



# Analisis Sentimen Ulasan Aplikasi *Smart Campus* Unisbank di Google Playstore Menggunakan Algoritma *Naive Bayes*

Dwi Rahma Firmansyah <sup>1\*</sup>, Endang Lestariningsih <sup>2</sup>

<sup>1,2</sup> Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Stikubank, Kota Semarang, Provinsi Jawa Tengah, Indonesia.

## article info

### Article history:

Received 1 December 2023  
Received in revised form  
28 February 2024  
Accepted 25 March 2024  
Available online April 2024.

### DOI:

<https://doi.org/10.35870/jtik.v8i2.1882>.

### Keywords:

Naive Bayes; Python;  
Smartcampus; Preprocessing;  
TF-IDF.

### Kata Kunci:

Naive Bayes; Python;  
Smartcampus; Preprocessing;  
TF-IDF.

## abstract

This study explores sentiment analysis on SmartCampus Unisbank application reviews on Google Play Store using the Naive Bayes classification method. Through Python programming language and web scraping techniques employing the google-play-scraper library, review data was automatically obtained and organized in CSV format. Text preprocessing techniques such as case folding, stopwords removal, tokenization, and stemming were applied to ensure accurate analysis. The data was divided into training and testing sets, and TF-IDF (Term Frequency-Inverse Document Frequency) was used for feature extraction. A Naive Bayes model was constructed and evaluated, achieving an accuracy of 84.6%. While the model demonstrated proficiency in identifying negative sentiments with 100% precision, it requires refinement for recognizing positive sentiments. These findings provide valuable insights for SmartCampus Unisbank developers to understand user perspectives and improve application quality.

## abstract

Penelitian ini mengeksplorasi analisis sentimen pada ulasan aplikasi SmartCampus Unisbank di Google Play Store menggunakan metode klasifikasi Naive Bayes. Melalui bahasa pemrograman Python dan teknik web scraping dengan library google-play-scraper, data ulasan diperoleh secara otomatis dan diatur dalam format CSV. Teknik preprocessing teks seperti case folding, penghapusan stopwords, tokenisasi, dan stemming diterapkan untuk memastikan analisis yang akurat. Data dibagi menjadi set pelatihan dan pengujian, dan TF-IDF (Term Frequency-Inverse Document Frequency) digunakan untuk ekstraksi fitur. Model Naive Bayes dibuat dan dievaluasi, mencapai akurasi sebesar 84.6%. Meskipun model menunjukkan keahlian dalam mengidentifikasi sentimen negatif dengan presisi 100%, perlu disempurnakan untuk mengenali sentimen positif. Temuan ini memberikan wawasan berharga bagi pengembang SmartCampus Unisbank untuk memahami pandangan pengguna dan meningkatkan kualitas aplikasi.

\*Corresponding Author. Email: [dwirahmafirmsyah@gmail.com](mailto:dwirahmafirmsyah@gmail.com) <sup>1\*</sup>.

© E-ISSN: 2580-1643.

Copyright © 2024 by the authors of this article. Published by Lembaga Otonom Lembaga Informasi dan Riset Indonesia (KITA INFO dan RISET). This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License. 

## 1. Latar Belakang

Perkembangan teknologi yang kian melesat, kini menjalar pada semua aspek kehidupan di dunia termasuk pendidikan. Pendidikan masa kini telah terbantu oleh adanya kecanggihan fitur-fitur teknologi. Para tenaga pendidik mampu memanfaatkan teknologi menjadi media pembelajaran dalam menyampaikan ilmu-ilmu pengetahuan kepada peserta didik. Media yang umumnya digunakan terdiri dari zoom, google classroom, google meeting atau whatsapp grup. Selain menjadi media pembelajaran, kecanggihan teknologi dimanfaatkan menjadi media informasi dan pembedahan dalam pendidikan di area pendidikan seperti sekolah dan kampus. Pemanfaatan yang dimaksud adalah terbentuknya aplikasi Smartcampus [1]. Aplikasi smartcampus mengacu pada segala aspek pendukung termasuk fasilitas dalam menunjang berjalannya kegiatan sivitas akademika dalam melaksanakan kewajiban Tridarma Perguruan Tinggi [2]. Pada smartcampus para peserta didik atau mahasiswa dan mahasiswi dapat mengakses segala fitur yang tertera bertujuan untuk mendapatkan informasi dan dapat memberikan penilaian atau evaluasi terhadap kinerja bahkan mekanisme sivitas akademika. Penggunaan aplikasi ini dimulai dengan mengunduh pada halaman Google Play Store. Umumnya para pengguna aplikasi akan memberikan ulasan terkait kinerja aplikasi. Ulasan - ulasan yang diberikan dapat berbentuk saran, kritikan, dan ucapan kepuasan. Developer aplikasi memerlukan ulasan - ulasan tersebut untuk mengetahui kelebihan dan kekurangan dari aplikasinya sehingga dapat memperbaiki bahkan mengembangkan. Salah satu cara untuk mengetahui hal tersebut yakni dapat dengan analisis sentimen berbasis aspek.

