

IMPLEMENTASI ALGORITMA NAÏVE BAYES UNTUK ANALISIS ULASAN PENGGUNA UNTUK APLIKASI SEABANK DI GOOGLE PLAY STORE

Elisabeth Dwi Kurnia Wardani^{*1}, Febi Febiola Yo², Wayan Nesa Meylugita³

^{1,2,3}Fakultas Sains dan Teknologi, Universitas Katolik Misi Charitas, Palembang, Indonesia.

Email: 1niaelisabeth418@gmail.com, 2olafebi905@gmail.com, 3meylugita.nesa@gmail.com

(Diterima : 25 November 2024, Direvisi : 8 Desember 2024, Disetujui : 21 Desember 2024)

Abstrak

Analisis sentimen terhadap layanan digital semakin penting di era transformasi digital karena dapat memberikan wawasan yang mendalam mengenai persepsi pelanggan terhadap layanan yang disediakan. Penelitian ini menggunakan metode klasifikasi untuk mengklasifikasikan sentimen pelanggan berdasarkan data ulasan yang dikumpulkan dari platform layanan digital Seabank. Proses analisis dilakukan dengan mengelompokkan sentimen menjadi kategori positif, negatif, dan netral, di mana evaluasi akurasi model dilakukan menggunakan metrik evaluasi seperti *presisi*, *recall*, dan *F1-score*. Hasil penelitian menunjukkan bahwa model mampu mengklasifikasikan sentimen dengan akurasi sebesar 85%, di mana kategori sentimen negatif didominasi oleh kata-kata seperti "buruk" dan "tidak puas," sementara sentimen positif didominasi oleh kata-kata seperti "sangat baik" dan "memuaskan." Penelitian ini bertujuan untuk memberikan pemahaman yang lebih baik kepada penyedia layanan terkait kebutuhan dan ekspektasi pelanggan, sehingga dapat digunakan untuk meningkatkan kualitas layanan di masa mendatang.

Kata kunci: analisis sentimen, ulasan pengguna, google play store, *naïve bayes*.

IMPLEMENTATION OF THE NAIVE BAYES ALGORITHM FOR USER REVIEW ANALYSIS OF THE SEABANK APPLICATION ON GOOGLE PLAY STORE

Abstract

Sentiment analysis of digital services is increasingly important in the era of digital transformation as it provides valuable insights into customer perceptions of the services offered. This research utilizes the Naive Bayes Classifier method to classify customer sentiment based on review data collected from Seabank's digital service platform. Sentiments are categorized into positive, negative, and neutral, with the model's accuracy evaluated using metrics such as precision, recall, and F1-score. The results indicate that the model achieves an accuracy of 85%, with negative sentiment dominated by words like "bad" and "dissatisfied," while positive sentiment is characterized by terms such as "very good" and "satisfactory." This study aims to help service providers better understand customer needs and expectations, enabling them to enhance service quality in the future.

Keywords: *google play store, tf-idf, user reviews, sentiment analysis, and naive bayes*

1. PENDAHULUAN

Di era teknologi yang semakin canggih, berbagai inovasi telah memberikan kemudahan bagi kehidupan manusia. Salah satu penemuan terbesar adalah internet, yang menjadikan hidup lebih sederhana, praktis, dan efisien[1]. Tanpa disadari Aplikasi sekarang sangat penting bagi kehidupan manusia. Sektor perbankan merupakan salah satu contoh Aplikasi perkembangan yang sejalan dengan tren kemajuan teknologi ini[2].

Industri perbankan telah mengalami kemajuan besar dalam teknologi informasi dan komunikasi. Seabank, bank digital yang inventif, menawarkan berbagai kemudahan perbankan. Perilaku keuangan masyarakat dan siswa dipengaruhi oleh perbankan digital ini. Seabank digital menawarkan layanan perbankan yang lebih efisien dan mudah digunakan, seperti perbankan untuk mahasiswa. Untuk menyesuaikan diri dengan teknologi dan menikmati kecepatan dan kenyamanan dalam berbagai aspek kehidupan, generasi muda memiliki lebih banyak kesempatan. Mahasiswa perbankan dapat memanfaatkan banyak fitur Seabank, seperti pembukaan rekening secara online, transfer antar bank murah atau gratis, dan layanan 24 jam[3]. Selain itu, ulasan dan rating pengguna dapat ditemukan

di setiap aplikasi mengenai layanan dan fiturnya. Komentar, yang dapat mencakup kritik, saran, atau keluhan, akan sangat bermanfaat bagi pengguna aplikasi berikutnya[4].

