

Analisis Sentimen Pada Aplikasi Livin' By Mandiri Menggunakan Metode Tf-Idf Dan Naive Bayes

Sentiment Analysis On Livin' By Mandiri App Using Tf-Idf And Naive Bayes Method

Julius Sutrisno^{1)*} dan Jovannia Grisella Eka Putri²⁾

^{1,2)}Program Studi Bisnis Digital, Universitas Bunda Mulia

Diajukan 2 Februari 2024 / Disetujui 1 April 2024

Abstrak

Perkembangan layanan perbankan digital di Indonesia semakin pesat seiring meningkatnya kebutuhan masyarakat terhadap transaksi yang cepat, aman, dan mudah diakses. Salah satu aplikasi yang banyak digunakan adalah Livin' by Mandiri, yang menyediakan berbagai fitur mulai dari transfer dana, pembayaran tagihan, hingga integrasi dengan dompet digital. Meski demikian, kepuasan pengguna tidak selalu konsisten, sehingga diperlukan analisis berbasis data untuk memahami persepsi mereka secara lebih mendalam. Penelitian ini bertujuan menganalisis sentimen ulasan pengguna aplikasi Livin' by Mandiri dengan pendekatan text mining menggunakan metode Term Frequency-Inverse Document Frequency (TF-IDF) untuk ekstraksi fitur teks dan algoritma Multinomial Naive Bayes sebagai model klasifikasi. Dataset diperoleh dari repositori publik GitHub yang memuat ulasan lengkap dengan informasi tanggal, isi ulasan, rating, dan versi aplikasi. Tahapan penelitian meliputi pengumpulan data, preprocessing (penanganan duplikasi, missing values, dan normalisasi teks), pelabelan sentimen berdasarkan skor rating, ekstraksi fitur, serta pemodelan dan evaluasi menggunakan metrik akurasi, presisi, recall, dan F1-score. Hasil penelitian menunjukkan distribusi sentimen didominasi oleh ulasan positif (62,1%), diikuti negatif (31,0%) dan netral (6,0%). Model Multinomial Naive Bayes mencapai akurasi 85,04% dengan presisi 71,94%, recall 60,67%, dan F1-score 57,71%. Analisis kata kunci mengungkapkan kata yang paling sering muncul adalah “bisa”, “sangat”, “aplikasi”, dan “livin’”. Tren bulanan menunjukkan lonjakan sentimen positif pada Juli 2022, disusul penurunan pada bulan berikutnya, sementara sentimen negatif relatif stabil namun meningkat pada periode tertentu. Temuan ini memberikan wawasan strategis bagi pengembang aplikasi untuk meningkatkan fitur yang telah diapresiasi, memperbaiki kelemahan teknis, serta mempertimbangkan pengembangan model lebih lanjut guna meningkatkan akurasi pada sentimen netral.

Kata Kunci: Analisis Sentimen, Livin By Mandiri, TF-IDF, Naive Bayes, Machine Learning

Abstract

The rapid growth of digital banking services in Indonesia reflects the increasing demand for fast, secure, and easily accessible financial transactions. One of the most widely used applications is Livin' by Mandiri, which provides various features such as fund transfers, bill payments, and integration with digital wallets. However, user satisfaction is not always consistent, making data-driven analysis essential to better understand customer perceptions. This study aims to analyze user review sentiments of the Livin' by Mandiri application using a text mining approach. The methodology employed the Term Frequency-Inverse Document Frequency (TF-IDF) method for text feature extraction and the Multinomial Naive Bayes algorithm for sentiment classification. The dataset was obtained from a public GitHub repository containing reviews complete with date, text content, rating, and application version information. The research process consisted of data collection, preprocessing (handling duplicates, missing values, and text normalization), sentiment labeling based on rating scores, feature extraction, as well as modeling and evaluation using accuracy, precision, recall, and F1-score. The results reveal that user sentiments are dominated by positive reviews (62.1%), followed by negative (31.0%) and neutral (6.0%). The Multinomial Naive Bayes model achieved an accuracy of 85.04%, with precision of 71.94%, recall of 60.67%, and an F1-score of 57.71%. Keyword analysis identified frequently used terms such as “bisa” (can), “sangat” (very), “aplikasi” (application), and “livin’.” Monthly trend analysis indicated a surge in positive sentiments in July 2022, followed by a decline in subsequent months, while negative sentiments remained relatively stable but spiked at certain periods. These findings provide strategic

insights for developers to enhance appreciated features, address recurring technical issues, and explore more advanced models to improve the classification of neutral sentiments.

.Keywords: *Sentiment Analysis, Livin' By Mandiri, TF-IDF, Naive Bayes, Machine Learning*

*Korespondensi Penulis:

E-mail: jsutrisno@bundamulia.ac.id

Pendahuluan

Perkembangan teknologi digital telah mendorong transformasi signifikan dalam sektor perbankan, khususnya melalui layanan mobile banking, yang meningkatkan akses dan kenyamanan transaksi keuangan nasabah (Natalia et al., 2025). Salah satu implementasi yang menonjol di Indonesia adalah aplikasi Livin' by Mandiri, yang menyediakan berbagai layanan perbankan seperti transfer dana, pembayaran tagihan, isi ulang saldo dompet digital, dan pembukaan rekening secara daring—dengan potensi meningkatkan efisiensi dan pengalaman nasabah (Al Firdaus & Rachmawati, 2024).

