

**Jutisi:** Jurnal Ilmiah Teknik Informatika dan Sistem Informasi  
<https://ojs.stmik-banjarbaru.ac.id/index.php/jutisi/index>  
 Jl. Ahmad Yani, K.M. 33,5 - Kampus STMIK Banjarbaru  
 Loktabat – Banjarbaru (Tlp. 0511 4782881), e-mail: [puslit.stmikbjb@gmail.com](mailto:puslit.stmikbjb@gmail.com)  
 e-ISSN: 2685-0893

## Analisis Sentimen Deepseek Berdasarkan Ulasan Google Play Store Menggunakan Metode *Naïve Bayes*

DOI: <http://dx.doi.org/10.35889/jutisi.v14i2.2778>

Creative Commons License 4.0 (CC BY – NC)



**Khalid Al Mas'ud<sup>1\*</sup>, Muhammad Izzan Fieldi<sup>2</sup>, M. Hadi Al-Farisy<sup>3</sup>, M. Alfarizi<sup>4</sup>, Fathoni<sup>5</sup>, Ali Ibrahim<sup>6</sup>**

Sistem Informasi, Universitas Sriwijaya, Palembang, Indonesia

\*e-mail *Corresponding Author*: [khalidal78673@gmail.com](mailto:khalidal78673@gmail.com)

### Abstrak

*The advancement of artificial intelligence technologies, including the Deepseek application, has increasingly emphasized the importance of user experience. User reviews on the Google Play Store serve as a key source for evaluating application quality. However, the large volume of reviews presents challenges for effective sentiment analysis. This study investigates user sentiment toward Deepseek by applying the Naïve Bayes algorithm, combined with the Term Frequency-Inverse Document Frequency (TF-IDF) weighting technique. The dataset, obtained from Kaggle, contains 15,124 reviews categorized into positive, neutral, and negative sentiments. Model evaluation was conducted using 5-Fold Cross Validation. Results indicate that the model achieved an average accuracy of 87%, with the highest performance observed in the positive sentiment category. These findings may serve as a valuable reference for developers aiming to improve the overall quality and user satisfaction of the Deepseek application.*

**Key words:** *Data Mining; Sentiment Analysis; Naïve Bayes; Deepseek*

### Abstrak

Perkembangan teknologi kecerdasan buatan, termasuk aplikasi Deepseek, semakin menekankan pentingnya pengalaman pengguna. Ulasan pengguna di Google Play Store menjadi sumber data utama dalam mengevaluasi kualitas aplikasi. Namun, banyaknya ulasan yang tersedia menyulitkan proses analisis sentimen secara manual. Penelitian ini menganalisis sentimen pengguna terhadap Deepseek dengan menggunakan algoritma *Naïve Bayes* yang dikombinasikan dengan metode pembobotan *Term Frequency-Inverse Document Frequency* (TF-IDF). Dataset yang digunakan berasal dari Kaggle, terdiri dari 15.124 ulasan dan diklasifikasikan ke dalam tiga kategori sentimen: positif, netral, dan negatif. Validasi model dilakukan menggunakan metode *5-Fold Cross Validation*. Hasil menunjukkan model memiliki akurasi rata-rata sebesar 87%, dengan performa tertinggi pada kategori sentimen positif. Temuan ini diharapkan dapat menjadi acuan bagi pengembang dalam meningkatkan kualitas layanan dan kepuasan pengguna aplikasi Deepseek secara berkelanjutan.

**Kata kunci:** *Data Mining; Analisis Sentimen; Naïve Bayes; Deepseek*

### 1. Pendahuluan

Adanya kemajuan teknologi informasi termasuk yang mendorong timbulnya berbagai aplikasi berbasis kecerdasan buatan, termasuk aplikasi Deepseek. Seiring berkembangnya jumlah pengguna, penilaian terhadap kualitas penyajian dan kepuasan pengguna menjadi aspek penting yang harus diprioritaskan oleh developer aplikasi. Analisa terhadap sentimen pengguna melalui ulasan di media seperti Google Play Store menjadi pendekatan penting dalam menghantam persepsi pengguna terhadap kinerja aplikasi [1].

Karena itu, penyelidikan yang mengkaji sentimen pengguna terhadap aplikasi Deepseek sebagai fenomena yang diteliti memiliki jumlah pendayagunaan yang diperkirakan cukup tinggi dalam komunitas Android, yang tercermin dalam ribuan ulasan pengguna di Google Play Store. Namun, timbullah masalah utama yaitu belum adanya analisis sistematis terkait persepsi

pengguna apakah ulasan tersebut positif, netral, atau negatif. Sebab, meskipun di dalam jumlah ulasannya banyak, belum ada analisis sistematis terkait persepsi pengguna terhadap aplikasi ini. Hal itu memicu masalah utama, yaitu belum adanya indikasi yang jelas dan terstruktur.