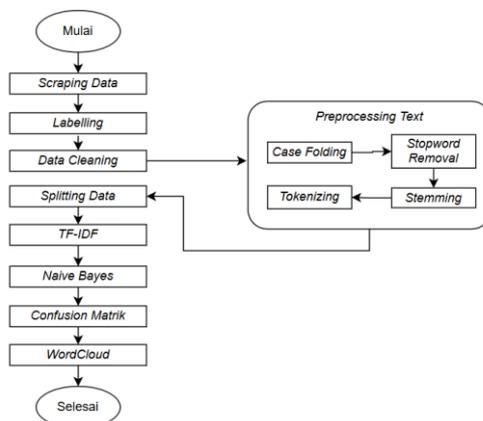
Analisis sentimen adalah teknik penelitian berbasis komputer yang bertujuan untuk mempelajari perasaan, pandangan, dan emosi yang terkandung dalam teks tentang hal-hal atau atribut tertentu. Proses analisis sentimen umumnya terdiri dari lima tahap utama: pengumpulan data, persiapan data, pemilihan fitur, proses klasifikasi, dan evaluasi hasil. Dalam berbagai situasi, analisis sentimen dapat digunakan sebagai konsep atau penilaian[5]. Analisis sentimen mengubah data kualitatif menjadi data kuantitatif yang bernilai dengan menghilangkan kata dan simbol yang tidak relevan. Data ulasan pengguna kemudian diklasifikasikan untuk mendapatkan ulasan positif dan negatif[6]. Analisis sentimen membantu bisnis memahami reaksi pelanggan terhadap merek atau produk tertentu dan memahami kebutuhan pelanggan sehingga mereka dapat membuat pilihan bisnis yang lebih baik[7].

Berdasarkan penelitian sebelumnya, peneliti menemukan berapa banyak ulasan positif dan negatif yang diberikan pengguna aplikasi Seabank melalui analisis sentimen dengan metode [8]. Penggunaan algoritma ini memberikan keuntungan karena menghitung proses klasifikasi data membutuhkan lebih sedikit data[9]. Untuk menilai perasaan pelanggan terhadap barang dan jasa, data evaluasi pelanggan yang diolah dari halaman web digunakan untuk mengetahui perubahan preferensi pelanggan. Dalam analisis pendapat, algoritma sering digunakan karena lebih akurat[10].

Berdasarkan latar belakang tersebut, penelitian ini bertujuan untuk menganalisis ulasan pengguna aplikasi Seabank menggunakan metode guna mengidentifikasi ulasan positif dan negatif yang dapat memberikan wawasan mengenai kepuasan pengguna. Hasil penelitian ini diharapkan dapat membantu Seabank dalam meningkatkan kualitas layanan berdasarkan preferensi dan kebutuhan pelanggan.

2. METODE PENELITIAN

Pada kajian berikut, ulasan aplikasi SeaBank yang tersedia di Google Play Store dipakai sebagai sumber data. Proses dimulai dengan pengumpulan data melalui *scraping*. Selanjutnya, teks dibersihkan dan diproses sebelum dipisahkan. Sementara algoritma *Naive Bayes* dipakai dalam tahapan klasifikasi, metoda TF-IDF digunakan guna menilai bobot kata [11], seperti yang digambarkan dalam Gambar 1 berikut.



Gambar 1. Metodologi Penelitian

2.1 Scraping Data

Scraping data adalah proses pengumpulan data secara otomatis dari sumber tertentu, dalam hal ini data aplikasi Seabank di Google Play Store. Proses ini dilakukan menggunakan platform Google Colab, paket Google-play-scraper, serta bahasa pemrograman Python[12]. Data yang diambil mencakup konten, ulasan, dan skor atau rating yang diberikan pengguna, yang kemudian disimpan dalam file CSV untuk keperluan analisis lebih lanjut[13]. Seluruh data awal diambil, tetapi data akhir yang digunakan dalam analisis ini hanya ulasan dan skor pengguna. Berikut adalah contoh kode yang digunakan untuk melakukan scraping data:

```
from google_play_scraper import Sort, reviews

result, continuation_token = reviews(
    'id.co.bankbkemobile.digitalbank',
    lang='id',
    country='id',
    sort=Sort.MOST_RELEVANT,
    count=1000,
    filter_score_with=None
)
```

Gambar 2. Code Python Scraping Data

2.2 Labelling

Penandaan atau *labelling* adalah pendekatan untuk membagi data menjadi dua kategori berdasarkan nilai atau skor [14]. Nilai dengan skor kurang dari tiga dianggap Negatif, sedangkan nilai dengan skor lebih dari tiga dianggap Positif [15]. Gambar 3 menunjukkan kode proses penandaan, dan skor 3 diabaikan karena dianggap ambigu.

```
def pelabelan(score):  
    if score < 3:  
        return 'Negatif'  
    elif score == 4 :  
        return 'Positif'  
    elif score == 5 :  
        return 'Positif'  
my_df['Label'] = my_df ['score'].apply(pelabelan)  
my_df.head(50)
```