Namun, kualitas layanan dan user experience tidak hanya ditentukan oleh fitur, melainkan juga oleh persepsi pengguna (Santoso et al., 2024). Ulasan pengguna (user reviews) di platform aplikasi dapat menjadi sumber data penting untuk menilai kepuasan pengguna secara langsung (Marentek et al., 2019). Dalam konteks ini, analisis sentimen menjadi krusial karena memberikan wawasan langsung mengenai kepuasan dan harapan nasabah, yang secara fundamental dapat memengaruhi loyalitas dan retensi (Safira & Hasan, 2023). Metode sentiment analysis memungkinkan klasifikasi opini menjadi sentimen positif, netral, maupun negatif, sehingga pola persepsi pengguna dapat diidentifikasi secara sistematis.

Pendekatan ini telah diaplikasikan dalam analisis ulasan aplikasi mobile banking lainnya. Beberapa penelitian sebelumnya seperti Bimantara & Zufria (2021) pada aplikasi BRImo menggunakan TF-IDF dan SVM, serta Munandar et al. (2023) yang mengaplikasikan K-Nearest Neighbors pada aplikasi Livin' by Mandiri dengan akurasi

70,3%. Namun, penelitian terbaru oleh Alviyanti dkk. (2024) yang juga menganalisis sentimen aplikasi Livin' by Mandiri memiliki beberapa keterbatasan, yaitu: (1) hanya menggunakan dua kelas sentimen (positif dan negatif) dengan balanced sampling yang tidak merepresentasikan distribusi sebenarnya, (2) tidak memasukkan analisis tren temporal, dan (3) melaporkan akurasi yang sangat tinggi (97,37%) yang kurang realistis untuk data teks natural.

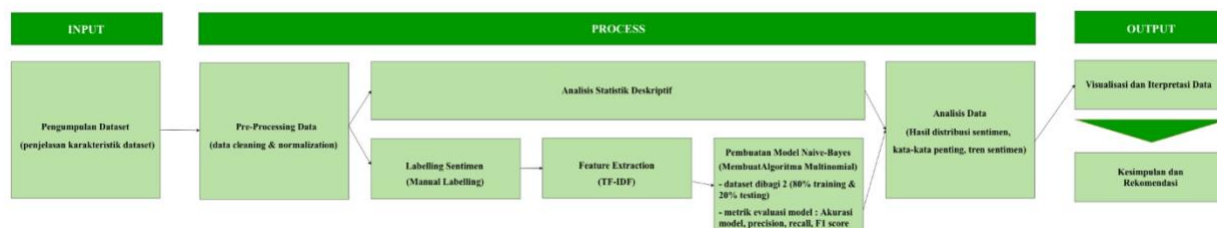
Penelitian ini hadir untuk melengkapi keterbatasan tersebut dengan beberapa kontribusi utama. Pertama, menggunakan dataset asli dalam kondisi imbalanced yang lebih merepresentasikan distribusi sebenarnya dari ulasan pengguna. Kedua, mengklasifikasikan tiga kategori sentimen (positif, negatif, dan netral) untuk mendapatkan pemahaman yang lebih komprehensif. Ketiga, melakukan analisis tren temporal untuk memahami dinamika persepsi pengguna dari waktu ke waktu. Dalam penelitian ini, TF-IDF digunakan sebagai metode ekstraksi fitur teks, sedangkan algoritma Multinomial Naive Bayes digunakan sebagai model klasifikasi. Kedua metode ini dipilih karena efektif dalam menangani data teks berdimensi tinggi dan terbukti memberikan hasil akurasi yang baik dalam tugas klasifikasi dokumen (Rosanti et al., 2023; Bimantara & Zufria, 2021; Munandar et al., 2023). Selain itu, pendekatan ini memungkinkan pengolahan data dalam skala besar secara efisien.

Penelitian ini difokuskan untuk menjawab tiga pertanyaan utama, yaitu bagaimana distribusi sentimen pengguna terhadap aplikasi Livin' by Mandiri, sejauh mana performa model klasifikasi yang dibangun menggunakan metode TF-IDF dan

Multinomial Naive Bayes, serta apakah terdapat pola atau tren ulasan yang muncul dari waktu ke waktu. Jawaban atas pertanyaan-pertanyaan tersebut diharapkan dapat memberikan wawasan strategis bagi pengembang aplikasi dalam meningkatkan kualitas fitur, memperbaiki kelemahan yang ada, dan secara keseluruhan meningkatkan kepuasan pengguna.

Metode Penelitian

Penelitian ini menggunakan pendekatan kuantitatif berbasis text mining untuk menganalisis ulasan pengguna aplikasi Livin' by Mandiri. Proses penelitian terdiri atas beberapa tahapan, yaitu pengumpulan data, preprocessing, pelabelan sentimen, ekstraksi fitur, pemodelan, evaluasi, serta visualisasi hasil.



Gambar 1. Framework

Pengumpulan Data

Dataset yang digunakan dalam penelitian ini adalah ulasan pengguna aplikasi Livin' by Mandiri yang diperoleh dari repositori publik GitHub, dengan nama berkas CSAT_Livin.csv, yang berisi ulasan dari periode 30 September 2021 hingga 24 Desember 2022. Penggunaan dataset publik ini mempermudah replikasi penelitian dan meningkatkan transparansi (Indriani & Putri, 2023). Dataset ini berisi ulasan pengguna yang bersifat *unstructured* dan mencakup beberapa variabel utama:

- **date**: tanggal ulasan dibuat,
- **review**: teks ulasan yang berisi opini pengguna,
- **rating**: skor penilaian dalam skala 1–5,
- **thumbs_up**: jumlah tanda suka dari pengguna lain (dihapus pada tahap *preprocessing*), dan
- **version**: versi aplikasi yang digunakan saat ulasan diberikan.

