rating 1 atau kurang memuaskan dan masih terdapat problem yang harus diperbaiki.

D. TF/IDF

Tahap ini merupakan persiapan penting sebelum menerapkan Naive Bayes karena metode tersebut hanya dapat memproses data yang diwakili dalam bentuk numerik. Oleh karena itu, digunakan TF-IDF sebagai alat untuk mengubah dokumen teks menjadi representasi numeris.[24] Berikut adalah langkah-langkah yang harus diikuti dalam proses vektorisasi menggunakan TF-IDF

1. Dalam penyiapan kode program seperti yang ditunjukkan dalam gambar 7, peneliti mengambil fitur dan label dari DataFrame yang tersedia. Fitur yang terpilih terdiri dari kolom 'lemmatized_review', 'content_len', dan 'punct', yang mewakili teks yang telah dilematkan, panjang teks tanpa spasi, dan persentase tanda baca dalam teks, masing-masing. Label diambil dari kolom 'label'. Setelah pemilihan fitur dan label, penggunaan perintah print dilakukan untuk menampilkan dimensi atau bentuk dari fitur (X) dan label (y), bertujuan untuk memastikan kebenaran pengaturan data sebelum proses analisis atau pemodelan dimulai..

```
[17] # Memilih fitur dan label dari dataframe
X = df[['lemmatized_review', 'content_len', 'punct']]
y = df['label']

# Menampilkan dimensi (bentuk) dari fitur (X) dan label (y)
print(X.shape)
print(y.shape)
```

```
(22642, 3)
(22642,)
```

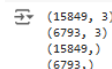
Gambar. 7. Listing program pemiloihan fitur dan label Data Frame

- Dalam implementasi kode yang serupa dengan gambar 8, peneliti memanfaatkan fungsi `train_test_split` dari pustaka `scikit-learn` untuk memisahkan dataset menjadi dua bagian, yaitu data latih dan data uji. Variabel `X` dan `y` digunakan untuk mewakili fitur dan label. Dengan menetapkan nilai `test_size=0.3`, 30% dari data dialokasikan untuk pengujian, sementara sisanya, yaitu 70%, digunakan untuk pelatihan model. Parameter `random_state=0` diatur untuk memastikan konsistensi dalam pembagian data setiap kali kode dieksekusi. Setelah proses pemisahan, perintah `print` digunakan untuk menampilkan dimensi dari data latih dan data uji, termasuk jumlah baris dan kolom untuk fitur serta jumlah baris untuk label. Langkah ini penting untuk memastikan kesesuaian pembagian data dengan kebutuhan model pembelajaran mesin.

```

# Memisahkan data menjadi data pelatihan dan data pengujian menggunakan train_test_split
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=0)

# Menampilkan dimensi (bentuk) data pelatihan dan data pengujian
print(X_train.shape) # Menampilkan dimensi data pelatihan fitur (jumlah baris, jumlah kolom)
print(X_test.shape)  # Menampilkan dimensi data pengujian fitur (jumlah baris, jumlah kolom)
print(y_train.shape) # Menampilkan dimensi data pelatihan label (jumlah baris)
print(y_test.shape)  # Menampilkan dimensi data pengujian label (jumlah baris)
    
```



Gambar. 8. Implementasi Kode untuk Pemisahan Dataset

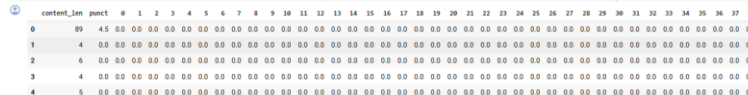
- Dalam rangka penelitian ini, `TfidfVectorizer` dari library `scikit-learn` telah dimanfaatkan oleh peneliti untuk mengubah teks ulasan yang sudah dilamatisasi menjadi representasi vektor TF-IDF. Penggunaan parameter `max_df` dan `min_df` bertujuan untuk mengesampingkan kata-kata yang muncul terlalu sering atau terlalu jarang dalam dokumen. Setelah proses vektorisasi diterapkan pada data pelatihan (`X_train['lemmatized_review']`), matriks TF-IDF untuk data pelatihan (`tfidf_train`) berhasil diperoleh. Langkah yang sama dilakukan untuk data pengujian (`X_test['lemmatized_review']`), menghasilkan matriks TF-IDF untuk data pengujian (`tfidf_test`). Setelah itu, dilakukan penggabungan matriks TF-IDF dengan fitur tambahan, yakni '`content_len`' dan '`punct`', menggunakan fungsi `pd.concat`. Ini menghasilkan DataFrame baru (`X_train_vect` dan `X_test_vect`) yang mencakup semua fitur tersebut bersama dengan matriks TF-IDF yang sesuai. Dengan menampilkan beberapa baris pertama dari `X_train_vect` menggunakan fungsi `head()`, peneliti dapat memverifikasi keberhasilan proses vektorisasi dan mendapatkan gambaran awal tentang representasi data yang akan digunakan dalam tahap analisis atau pemodelan lebih lanjut..

```

[ ] from sklearn.feature_extraction.text import TfidfVectorizer
tfidf = TfidfVectorizer(max_df = 0.5, min_df = 2)
# ignore terms that occur in more than 50% documents and the ones that occur in less than 2
tfidf_train = tfidf.fit_transform(X_train['lemmatized_review'])
tfidf_test = tfidf.transform(X_test['lemmatized_review'])

X_train_vect = pd.concat([X_train[['content_len', 'punct']].reset_index(drop=True),
pd.DataFrame(tfidf_train.toarray()), axis=1]
X_test_vect = pd.concat([X_test[['content_len', 'punct']].reset_index(drop=True),
pd.DataFrame(tfidf_test.toarray()), axis=1]

X_train_vect.head()
    
```