Gambar 3. Code Python Labelling

2.3 Data Cleaning

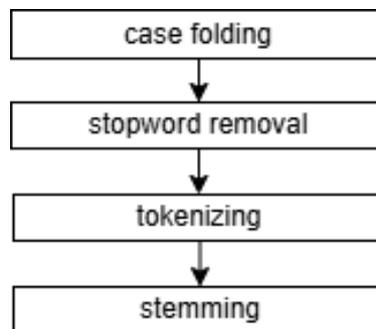
Proses pembersihan data dimulai dengan memeriksa hasil set data yang diambil dan dilabeli. Kemudian, data ditampilkan dengan nilai null dengan fitur *isna()* dan fitur *dropna()*. Kemudian simpan ke dalam file.csv [16]. Gambar 4 menunjukkan kode yang digunakan dalam proses *data cleaning*.

```
my_df.dropna(subset=['Label'], inplace=True)
```

Gambar 4. Code Python Data Cleaning

2.4 Preprocessing Text

Preprocessing teks adalah proses berikutnya di mana data yang telah diambil dan dibersihkan diformat sehingga dapat diproses ke tahap berikutnya [17]. Proses *preprocessing* teks terdiri dari empat tahap: *Case Folding*, *stopword removal*, *tokenization*, dan *stemming*, Sebagaimana ditunjukkan pada Gambar 5 di bawah ini.



Gambar 5. Alur Proses *Preprocessing* Text

- a. *Case Folding*
Semua huruf kapital dari A hingga Z diubah menjadi huruf kecil dari A hingga Z pada tahap *Case Folding*, yang memastikan bahwa format teks tetap konsisten [18].
- b. *Stopword Removal*
Langkah berikutnya adalah menghapus kata-kata yang memiliki informasi minimal atau tidak relevan dalam teks, yang dikenal sebagai stopwords, untuk menyaring elemen yang tidak penting dalam analisis [19].
- c. *Tokenizing*
Tokenizing, yang memecah teks atau kalimat menjadi bagian-bagian kata untuk dianalisis lebih lanjut, dilakukan setelah tahap penghapusan stopwords [20].
- d. *Stemming*
Tahap selanjutnya adalah stemming, yang berarti mengembalikan kata yang memiliki afiks ke bentuk aslinya. Kemudian simpan data dalam file.csv [21].

2.5 Splitting Data

Text Preprocessing: Setelah proses preprocessing selesai, langkah berikutnya adalah pembagian data. Sebelum melanjutkan ke tahap berikutnya, tujuan adalah untuk membagi data menjadi dua set: 80% untuk set pelatihan dan 20% untuk set uji [22]. Gambar 6 menunjukkan kode proses pembagian data:

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

Gambar 6. Code Python Splitting Data

2.6 Term Frequency inverse Document Frequency (TF-IDF)

Setelah pembagian data, TF-IDF digunakan untuk menganalisis data pelatihan guna mengidentifikasi istilah yang kerap muncul serta memberi masing-masing nilai yang ditentukan oleh seberapa sering istilah tersebut muncul pada dokumen, serta relevansinya terhadap keseluruhan dataset. Teknik ini membantu menilai pentingnya kata-kata tertentu dalam konteks analisis sentimen[23]. Nilai atau berat kata akan meningkat jika muncul lebih sering dalam dokumen. Gambar 7 di bawah ini menunjukkan kode proses pembobotan TF-IDF :

```
from sklearn.feature_extraction.text import TfidfVectorizer

tfidf_vectorizer = TfidfVectorizer()
tfidf_train = tfidf_vectorizer.fit_transform(X_train)
tfidf_test = tfidf_vectorizer.transform(X_test)
```

Gambar 7. Code Python TF-IDF

2.7 Naive Bayes

Setelah tahap TF-IDF, algoritma *Naive Bayes* digunakan untuk proses klasifikasi. *Naive Bayes* adalah metode klasifikasi sederhana yang menghitung peluang berdasarkan kombinasi data yang ada. Sering kali, metode ini digunakan untuk mengklasifikasikan data dengan memanfaatkan teori probabilitas. Dalam analisis sentimen, setiap kata dalam dokumen atau kumpulan data akan dihitung untuk menentukan apakah sentimennya positif atau negatif menggunakan bobot TF-IDF[24].

Rumus dasar algoritma adalah:

$$P(C|X) = \frac{P(X|C) \cdot P(C)}{P(X)}$$

Di mana:

- P(C|X)P(C|X) : Probabilitas suatu kelas CC (misalnya, sentimen positif atau negatif) diberikan data XX.
- P(X|C)P(X|C) : Probabilitas data XX muncul dalam kelas CC.
- P(C)P(C) : Probabilitas awal dari kelas CC.
- P(X)P(X) : Probabilitas data XX secara keseluruhan.

mengasumsikan bahwa setiap fitur atau variabel independen satu sama lain, yang membuat perhitungan menjadi lebih sederhana. Salah satu keunggulan utama algoritma ini adalah membutuhkan sedikit data pelatihan untuk menghasilkan model yang akurat.