Untuk memecahkan masalah tersebut, penelitian ini menjadikan pendekatan analisis sentimen berbasis machine learning berdasarkan algoritma *Naïve Bayes* yang digabungkan dengan teknik pembobotan TF-IDF. Pasangan metode tersebut sering digunakan dalam penelitian-penelitian sebelumnya dan terbukti berlaku berhasil dalam klasifikasinya atas data teks dalam jumlah besar [2]. *Naïve Bayes* ditemukan andal dalam klasifikasi efisien dan cepat, sedangkan TF-IDF berfungsi berusaha untuk meningkatkan relevansi teks atribut yang dianalisa. Penggunaan kedua metode tersebut dipilih karena berhasil memberi hasil akurat cukup, walau data yang digunakan tidak seimbang, serta memiliki keunggulan dari sisi efisiensi komputasi [3].

Penelitian ini menjadikan tujuan untuk menganalisa sentimen pengguna terhadap aplikasi Deepseek berdasarkan hasil ulasan di Google Play Store menggunakan metode *Naïve Bayes* serta TF-IDF. Hasil penelitian ini diharapkan bisa berkontribusi sebagai bahan evaluasi dan saran bagi developer aplikasi untuk membenahi kualitas penyajian dan pengalaman pengguna [4]. Selain itu, penelitian ini juga berkontribusi terhadap penyempurnaan metode analisa sentimen efisien dan akurat, terutama dalam konteks aplikasi kecerdasan buatan.

## 2. Tinjauan Pustaka

Seiring dengan meningkatnya penggunaan aplikasi berbasis kecerdasan buatan, banyak penelitian yang telah dilakukan untuk menganalisis sentimen pengguna terhadap platform AI maupun menggunakan metode berbasis AI untuk tujuan klasifikasi opini. Salah satunya ditunjukkan oleh Samanmali dan Rupasingha, yang melakukan analisis sentimen pada ulasan pengguna dari 15 aplikasi populer di Google Play Store dengan menggunakan pendekatan *deep learning* [5]. Penelitian ini memproses ulasan teks mentah untuk mengidentifikasi opini pengguna terhadap performa aplikasi. Mereka menerapkan model neural network untuk mengklasifikasikan sentimen positif, negatif, dan netral, dengan parameter utama berupa fitur hasil preprocessing teks dan tingkat akurasi klasifikasi.

Ahammad et al. menganalisis lebih dari 122.000 ulasan dari aplikasi berbasis kecerdasan buatan yang tersedia di Google Play Store [6]. Mereka menggunakan model *deep learning* berbasis LSTM dan CNN untuk mendeteksi sentimen pengguna. Penelitian ini menekankan pentingnya tahapan preprocessing, tokenisasi, serta pemilihan arsitektur model dalam meningkatkan akurasi. Evaluasi dilakukan dengan parameter seperti akurasi, *precision*, *recall*, dan *F1-score*.

Sagala dan Samuel menganalisis ulasan pengguna terhadap aplikasi ChatGPT dengan menggunakan tiga metode klasifikasi: *Naïve Bayes*, *Random Forest*, dan *Support Vector Machine* [7]. Dari 2.652 ulasan, mereka melakukan evaluasi model menggunakan akurasi dan confusion matrix. Hasilnya menunjukkan bahwa *Naïve Bayes* tetap kompetitif dan efisien dalam pengolahan teks berukuran besar untuk klasifikasi sentimen terhadap aplikasi berbasis AI.

Metode ini juga relatif efektif untuk mengolah *dataset* yang tidak seimbang, seperti jumlah ulasan positif dan negatif yang tidak berproporsi [8]. Meskipun begitu, metode *Naïve Bayes* juga memiliki beberapa kelemahan, seperti ketidakmampuannya dalam menampung konteks antar kata, serta kekurangan dalam mengklasifikasikan kalimat yang mengandung sarkasme atau ambiguitas [9]. Untuk mengatasi kekurangan tersebut, kombinasi teknik TF-IDF dengan *Naïve Bayes* digunakan secara meluas untuk menambah representasi fitur teks dan melanjutkan ke akurasi klasifikasi [1].

Penelitian ini membawa pendekatan yang berbeda dengan menargetkan analisis sentimen terhadap aplikasi Deepseek, yaitu aplikasi AI yang secara khusus belum banyak dikaji sebelumnya. Penelitian ini juga mengklasifikasikan tiga jenis sentimen (positif, netral, negatif), bukan hanya dua seperti pada beberapa penelitian terdahulu. Kombinasi metode *Naïve Bayes* dan TF-IDF digunakan untuk ekstraksi dan klasifikasi fitur teks, serta divalidasi melalui teknik *5-Fold Cross Validation*. Dengan demikian, penelitian ini menawarkan kebaruan dalam objek kajian, kedalaman kategori sentimen, serta integrasi teknik validasi yang lebih sistematis.