Analisis sentimen sendiri memiliki makna yaitu sebuah pandangan, pendapat atau emosi yang dijumpai pada suatu unggahan teks, ucapan, post (aktivitas di Internet), atau basis data dengan pendekatan *Natural Language Processing* (NLP). Analisis sentimen terdiri dari beberapa tugas yang diantaranya yaitu, ekstraksi sentimen, klasifikasi sentimen, dan deteksi spam [3]. Analisa sentimen ulasan aplikasi dapat dilakukan dengan algoritma Naive bayes. Naive bayes merupakan suatu algoritma yang sederhana dan memiliki tingkat pengklasifikasian teks yang bagus [4]. *Naive bayes*

adalah algoritma klasifikasi teks yang memiliki kecepatan proses dan tingkat akurasi cukup tinggi jika diaplikasikan pada kasus dengan data dalam jumlah besar dan beragam atau variatif [5]. Pada penelitian pertama yang telah dilakukan oleh [6] yang berjudul "Analisis Sentimen Pada Review Aplikasi Grab di Google Play Store Menggunakan *Support Vector Machine*" yang bertujuan untuk melihat dan menilai ulasan yang diberikan pengguna dalam aplikasi Grab di Google Play dengan menggunakan metode klasifikasi *Support Vector Machine* (SVM). Metode dalam penelitian tersebut bertahap dimulai dari pengumpulan data, tahap *preprocessing*, tahap pembagian data, tahap klasifikasi, perbandingan akurasi dan validasi, serta tahap hasil dan visualisasi. Hasil penelitian dengan proses analisis sentimen terhadap dengan data yang diperoleh sejumlah 1000 ulasan dengan hasil akurasi yang diperoleh sebesar 85,54% dengan ulasan positif yang paling sering diberikan adalah kata "Ovo", sedangkan ulasan negatif yang paling sering muncul tentang "driver".

Penelitian kedua yang dilakukan oleh [7] dengan judul "Analisis Sentimen Ulasan Aplikasi Mola Pada Google Play Store Menggunakan Algoritma *Support Vector Machine*". Metode yang digunakan pada penelitian ini adalah *Knowledge Discovery in Database* (KDD) dengan data penelitian yang mencakup 520 ulasan aplikasi MOLA, yang terdiri dari 312 ulasan positif dan 208 ulasan negatif. Hasil yang paling optimal tercapai pada skenario 1, di mana data dibagi dalam perbandingan 90:10, dengan penggunaan kernel RBF (Radial Basis Function). Hasil evaluasi mencatat akurasi sebesar 92,31%, presisi sebesar 96,3%, recall sebesar 89,66%, dan f1-score sebesar 92,86%.

Hasil penelitian ini menunjukkan bahwa analisis sentimen menggunakan algoritma *Support Vector Machine* dapat memberikan hasil yang akurat dalam mengklasifikasikan ulasan pengguna terhadap aplikasi MOLA. Hasil evaluasi kinerja juga menunjukkan bahwa kernel RBF memberikan performa yang lebih baik dibandingkan kernel lainnya seperti Kernel Linear, Kernel Sigmoid, dan Kernel Polynomial. Berdasarkan visualisasi frasa yang sering muncul pada ulasan positif, ditemukan kata-kata seperti "bagus", "mola", "mantap", dan "gratis". Sedangkan frasa yang sering muncul pada ulasan negatif adalah "aplikasi", "mola", "error", dan "macet". Penelitian ini

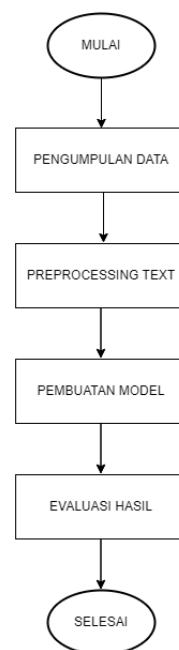
menunjukkan bahwa analisis sentimen menggunakan algoritma *Support Vector Machine* dapat digunakan untuk mengidentifikasi persepsi pengguna terhadap aplikasi MOLA di Google Play Store dan Kernel RBF merupakan kernel terbaik dalam klasifikasi ulasan pengguna, yang memberikan presisi, akurasi, recall, dan f1-score yang tinggi. Kata-kata yang sering muncul dalam ulasan positif dan negatif juga memberikan gambaran tentang kepuasan pengguna terhadap aplikasi MOLA.

Adapun tujuan dari penelitian untuk memberikan wawasan yang lebih dalam kepada pengembang Aplikasi Smartcampus Unisbank mengenai pandangan pengguna terhadap aplikasi tersebut. Analisis yang dihasilkan diharapkan mampu mengidentifikasi kelebihan dan kekurangan aplikasi, memberikan pemahaman yang lebih baik kepada pengembang, dan membantu dalam memperbaiki kualitas aplikasi. Penelitian ini juga bertujuan untuk mengevaluasi akurasi algoritma yang digunakan dalam memprediksi sentimen ulasan, apakah ulasan tersebut bersifat positif atau negatif. Evaluasi ini penting untuk memahami sejauh mana algoritma yang digunakan mampu melakukan klasifikasi dengan tepat, memberikan pandangan yang lebih mendalam terhadap performa algoritma dalam konteks aplikasi yang digunakan. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi yang berharga dalam pengembangan aplikasi berbasis pengguna dan algoritma yang dapat meningkatkan pengalaman pengguna secara keseluruhan.

## 2. Metode Penelitian

Penelitian ini bertujuan untuk menyelidiki sentimen yang terkandung dalam ulasan pengguna terhadap aplikasi SmartCampus Unisbank di Google Play Store dengan menggunakan bahasa pemrograman python. Melalui penggunaan teknik web scraping menggunakan library *google-play-scraper*, data berupa ulasan akan diperoleh secara otomatis dari platform tersebut dan diorganisir dalam format CSV. Sebelum data dianalisis akan dilakukan preprocessing text yaitu data ulasan kemudian akan mengalami proses pembersihan untuk memastikan keakuratan analisis, termasuk langkah-langkah penting seperti case folding, penghapusan stopwords, tokenisasi untuk memecah teks menjadi unit-unit yang lebih

kecil, dan stemming untuk mengonversi kata-kata ke bentuk dasarnya. Setelah proses pembersihan, data akan dibagi menjadi dua bagian: data pelatihan dan data pengujian. Fitur dari teks akan diekstraksi menggunakan nilai TF-IDF untuk menangkap pentingnya kata-kata dalam setiap ulasan. Model sentimen menggunakan metode klasifikasi Naive Bayes akan diaplikasikan pada data pengujian untuk menilai kinerja model dalam mengklasifikasikan ulasan sebagai positif atau negatif. Evaluasi model akan dilakukan dengan menggunakan sejumlah metrik, termasuk akurasi, presisi, recall, dan F1-score, untuk memberikan pemahaman yang komprehensif tentang seberapa baik model mampu menginterpretasikan sentimen dari ulasan pengguna. Untuk membantu memahami alur atau desain penelitian dapat dilihat pada gambar flowchart berikut.