Dengan menerapkan algoritma ini, data sentimen dari ulasan pengguna dapat diklasifikasikan secara efisien untuk menghasilkan wawasan yang bermanfaat.

2.8 Confusion matrix

Selanjutnya, evaluasi dilakukan menggunakan matriks kebingungan, yang sering disebut sebagai *confusion matrix*. Tujuan dari evaluasi berikut ialah guna menilai kinerja metode klasifikasi. Empat istilah utama yang dipakai pada penilaiannya yaitu *True Positives (TP)*, *True Negatives (TN)*, *False Positives (FP)*, serta *False Negatives (FN)*. Setelah menghitung nilai-nilai ini, terdapat perumusan yang diterapkan untuk menghitung akurasi, presisi, dan recall guna mengukur performa metode klasifikasi..

$$Accuracy = \frac{TP+TN}{Total} \quad (1)$$

Keterangan :

TP (True Positive) = Data prediksi serta nilai aktual positif dapat disingkat menjadi data prediksi positif atau data hasil nyata positif.

TN (True Negative) = Data dengan prediksi serta nilai aktual negatif bisa disingkat menjadi data prediksi negatif atau data hasil nyata negatif.

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

Keterangan :

TP (True Positive) = Data dengan prediksi dan nilai aktual positif bisa disebut data prediksi dan hasil aktual positif.

FP (False Positive) = Data dengan prediksi positif namun nilai aktual negatif bisa disebut data prediksi positif dengan hasil aktual negatif.

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

Keterangan :

TP (True Positive) = Data dengan prediksi dan nilai aktual positif bisa disingkat menjadi data prediksi dan hasil aktual positif.

FN (False Negative) = Data dengan prediksi negatif, meskipun nilai aktualnya positif, dapat disebut sebagai data prediksi negatif dengan hasil aktual positif.

2.9 Word Cloud

Setelah tahap evaluasi, langkah selanjutnya adalah presentasi visual melalui *Word Cloud* dan diagram. Pada tahap ini, awan kata serta grafik akan digunakan untuk mengilustrasikan istilah-istilah yang umum digunakan dalam ulasan yang menguntungkan dan tidak menguntungkan dari aplikasi Seabank, sehingga memudahkan identifikasi kata-kata yang kerap muncul dalam keseluruhan sentimen [25].

3. HASIL DAN PEMBAHASAN

Analisis sentimen pada aplikasi SeaBank dilakukan secara terstruktur menggunakan teknik yang diimplementasikan melalui Google Colab. Bagian berikutnya akan menguraikan contoh data yang didapat melalui tahapan pengumpulan data, pemberian label, pembersihan, pemrosesan teks, pembagian dataset, penggunaan TF-IDF, klasifikasi dengan , evaluasi menggunakan *Confusion Matrix*, serta visualisasi hasil menggunakan.

3.1 Scraping Data

Ulasan nasabah Seabank di Google Play Store menjadi sumber data studi berikut. Sebanyak 1000 data dihimpun dengan teknik scraping, seperti yang ditunjukkan pada Gambar 8.

	content	score
80	Sangat berguna dalam segala hal tentang keuang...	5
3	Pranktis dan simpel, mudah di gunakan, proses ...	5
530	Top up ewallet skrg yg gratis cuma dana sama s...	3
393	sangat bagus dan terpercaya tanpa bunga admin ...	5
1	Aplikasi dana digital the best pokonya.. sanga...	5

Gambar 8. Hasil Scraping Data

3.2 Labelling

Setelah *scraping*, data dilabelkan dengan dua kategori: Positif dan Negatif. Hasilnya bisa ditelaah pada Gambar 9 di bawah ini.

	content	score	Label
0	Sangat berguna dalam segala hal tentang keuang...	5	Positif
1	Pranktis dan simpel, mudah di gunakan, proses ...	5	Positif
3	sangat bagus dan terpercaya tanpa bunga admin ...	5	Positif
4	Aplikasi dana digital the best pokonya.. sanga...	5	Positif
5	Saya pengguna baru kirain mudah kirim kemanapu...	2	Negatif
6	Overall memuaskan aplikasinya... Dari UI sampa...	5	Positif
7	lebih ditingkatkan lagi pelayanannya kadang su...	4	Positif

Gambar 9. Hasil Labelling

3.3 Data Cleaning

Langkah pertama dalam proses pembersihan adalah memeriksa dataset yang diperoleh dari proses pengambilan data. Data ini diperiksa untuk mengetahui apakah terdapat nilai yang hilang atau null dengan menggunakan fungsi *isna()* dari *pandas*. Setelah itu, data yang mengandung nilai kosong atau null akan dihapus agar dataset menjadi

bersih dan siap digunakan. Dataset yang sudah dibersihkan kemudian disimpan kembali dalam format file .csv. Hasil data sebelum dan setelah proses *data cleaning* Gambar 10 dan 11 di bawah ini menunjukkan hal ini.

```

0
content 0
score 0
Label 47
dtype: int64
    
```