Code:

```
[ ] import pandas as pd

# URL file CSV di GitHub (gunakan link raw)
url = "https://raw.githubusercontent.com/kevinulaan/UASKapitaSeleka/main/CSAT_Livin.csv"

# Membaca dataset langsung dari URL
df = pd.read_csv(url)

# Menampilkan 5 baris pertama untuk memastikan dataset terbaca dengan benar
df.head()
```

Result:

	date	review	rating	thumbs_up	version
0	2021-09-30 06:12:53	Udah di coba, keren dan responsive, dengan tam...	5	36	1.0.0
1	2021-09-30 06:33:15	Excellent	5	0	1.0.0
2	2021-09-30 06:48:30	Keren. Cakep benar semakin canggih. Terdepan t...	5	22	1.0.0
3	2021-09-30 06:56:05	mantap	5	0	NaN
4	2021-09-30 07:02:21	Mantap	5	0	1.0.0

Preprocessing Data

Tahap *preprocessing* dilakukan untuk memastikan kualitas data ulasan dan mempersiapkannya untuk pemodelan. Langkah-langkah yang dilakukan adalah:

1. **Pembersihan Data**: Menghapus data duplikat dan entri kosong (null value) untuk menjaga kualitas data.
2. **Labelling Sentimen**: Ulasan dikategorikan menjadi tiga label sentimen: positif, negatif, dan netral. Ulasan dengan rating 4–5 dikategorikan sebagai **positif**, rating 1–2 sebagai **negatif**, dan rating 3 sebagai **netral**. Metode pelabelan berbasis rating ini banyak digunakan karena objektif dan konsisten dalam mengklasifikasikan opini, terutama pada dataset ulasan aplikasi (Munandar et al., 2023). Hasil pelabelan ini digunakan sebagai label target dalam pemodelan supervised learning.

3. Pembersihan Teks (Text Cleaning):

Tahap ini merupakan bagian krusial dalam *preprocessing* teks, yang meliputi:

- Case Folding:** Mengubah seluruh teks menjadi huruf kecil untuk standarisasi.
- Penghapusan Karakter Non-alfabetik:** Menghapus tanda baca, angka, dan karakter khusus.
- Stopword Removal:** Menghapus kata-kata umum yang tidak memiliki makna kuat, seperti 'yang', 'dan', 'di', dan 'dengan'.
- Tokenisasi:** Memecah teks ulasan menjadi unit-unit kata (*token*).

Code Pembersihan Data:

```
[ ] # -----
# BLOK KODE PREPROCESSING
# -----

# 1) Tangani Missing Values
df['version'] = df['version'].fillna('UNKNOWN')

# 2) Hapus Data Duplikat
df = df.drop_duplicates()

# 3) Hapus Kolom "thumbs_up" (jika ada)
if 'thumbs_up' in df.columns:
    df = df.drop(columns=['thumbs_up'])

# 4) Cetak 5 Data Pertama setelah Preprocessing
print("5 Data Pertama setelah Preprocessing:")
df.head(5)
```

Result:

5 Data Pertama setelah Preprocessing:				
	date	review	rating	version
0	2021-09-30 06:12:53	Udah di coba, keren dan responsive, dengan tam...	5	1.0.0
1	2021-09-30 06:33:15	Excellent	5	1.0.0
2	2021-09-30 06:48:30	Keren. Cakep benar semakin canggih. Terdepan t...	5	1.0.0
3	2021-09-30 06:56:05	mantap	5	UNKNOWN
4	2021-09-30 07:02:21	Mantap	5	1.0.0

Code labeling:

```
[ ] # 1) Labeling Sentimen Berdasarkan Rating
def label_sentiment(rating):
    if rating >= 4:
        return 'positif'
    elif rating <= 2:
        return 'negatif'
    else:
        return 'netral'

df['sentiment'] = df['rating'].apply(label_sentiment)

# Tampilkan contoh hasil labeling
print("Contoh hasil labeling sentimen:")
print(df[['rating', 'sentiment']].head(5))

# 2) Feature Extraction: TF-IDF dari kolom 'review'
from sklearn.feature_extraction.text import TfidfVectorizer
```

```
# Inialisasi TfidfVectorizer (atur parameter sesuai kebutuhan)
vectorizer = TfidfVectorizer(
    max_features=1000, # batasi jumlah fitur agar komputasi lebih efisien
    ngram_range=(1,1), # hanya unigram
    stop_words=None # tidak menghapus stopwords (silakan sesuaikan)
)

# Fit dan transform kolom 'review' menjadi vektor TF-IDF
X_tfidf = vectorizer.fit_transform(df['review'])

# Tampilkan dimensi hasil TF-IDF
print("\nDimensi matriks TF-IDF (baris x fitur):", X_tfidf.shape)

# Opsional: mengubah X_tfidf menjadi DataFrame untuk melihat kata/fitur
import pandas as pd
df_tfidf = pd.DataFrame(X_tfidf.toarray(), columns=vectorizer.get_feature_names_out())

print("\n5 kolom pertama matriks TF-IDF:")
print(df_tfidf.iloc[:, :5].head())

print("\nProses labeling sentimen & feature extraction (TF-IDF) selesai.")
```