## 3. Metodologi

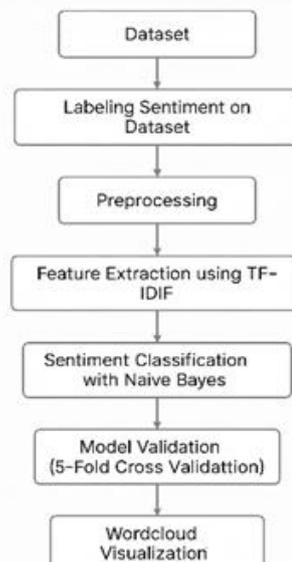
Metode pendekatan pada penelitian ini adalah melalui analisis sentimen berbasis *machine learning*. Penelitian ini melibatkan metode statistik kuantitatif karena salah satu kelebihan

adalah efektivitasnya dalam mengolah data meskipun volumenya besar, dan dalam implementasinya, seluruh proses analisis dilakukan sepenuhnya menggunakan bahasa pemrograman Python. Python dipilih karena kemampuannya yang handal dalam mengolah data dan mendukung kebutuhan [10] [11]. Semua tahapan, mulai dari membaca data, membersihkan teks, memecah kalimat menjadi token, menghapus kata-kata umum yang kurang penting, hingga mengekstraksi fitur penting menggunakan TF-IDF, dilakukan menggunakan Python.

Penelitian ini menggunakan *dataset* dari Kaggle yang berisi 15.124 ulasan pengguna aplikasi Deepseek pada Google Play Store. Kaggle dipilih untuk mempercepat proses pengumpulan data melalui cara yang sistematis, singkat, dan terstruktur, dianjurkan dalam penelitian berbasis analisis sentimen otomatis. Data yang telah diperoleh kemudian diproses dengan beberapa teknik *preprocessing* teks. Setelah data dalam kondisi siap, metode TF-IDF digunakan untuk mengekstraksi fitur secara efisien, dengan menghitung frekuensi dan distribusi kata [12] [13].

Selanjutnya, model *Naïve Bayes* diterapkan untuk melakukan klasifikasi sentimen. *Naïve Bayes* dikenal memiliki performa klasifikasi yang cocok untuk pengujian sentimen dalam skala besar [14]. Sentimen dikategorikan ke dalam tiga kelas: positif, netral, dan negatif, untuk menilai apakah ulasan tersebut baik (*good*), buruk (*bad*), atau netral. Metode ini merupakan salah satu teknik klasifikasi teks yang paling banyak digunakan dan terbukti efektif, terutama dengan fitur yang dihasilkan dari TF-IDF [15].

Proses klasifikasi tersebut kemudian divalidasi menggunakan metode *cross-validation*. Model selanjutnya dievaluasi berdasarkan akurasi, presisi, *recall*, dan *F1-score*, yang mencerminkan sejauh mana kualitas prediksi model dalam mengklasifikasikan sentimen secara menyeluruh [16]. Untuk memastikan model dapat digunakan secara umum dan tidak hanya sesuai untuk satu jenis data, evaluasi performa dilakukan dengan metode *5-Fold Cross Validation*. Hasil analisis dan klasifikasinya kemudian divisualisasikan menggunakan pustaka seperti *matplotlib*, *seaborn*, dan *wordcloud*. Karena kelengkapan pustaka serta kemudahan yang ditawarkan Python, seluruh proses berjalan lebih praktis, otomatis, dan mudah untuk diulang jika diperlukan.



Gambar 1. Alur penelitian

Gambar 1 menerangkan aliran penelitian yang berisi tujuh tahap utama. Tahap pertama yaitu pengumpulan *dataset*, yaitu ulasan pengguna terhadap aplikasi Deepseek yang diambil dari *website* Kaggle. Data berisi lebih dari 15.000 ulasan sebagai awal proses analisis sentimen. Selanjutnya, dilaksanakan pelabelan sentimen berdasarkan skoring ulasan pengguna. Skoring sebesar 1–2 digolongkan sebagai sentimen negatif, skoring sebesar 3 sebagai netral, serta skoring sebesar 4–5 sebagai sentimen positif.

Tahap ketiga yaitu *preprocessing*, yaitu pembersihan data teks agar diolah untuk dianalisis. Tahap yang disediakan mencakup konversi huruf menjadi huruf kecil (*lowercasing*), eliminasi karakter non-alfabet, penghilangan kata berhenti, serta *stemming* untuk mempermudah bentuk

kata. Setelah bersih, dilakukan ekstraksi fitur menggunakan metode *Term Frequency-Inverse Document Frequency* (TF-IDF). Teknik ini membobot kata dalam dokumen agar bisa direpresentasikan dalam bentuk numerik yang menggambarkan kepentingan tiap kata secara kontekstual.

Data sentimen hasil ekstraksi kemudian diproses di tahap klasifikasi sentimen menggunakan algoritma *Naïve Bayes*. Pada algoritma ini digunakan karena performanya tinggi untuk klasifikasi teks, terutama untuk data berukuran besar dan tidak seimbang. Setelah proses pembentukan model, dilakukan test akurasi menggunakan metode *5-Fold Cross Validation*. Prosedur ini memecah data menjadi lima bagian untuk membuktikan adanya konsistensi dan generalisasi model. Evaluasi dilakukan berdasarkan akurasi, precision, recall, dan F1-score. Tahapan akhir adalah pengembangan wordcloud yang memvisualisasikan kata dominan masing-masing kategori sentimen, sehingga mempermudah interpretasi persepsi pengguna secara visual dan intuitif.