Gambar 10. Hasil Sebelum di lakukan *Cleaning Data*

```

0
content 0
score 0
Label 0
dtype: int64
    
```

Gambar 11. Hasil Sesudah di lakukan *Cleaning Data*

3.4 Preprocessing Text

Preprocessing, yang merupakan tahap penting dalam analisis data, terutama untuk data teks, dilakukan setelah pelabelan selesai. Tujuan dari *preprocessing* adalah untuk membuat data siap untuk analisis lebih lanjut.

Berikut adalah tahapan-tahapan yang dilakukan:

- Case Folding*: Ini merupakan proses mengubah semua huruf dalam data menjadi huruf kecil tanpa mempertimbangkan apakah huruf tersebut kapital atau tidak.
- Stop Word Removal: Kata-kata umum seperti "dan", "yang", "di", "untuk", dan sebagainya dihilangkan selama proses.
- Tokenizing: proses pembagian kalimat menjadi bagian yang lebih kecil yang terdiri dari kata-kata individu.
- Stemming: Proses ini bertujuan untuk mengubah kata-kata menjadi bentuk dasar atau akar kata dari kata tersebut.

Hasil dari tahapan-tahapan di atas dapat dilihat pada tabel berikut.

Tabel 1. Hasil Proses Preprocessing Text

Proses	Hasil
<i>Case Folding</i>	[saya top up saldo lewat dana, di aplikasi dana transaksi berhasil, tapi di aplikasi seabank saldonya belum masuk]
StopWord Removal	[top up, saldo, dana, aplikasi, dana, transaksi, berhasil, aplikasi, seabank, saldonya, masuk]
Tokenizing	[saya, top up, saldo, lewat, dana, di, aplikasi, dana, transaksi, berhasil, tapi, di, aplikasi, seabank, saldonya, belum, masuk]
Stemming	[topup, saldo, dana, aplikasi, dana, transaksi, berhasil, aplikasi, seabank, saldonya, masuk]

3.5 Splitting Data

Data pelatihan (80%) dan pengujian (20%) terdiri dari data ulasan aplikasi Seabank saat ini. Pembagian ini menggunakan metode *train_test_split* untuk menjamin distribusi kelas yang proporsional. Metode ini memungkinkan model untuk dilatih dengan data pelatihan dan diuji dengan data pengujian. Pembagian yang tepat sangat penting untuk menilai kinerja model dan mencegah *overfitting*.

Tabel 2. Hasil Proses Splitting Data

Kelas Sentimen	Data Latih (80%)	Data Uji (20%)
Negatif	528	142
Positif	232	49

Total	760	191
-------	-----	-----

3.6 TF-IDF

Untuk menghasilkan fitur data, metode Term Frequency-Inverse Document Frequency (TF-IDF) menggunakan hasil stemming. TF-IDF menghitung pentingnya setiap kata dengan mempertimbangkan frekuensi kemunculannya baik dalam dokumen tertentu maupun dalam kumpulan dokumen secara keseluruhan. TfidfVectorizer dari scikit-learn digunakan untuk menghitung bobot masing-masing kata. Tabel yang dihasilkan mencakup istilah, frekuensi kemunculan, dan bobot TF-IDF. Tabel ini dapat digunakan untuk analisis teks lanjutan atau untuk mengembangkan model pembelajaran yang lebih baik.

term	count	weight
aplikasi	1.557217	1.557217
berhasil	1.557217	1.557217
dana	1.557217	1.557217
saldo	1.557217	1.557217
topup	1.557217	1.557217
transaksi	1.537934	1.537934
seabank	1.373700	1.373700
masuk	1.272542	1.272542
tidak	0.557399	0.557399
di	0.538172	0.538172
untuk	0.538172	0.538172

Gambar 12. Hasil Proses TF-IDF

3.7 Hasil Pengujian Naive Bayes

Setelah tahap TF-IDF, algoritma digunakan untuk mengklasifikasikan data. Algoritma ini biasanya digunakan untuk mengklasifikasikan data dengan menghitung probabilitas. Pada proses ini, setiap kata dalam dokumen atau kumpulan data dianalisis untuk menentukan apakah sentimen yang terkandung adalah positif atau negatif, berdasarkan hasil perhitungan yang diperoleh dari TF-IDF.

```
[70]
from sklearn.naive_bayes import MultinomialNB

nb = MultinomialNB()
nb.fit(tfidf_train, y_train)
```

Gambar 13. Code Python Klasifikasi Navie Bayes

Hasil pengujian klasifikasikan menggunakan data TF-IDF:

- Input Data Fitur (X): Teks berisi kata-kata seperti *aplikasi*, *berhasil*, *saldo*, dsb., dengan nilai TF-IDF sebagai input model.
- Likelihood ($P(x_i|C)$): robabilitas setiap fitur diberikan kelas tertentu dihitung dari nilai weight TF-IDF.