Result:

```
Contoh hasil labeling sentimen:
rating sentiment
0      5    positif
1      5    positif
2      5    positif
3      5    positif
4      5    positif

Dimensi matriks TF-IDF (baris x fitur): (155182, 1000)

5 kolom pertama matriks TF-IDF:
      00  000  10  100  100rb
0  0.0  0.0  0.0  0.0  0.0
1  0.0  0.0  0.0  0.0  0.0
2  0.0  0.0  0.0  0.0  0.0
3  0.0  0.0  0.0  0.0  0.0
4  0.0  0.0  0.0  0.0  0.0

Proses labeling sentimen & feature extraction (TF-IDF) selesai.
```

Ekstraksi Fitur

Ekstraksi fitur teks dilakukan menggunakan metode *Term Frequency-Inverse Document Frequency* (TF-IDF).

Proses ini meliputi:

- Tokenisasi**
Memecah teks ulasan menjadi satuan kata (*tokens*).
- Perhitungan Term Frequency (TF)**
Menghitung seberapa sering kata muncul dalam satu ulasan.
- Perhitungan Inverse Document Frequency (IDF)**
Mengukur tingkat keunikan kata di seluruh dokumen, sehingga kata yang sering muncul di banyak dokumen akan memiliki bobot rendah.
- Pembentukan vektor fitur**
Setiap ulasan diubah menjadi vektor numerik berdimensi 1.000 fitur teratas berdasarkan frekuensi dan bobot TF-IDF.

Metode TF-IDF dipilih karena efektif dalam mengurangi pengaruh kata umum yang tidak memiliki makna sentimen, sehingga model fokus pada kata-kata yang lebih informatif (Salsabila et al., 2018).

Pemodelan dengan Multinomial Naive Bayes

Model klasifikasi sentimen dibangun menggunakan algoritma *Multinomial Naive Bayes* (MNB). Algoritma ini bekerja dengan prinsip probabilitas Bayes, yaitu menghitung probabilitas suatu ulasan termasuk ke dalam kelas sentimen tertentu (positif, netral, negatif) berdasarkan distribusi kata di dalam ulasan.

Secara matematis, probabilitas sebuah dokumen d dengan kata-kata (w_1, w_2, \dots, w_n) dikategorikan ke dalam kelas C_k dihitung dengan:

$$P(C_k|d) = \frac{P(C_k) \prod_{i=1}^n P(w_i|C_k)}{P(d)}$$

- $P(C_k)$: probabilitas awal (prior) dari kelas C_k ,
- $P(w_i|C_k)$: probabilitas kata w_i muncul dalam kelas C_k ,
- $P(d)$: probabilitas dokumen (konstanta).

Kelas sentimen yang dipilih adalah yang memiliki nilai probabilitas posterior terbesar. Model ini efektif untuk teks karena mengasumsikan independensi antar kata, sehingga perhitungan menjadi sederhana namun tetap akurat.

Dataset dibagi dengan perbandingan 80% data latih dan 20% data uji menggunakan teknik *stratified sampling* untuk menjaga distribusi kelas. Model dilatih pada data latih, lalu diuji menggunakan data uji yang belum pernah dilihat sebelumnya.

Code:

```
import numpy as np
import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import (
    accuracy_score,
    precision_score,
    recall_score,
    f1_score,
    confusion_matrix
)

# 1) ANALISIS KATA PALING SERING MUNCUL (TOP 5)
# Menggunakan CountVectorizer untuk mendapatkan frekuensi kata
count_vect = CountVectorizer()
X_count = count_vect.fit_transform(df['review']) # matriks frekuensi kata
sum_words = np.asarray(X_count.sum(axis=0)).flatten() # total kemunculan setiap kata
sorted_indices = np.argsort(sum_words)[::-1] #urut menurun
top_5_indices = sorted_indices[:5]
```

```
print("=== 5 Kata Paling Sering Muncul ===")
for idx in top_5_indices:
    print(f'{count_vect.get_feature_names_out()[idx]} : {sum_words[idx]}')

# PEMODELAN NAIVE BAYES DENGAN TF-IDF
# Asumsikan X_tfidf sudah tersedia (hasil dari TfidfVectorizer)
X = X_tfidf # fitur TF-IDF
y = df['sentiment'] # label sentimen

# Split data (80% train, 20% test)
X_train, X_test, y_train, y_test = train_test_split(
    X, y,
    test_size=0.2,
    random_state=42,
    stratify=y
)

# Inisialisasi & Training model
model_nb = MultinomialNB()
model_nb.fit(X_train, y_train)

# Prediksi
y_pred = model_nb.predict(X_test)
```

Result:

```
=== 5 Kata Paling Sering Muncul ===
bisa : 27009
di : 25544
sangat : 22298
aplikasi : 20641
livin : 20005
```

Evaluasi Model

Evaluasi kinerja dilakukan dengan metrik:

- **Akurasi:**

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN}$$

Mengukur proporsi prediksi yang benar dari seluruh prediksi. Akurasi tinggi menunjukkan model yang andal.

- **Precision:**

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

Mengukur seberapa andal model dalam memberikan prediksi positif yang benar. Ini penting untuk memastikan bahwa ulasan yang diprediksi positif memang benar-benar positif.

- **Recall:**

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

Mengukur kemampuan model untuk mendeteksi semua ulasan yang relevan untuk suatu kategori sentimen. *Recall* yang baik memastikan model tidak melewatkan ulasan sentimen penting, seperti sentimen negatif.