### 3.1. *Naïve Bayes*

*Naïve Bayes* adalah metode klasifikasi dalam *machine learning* yang berdasarkan teorema Bayes dengan asumsi bahwa setiap pasangan fitur adalah independen satu sama lain [17].

### 3.2 *Term Frequency-Inverse Document Frequency* (TF-IDF)

TF-IDF adalah vektor dari representasi teks metode yang membandingkan kemunculan spesifik kata dalam dokumen yang spesifik ke seluruh sekumpulan dokumen [18].

### 3.3 Validasi Model: *5-Fold Cross Validation*

*5-Fold Cross Validation* adalah teknik evaluasi model yang terdiri dari membagi *dataset* menjadi lima bagian dengan ukuran yang hampir sama. Model dilatih sebanyak lima kali, dengan kombinasi data pelatihan dan data validasi yang berbeda pada setiap iterasi [19].

## 4. Hasil dan Pembahasan

Penelitian yang telah kami selesaikan adalah analisis sentimen terhadap ulasan pengguna Aplikasi Deepseek di Google Play Store. Proses analisis sentimen ini diimplementasikan menggunakan bahasa pemrograman Python. Seluruh proses dari pengumpulan data, preprocessing kata, hingga ekstraksi fitur TF-IDF, proses klasifikasi *Naïve Bayes* serta divalidasi menggunakan teknik *5-Fold Cross Validation* dengan bantuan pustaka Python seperti *pandas*, *scikit-learn*, *nltk*, dan *matplotlib*.

### 4.1 Pemuatan dan Eksplorasi *Dataset*

*Dataset* yang digunakan diambil dari platform Kaggle dan dimuat ke dalam Python menggunakan pustaka *pandas*. Data ini terdiri dari ulasan pengguna, skor, dan waktu unggahan ulasan

```
df = pd.read_csv("Deepseek Reviews Analysis.csv")
print("Kolom dataset:", df.columns)
print("Contoh data:")
print(df.head())
```

Gambar 2. Algoritma pemuatan *dataset*

Data memiliki 5 kolom utama: *content*, *score*, *thumbsUpCount*, *at*, dan *Unnamed: 0*. Kolom *content* berisi ulasan yang akan menjadi objek utama analisis. Berikut contoh data yang dibaca dan dibuat.

```
Kolom dataset: Index(['Unnamed: 0', 'content', 'score',
 'thumbsUpCount', 'at'], dtype='object')
Contoh data:
Unnamed: 0                                     content
score thumbsUpCount                               at
0          0  It's Great app, but an organization feature wo...
5          54  2025-01-29 16:05:38
```

```

1          1 DeepSeek is an absolute game-changer! This app..
5          38 2025-01-28 19:34:25
2          2 I hope this message finds you well. I'm a big ...
5          45 2025-02-01 10:55:53
3          3 Subject: Issue with App Performance - Server B...
1          365 2025-02-01 17:58:11
4          4 DeepSeek is a game-changer! Its AI-powered sea...
5          10 2025-02-01 17:01:23

```

Gambar 3. Contoh data yang dibaca dan dimuat

#### 4.2 Pelabelan Sentimen Berdasarkan Skor

Langkah selanjutnya adalah memberikan label sentimen berdasarkan skor. Berikut adalah pendekatan kategorisasi:

- Skor 1–2 dikategorikan sebagai sentimen negatif.
- Skor 3 dikategorikan sebagai sentimen netral.
- Skor 4–5 dikategorikan sebagai sentimen positif.

```

def label_sentiment (score): if score <= 2: return 'Negatif' elif score ==
3: return 'Netral' else: return 'Positif'
df["sentimen"] = df["score"].apply(label_sentiment)
print(df['sentimen'].value_counts())

```

Gambar 4. Algoritma pelabelan sentimen

```

          Positif      10851
          Negatif      3651
          Netral        622
Name: count, dtype: int64

```

Gambar 5. Output pelabelan sentimen

Dari pengukuran tersebut pendistribusian sentimen yaitu positif sebanyak 10.851 ulasan, negatif sebanyak 3.651 ulasan, dan netral sebanyak 622 ulasan. Dari jumlah tersebut, dapat dikatakan bahwa mayoritas pengguna memberikan ulasan positif.