Sebagai contoh:

$$P(\text{aplikasi}|C_1) = 1.557217$$

$$P(\text{saldo}|C_1) = 1.557217$$

$$P(\text{tidak}|C_2) = 0.553999$$

- Posterior Probabilitas ($P(C|X)$): Dihitung dengan rumus:

$$P(C|X) \propto P(C) \cdot \prod_{\{i=1\}}^{\{n\}} P(x_i|C)$$

- Klasifikasi: Pilih kelas C_{maks} dengan probabilitas posterior tertinggi.

- Perhitungan 1 data

Teks yang akan diklasifikasi mengandung kata aplikasi, saldo, dan tidak. Dengan data TF-IDF:

$$P(\text{aplikasi}|C_1) = 1.557217$$

$$P(\text{saldo}|C_1) = 1.557217$$

$$P(\text{tidak}|C_2) = 0.553999$$

Dan prior

$$P(C_1) = 0.6,$$

$$P(C_2) = 0.4P$$

Untuk C_1 :

$$P(C_1|X) = P(C_1) \cdot P(\text{saldo}|C_1) \cdot P(\text{aplikasi}|C_1)$$

$$P(C_1|X) = 0.6 \cdot 1.557217 \cdot 1.557217$$

$$P(C_1|X) = 1.454$$

Untuk C_2 :

$$P(C_2|X) = P(C_2) \cdot P(\text{tidak}|C_2)$$

$$P(C_2|X) = 0.4 \cdot 0.553999$$

$$P(C_2|X) = 0.222$$

Karena $P(C_1|X) > P(C_2|X)$ teks diklasifikasikan ke dalam kelas C_1 .

3.8 Confusion matrix

Seperti yang ditunjukkan di bawah ini, tahap berikutnya adalah tahap evaluasi menggunakan matriks kecacauan. Tujuan dari tahap ini adalah untuk memastikan bahwa hasil klasifikasi data pemeriksaan benar.

Tabel 3. Hasil Proses TF-IDF

Aktual/Prediksi	False	True	Total Aktual
Negatif	26	27	53
Positif	3	135	138
Total Prediksi	29	162	191

Setelah memperoleh nilai dari *True Positives (TP)*, *True Negatives (TN)*, *False Positives (FP)*, dan *False Negatives (FN)*, dapat dihitung *Accuracy* dengan persamaan (1), *Precision* dengan persamaan (2) dan *Recall* dengan persamaan (3) sebagai berikut :

$$Accuracy = \frac{135+27}{162} = \frac{162}{162} = 1$$

$$Precision = \frac{135}{135+3} = \frac{135}{138} = 0,97$$

$$Recall = \frac{135}{135+26} = \frac{135}{161} = 0,84$$

Langkah berikutnya adalah melakukan pengujian kinerja model dalam bahasa pemrograman Python untuk menghitung metrik evaluasi seperti akurasi, presisi, dan recall. Dengan menggunakan rumus-rumus tersebut, kita dapat mengukur sejauh mana model klasifikasi dapat mengklasifikasikan data dengan benar, serta mengidentifikasi kesalahan klasifikasi yang terjadi. Pengujian ini memberikan gambaran yang lebih jelas tentang keefektifan model dalam menangani data yang diberikan. Gambar 14 menunjukkan kode script pengujian, dan Tabel 4 menunjukkan hasilnya.

```
# Data precision, recall, f1-score, dan support
data_metrics = {
    "Kelas Sentimen": ["Negatif", "Positif"],
    "Precision": [0.90, 0.83],
    "Recall": [0.49, 0.98],
    "F1-Score": [0.63, 0.90],
    "Support": [53, 138]
}

# Membuat DataFrame
metrics_df = pd.DataFrame(data_metrics)

# Menambahkan baris untuk total
total = pd.DataFrame({"Kelas Sentimen": ["Total"],
    "Precision": [0.85],
    "Recall": [0.84],
    "F1-Score": [0.83],
    "Support": [191]})
metrics_df = pd.concat([metrics_df, total], ignore_index=True)

# Menampilkan tabel
print(metrics_df.to_string(index=False))
```

Gambar 14. Code Python Confusion matrix

Tabel 4. Hasil Proses Perhitungan *Confusion matrix* dengan menggunakan python

Kelas Sentimen	Precision	Recall	F1-Score	Support
Negatif	0.90	0.49	0.63	53
Positif	0.83	0.98	0.90	138

pelatihan dan pengujian model. Selain itu, distribusi ini membantu memberikan gambaran awal mengenai persepsi pengguna terhadap layanan, yang cenderung lebih positif.