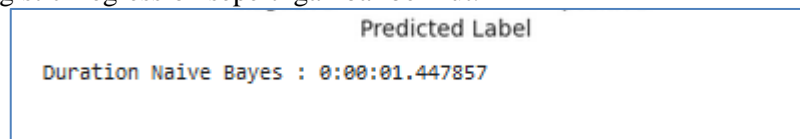


Gambar. 9. Menampilkan kode program yang mengubah kata-kata menjadi representasi vektor

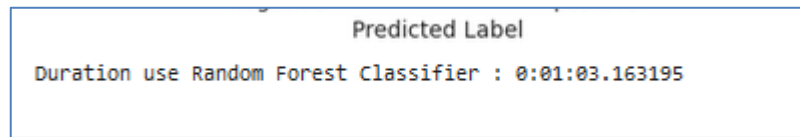
E. Hasil Naïve Bayes

Setelah melalui beberapa tahapan seperti preprocessing, EDA dan pengujian TF/IDF dilakukan, maka selanjutnya proses klasifikasi, dilakukan penentuan apakah data yang diuji memiliki sentimen positif atau negatif. Algoritma Naïve Bayes digunakan pada tahap ini. Hasil klasifikasi kemudian dikelompokkan sebagai memiliki kecenderungan positif atau negatif berdasarkan perbandingan nilai probabilitas posterior. Jika nilai probabilitas posterior positif lebih besar daripada nilai probabilitas posterior negatif, maka hasil klasifikasi akan memiliki kecenderungan positif, demikian pula sebaliknya. Hasil penjabarannya akan dijelaskan pada tahapan Confusion Matrix, kemudian setelah dilakukan pengujian menggunakan naïve bayes dimana metode ini terkenal dengan

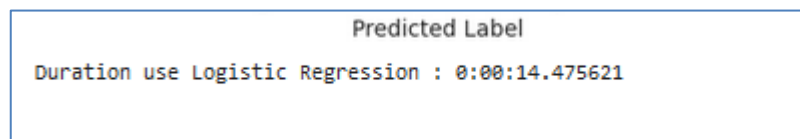
kecepatannya makan diperoleh hasil yang sangat cepat jika dibandingkan dengan metode lain seperti Random Forest Classifier dan Logistic Regression seperti gambar berikut:



Gambar. 10. Duration use Naive Bayes



Gambar. 11. Duartion use Random Forest Classifier

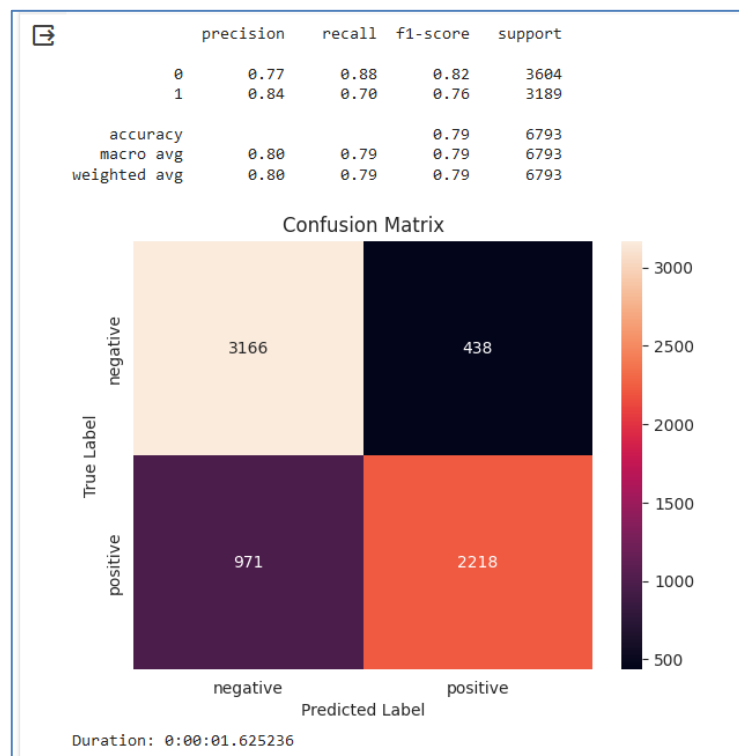


Gambar. 12. Duration use Logistic Regression

Setelah menguji data, ditemukan bahwa waktu pengujiannya adalah 1.6 detik. Ini menunjukkan bahwa algoritma Naïve Bayes berkinerja sangat baik dalam hal durasi pengujian.

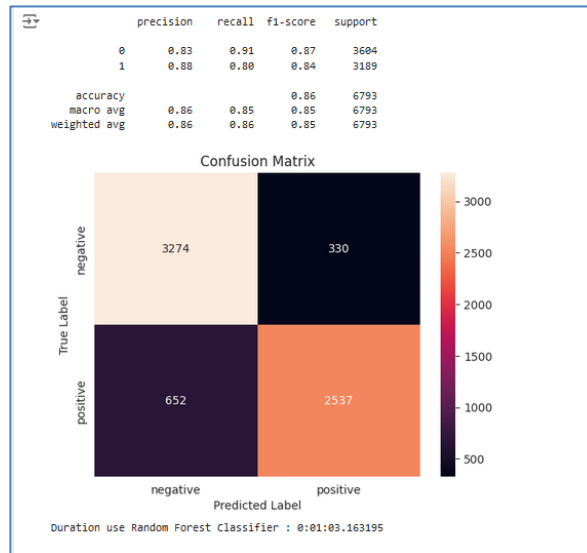
F. Hasil Confusion Matrix

Setelah pengujian menggunakan Naïve Bayes kemudian masuk ke tahap evaluasi menggunakan Confusion Matrix.

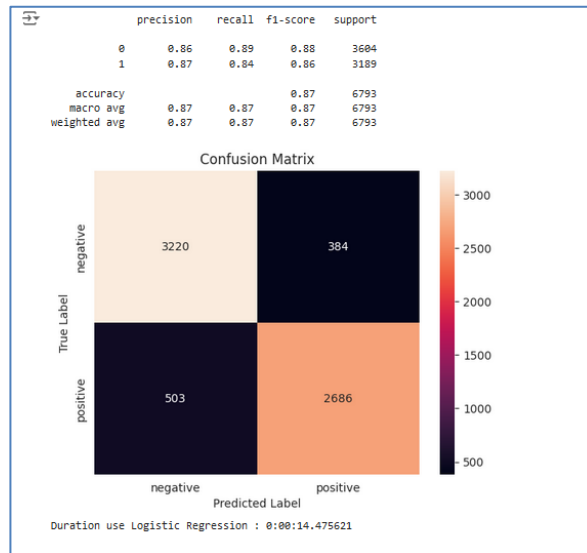


Gambar. 13. Hasil Confusion Matrix Naïve Bayes

Setelah mendapatkan nilai True Positive (TP), True Negative (TN), False Positive (FP), dan False Negative (FN), dilakukan pengujian pada Python dengan mengimplementasikan rumus-rumus yang telah dijelaskan sebelumnya. Hasil pengujian menunjukkan tingkat akurasi sebesar 79%, presisi sebesar 80%, dan recall sebesar 79%.



Gambar. 16. Confussion Matrix Random Forest Classifier



Gambar. 17. Confussion Matrix Logistic Regression

Dari hasil analisis pada gambar di atas, terlihat bahwa metode Random Forest Classifier dan Logistic Regression menghasilkan tingkat akurasi yang tinggi, melebihi 80%. Namun, jika dilihat dari segi durasi, metode Naïve Bayes tetap unggul dalam kecepatan pengujian.

Selain itu, dalam penelitiannya, Made Widiarta menyimpulkan bahwa kualitas Telkomsel Orbit Star 2 lebih baik daripada Modem USB TP-Link MR3420. Temuan ini memiliki potensi untuk memengaruhi kinerja dan kualitas produk, sehingga mendorong pengguna untuk memberikan ulasan positif terhadap aplikasi tersebut.