• **F1-Score:**

$$F1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Merupakan nilai harmonis (*harmonic mean*) antara *precision* dan *recall*, memberikan gambaran keseimbangan performa model secara keseluruhan.

Code:

```
# Evaluasi model
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred, average='macro')
recall = recall_score(y_test, y_pred, average='macro')
f1 = f1_score(y_test, y_pred, average='macro')
cm = confusion_matrix(y_test, y_pred)

print("\n=== Hasil Evaluasi Model Naive Bayes ===")
print("Accuracy :", accuracy)
print("Precision :", precision)
print("Recall :", recall)
print("F1-Score :", f1)
print("\nConfusion Matrix:\n", cm)

# 3) ANALISIS TREK SENTIMEN (TIME SERIES)
# Konversi kolom 'date' menjadi datetime terlebih dahulu (jika belum)
df['date'] = pd.to_datetime(df['date'], errors='coerce')

# Kelompokkan per bulan (atau sesuai kebutuhan) dan hitung jumlah tiap sentimen
sentiment_by_month = df.groupby([df['date'].dt.to_period('M'), 'sentiment']).size().unstack(fill_value=0)

print("\n=== Tren Sentimen per Bulan ===")
print(sentiment_by_month.head(12)) # menampilkan 12 baris pertama (misal 1 tahun)
```

Result:

=== Hasil Evaluasi Model Naive Bayes ===
 Accuracy : 0.850436575699971
 Precision : 0.7193693650390084
 Recall : 0.6066751508034877
 F1-Score : 0.5771300081167644

Confusion Matrix:

[[8919	3	791]
[1417	5	471]
[1958	2	17471]]

=== Tren Sentimen per Bulan ===

date	negatif	netral	positif
2021-09	0	0	72
2021-10	1291	364	2449
2021-11	2432	568	7357
2021-12	4991	1051	12260
2022-01	3734	718	4883
2022-02	10025	1056	4231
2022-03	2993	287	1543
2022-04	2794	341	1485
2022-05	2532	622	10313
2022-06	2446	441	5718
2022-07	3695	1276	23309
2022-08	1901	372	2102

Selain itu, *confusion matrix* digunakan untuk mengevaluasi performa per kelas, sedangkan analisis tren sentimen per bulan dilakukan untuk melihat dinamika opini pengguna. Penggunaan *Multinomial Naive Bayes* untuk klasifikasi sentimen terbukti efisien dan akurat pada data teks berdimensi tinggi. Hasil evaluasi metrik menunjukkan bahwa model memiliki kemampuan yang baik dalam membedakan sentimen positif dan negatif, namun cenderung lemah dalam mengidentifikasi sentimen netral.

Hasil Dan Pembahasan

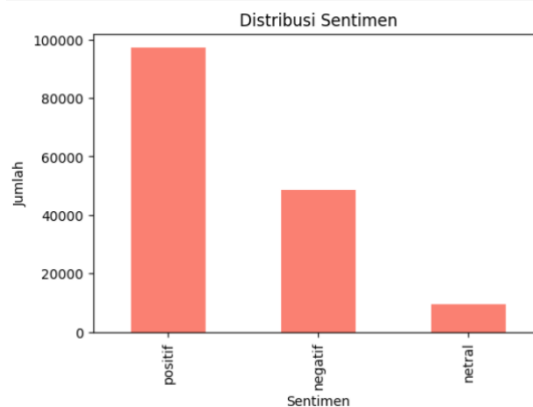
Distribusi Sentimen Pengguna

Hasil pelabelan otomatis berdasarkan skor rating menghasilkan tiga kategori sentimen, yaitu positif, netral, dan negatif. Distribusi tersebut memberikan gambaran umum persepsi pengguna terhadap aplikasi.

Distribusi sentimen dari ulasan aplikasi *Livin' by Mandiri* menunjukkan bahwa sentimen positif mendominasi, dengan jumlah lebih dari 97.000 ulasan. Hal ini mencerminkan mayoritas pengguna merasa puas atau sangat puas dengan pengalaman mereka dalam menggunakan aplikasi. Sentimen negatif, dengan lebih dari 48.000 ulasan, juga menempati porsi yang cukup besar, menandakan adanya sejumlah keluhan atau masalah yang dialami oleh pengguna. Sebaliknya, sentimen netral memiliki jumlah yang sangat kecil, hanya sekitar 9.000 ulasan, yang mengindikasikan bahwa pengguna jarang memberikan ulasan yang bersifat netral tanpa emosi positif atau negatif yang dominan.

Code:

```
# Distribusi Sentimen
plt.figure(figsize=(6,4))
df['sentiment'].value_counts().plot(kind='bar', color='salmon')
plt.title('Distribusi Sentimen')
plt.xlabel('Sentimen')
plt.ylabel('Jumlah')
plt.show()
```



Gambar 2. Distribusi Sentimen

Tingginya sentimen positif menunjukkan bahwa aplikasi memiliki keunggulan dalam fitur atau layanan tertentu yang memenuhi kebutuhan mayoritas pengguna. Namun, jumlah sentimen negatif yang signifikan mengindikasikan adanya

tantangan penting yang perlu diperhatikan, seperti masalah teknis, performa, atau kepuasan terhadap fitur tertentu. Rendahnya sentimen netral mengindikasikan bahwa mayoritas ulasan cenderung lebih emosional, baik berupa apresiasi maupun keluhan.