4. KESIMPULAN

Berdasarkan hasil analisis menggunakan metode Naïve Bayes, model mampu mencapai tingkat akurasi sebesar 85%, yang menunjukkan bahwa metode ini efektif dalam mengklasifikasikan sentimen pelanggan terhadap layanan digital Seabank. Hasil perhitungan *confusion matrix* menunjukkan performa klasifikasi yang cukup baik dengan distribusi sentimen ke dalam tiga kategori utama: "Baik Sekali," "Baik," dan "Buruk." Pada kategori sentimen positif ("Baik Sekali" dan "Baik"), model berhasil mengidentifikasi sebanyak 88% data dengan benar, sementara pada kategori negatif ("Buruk"), akurasi mencapai 82%. Hasil ini menunjukkan bahwa model memiliki kemampuan yang handal dalam mengenali pola-pola sentimen, terutama pada kategori positif. Dengan demikian, penelitian ini memberikan kontribusi signifikan dalam memahami persepsi pelanggan dan dapat digunakan sebagai dasar untuk meningkatkan kualitas layanan digital yang lebih responsif terhadap kebutuhan pengguna.

DAFTAR PUSTAKA

- [1] Alfandi Safira and F. N. Hasan, "Analisis Sentimen Masyarakat Terhadap Paylater Menggunakan Metode Classifier," *Zo. J. Sist. Inf.*, vol. 5, no. 1, pp. 59–70, 2023, doi: 10.31849/zn.v5i1.12856.
- [2] R. Adyatma Subagja, Y. Widiastiwi, and N. Chamidah, "Klasifikasi Ulasan Aplikasi Jenius pada Google Play Store Menggunakan Algoritma," *Inform. J. Ilmu Komput.*, vol. 17, no. 3, p. 197, 2021, doi: 10.52958/iftk.v17i3.3652.
- [3] Jesslyn, B. Maitri, C. Hartono, F. Jennifer, J. Liana, and A. N. Fatyandri, "Analisis Strategi Bisnis dan Tingkat Persaingan dengan Kompetitor pada Perusahaan Seabank dalam Dunia Perbankan," *J. Mirai Manag.*, vol. 8, no. 2, pp. 48–52, 2023, [Online]. Available: <https://www.journal.stieamkop.ac.id/index.php/mirai/article/view/4495%0A>
- [4] M. Suhendra, W. Swastika, and M. Subianto, "Analisis Sentimen Pada Ulasan Aplikasi Video Conference Menggunakan," *Sainsbertek J. Ilm. Sains Teknol.*, vol. 2, no. 1, pp. 1–9, 2021, doi: 10.33479/sb.v2i1.145.
- [5] M. K. Insan, U. Hayati, and O. Nurdiawan, "Analisis Sentimen Aplikasi Brimo Pada Ulasan Pengguna Di," *J. Mhs. Tek. Inform.*, vol. 7, no. 1, pp. 478–483, 2023.
- [6] B. Z. Ramadhan, I. Riza, and I. Maulana, "Analisis Sentimen Ulasan Pada Aplikasi E-Commerce Dengan Menggunakan Algoritma Naïve Bayes," 2022.
- [7] A. Nadira, N. Y. Setiawan, and W. Purnomo, "Analisis Sentimen Pada Ulasan Aplikasi Mobile Banking Menggunakan Metode Naïve Bayes Dengan Kamus Inset," *Indexia*, vol. 5, no. 01, p. 35, 2023, doi: 10.30587/indexia.v5i01.5138.
- [8] O. Irnawati and K. Solecha, "Analisis Sentimen Ulasan Aplikasi Flip Menggunakan Naïve Bayes dengan Seleksi Fitur PSO," *J. Ilm. Intech Inf. Technol. J. UMUS*, vol. 4, no. 02, pp. 189–199, 2022, doi: 10.46772/intech.v4i02.868.
- [9] R. Apriani and D. Gustian, "Analisis Sentimen Dengan Naïve Bayes Terhadap Komentar Aplikasi Tokopedia," *J. Rekayasa Teknol. Nusa Putra*, vol. 6, no. 1, pp. 54–62, 2019, doi: 10.52005/rekayasa.v6i1.86.
- [10] T. A. Sari, E. Sinduningrum, and F. Noor Hasan, "KLIK: Kajian Ilmiah Informatika dan Komputer Analisis Sentimen Ulasan Pelanggan Pada Aplikasi Fore Coffee Menggunakan Metode Naïve Bayes," *Media Online*, vol. 3, no. 6, pp. 773–779, 2023, doi: 10.30865/klik.v3i6.884.
- [11] A. Nurian, "Analisis Sentimen Ulasan Pengguna Aplikasi Google Play Menggunakan Naïve Bayes," *J. Inform. dan Tek. Elektro Terap.*, vol. 11, no. 3s1, pp. 829–835, 2023, doi: 10.23960/jitet.v11i3s1.3348.
- [12] Ernianti Hasibuan and Elmo Allistair Heriyanto, "Analisis Sentimen Pada Ulasan Aplikasi Amazon Shopping Di Google Play Store Menggunakan Classifier," *J. Tek. dan Sci.*, vol. 1, no. 3, pp. 13–24, 2022, doi: 10.56127/jts.v1i3.434.
- [13] N. Lestari, E. Haerani, and R. M. Candra, "Analisa Sentimen Ulasan Aplikasi Wetv Untuk Peningkatan Layanan Menggunakan Metode Naïve Bayes," *J. Inf. Syst. Res.*, vol. 4, no. 3, pp. 874–882, 2023, doi: 10.47065/josh.v4i3.3355.
- [14] G. K. Locarso, "Analisis Sentimen Review Aplikasi Pedulilindungi Pada Google Play Store Menggunakan Nbc," *JTIK (Jurnal Tek. Inform. Kaputama)*, vol. 6, no. 2, pp. 353–361, 2022, doi: 10.59697/jtik.v6i2.207.
- [15] S. A. Azzahra and A. Wibowo, "Analisis Ulasan Wisatawan," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 4, p. 737, 2020, doi: 10.25126/jtiik.202071907.
- [16] E. Fitri, "Sentiment Analysis of the Ruangguru Application Using , Random Forest and Support Vector Machine Algorithms," *J. Transform.*, vol. 18, no. 1, p. 71, 2020.
- [17] Meliyawati and F. N. Hasan, "Analisis Sentimen Pengguna Aplikasi CapCut Pada Ulasan di Play Store Menggunakan Metode Naïve Bayes," *Media Online*, vol. 4, no. 4, pp. 2272–2280, 2024, doi: 10.30865/klik.v4i4.1555.