Gambar 1. Rancangan Penelitian

Data yang digunakan untuk penelitian ini diambil dari ulasan aplikasi Smart Campus Unisbank yang tersedia di Google Play Store. yang meliputi kalimat ulasan dari pengguna dan rating yang telah diberikan. Data yang didapat dilakukan dengan cara melakukan scraping data dengan menggunakan *library python* yaitu *google play scraper* yang kemudian akan disimpan dalam bentuk formant berkas CSV (chairunnisa *et al.*, 2022). Untuk memahami lebih lanjut proses pengambilan data dapat dilihat pada gambar dibawah ini.

```
!pip install google-play-scraper
from google_play_scraper import app
import pandas as pd
import numpy as np
from google_play_scraper import Sort, reviews_all
result = reviews_all(
    'id.ac.unisbank.smartcampusapp',
    lang='id',
    country='id',
    sort=Sort.NEWEST,
    filter_score_with=None
)
df_busu = pd.DataFrame(np.array(result), columns=['reviews'])
df_busu = df_busu.join(pd.DataFrame(df_busu.pop('reviews').tolist()))
df_busu[['userName', 'score', 'at', 'content']]
new_df = df_busu[['userName', 'score', 'at', 'content']]
sorted_df = new_df.sort_values(by='at', ascending=False)
sorted_df
```

Gambar 2. *Scraping data*

Setelah data yang diperoleh dari melakukan *scraping*, data yang diperoleh dilakukan pelabelan disetiap ulasannya. pelabelan data menggunakan positif dan negatif [8]. Dapat dilihat pada gambar 2, pelabelan berdasarkan score dengan pengkategorian jika nilai 'score' kurang dari sama dengan 3 maka label berupa 'Negatif'. Jika nilai 'score' lebih dari sama dengan 4 maka label berupa 'Positif'.

```
my_df=sorted_df[['content', 'score']]

def pelabelan(score):
    if score <= 3:
        return 'Negatif'
    elif score >= 4 :
        return 'Positif'
my_df['Label'] = my_df['score'].apply(pelabelan)
my_df
```

Gambar 3. Pelabelan

Setelah melakukan pelabelan maka dilakukanlah data cleaning atau Pembersihan data mengacu pada proses mengidentifikasi bagian data yang tidak lengkap, tidak benar, tidak akurat, atau tidak relevan, lalu menggantinya, memodifikasinya, atau menghapus data yang kurang baik atau kasar tersebut [9]. Data cleansing yang dilakukan yaitu mencari data atau ulasan yang memiliki nilai null atau tidak terdefinisi yang nantinya data tersebut tidak akan digunakan dalam pengujian model dengan menghapus yang dapat dilihat pada gambar 4.

```
my_df.info()
my_df.isnull().sum()
my_df.isna().any()
my_df.dropna(subset=['Label'], inplace = True)
```

Gambar 4. *Data cleaning*

Dari hasil pelabelan, kemudian data akan melalui *Preprocessing text* terlebih dahulu. *Preprocessing text*

adalah tahap penting dalam proses penambangan data karena data yang digunakan tidak selalu ideal untuk diolah. Beberapa kali, data memiliki berbagai isu yang dapat mengganggu hasil dari proses penambangan data itu sendiri, seperti keberadaan nilai yang hilang, kelimpahan data yang tidak terlalu berguna, data yang tidak sesuai dengan kebutuhan sistem, dan kehadiran data yang berbeda secara signifikan dari nilai-nilai umum, yang dikenal sebagai *outliers* [10]. Pada tahap ini data atau teks akan diolah untuk menghilangkan kata yang tidak relevan yang tidak mewakili sentiment apapun seperti tanda baca, simbol, atau tautan [11].

### *Case folding*

Proses *case folding* merupakan langkah untuk mengubah semua huruf dalam teks menjadi huruf kecil. Pada proses ini, karakter-karakter 'A' sampai 'Z' yang terdapat dalam data diubah menjadi karakter 'a' sampai 'z'. *Case folding* merupakan salah satu metode paling sederhana dan efektif dalam pra-pemrosesan teks, walaupun sering kali diabaikan. Tujuannya adalah untuk mengubah semua huruf dalam dokumen menjadi huruf kecil. Hanya karakter dari 'a' hingga 'z' yang dipertahankan, sedangkan karakter selain huruf akan dihapus dan dianggap sebagai pembatas (Muhammad Harris Syafa'at *et al.*, 2021).

### *Stopwords removal*

Dalam proses penghapusan stopwords, kata-kata yang bukan baku akan dihilangkan. Metodenya menggunakan daftar kata-kata baku, dimana jika kata-kata setelah dipisah-pisahkan tidak termasuk dalam daftar kata baku tersebut, maka kata-kata tersebut akan dihapus. Dasar pembuatan daftar kata-kata baku ini diambil dari Kamus Besar Bahasa Indonesia (KBBI) dan diperbarui secara berkala sesuai dengan KBBI yang berlaku. Stopwords removal merupakan penghapusan kata-kata umum yang sering muncul namun dianggap tidak memiliki makna penting dalam teks, seperti contohnya "yang", "dan", "di", "dari", dan sebagainya. Dengan menghilangkan kata-kata yang kurang informatif, perhatian dapat difokuskan pada kata-kata yang lebih signifikan [12].