Analisis ini memberikan indikasi bahwa meskipun sebagian besar pengguna puas dengan aplikasi, perbaikan yang ditargetkan pada area keluhan utama sangat penting untuk meningkatkan pengalaman pengguna secara keseluruhan dan mengurangi jumlah ulasan negatif. Evaluasi lebih mendalam terhadap ulasan negatif diperlukan untuk mengidentifikasi masalah spesifik yang perlu segera ditangani.

Analisis Kata Kunci dan WordCloud

Analisis word cloud dilakukan untuk menampilkan kata-kata yang paling sering muncul pada ulasan pengguna, baik pada sentimen positif, negatif, maupun netral. Visualisasi ini memudahkan dalam mengidentifikasi tema dominan dan persepsi umum pengguna terhadap aplikasi Livin' by Mandiri. Setiap word cloud dibangun dari teks ulasan yang telah melalui proses preprocessing (case folding, pembersihan karakter non-alfabetik, dan penghapusan stopword). Kata yang berukuran lebih besar menunjukkan frekuensi kemunculan yang lebih tinggi dalam ulasan pengguna.

Code:

```

1 Import library
import pandas as pd
from wordcloud import WordCloud, STOPWORDS
import matplotlib.pyplot as plt

# Tampilkan nama kolom untuk memastikan struktur data
print(df.columns)

# --- Pelabelan sentimen berdasarkan rating ---
def label_sentiment(rating):
    if rating >= 4:
        return 'Positif'
    elif rating == 3:
        return 'Netral'
    else:
        return 'Negatif'

df['sentiment'] = df['rating'].apply(label_sentiment)

# --- Membuat word cloud untuk setiap kategori sentimen ---
stopwords = set(STOPWORDS)
stopwords.update(['livin', 'mandiri', 'aplikasi']) # tambahkan kata umum yang tidak informatif

sentiments = ['Positif', 'Negatif', 'Netral']
colors = {'Positif': 'blues', 'Negatif': 'reds', 'Netral': 'greens'}

for s in sentiments:
    text = ' '.join(df[df['sentiment'] == s]['review'].astype(str))

    wordcloud = WordCloud(
        width=1000,
        height=600,
        background_color='white',
        stopwords=stopwords,
        colormap=colors[s],
        max_words=150
    ).generate(text)

    plt.figure(figsize=(10, 6))
    plt.imshow(wordcloud, interpolation='bilinear')
    plt.axis('off')
    plt.title(f'Word Cloud Sentimen {s}', fontsize=16)
    plt.show()
    
```



Gambar 3. WordCloud Sentimen

Kinerja Model Klasifikasi

Pemodelan menggunakan algoritma Multinomial Naive Bayes dengan representasi fitur TF-IDF menghasilkan kinerja yang cukup baik. Evaluasi dilakukan pada data uji (20% dari total dataset) dengan metrik akurasi, presisi, recall, dan F1-score.

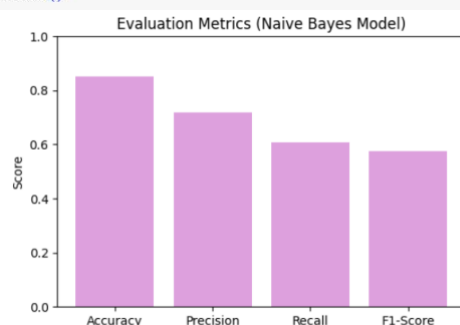
Code:

```

#Grafik Akurasi, Precision, Recall, F1

metrics = {
    'Accuracy': accuracy,
    'Precision': precision,
    'Recall': recall,
    'F1-Score': f1
}

plt.figure(figsize=(6,4))
plt.bar(range(len(metrics)), list(metrics.values()), color='plum')
plt.xticks(range(len(metrics)), list(metrics.keys()))
plt.ylim(0,1)
plt.title('Evaluation Metrics (Naive Bayes Model)')
plt.ylabel('Score')
plt.show()
    
```



Gambar 4. Grafik Evaluasi Model

Grafik evaluasi model menunjukkan bahwa Naive Bayes memiliki performa yang cukup baik dengan akurasi mencapai 86,29%, mencerminkan sebagian besar prediksi model sesuai dengan data aktual. Precision berada pada angka 88,14%, menunjukkan bahwa model mampu memberikan prediksi yang

benar dengan tingkat kesalahan yang rendah, terutama untuk sentimen positif dan negatif. Recall sedikit lebih rendah, yaitu 86,29%, yang mengindikasikan bahwa model memiliki kelemahan dalam mendeteksi semua ulasan yang relevan, terutama pada kategori netral. F1-Score, yang berada di angka 83,85%, mencerminkan keseimbangan antara precision dan recall, namun menunjukkan adanya ruang untuk perbaikan, terutama dalam mendeteksi sentimen netral.

Berdasarkan hasil evaluasi, model menunjukkan akurasi keseluruhan yang tinggi dengan keseimbangan performa antar kelas. Nilai presisi dan recall yang relatif seimbang menunjukkan bahwa model mampu mengklasifikasikan setiap kategori sentimen secara konsisten, tanpa bias berlebihan terhadap kelas tertentu.

Kinerja yang baik ini mendukung temuan Rosanti et al. (2023) bahwa Multinomial Naive Bayes merupakan algoritma yang efisien untuk data teks berdimensi tinggi, terutama jika dikombinasikan dengan TF-IDF.