- [18] M. T. Nitamia and H. Februariyanti, “Analisis Sentimen Ulasan Ekpedisi J&T Expres Menggunakan Algoritma,” *J. Manaj. Inform. Sist. Inf.*, vol. 5, no. 1, pp. 20–29, 2022.
- [19] Y. Asri, W. N. Suliyanti, D. Kuswardani, and M. Fajri, “Pelabelan Otomatis Lexicon Vader dan Klasifikasi dalam menganalisis sentimen data ulasan PLN Mobile,” *Petir*, vol. 15, no. 2, pp. 264–275, 2022, doi: 10.33322/petir.v15i2.1733.
- [20] D. Darwis, N. Siskawati, and Z. Abidin, “Penerapan Algoritma Untuk Analisis Sentimen Review Data Twitter Bmkg Nasional,” *J. Tekno Kompak*, vol. 15, no. 1, p. 131, 2021, doi: 10.33365/jtk.v15i1.744.
- [21] M. Rezki, D. N. Kholifah, M. Faisal, P. Priyono, and R. Suryadithia, “Analisis Review Pengguna Google Meet dan Zoom Cloud Meeting Menggunakan Algoritma Naïve Bayes,” *J. Infortech*, vol. 2, no. 2, pp. 264–270, 2020, doi: 10.31294/infortech.v2i2.9286.
- [22] G. Darmawan, S. Alam, and M. I. Sulisty, “Analisis Sentimen Berdasarkan Ulasan Pengguna Aplikasi Mypertamina Pada Google Playstore Menggunakan Metode Naïve Bayes,” *STORAGE – J. Ilm. Tek. dan Ilmu Komput.*, vol. 2, no. 3, pp. 100–108, 2023.
- [23] U. Muhammadiyah Jember, M. Izunnahdi, G. Aburrahman, and A. Eko Wardoyo, “Sentimen Analisis Pada Data Ulasan Aplikasi KAI Access Di Google PlayStore Menggunakan Metode Multinomial Sentiment Analysis on KAI Access Application Review Data on Google PlayStore Using Multinomial Method,” *J. Smart Teknol.*, vol. 4, no. 2, pp. 2774–1702, 2023, [Online]. Available: <http://jurnal.unmuhjember.ac.id/index.php/JST>
- [24] Merinda Lestandy, Abdurrahim Abdurrahim, and Lailis Syafa’ah, “Analisis Sentimen Tweet Vaksin COVID-19 Menggunakan Recurrent Neural Network dan Naïve Bayes,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 4, pp. 802–808, 2021, doi: 10.29207/resti.v5i4.3308.
- [25] S. Juanita, “Analisis Sentimen Persepsi Masyarakat Terhadap Pemilu 2019 Pada Media Sosial Twitter Menggunakan,” *J. Media Inform. Budidarma*, vol. 4, no. 3, p. 552, 2020, doi: 10.30865/mib.v4i3.2140.