### *Tokenizing*

Tokenisasi adalah proses membagi teks menjadi bagian-bagian kecil yang disebut sebagai token. Token dapat berupa kata, angka, simbol, tanda baca, dan entitas penting lainnya dalam teks. Dalam pengolahan bahasa alami (NLP), token biasanya diidentifikasi



sebagai "kata", meskipun proses tokenisasi juga dapat diterapkan pada tingkat paragraf atau kalimat. Dalam konteks NLP, tokenisasi memungkinkan teks untuk diurai ke dalam unit-unit yang lebih kecil agar bisa dianalisis dengan lebih baik [12].

### Stemming

Pada langkah *stemming*, setelah data komentar melalui proses *filtering* dan penghapusan *stopwords*, potongan-potongan kata yang sudah menjadi bentuk dasar akan diubah menjadi bentuk kata baku yang tepat dalam bahasa Indonesia. *Stemming* adalah proses pengenalan dan penguraian kata ke dalam bentuk dasarnya. *Library Python* Sastrawi yang telah dipersiapkan sebelumnya dapat digunakan untuk melakukan *stemming* dalam bahasa Indonesia. Sastrawi menggunakan Algoritma Nazief dan Adriani untuk memetakan kata-kata ke bentuk dasarnya dalam bahasa Indonesia [13].

Setelah melakukan pembersihan data, langkah selanjutnya adalah melakukan pembagian data. Data ulasan dibagi menjadi dua kelompok, yaitu data pelatihan (*training*) dan data pengujian (*testing*). Pembagian data ini umumnya dilakukan dengan rasio tertentu, misalnya, 80% data untuk pelatihan dan 20% data untuk pengujian. Hal ini dilakukan untuk melatih model dan menguji kinerjanya. Setelah data ulasan dibagi menjadi dua, langkah selanjutnya adalah melakukan pembobotan kata. Pembobotan kata adalah proses memberikan nilai atau bobot pada kata berdasarkan tingkat kepentingannya dalam suatu kumpulan dokumen atau teks [14]. Terdapat beberapa metode yang dapat digunakan untuk melakukan pembobotan kata. Dalam penelitian ini pembobotan menggunakan metode ekstraksi fitur berdasarkan penggunaan TF-IDF (*Term Frequency-Inverse Document Frequency*), mengukur seberapa sering kata muncul dalam satu dokumen (TF) dan seberapa umum kata tersebut dalam seluruh koleksi dokumen (IDF). Dengan menggunakan *TfidfVectorizer* dari *library scikit-learn* yang digunakan untuk melakukan ekstraksi fitur dan menghasilkan representasi TF-IDF untuk setiap kata dalam teks. *DataFrame* yang sudah dibuat kemudian mencakup informasi DF (*Document Frequency*) dan IDF (*Inverse Document Frequency*) untuk setiap kata yang dapat membantu dalam memahami kontribusi kata terhadap keseluruhan konten ulasan. Fitur-fitur ini kemudian

digunakan sebagai input untuk model klasifikasi sentimen.

Dalam langkah terakhir, pembuatan model sentimen menggunakan algoritma *Naive Bayes*. *Naive Bayes* merupakan suatu metode atau algoritma klasifikasi dengan probabilitas dan statistik dalam pembelajaran induktif yang umumnya digunakan machine learning dan data mining. Metode ini dikemukakan oleh seorang ilmuwan Inggris Thomas Bayes saat ingin melakukan prediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya. *Naive* diasumsikan sebagai antar atribut saling bebas [15]. Proses pembuatan model ini merupakan langkah penting dalam analisis sentimen teks, yang memungkinkan untuk memahami dan mengklasifikasikan sentimen pada teks baru berdasarkan pengetahuan yang diperoleh dari teks yang telah terlabel sebelumnya. Pada tahap ini dilakukan pembuatan model untuk melakukan analisis sentimen berdasarkan teks yang dapat dilihat pada gambar dibawah ini.

```
from sklearn.naive_bayes import MultinomialNB
nb = MultinomialNB()
nb.fit(tfidf_train, y_train)

tfidf_train.toarray()

y_pred = nb.predict(tfidf_test)
```

Gambar 5. Pembuatan model

Metode yang digunakan dalam pembuatan model adalah *Naive Bayes* dengan varian *MultinomialNB*, sebuah algoritma klasifikasi yang memanfaatkan vektor TF-IDF untuk mewakili teks. Proses dimulai dengan pembagian dataset menjadi data latih dan data uji. Data latih digunakan untuk melatih model, sementara data uji digunakan untuk menguji kinerja model yang telah dilatih. Tahapan selanjutnya adalah pembuatan vektor TF-IDF dari teks yang telah diproses, yang merupakan representasi numerik dari teks yang digunakan oleh model. Setelah pembobotan kata menggunakan TF-IDF, dilakukan inisialisasi model *Naive Bayes* (*MultinomialNB*) yang kemudian dilatih dengan data latih menggunakan `nb.fit(tfidf_train, y_train)`. Proses pelatihan ini memungkinkan model untuk memahami hubungan antara representasi vektor TF-IDF dari teks dengan label sentimen yang sesuai. Selanjutnya, dilakukan prediksi pada data uji menggunakan model yang telah

dilatih sebelumnya ( $y\_pred = nb.predict(tfidf\_test)$ ). Prediksi ini bertujuan untuk mengevaluasi seberapa baik model yang telah dibuat dapat mengklasifikasikan sentimen pada teks yang belum pernah dilihat sebelumnya. Hasil pengujian termasuk akurasi, presisi, recall, dan skor F1 digunakan untuk mengevaluasi kinerja model dalam mengklasifikasikan ulasan sebagai positif atau negatif. Dengan demikian, dapat diukur sejauh mana model dapat memprediksi sentimen ulasan dengan benar.

