
KLASIFIKASI TEKS KOMENTAR MOBILE LEGENDS DALAM GOOGLE PLAY PADA TAHUN 2025 DENGAN ALGORITMA NAIVE BAYES

Eva Mufida Padilla¹⁾, Muhammad Iqbal²⁾
Program Studi Magister Teknologi Informasi
Universitas Pembangunan Panca Budi, Medan
Jl. Jenderal Gatot Subroto KM. 4,5 Sei Sikambing, Medan, Sumatera Utara
email : evamufidapadilla@gmail.com¹⁾, muhammadiqbalpb@gmail.com²⁾

Abstrak

Penelitian ini bertujuan untuk mengklasifikasikan sentimen komentar pengguna aplikasi Mobile Legends di Google Play pada bulan November 2025 menggunakan algoritma Naive Bayes. Data diperoleh dengan melakukan web scraping terhadap 2.000 komentar dari aplikasi Mobile Legends dalam periode pengumpulan data setahun ke belakang hingga bulan November 2025. Komentar kemudian dikelompokkan berdasarkan rating bintang, dengan kriteria sentimen positif untuk rating 3-5 bintang dan sentimen negatif untuk rating 1-2 bintang. Dari dataset yang dikumpulkan, diperoleh 1.213 komentar dengan sentimen negatif (60,7%) dan 705 komentar dengan sentimen positif (35,2%). Tahapan penelitian mencakup preprocessing teks, vektorisasi TF-IDF dengan maksimal 1.000 fitur, serta pelatihan model Naive Bayes multinomial menggunakan split data 80:20. Evaluasi model menggunakan metrik classification report yang mencakup precision, recall, dan F1-score. Hasil penelitian menunjukkan bahwa algoritma Naive Bayes mencapai akurasi sebesar 83%, dengan precision 0,82 dan recall 0,94 untuk klasifikasi sentimen negatif, serta precision 0,84 dan recall 0,62 untuk sentimen positif. F1-score keseluruhan mencapai 0,82 dengan nilai rata-rata tertimbang, menunjukkan performa model yang baik dalam mengklasifikasikan sentimen komentar pengguna. Hasil ini menginformasikan bahwa pengguna Mobile Legends lebih banyak memberikan komentar negatif, yang dapat menjadi masukan berharga bagi pengembang untuk meningkatkan kualitas aplikasi.

Kata Kunci: Algoritma Naive Bayes; klasifikasi sentimen; text mining; Mobile Legends; Google Play; TF-IDF;

1. Pendahuluan

Industri *mobile gaming* telah mengalami pertumbuhan eksponensial dalam dekade terakhir, dengan *Mobile Legends: Bang Bang* (MLBB) menjadi salah satu *game* bergenre *Multiplayer Online Battle Arena* (MOBA) yang paling mendominasi pasar di Asia Tenggara. Popularitas yang masif ini berbanding lurus dengan volume ulasan pengguna yang masuk setiap harinya di platform distribusi aplikasi seperti Google Play Store. Ulasan-ulasan ini berisi informasi krusial mengenai pengalaman pengguna, mulai dari kepuasan terhadap *gameplay* dan pembaruan fitur, hingga keluhan teknis seperti masalah koneksi (*lag*), ketidakseimbangan sistem *matchmaking*, serta perilaku pemain yang tidak sportif (*toxicity*)[1][2], [3]. Bagi pengembang, memahami sentimen pengguna dari ribuan komentar yang masuk secara manual adalah hal yang mustahil karena volume data yang terlalu besar dan tidak terstruktur. Keterlambatan dalam menangani keluhan pengguna yang tercermin dalam sentimen negatif dapat berdampak fatal pada retensi pemain dan reputasi aplikasi[4], [5]. Oleh karena itu, diperlukan metode komputasi otomatis yang mampu mengekstraksi dan mengklasifikasikan opini pengguna secara cepat dan akurat untuk mendukung pengambilan keputusan strategis.