#### 4.3 Preprocessing Teks

Data sebelum dilakukan klasifikasi dibersihkan terlebih dahulu sehingga dapat dimodelkan lebih akurat oleh algoritma pembelajaran mesin. Beberapa *preprocessing* yang dilakukan diantaranya:

- Mengubah seluruh teks menjadi huruf kecil (*lowercasing*).
- Menghapus karakter non-alfabet dan simbol-simbol.
- Menghapus kata-kata umum yang tidak mengandung informasi sentimen (*stopwords*).
- Tokenizing* dan *stemming* menggunakan pustaka *nlTK*.

```

stop_words = set(stopwords.words('english'))
stemmer = PorterStemmer()
def clean_text(text):
    text = re.sub(r'^[a-zA-Z]', ' ', str(text))
    text = text.lower()
    tokens = text.split()
    tokens = [stemmer.stem(word) for word in tokens if word not in
stop_words]
    return ' '.join(tokens)
df['clean_content'] = df['content'].apply(clean_text)

```

Gambar 6. Algoritma *preprocessing*

=== Contoh Hasil Preprocessing ===

Teks asli:

It's Great app, but an organization feature would be amazing! I absolutely love using DeepSeek, but I think one feature that would make the app even better is the ability to organize chat history into folders or sections. Currently, it's hard to manage multiple conversations, especially for users who have a lot of ongoing discussions on different topics. Being able to categorize, label, or move chats into specific folders would be a huge time-saver and make the app much more user-friendly! Best!

Teks bersih:

great app organ featur would amaz absolut love use deepseek think one featur would make app even better abil organ chat histori folder section current hard manag multipl convers espec user lot ongo discuss differ topic abl categor label move chat specif folder would huge time saver make app much user friendli best

Gambar 7. Contoh *output preprocessing*

Hasil dari langkah ini adalah *noise* pada model yang dapat memengaruhi akurasi hasil secara keseluruhan telah dibersihkan sehingga dapat dilakukan ekstraksi menggunakan TF-IDF.

#### 4.4. Ekstraksi Fitur menggunakan TF-IDF

Setelah proses *preprocessing* selesai dilakukan, selanjutnya data teks tersebut diubah menjadi representasi numerik dengan metode token dan TF-IDF. Metode ini digunakan untuk menghitung kepentingan kata dalam suatu dokumen dibandingkan dengan semua dokumen yang ada.

```
X = df['clean_content']
y = df['sentimen']
tfidf = TfidfVectorizer(max_features=3000)
X_tfidf = tfidf.fit_transform(X)
X_train, X_test, y_train, y_test = train_test_split(X_tfidf, y,
test_size=0.2, random_state=42)
```

Gambar 8. Algoritma ekstraksi fitur menggunakan TF-IDF dan *split data*

=== Contoh Fitur TF-IDF ===

```
0 1 2 3 4
folder 0.340711 0.000000 0.000000 0.000000 0.000000
would 0.323215 0.000000 0.105029 0.128542 0.000000
organ 0.310333 0.166801 0.000000 0.000000 0.000000
user 0.202984 0.109102 0.197880 0.121089 0.148530
featur 0.192757 0.103605 0.000000 0.114989 0.000000
```

Gambar 9. Contoh *output* ekstraksi fitur menggunakan TF-IDF

TF-IDF digunakan untuk mengubah teks menjadi vektor numerik. Data dibagi 80:20 untuk *training* dan *testing*. Karena itulah TF-IDF sering digunakan dalam proses klasifikasi untuk meningkatkan akurasi model, karena data yang digunakan sebagai fitur menjadi lebih relevan dan informatif.

#### 4.5. Klasifikasi Sentimen dengan *Naïve Bayes*

Model yang digunakan pada penelitian ini yaitu *Multinomial Naïve Bayes* karena model ini cocok saat *dataset* teks telah direpresentasikan sebagai matriks yang berisi frekuensi kata atau T frekuensi.

```
model = MultinomialNB()
model.fit(X_train, y_train)
```

Gambar 10. Algoritma klasifikasi sentimen

Tabel 1. Hasil evaluasi sebelum validasi

	<i>Precision</i>	<i>Recall</i>	<i>F1-score</i>	<i>Support</i>
Negatif	0.81	0.75	0.78	756
Netral	0.00	0.00	0.00	128
Positif	0.89	0.97	0.93	2141
<i>Accuracy</i>			0.87	3025
<i>Macro AVG</i>	0.57	0.57	0.57	3025
<i>Weighted AVG</i>	0.84	0.87	0.85	3025

Model klasifikasi sentimen yang menggunakan algoritma *Naïve Bayes* berhasil meraih akurasi sebesar 87,4%. Evaluasi dilakukan dengan metrik *precision*, *recall*, dan *F1-score* untuk menilai kemampuan model dalam mengenali tiga jenis sentimen: negatif, netral, dan positif. Setelah tahap tersebut, dilakukan model klasifikasi yang melatih data larik teks sebelumnya dan menggunakan algoritma *Multinomial Naïve Bayes*. Model ini kondusif untuk data teks, terutama data himpunan fitur TF-IDF yang dipisahkan. Ketika model dijalankan pada metode *fit()* data latih, itu melatihnya untuk mengenali pola data fitur label sentimen. Meskipun operasi ini tidak mengeluarkan *output*, ini menjadi model prediksi karena dapat memprediksi data dari uji *dataset*.