IV. KESIMPULAN

Berdasarkan penelitian yang sudah dilakukan dapat disimpulkan bahwa menggunakan algoritma naïve bayes berkinerja baik dan cepat dalam waktu 1.6 detik dan hasil dari confusion matrix nya menunjukkan tingkat akurasi sebesar 79%, presisi sebesar 80%, dan recall sebesar 79%.

Tingkat akurasi sebesar 79% mencerminkan seberapa baik model dapat mengklasifikasikan data secara benar secara keseluruhan. Presisi sebesar 80% memberikan informasi tentang seberapa tepat model dalam mengidentifikasi hasil positif, sementara recall sebesar 79% menggambarkan seberapa efektif model dapat mendeteksi seluruh kejadian positif dalam dataset.

DAFTAR PUSTAKA

- [1] I. Made Widiarta, D. Ardiansyah, L. Putu, S. Yuliasuti, and H. Artikel, "Analisis Perbandingan Performa Modem Usb Tp-Link Dan Telkomsel Orbit Menggunakan Metode QoS," *Digital Transformation Technology (Digitech) | e*, vol. 3, no. 1, 2023, doi: 10.47709/digitech.v3i1.2512.

- [2] P. Br Sihotang, F. Dameka Br Sitanggang, N. Azriansyah, and E. Indra, "PENERAPAN NATURAL LANGUAGE PROCESSING UNTUK ANALISIS SENTIMEN TERHADAP APLIKASI STREAMING," 2023.
- [3] 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.
- [4] G. Darmawan, S. Alam, M. Imam Sulisty, P. Studi Teknik Informatika, S. Tinggi Teknologi Wastukencana Purwakarta, and R. Artikel, "ANALISIS SENTIMEN BERDASARKAN ULASAN PENGGUNA APLIKASI MYPERTAMINA PADA GOOGLE PLAYSTORE MENGGUNAKAN METODE NAIVE BAYES INFO ARTIKEL ABSTRAK," vol. 2, no. 3, pp. 100–108, 2023, doi: 10.55123.
- [5] N. H. Aini, M. Lastrim, E. Mailasari, and E. Damayanti, "Metode Naive Bayes Classifier (NBC) Pada Klasifikasi Rekomendasi Promosi Internet Telkomsel Orbit Studi Kasus Kecamatan Prabumulih Selatan".
- [6] U. Esa and U. Jakarta, "STRATEGI PUBLIC RELATIONS DALAM MENINGKATKAN CITRA PERUSAHAAN PADA PT. TELKOMSEL TBK Sartika, Ikbal Rachmat", [Online]. Available: <http://sosains.greenvest.co.id>
- [7] A. Jalu Narendra Kisma and C. Raras Ajeng Widiawati, "Analisis Aplikasi Di Playstore Berdasarkan Rating Dan Type Menggunakan Naive Bayes Dan Logistik Regresi," vol. 10, no. 2, pp. 174–184, 2023, [Online]. Available: <http://jurnal.mdp.ac.id>
- [8] A. Wibisono Informatika, "FILTERING SPAM EMAIL MENGGUNAKAN METODE NAIVE BAYES."
- [9] N. Haqqizar and T. Nur Larasyanti, "Analisis Sentimen Terhadap Layanan Provider Telekomunikasi Telkomsel Di Twitter Dengan Metode Naive Bayes," *Seminar Nasional Rekayasa dan Teknologi*, vol. 27, 2019, [Online]. Available: <https://t.co/TTpms24klj>
- [10] B. Kurniawan Rachmat, A. Suwarisman, I. Afriyanti, A. Wahyudi, and D. D. Saputra, "Analisis Sentimen Complain dan Bukan Complain pada Twitter Telkomsel dengan SMOTE dan Naive Bayes," *Jurnal Teknologi Informasi dan Komunikasi*, vol. 7, no. 1, 2023, doi: 10.35870/jti.
- [11] "Analisis Sentimen Pada Ulasan Aplikasi Home Credit dengan Metode SVM dan KNN", [Online]. Available: <https://ejournal.mediaantartika.id/index.php/jka>
- [12] M. Y. Siregar, A. Davy Wiranata, and R. A. Saputra, "Analisis Sentimen Pada Ulasan Pengguna Aplikasi Streaming Vidio Menggunakan Metode Naive Bayes," *Media Online*, vol. 4, no. 5, pp. 2419–2429, 2024, doi: 10.30865/klik.v4i5.1787.