### 3. Hasil dan Pembahasan

#### *Hasil pengumpulan data*

Setelah melaksanakan proses pengumpulan data ulasan aplikasi SmartCampus di Google Play Store dengan memanfaatkan library Google Play Scraper,

Peneliti berhasil mengumpulkan data sebanyak 64 ulasan. Dalam upaya menyelaraskan data dengan keperluan analisis sentimen, Peneliti melakukan penyaringan data untuk menampilkan hanya kolom-kolom yang relevan, seperti username atau nama pengguna, skor atau rating yang diberikan oleh pengguna, waktu ulasan diberikan, dan konten atau isi ulasan yang disampaikan oleh pengguna. Pendekatan ini memungkinkan Peneliti untuk fokus pada informasi yang paling penting untuk evaluasi kualitas aplikasi. Dari hasil pengumpulan data, Peneliti menghasilkan tabel yang memuat 3 ulasan teratas, yang diurutkan berdasarkan rentang waktu terbaru. Hal ini memberikan gambaran yang jelas tentang persepsi pengguna terhadap aplikasi SmartCampus secara kontemporer. Berikut hasil scraping data 3 ulasan teratas yang sudah diurutkan berdasarkan rentang waktu terbaru yang dapat dilihat pada tabel 1.

Tabel 1. Hasil pengumpulan data

No	username	score	At	Content
1	Auwlaa Fill	5	2022-10-08 01:25:10	Sangat penting bagi Mahasiswa/i Unisbank
2	Ibnu Aryaa	4	2022-09-12 07:44:26	Upload filenya gak bisaa, kecewa akuu
3	Adzan Assharies	1	2022-09-08 01:08:32	Aplikasi eror banyak membuat mahasiswa rugi. Mana kalau telat bayar harus bayar denda 200.000 Lo itu sama aja 23% dari SPP. Seharusnya harga dengan kualitas harus sama kan. Payah ni kampus salah pilih kampus saya menyesal saya

#### *Hasil pelabelan*

Setelah melakukan pengumpulan data maka dilakukanlah pelabelan agar lebih mudah untuk melakukan analisis sentimen. Pelabelan dilakukan berdasarkan score yang diberikan pengguna yang diklasifikasikan sebagai ulasan negatif dan positif yaitu jika score kurang atau sama dengan tiga maka akan diklasifikasikan sebagai ulasan negatif dan jika score lebih dari tiga maka ulasan dikategorikan

sebagai ulasan positif dan hasil klasifikasi akan ditulis dengan menambahkan kolom baru yaitu label. Setelah dilakukan pelabelan akan ditampilkan dalam bentuk tabel dan hanya akan ditampilkan kolom content yang berisi ulasan dari para pengguna dan score atau rating ulasan yang diberikan pengguna dan kolom label yang berisi negatif atau positif dari ulasan yang diberikan.

Tabel 2. Hasil pelabelan

No	Content	score	Label
1	Sangat penting bagi Mahasiswa/i Unisbank	5	Positif
2	Aplikasi eror banyak membuat mahasiswa rugi. Mana kalau telat bayar harus bayar denda 200.000 Lo itu sama aja 23% dari SPP. Seharusnya harga dengan kualitas harus sama kan. Payah ni kampus salah pilih kampus saya menyesal saya	3	Negatif
3	Gabisa dibuka	1	Negatif

*Hasil data cleaning*

Setelah melakukan pelabelan data kemudian dilakukan data cleaning pada kolom yang digunakan yaitu content, score dan label. Semua data atau ulasan yang sudah dicek bernilai non-null atau tidak ada yang memiliki nilai null atau tidak terdefinisi yang dapat dilihat pada tabel dibawah ini.

Tabel 3. Data cleaning

Column	Non-Null Count	Tipe Data
--------	----------------	-----------

content	64 non-null	object
Score	64 non-null	int64
Label	64 non-null	Object

*Hasil preprocessing*

Setelah melakukan *preprocessing text* yaitu *case folding*, *stopwords removal*, tokenizing, *stemming* didapatkan data yang sudah bersih yang dapat dilihat pada tabel dibawah ini.

Tabel 4. Preprocessing

<i>Case folding</i>	<i>Stopwords removal</i>	<i>Tokenizing</i>	<i>Stemming</i>
sangattt penting bagi mahasiswa unisbank	sangattt mahasiswa unisbank	[sangattt, mahasiswa, unisbank]	sangattt mahasiswa unisbank
upload filenya gak bisaa kecewa akuu	upload filenya gak bisaa kecewa akuu	[upload, filenya, gak, bisaa, kecewa, akuu]	upload filenya gak bisaa kecewa akuu
gabisa dibuka	gabisa dibuka	[gabisa, dibuka]	gabisa buka