#### 4.6. Validasi Model (5-Fold Cross Validation)

Untuk menguji seberapa stabil dan umum model yang dibuat, digunakan teknik validasi *5-Fold Cross Validation*. Dengan metode ini, data dibagi menjadi lima bagian, dan setiap bagian bergiliran menjadi data uji, sedangkan sisanya menjadi data latih.

```
print("\n=== EVALUASI DENGAN CROSS VALIDATION 5-FOLD ===")
skf = StratifiedKFold(n_splits=5, shuffle=True, random_state=42)

precision_list = []
recall_list = []
f1_list = []
accuracy_list = []

for train_idx, test_idx in skf.split(X_tfidf, y):
    X_train_cv, X_test_cv = X_tfidf[train_idx], X_tfidf[test_idx]
    y_train_cv, y_test_cv = y.iloc[train_idx], y.iloc[test_idx]

    model_cv = MultinomialNB()
    model_cv.fit(X_train_cv, y_train_cv)
    y_pred_cv = model_cv.predict(X_test_cv)

    precision, recall, f1, _ =
precision_recall_fscore_support(y_test_cv, y_pred_cv, average=None,
labels=model_cv.classes_)
    accuracy = accuracy_score(y_test_cv, y_pred_cv)

    precision_list.append(precision)
    recall_list.append(recall)
    f1_list.append(f1)
    accuracy_list.append(accuracy)

# Hitung rata-rata per label
avg_precision = np.mean(precision_list, axis=0)
avg_recall = np.mean(recall_list, axis=0)
```



Penggunaan kalimat seperti "best ai" mengandung indikasi bahwa konsumen punya ekspektasi yang tinggi terhadap kegunaan AI dan menyatakan aplikasi ini memiliki keunggulan dari aplikasi jenis yang serupa. Totalitas stereotip positif tersebut mencerminkan kesuksesan Deepseek untuk menyampaikan pengalaman konsumen terbaik, dan menandakan adanya potensi untuk meningkatkan loyalitas konsumen dan meningkatkan adopsi aplikasi di masa mendatang.



Gambar 9. Wordcloud negatif

Berbeda halnya dengan visualisasi wordcloud pada kategori sentimen negatif, yang menampilkan munculnya kata "work", "slow", "busy", dan "answer". Istilah-istilah tersebut memberikan kesan bahwa beberapa pengguna mengalami kekurangan ketika mereka menggunakan aplikasi, terutama di aspek teknis seperti kecepatan akses dan kecepatan respons sistem. Kata "slow" dan "busy" memberi isyarat adanya hambatan dari aspek performa server atau ketersediaan kapasitas layanan. Sedangkan, "answer" memberikan isyarat kekecewaan terhadap hasil yang ditawarkan oleh sistem, mungkin tidak sejalan dengan konteks atau harapannya sendiri. Review negatif tersebut memberikan isyarat adanya celah yang harus diratapi dan dibuktikan dengan peningkatan perform.

Berdasarkan hasil analisis, pengembang aplikasi Deepseek disarankan untuk meningkatkan performa server guna mengurangi keluhan terkait lambat dan padatnya aplikasi, memperbaiki kualitas respons AI melalui penyempurnaan model, serta menambahkan fitur interaktif seperti penilaian jawaban dan pengorganisasian riwayat percakapan. Selain itu, penerapan metode klasifikasi yang lebih canggih seperti *ensemble* atau *deep learning* juga penting untuk meningkatkan akurasi, terutama dalam mengidentifikasi sentimen netral yang saat ini masih kurang akurat

#### 4.8. Pembahasan Performa Metode

Hasil validasi menggunakan metode *5-Fold Cross Validation* diperoleh bahwa model *Naïve Bayes* dalam penelitian ini mencapai rata-rata akurasi sebesar 86,94%, termasuk *F1-score* tertinggi di sentimen positif (92,7%), diikuti sentimen negatif (76,3%), sementara performa klasifikasi sentimen netral tetap lemah (0,6%). Capaian ini menunjukkan bahwa walaupun model sangat berprestasi di dua kelas dominan, masih terdapat ruang untuk peningkatan dalam klasifikasi sentimen netral.

Temuan ini menunjukkan performa yang komparatif dan bahkan lebih tinggi dibandingkan beberapa penelitian terdahulu yang juga menggunakan pendekatan *Naïve Bayes*. Misalnya, Hasanah dan Sari dalam analisis ulasan pengguna aplikasi Maxim melaporkan akurasi sebesar 81,9% dengan dua kelas sentimen [2]. Sementara Nurian et al. yang menganalisis aplikasi Shopee menggunakan metode serupa mencatatkan akurasi sebesar 83,2% [1]. Di sisi lain, Sagala dan Samuel menunjukkan bahwa *Naïve Bayes* tetap bersaing dibandingkan *Random Forest* dan SVM dalam mengklasifikasikan ulasan terhadap ChatGPT, meskipun akurasinya berada pada kisaran 85% [7].