- [13] A. Jalu Narendra Kisma and C. Raras Ajeng Widiawati, "Analisis Aplikasi Di Playstore Berdasarkan Rating Dan Type Menggunakan Naive Bayes Dan Logistik Regresi," vol. 10, no. 2, pp. 174–184, 2023, [Online]. Available: <http://jurnal.mdp.ac.id>
- [14] Merinda Lestandy, Abdurrahim Abdurrahim, and Laili Syafa'ah, "Analisis Sentimen Tweet Vaksin COVID-19 Menggunakan Recurrent Neural Network dan Naive Bayes," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 4, pp. 802–808, Aug. 2021, doi: 10.29207/resti.v5i4.3308.
- [15] S. Ghosh and M. S. Desarkar, "Class Specific TF-IDF Boosting for Short-text Classification: Application to Short-texts Generated during Disasters," in *The Web Conference 2018 - Companion of the World Wide Web Conference, WWW 2018*, Association for Computing Machinery, Inc, Apr. 2018, pp. 1629–1637. doi: 10.1145/3184558.3191621.
- [16] Y. Cahyana and A. M. Siregar, "Analisis Sentiment Pembelajaran Tatap Muka Terbatas (PTMT) Selama Pandemi Covid-19 Menggunakan Algoritma Naive Bayes."
- [17] M. A. Djamaludin, A. Triayudi, and E. Mardiani, "Analisis Sentimen Tweet KRI Nanggala 402 di Twitter menggunakan Metode Naive Bayes Classifier," *Jurnal Teknologi Informasi dan Komunikasi*, vol. 6, no. 2, p. 2022, 2022, doi: 10.35870/jti.
- [18] I. Verawati and B. S. Audit, "Algoritma Naive Bayes Classifier Untuk Analisis Sentiment Pengguna Twitter Terhadap Provider By.u," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 6, no. 3, p. 1411, Jul. 2022, doi: 10.30865/mib.v6i3.4132.
- [19] G. Darmawan, S. Alam, M. Imam Sulisty, P. Studi Teknik Informatika, S. Tinggi Teknologi Wastukencana Purwakarta, and R. Artikel, "ANALISIS SENTIMEN BERDASARKAN ULASAN PENGGUNA APLIKASI MYPERTAMINA PADA GOOGLE PLAYSTORE MENGGUNAKAN METODE NAIVE BAYES INFO ARTIKEL ABSTRAK," vol. 2, no. 3, pp. 100–108, 2023, doi: 10.55123.
- [20] S. Alpin Rizaldi, S. Alam, and I. Kurniawan, "ANALISIS SENTIMEN PENGGUNA APLIKASI JMO (JAMSOSTEK MOBILE) PADA GOOGLE PLAY STORE MENGGUNAKAN METODE NAIVE BAYES 1)," vol. 2, no. 3, pp. 109–117, 2023, doi: 10.55123.
- [21] Z. Rais, F. T. T. Hakiki, and R. Aprianti, "Sentiment Analysis of Peduli Lindungi Application Using the Naive Bayes Method," *SAINSMAT: Journal of Applied Sciences, Mathematics, and Its Education*, vol. 11, no. 1, pp. 23–29, Mar. 2022, doi: 10.35877/sainsmat794.
- [22] J. Homepage, N. C. Agustina, D. Herlina Citra, W. Purnama, C. Nisa, and A. Rozi Kurnia, "Implementasi Algoritma Naive Bayes untuk Analisis Sentimen Ulasan Shopee pada Google Play Store," vol. 2, pp. 47–54, 2022.
- [23] H. Junianto, P. Arsi, B. A. Kusuma, D. Intan, and S. Saputra, "Evaluasi Aplikasi Raileo Melalui Analisis Sentimen Ulasan Playstore Dengan Metode Naive Bayes", [Online]. Available: <https://doi.org/10.31598>
- [24] J. Khatib Sulaiman, A. Shopee Salman Alfaris, and U. Amikom Yogyakarta, "Komparasi Metode KNN dan Naive Bayes Terhadap Analisis Sentimen Pengguna," *Indonesian Journal of Computer Science Attribution*, vol. 12, no. 5, pp. 2023–2766.
- [25] I. Made Widiarta, D. Ardiansyah, L. Putu, S. Yuliasuti, and H. Artikel, "Analisis Perbandingan Performa Modem Usb Tp-Link Dan Telkomsel Orbit Menggunakan Metode QoS," *Digital Transformation Technology (Digitech) | e*, vol. 3, no. 1, 2023, doi: 10.47709/digitech.v3i1.2512.