*Hasil pembagian data dan pembobotan kata*

Setelah pembagian data dilakukan dengan rasio 80% untuk pelatihan dan 20% untuk pengujian, proses selanjutnya melibatkan pembentukan model. Pembagian data ini penting untuk memastikan model yang dikembangkan dapat diuji dengan baik menggunakan data yang tidak digunakan dalam proses pelatihan. Dengan demikian, hasil evaluasi dapat memberikan gambaran yang lebih akurat tentang kinerja model pada data yang belum pernah dilihat sebelumnya. selanjutnya dari hasil pembobotan kata menghasilkan data yang dapat dilihat pada tabel 5 yang. Terdapat kolom-kolom yang menunjukkan frekuensi dokumen, yaitu berapa kali sebuah kata muncul dalam dokumen secara total, dan juga nilai Inverse Document Frequency (IDF) untuk setiap kata dalam kumpulan dokumen yang digunakan. Tabel tersebut menampilkan kata-kata beserta informasi penting terkait penggunaan kata-kata dalam kumpulan dokumen. Kolom "Word" menunjukkan kata-kata yang diekstraksi dari teks yang telah dibersihkan, sedangkan "Document Frequency" memberikan informasi tentang seberapa sering kata tersebut muncul dalam keseluruhan dokumen. Nilai IDF dalam kolom "Inverse Document Frequency" menggambarkan seberapa pentingnya sebuah kata dalam koleksi dokumen, dengan mengurangi bobot kata-kata yang muncul secara umum dalam banyak dokumen. Semakin tinggi nilai IDF, semakin besar kontribusi kata tersebut terhadap pemahaman konten dokumen secara keseluruhan.

Tabel 5. Hasil pembobotan kata

Word	Document Frequency	Inverse Document Frequency
Terimakasih	1	4.258097
Kesalahan	1	4.258097
Memudahkan	1	4.258097
Dan	9	2.648659
saya	8	2.754019
Yang	4	3.341806

Frekuensi dan bobot kata-kata ini sangat berguna dalam pemrosesan teks dan pemodelan seperti klasifikasi sentimen. Kata-kata yang memiliki nilai IDF yang tinggi (mengindikasikan kemunculan yang kurang umum) cenderung memberikan kontribusi yang lebih besar dalam memahami esensi atau tema khusus dalam suatu dokumen atau kelompok dokumen. Sebagai contoh, kata-kata dengan nilai IDF tinggi seperti "terimakasih", "kesalahan", dan "memudahkan" cenderung menjadi kata-kata yang memberikan informasi yang lebih spesifik atau unik dalam dokumen-dokumen yang ada. Sementara kata-kata dengan nilai IDF yang lebih rendah atau frekuensi kemunculan yang tinggi seperti "dan", "saya", atau "yang" memiliki nilai IDF yang lebih rendah karena muncul secara umum dalam banyak dokumen dan kurang memberikan informasi khusus terkait konten.

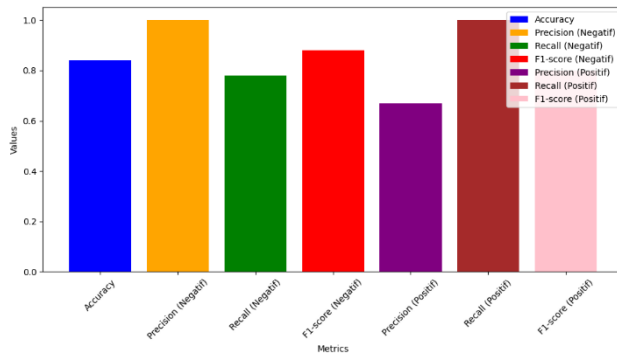
*Evaluasi Hasil Model Naive Bayes*

Tahap selanjutnya atau terakhir adalah melakukan evaluasi Hasil model *Naive Bayes* pada analisis

sentimen ulasan aplikasi Smartcampus Unisbank di Google Play Store. Hasil evaluasi mode dapat dilihat pada tabel 6 dan juga ditampilkan dalam bentuk diagram batang pada gambar 6 menunjukkan akurasi sebesar 84.6%. Evaluasi ini menggambarkan bahwa model memiliki tingkat presisi 100% dalam mengidentifikasi sentimen negatif, dengan recall sebesar 77.8%. Sementara untuk sentimen positif, model memiliki presisi sebesar 66.7% dengan recall 100%.

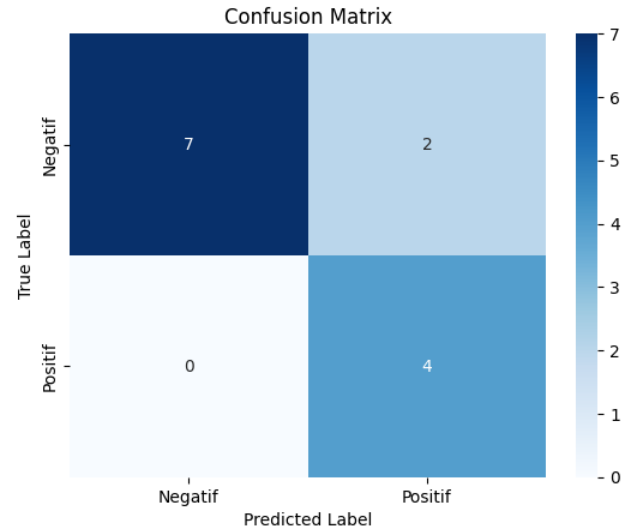
Tabel 6. Hasil naive bayes

<i>Accuracy</i>	Negatif	Positif
<i>Precision</i>	1.0	0.6666
<i>Recall</i>	0.7777	1.00
<i>F1_score</i>	0.8750	0.80