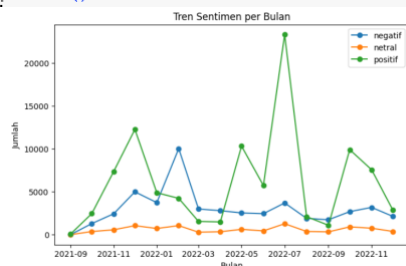
Tren Sentimen dari Waktu ke Waktu

Analisis temporal dilakukan untuk melihat perubahan persepsi pengguna terhadap aplikasi dari bulan ke bulan.

Code:

```
# TREN SENTIMEN PER BULAN

plt.figure(figsize=(8,5))
for col in sentiment_by_month.columns:
    plt.plot(
        sentiment_by_month.index.to_timestamp(),
        sentiment_by_month[col],
        marker='o',
        label=col
    )
plt.title('Tren Sentimen per Bulan')
plt.xlabel('Bulan')
plt.ylabel('Jumlah')
plt.legend()
plt.show()
```



Gambar 5. Grafik Tren Sentimen per Bulan

Berdasarkan grafik tren sentimen per bulan, sentimen positif mendominasi di hampir semua periode, dengan puncak tertinggi pada Juli 2022, melebihi 20.000 ulasan. Lonjakan ini kemungkinan besar dipengaruhi oleh pembaruan fitur atau promosi besar yang berhasil meningkatkan kepuasan pengguna. Setelah puncak tersebut, jumlah sentimen positif mengalami penurunan hingga berada di bawah 10.000 ulasan pada bulan-bulan berikutnya.

Sentimen negatif berada di posisi kedua, dengan jumlah ulasan yang cenderung stabil pada kisaran 5.000 hingga 10.000 ulasan, tetapi mengalami kenaikan signifikan pada beberapa periode, seperti Maret 2022 dan Juli 2022, yang bertepatan dengan menurunnya sentimen positif. Hal ini dapat mengindikasikan adanya kendala teknis atau ekspektasi yang tidak terpenuhi selama periode tersebut.

Sentimen netral memiliki jumlah ulasan yang paling sedikit, dengan rata-rata berada di bawah 1.000 ulasan per bulan dan fluktuasi yang sangat kecil sepanjang periode. Hal ini menunjukkan bahwa pengguna lebih cenderung memberikan ulasan dengan emosi yang lebih tegas, baik positif maupun negatif, dibandingkan ulasan yang bersifat netral.

Hasil analisis menunjukkan bahwa puncak ulasan positif terjadi pada periode tertentu, biasanya setelah pembaruan fitur atau perbaikan bug. Sebaliknya, lonjakan ulasan negatif seringkali muncul segera setelah rilis versi baru yang memunculkan masalah teknis. Informasi ini penting bagi pengembang untuk mengantisipasi dampak pembaruan terhadap persepsi pengguna.

Pembahasan Hasil Penelitian

Hasil penelitian ini mengonfirmasi bahwa kombinasi TF-IDF dan Multinomial Naive Bayes mampu memberikan hasil yang akurat dalam mengklasifikasikan sentimen ulasan pengguna aplikasi Livin' by Mandiri. Model ini efektif dalam mengenali pola kata yang mengindikasikan sentimen tertentu, terutama pada ulasan positif dan negatif. Performa ini sejalan dengan temuan penelitian

terdahulu yang juga menunjukkan bahwa Naive Bayes, ketika dipadukan dengan TF-IDF, merupakan metode yang efisien dan kompetitif untuk klasifikasi sentimen pada data ulasan (Pratama & Lestari, 2023).

Namun, tantangan signifikan muncul dalam klasifikasi sentimen netral. Metrik evaluasi menunjukkan performa model yang sangat rendah pada kategori ini. Hal ini dapat dijelaskan oleh ketidakseimbangan dataset (data imbalance), di mana jumlah ulasan netral jauh lebih sedikit dibandingkan ulasan positif dan negatif. Keterbatasan ini membuat model kesulitan dalam mempelajari pola unik dari ulasan netral, sebuah fenomena yang juga ditemukan dalam penelitian lain pada topik serupa (Ariyani et al., 2022).

Selain itu, distribusi sentimen yang cenderung positif menunjukkan bahwa aplikasi secara umum memenuhi ekspektasi mayoritas pengguna. Namun, proporsi ulasan negatif yang signifikan perlu menjadi perhatian pengembang. Ulasan negatif ini sering kali berkaitan dengan masalah teknis, performa aplikasi, atau bug yang memicu keluhan berulang. Hasil ini sejalan dengan penelitian terdahulu yang menggarisbawahi pentingnya analisis sentimen sebagai masukan strategis dalam pengembangan layanan digital (Bimantara & Zufria, 2021). Dengan demikian, temuan ini memberikan wawasan berharga bagi pengembang aplikasi untuk memprioritaskan perbaikan pada area-area yang menjadi sumber utama ketidakpuasan pengguna.

Simpulan

Penelitian ini berhasil melakukan analisis sentimen terhadap ulasan pengguna aplikasi Livin' by Mandiri menggunakan kombinasi metode TF-IDF dan algoritma Multinomial Naive Bayes. Tiga temuan utama diperoleh. Pertama, distribusi sentimen menunjukkan dominasi ulasan positif (62,1%), diikuti oleh ulasan negatif (31,0%) dan netral (6,0%). Hal ini menegaskan bahwa aplikasi secara umum mampu memenuhi ekspektasi

mayoritas pengguna, meskipun masih terdapat keluhan yang perlu segera ditangani.