Performa model di sini juga serempak dengan Kevin et al., yang melaporkan *F1-score* sebesar 89% dalam klasifikasi dua kelas (negatif dan positif) terhadap ulasan aplikasi Kinemaster menggunakan *Naïve Bayes* [8]. Kontribusi yang diberi oleh penelitian ini adalah pada sisi penggunaan tiga kategori sentimen, yaitu positif, netral, dan negatif, yang kurang diperhatikan sebelumnya, serta adanya TF-IDF untuk memprioritaskan fitur teks sebelum diklasifikasikan. Hal ini menjadikan metode yang didasari lebih adaptif terhadap kompleksitas data nyata.

Secara keseluruhan, pendekatan gabungan *Naïve Bayes* dan TF-IDF terbukti mampu memberikan hasil yang kuat dalam teks ulasan pengguna aplikasi, sebagaimana juga ditunjukkan oleh Cherradi dan El Haddadi yang menempatkan *Naïve Bayes* sebagai salah satu algoritma yang efisien meskipun tidak selalu merupakan yang terbaik dalam perbandingan metode deep learning. Namun, termasuk dalam sumber daya yang lebih rendah dan kekompleksan, performa model ini tetap kompetitif dan layak dijadikan acuan dalam implementasi praktis [15].

#### 4. Simpulan

Penelitian ini berhasil menerapkan algoritma *Naïve Bayes* yang dikombinasikan dengan metode pembobotan Term Frequency–Inverse Document Frequency (TF-IDF) untuk melakukan analisis sentimen terhadap ulasan pengguna aplikasi Deepseek di Google Play Store. Dengan menggunakan dataset yang berisi 15.124 ulasan dan melalui tahapan preprocessing serta ekstraksi fitur, model ini menunjukkan performa klasifikasi yang cukup baik, dengan akurasi rata-rata sebesar 86,94% berdasarkan evaluasi *5-Fold Cross Validation*. Hasil terbaik diperoleh pada kategori sentimen positif, sementara sentimen netral masih menunjukkan kelemahan, yang mengindikasikan perlunya pendekatan lanjutan untuk meningkatkan representasi fitur pada kelas tersebut.

Dari sisi kontribusi ilmiah, penelitian ini memperkuat temuan sebelumnya terkait efektivitas algoritma *Naïve Bayes* dalam klasifikasi teks ulasan pengguna aplikasi, sekaligus memberikan nilai tambah melalui penggunaan tiga kategori sentimen yang lebih kompleks. Integrasi dengan TF-IDF terbukti mampu meningkatkan relevansi fitur serta efisiensi komputasi, menjadikan pendekatan ini layak diterapkan dalam konteks big data tekstual, khususnya di ranah ulasan pengguna aplikasi berbasis AI. Temuan ini juga konsisten dengan hasil-hasil studi sebelumnya yang menunjukkan bahwa metode ini tetap kompetitif jika dibandingkan dengan algoritma lain seperti SVM dan *Random Forest*, terutama dalam menghadapi data yang tidak seimbang.

Berdasarkan hasil analisis, pengembang aplikasi Deepseek disarankan untuk meningkatkan performa server guna mengurangi keluhan terkait lambat dan padatnya aplikasi, memperbaiki kualitas respons AI melalui penyempurnaan model, serta menambahkan fitur interaktif seperti penilaian jawaban dan pengorganisasian riwayat percakapan.

#### Daftar Referensi

- [1] A. Nurian, M. S. Ma'arif, I. N. Amalia, and C. Rozikin, "Analisis Sentimen Pengguna Aplikasi Shopee Pada Situs Google Play Menggunakan Naive Bayes Classifier," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 1, pp. 97–105, Jan. 2024, doi: 10.23960/jitet.v12i1.3631.
- [2] A. N. Hasanah and B. N. Sari, "Analisis Sentimen Ulasan Pengguna Aplikasi Jasa Ojek Online Maxim Pada Google Play Dengan Metode Naïve Bayes Classifier," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 1, pp. 90–96, Jan. 2024, doi: 10.23960/jitet.v12i1.3628.
- [3] C. Padurariu and M. E. Breaban, "Dealing with Data Imbalance in Text Classification," *Procedia Comput. Sci.*, vol. 159, pp. 736–745, 2019, doi: <https://doi.org/10.1016/j.procs.2019.09.229>.
- [4] D. P. Santoso and W. Wibowo, "Analisis Sentimen Ulasan Aplikasi Buzzbreak Menggunakan Metode Naïve Bayes Classifier pada Situs Google Play Store," *J. Sains dan Seni ITS*, vol. 11, no. 2, pp. D190–D196, 2022, doi: 10.12962/j23373520.v11i2.72534.
- [5] P. H. C. Samanmali and R. A. H. M. Rupasingha, "Sentiment analysis on google play store app users' reviews based on deep learning approach," *Multimed. Tools Appl.*, vol. 83, no. 36, pp. 84425–84453, 2024, doi: 10.1007/s11042-024-19185-w.
- [6] S. Ahammad, S. A. Sinthia, M. Ahmed, M. Hossain, N.-A.-A. Asif, and N. A. Ikram, "Deep Learning-based Sentiment Analysis of User Generated Reviews of Various AI Powered Mobile Applications," in *2024 International Conference on Inventive Computation Technologies (ICICT)*, 2024, pp. 505–512. doi: 10.1109/ICICT60155.2024.10544699.
- [7] G. Jeffson Sagala and Y. T. Samuel, "Sentiment Analysis on ChatGPT App Reviews on Google Play Store Using Random Forest Algorithm, Support Vector Machine and Naïve Bayes," *Int. J. Eng. Bus. Soc. Sci.*, vol. 2, no. 04, pp. 1194–1204, Mar. 2024, doi: 10.58451/ijebss.v2i04.148.
- [8] K. Kevin, M. Enjeli, and A. Wijaya, "Analisis Sentimen Penggunaan Aplikasi Kinemaster Menggunakan Metode Naive Bayes," *J. Ilm. Comput. Sci.*, vol. 2, no. 2, pp. 89–98, Jan. 2024, doi: 10.58602/jics.v2i2.24.