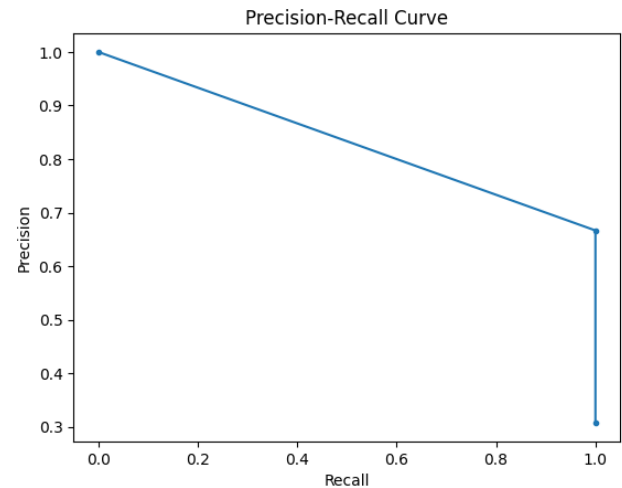
Gambar 6. Matrix naive bayes

Pada *Confusion matrix* menunjukkan bahwa dari total 13 sampel, terdapat 7 hasil klasifikasi benar untuk kelas 'Negatif', 4 hasil klasifikasi benar untuk kelas 'Positif', 2 kesalahan prediksi untuk kelas 'Negatif', dan tidak ada kesalahan prediksi untuk kelas 'Positif' yang artinya model berhasil mengklasifikasikan 11 sampel dengan benar, dengan 2 kesalahan klasifikasi. Selain itu, melalui validasi silang k-fold dengan 5 *fold*, rata-rata akurasi model sebesar 80.4%, menunjukkan konsistensi performa model pada data latih yang berbeda-beda. Evaluasi ini menggambarkan bahwa model mampu melakukan klasifikasi dengan benar pada sebagian besar data uji, dengan kemampuan yang baik dalam mengenali sentimen negatif. Namun, terdapat beberapa kesalahan klasifikasi yang perlu diperhatikan lebih lanjut untuk meningkatkan kinerja model, khususnya dalam mengenali sentimen positif.



Gambar 7. Confusion matrix

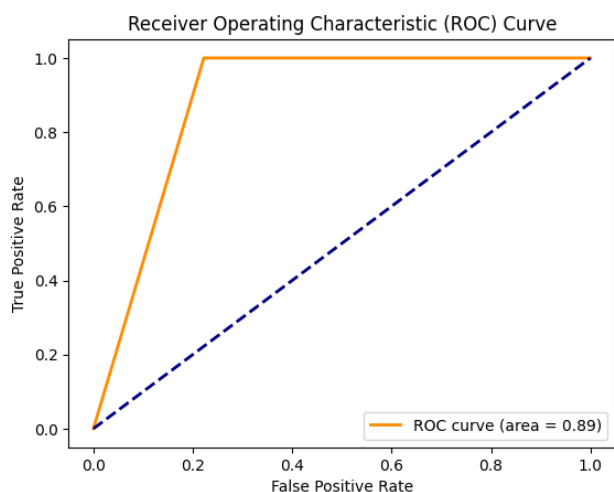
Selain evaluasi yang telah disebutkan sebelumnya, penambahan visualisasi dengan Kurva *Precision-Recall* atau *Receiver Operating Characteristic (ROC)* dapat memberikan pemahaman yang lebih mendalam tentang performa model dalam berbagai *threshold*.



Gambar 8. Precision recall

Seperti pada gambar 8, Kurva *Precision-Recall* akan menunjukkan trade-off antara presisi dan recall pada berbagai nilai *threshold* yang berbeda. Dalam konteks ini, visualisasi tersebut akan membantu memperlihatkan bagaimana perubahan *threshold* akan mempengaruhi performa model dalam mengklasifikasikan sentimen positif dan negatif.





Gambar 9. ROC

Sementara itu, *ROC curve* menampilkan perbandingan antara tingkat *true positive (recall)* dengan tingkat false positive pada berbagai threshold. Dari kurva ini, kita bisa mengevaluasi sejauh mana model dapat membedakan antara sentimen positif dan negatif, serta seberapa baik model dapat menghindari klasifikasi yang salah. Kemudian kata dari ulasan positif dan negatif dapat ditampilkan dalam bentuk *word clouds* atau grafik frekuensi seperti gambar dibawah untuk menunjukkan kata-kata yang paling umum muncul dalam setiap kategori yang dapat dilihat.



Gambar 10. *Wordcloud* Positif



Gambar 11. *Wordcloud* Negatif

Grafik tersebut bisa memberikan wawasan tentang kata-kata kunci yang dapat mempengaruhi klasifikasi seperti kata 'good' pada ulasan positif yang berarti cukup banyak user yang puas dengan performa aplikasi Smartcampus, dan juga kata 'diperbaiki' pada ulasan negatif yang banyak dari pengguna aplikasi juga mengeluhkan atas kurangnya dari aplikasi tersebut.

## 4. Kesimpulan

Berdasarkan penelitian yang telah dilakukan, analisis sentimen terhadap ulasan aplikasi SmartCampus Unisbank di Google Play Store, dapat disimpulkan bahwa penggunaan metode klasifikasi Naive Bayes dalam menganalisis sentimen ulasan aplikasi ini mampu memberikan hasil yang memadai. Dalam proses evaluasi model, tercatat bahwa model mampu mengklasifikasikan ulasan dengan akurasi sebesar 84.6%. Meskipun model menunjukkan kemampuan yang baik dalam mengidentifikasi sentimen negatif dengan presisi 100%, terdapat kebutuhan untuk lebih meningkatkan kemampuan dalam mengenali sentimen positif. Hal ini menunjukkan bahwa pengembangan lebih lanjut pada model dan penyesuaian terhadap kata-kata yang sering muncul dalam ulasan dapat meningkatkan performa klasifikasi sentimen. Selain itu, penelitian ini memberikan pandangan awal yang signifikan bagi pengembang aplikasi SmartCampus Unisbank untuk memahami pandangan dan persepsi pengguna, yang dapat menjadi landasan bagi perbaikan dan pengembangan aplikasi yang lebih adaptif serta responsif terhadap kebutuhan pengguna. Dalam lanjutan penelitian, fokus pada pengoptimalan model untuk mengenali sentimen positif secara lebih akurat serta pembaruan berkelanjutan aplikasi berdasarkan temuan dari analisis sentimen dapat menjadi langkah penting untuk meningkatkan kualitas dan kepuasan pengguna.