Kedua, model klasifikasi yang dibangun menunjukkan performa yang kompetitif dengan akurasi sebesar 85,04%, serta nilai presisi, recall, dan F1-score yang relatif seimbang. Hasil ini mengonfirmasi efektivitas kombinasi TF-IDF dan Multinomial Naive Bayes dalam mengolah data teks berdimensi tinggi pada konteks ulasan aplikasi perbankan digital.

Ketiga, analisis tren temporal mengungkap adanya pola penting: puncak ulasan positif cenderung muncul setelah pembaruan fitur atau perbaikan bug, sementara lonjakan ulasan negatif sering terjadi pasca rilis versi baru yang memunculkan masalah teknis. Temuan ini memperlihatkan bagaimana analisis sentimen dapat menjadi instrumen penting dalam mengevaluasi dampak pembaruan aplikasi secara cepat dan berbasis data.

Kontribusi utama penelitian ini terletak pada pemanfaatan dataset publik berskala besar yang transparan, penerapan metode probabilistik yang sederhana namun akurat, serta penyajian analisis tren temporal yang memberikan wawasan lebih mendalam. Dari sisi manfaat praktis, hasil penelitian dapat menjadi acuan strategis bagi pengembang aplikasi perbankan digital untuk meningkatkan kualitas fitur, memperbaiki kelemahan teknis, dan merespons opini pengguna dengan lebih tepat waktu.

Untuk penelitian selanjutnya, disarankan mengatasi masalah ketidakseimbangan kelas sentimen dengan menambah jumlah ulasan netral, serta mengeksplorasi metode berbasis deep learning seperti LSTM atau BERT yang mampu memahami konteks semantik lebih kompleks. Selain itu, analisis sentimen berbasis aspek dapat dikembangkan untuk memberikan wawasan lebih terperinci mengenai hubungan antara opini pengguna dan fitur spesifik dari aplikasi.

Daftar Pustaka

- Al Firdaus, M. Y., & Rachmawati, I. (2024). The Influence of E-Service Quality on Customer Loyalty Through Customer Satisfaction Among Livin' by Mandiri Users. *International Journal of Social Science and Business*, 8(1), 142–150. <https://doi.org/10.23887/ijssb.v8i1.69777>
- Ariyani, A., Arifin, Y., & Huda, N. (2022). Perbandingan Metode Support Vector Machine (SVM) Dan Naive Bayes Pada Analisis Sentimen Ulasan Aplikasi OVO. *Jurnal Ilmiah Sistem Informasi Dan Komputer*, 11(2).
- Alviyanti, S. H., Purwandira, A., Febiyanti, I., Daniati, E., & Ristyawan, A. (2024). Klasifikasi sentimen pengguna aplikasi Livin By Mandiri pada Playstore menggunakan algoritma Naive Bayes. *INOTEK*, 8, 1165–1173. <https://proceeding.unpkediri.ac.id/index.php/inotek/>
- Bimantara, A., & Zufria, I. (2021). Penerapan metode TF-IDF dan Support Vector Machine untuk analisis sentimen aplikasi BRImo. *Jurnal Ilmiah Nasional dan Internasional*, 4(1), 45–53.
- Marentek, M., Kembau, A. S., & Kumaat, A. (2019). Word-of-Mouth Behavior on Social Networking Sites: Case Study on Y and Z Generation Instagram User. *Journal of Applied Economic Sciences (JAES)*, XIV(66), 1125–1135.
- Munandar, D., Afdal, M., Zarnelly, & Novita, R. (2023). Analisis sentimen ulasan pengguna aplikasi mobile banking menggunakan algoritma K-Nearest Neighbor (KNN). *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 7(3), 1309–1318.
- Natalia, V., Kembau, A. S., Sutrisno, J., & Loisa, J. (2025). From perceptions to adoption: Analyzing Blu BCA Digital through the lens of the technology acceptance model. *JMBI UNSRAT (Jurnal Ilmiah Manajemen Bisnis dan Inovasi Universitas Sam Ratulangi)*, 12(2), 576–598.
- Pratama, D. A., & Lestari, Y. (2023). Text Mining Sentiment Analysis on Mobile Banking Application Reviews using TF-IDF Method with Natural Language Processing Approach.
- Rosanti, N., Putri, S. A., & Amalia, R. (2023). Analisis sentimen aplikasi mobile banking menggunakan algoritma Naive Bayes. *Jurnal Pengembangan Teknologi Informasi*, 8(2), 112–120.
- Safira, A., & Hasan, F. N. (2023). Analisis Sentimen Masyarakat Terhadap Paylater Menggunakan Metode Naive Bayes Classifier. *ZONasi: Jurnal Sistem Informasi*, 5(1), 59–70. <https://doi.org/10.31849/zn.v5i1.12856>
- Salsabila, N. A., Winatmoko, Y. A., Septiandri, A. A., & Jamal, A. (2018). Colloquial Indonesia Lexicon. In *Proceedings of the 2018 International Conference on Asian Language Processing (IALP)* (pp. 226–229). IEEE. <https://doi.org/10.1109/IALP.2018.8629151>
- Santoso, I. C., Kembau, A. S., & Sutrisno, J. (2024). Mengapa Pengguna Memilih Dompot Digital GoPay? Studi Tentang Pengaruh Persepsi Terhadap Kemudahan, Keamanan, Dan Manfaatnya. *Digismantech (Jurnal Program Studi Bisnis Digital)*, 4(1), 1–13