- [9] M. Raffi, A. Suharso, and I. Maulana, "Analisis Sentimen Ulasan Aplikasi Binar Pada Google Play Store Menggunakan Algoritma Naïve Bayes Sentiment Analysis of Binar Application Reviews on Google Play Store Using Naïve Bayes Algorithm," *J. Inf. Technol. Comput. Sci.*, vol. 6, no. 1, pp. 1–7, 2023, [Online]. Available: <https://pdfs.semanticscholar.org/d43d/517a3e660d7316ca43ce738593e033f92a90.pdf>
- [10] L. Oleshchenko and O. Melnychuk, "Ensemble Classification Methods of Machine Learning for Analyzing News Texts for Falsity BT - Advances in Computer Science for Engineering and Education VII," Z. Hu, F. Yanovsky, I. Dychka, and M. He, Eds., Cham: Springer Nature Switzerland, 2025, pp. 349–361.
- [11] S. Verma, S. Agrawal, S. Gulati, and G. P., "A Comparative Study on Spam Identification using Naive Bayes and SVM," in *2024 5th International Conference on Data Intelligence and Cognitive Informatics (ICDICI)*, IEEE, Nov. 2024, pp. 914–919. doi: 10.1109/ICDICI62993.2024.10810925.
- [12] S. Gadri, S. Chabira, S. Ould Mehieddine, and K. Herizi, "Sentiment Analysis: Developing an Efficient Model Based on Machine Learning and Deep Learning Approaches," 2022, pp. 237–247. doi: 10.1007/978-3-030-93247-3\_24.
- [13] H. Chauhan, K. S. Rathore, and V. G. Rajan, "Sentiment Analysis using Supervised Machine Learning," in *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, IEEE, Apr. 2022, pp. 1276–1279. doi: 10.1109/ICACITE53722.2022.9823681.
- [14] N. L. Adam, N. H. Rosli, and S. C. Soh, "Sentiment Analysis on Movie Review using Naïve Bayes," in *2021 2nd International Conference on Artificial Intelligence and Data Sciences (AiDAS)*, IEEE, Sep. 2021, pp. 1–6. doi: 10.1109/AiDAS53897.2021.9574419.
- [15] M. Cherradi and A. El Haddadi, "Comparative Analysis of Machine Learning Algorithms for Sentiment Analysis in Film Reviews," *Acadlore Trans. AI Mach. Learn.*, vol. 3, no. 3, pp. 137–147, 2024, doi: 10.56578/ataiml030301.
- [16] M. Alkaff, A. Rizky Baskara, and Y. Hendro Wicaksono, "Sentiment Analysis of Indonesian Movie Trailer on YouTube Using Delta TF-IDF and SVM," in *2020 Fifth International Conference on Informatics and Computing (ICIC)*, IEEE, Nov. 2020, pp. 1–5. doi: 10.1109/ICIC50835.2020.9288579.
- [17] N. Boyko and K. Boksho, "Application of the naive Bayesian classifier in work on sentimental analysis of medical data," *CEUR Workshop Proc.*, vol. 2753, pp. 230–239, 2020.
- [18] M. Liang and T. Niu, "Research on Text Classification Techniques Based on Improved TF-IDF Algorithm and LSTM Inputs," *Procedia Comput. Sci.*, vol. 208, pp. 460–470, 2022, doi: 10.1016/j.procs.2022.10.064.
- [19] I. R. Dina, M. A. Barata, and P. E. Yuwita, "Penerapan Data Mining pada Algoritma Multiple Linear Regression dalam Peramalan Harga Emas," *SMARTICS J.*, vol. 11, no. 1 SE-Article, pp. 1–7, Apr. 2025, doi: 10.21067/smartics.v11i1.11710.