## 5. Daftar Pustaka

- [1] Agustian, N., & Salsabila, U. H. (2021). Peran teknologi pendidikan dalam pembelajaran. *Islamika*, 3(1), 123-133. DOI: 10.36088/islamika.v3i1.1047.
- [2] Areni, I. S., Palantei, E., Suyuti, A., Baharuddin, M., Samman, F. A., & Umraeni, A. E. (2019). Pengembangan dan implementasi smart campus berbasis smart card di Institut Agama

- Islam Negeri Bone. *JURNAL TEPAT: Teknologi Terapan untuk Pengabdian Masyarakat*, 2(1), 1-7. DOI: [https://doi.org/10.25042/jurnal\\_tepat.v2i1.51](https://doi.org/10.25042/jurnal_tepat.v2i1.51).
- [3] Flores, V. A., Jasa, L., & Linawati, L. (2020). Analisis Sentimen untuk Mengetahui Kelemahan dan Kelebihan Pesaing Bisnis Rumah Makan Berdasarkan Komentar Positif dan Negatif di Instagram. *Majalah Ilmiah Teknologi Elektro*, 19(1), 49.
- [4] Wisnu, G. R. G., Muttaqi, A. R., Santoso, A. B., Putra, P. K., & Budi, I. (2020, October). Sentiment analysis and topic modelling of 2018 central java gubernatorial election using twitter data. In *2020 International Workshop on Big Data and Information Security (IW BIS)* (pp. 35-40). IEEE. DOI: 10.1109/IWBIS50925.2020.9255583.
- [5] Fitriyyah, S. N. J., Safriadi, N., & Pratama, E. E. (2019). Analisis Sentimen Calon Presiden Indonesia 2019 dari Media Sosial Twitter Menggunakan Metode Naive Bayes. *JEPIN (Jurnal Edukasi dan Penelitian Informatika)*, 5(3), 279-285. DOI: <http://dx.doi.org/10.26418/jp.v5i3.34368>.
- [6] Wahyudi, R., & Kusumawardhana, G. (2021). Analisis Sentimen pada review Aplikasi Grab di Google Play Store Menggunakan Support Vector Machine. *Jurnal Informatika*, 8(2), 200-207.
- [7] Hendriyanto, M. D., Ridha, A. A., & Enri, U. (2022). Analisis Sentimen Ulasan Aplikasi Mola Pada Google Play Store Menggunakan Algoritma Support Vector Machine. *INTECOMS: Journal of Information Technology and Computer Science*, 5(1), 1-7. DOI: <https://doi.org/10.31539/intecom.v5i1.3708>
- [8] Tanggraeni, A. I., & Sitokdana, M. N. (2022). Analisis Sentimen Aplikasi E-Government pada Google Play Menggunakan Algoritma Naïve Bayes. *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, 9(2), 785-795. DOI: <https://doi.org/10.35957/jatisi.v9i2.1835>.
- [9] Saini, S., Punhani, R., Bathla, R., & Shukla, V. K. (2019, April). Sentiment analysis on twitter data using R. In *2019 International Conference on Automation, Computational and Technology Management (ICACTM)* (pp. 68-72). IEEE.
- [10] Apriani, R., & Gustian, D. (2019). Analisis Sentimen dengan Naïve Bayes Terhadap Komentar Aplikasi Tokopedia. *Jurnal Rekayasa Teknologi Nusa Putra*, 6(1), 54-62. DOI: <https://doi.org/10.52005/rekayasa.v6i1.86>.
- [11] Pratama, M. O., Satyawan, W., Jannati, R., Pamungkas, B., Syahputra, M. E., & Neforawati, I. (2019, April). The sentiment analysis of Indonesia commuter line using machine learning based on twitter data. In *Journal of Physics: Conference Series* (Vol. 1193, No. 1, p. 012029). IOP Publishing.
- [12] Syafa'at, M. H., Setyaningsih, E. R., & Kristian, Y. (2021). Svm Untuk Sentiment Analysis Calon Kepala Daerah Berdasar Data Komentar Video Debat Pilkada Di Youtube. *Antivirus: Jurnal Ilmiah Teknik Informatika*, 15(2), 262-276.
- [13] Yusnitasari, T., Ikasari, D., Pratiwi, E. E. S., & Ramdani, N. S. (2017, November). Analisis Sentimen Terhadap Review Restoran Fish Streat pada Aplikasi Zomato Menggunakan Stemming Nazief Adriani dan Naive Bayes Classifier. In *Prosiding Sentrinov (Seminar Nasional Terapan Riset Inovatif)* (Vol. 3, No. 1, pp. EB163-174).
- [14] Kosasih, R., & Alberto, A. (2021). Analisis Sentimen Produk Permainan Menggunakan Metode TF-IDF Dan Algoritma K-Nearest Neighbor. *InfoTekJar: Jurnal Nasional Informatika dan Teknologi Jaringan*, 6(1), 134-139. DOI: <https://doi.org/10.30743/infotekjar.v6i1.3893>.
- [15] Muin, A. A. (2016). Metode Naive Bayes Untuk Prediksi Kelulusan (Studi Kasus: Data Mahasiswa Baru Perguruan Tinggi). *Jurnal Ilmiah Ilmu Komputer Fakultas Ilmu Komputer Universitas Al Azyariah Mandar*, 2(1), 22-26.