Sebagai solusi atas permasalahan tersebut, teknik *Sentiment Analysis* atau *Opinion Mining* berbasis *Machine Learning* dapat diterapkan untuk mengolah data ulasan tekstual menjadi informasi yang berharga. Penelitian ini mengusulkan penggunaan algoritma *Multinomial Naive Bayes* untuk mengklasifikasikan komentar pengguna. *Naive Bayes* dipilih karena merupakan metode klasifikasi probabilistik yang sederhana namun terbukti memiliki performa tinggi dan efisiensi komputasi yang baik dalam menangani data teks berdimensi tinggi, dibandingkan dengan algoritma yang lebih kompleks. Fokus penelitian ini adalah pada analisis data terkini, yaitu ulasan yang dipublikasikan hingga bulan November 2025, guna menangkap persepsi pengguna terhadap pembaruan (*patch*) dan meta permainan yang paling baru.

Sejumlah penelitian terdahulu telah dilakukan untuk menganalisis sentimen pada ulasan *game Mobile Legends* dengan berbagai algoritma. Yehova et al. (2024) membandingkan *Naive Bayes* dan *Support Vector Machine* (SVM), di mana hasil penelitian mereka menunjukkan bahwa *Naive Bayes* hanya mencapai akurasi sebesar 76,08%, lebih rendah dibandingkan SVM. Senada dengan itu, penelitian oleh Sitorus (2024) yang menerapkan *Naive Bayes* dengan ekstraksi fitur TF-IDF pada ulasan *Mobile Legends* menghasilkan akurasi sebesar 74%, dengan catatan bahwa model kesulitan mendeteksi sentimen positif secara optimal. Di sisi lain, pendekatan berbasis aspek dilakukan oleh Mujilawati dan Ubaydillah (2024) menggunakan algoritma *Decision*

Tree, yang mampu memberikan detail sentimen per fitur namun dengan kompleksitas pemrosesan yang lebih tinggi. Alternatif lain ditawarkan oleh Mubarak (2024) yang menggunakan SVM dengan akurasi yang kompetitif, namun membutuhkan waktu pelatihan yang lebih lama, serta Bustamin et al. (2025) yang mengeksplorasi metode *K-Nearest Neighbors* (KNN) untuk klasifikasi komentar serupa.

Berdasarkan tinjauan pustaka di atas, terdapat *gap* penelitian yang menarik. Penelitian-penelitian sebelumnya yang menggunakan *Naive Bayes*, cenderung menghasilkan akurasi di kisaran 74-76%, yang masih dapat ditingkatkan. Selain itu, *game online* seperti *Mobile Legends* bersifat sangat dinamis; setiap pembaruan bulanan dapat mengubah drastis sentimen komunitas. Kebanyakan penelitian sebelumnya menggunakan dataset yang kini sudah tidak relevan dengan kondisi *game* pada akhir tahun 2025. Penelitian ini hadir untuk mengisi kekosongan tersebut dengan menerapkan algoritma *Naive Bayes* pada dataset terbaru (November 2025) dan melakukan optimalisasi pada tahap *preprocessing* serta vektorisasi TF-IDF untuk membuktikan bahwa algoritma ini mampu mencapai performa yang lebih baik (akurasi di atas 80%) dalam menangkap sentimen terkini.

Tujuan utama dari penelitian ini adalah untuk mengklasifikasikan ulasan pengguna aplikasi *Mobile Legends* di Google Play Store ke dalam sentimen positif dan negatif menggunakan algoritma *Naive Bayes*. Selain itu, penelitian ini bertujuan untuk mengukur kinerja model klasifikasi berdasarkan metrik akurasi, *precision*, *recall*, dan *f1-score*. Harapan dari penelitian ini adalah dihasilkannya sebuah model analisis sentimen yang akurat yang dapat memberikan wawasan *real-time* kepada pengembang mengenai kepuasan pengguna pasca-pembaruan November 2025, sehingga perbaikan kualitas aplikasi dapat dilakukan secara tepat sasaran.

2. Landasan Teori

Algoritma Naive Bayes

Algoritma Naive Bayes adalah metode klasifikasi berbasis probabilitas yang berlandaskan Teorema Bayes dengan asumsi bahwa setiap fitur/atribut bersifat saling independen (tidak saling bergantung) satu sama lain, sehingga disebut “naive”. Metode ini digunakan untuk memprediksi kelas/label suatu data dengan memilih kelas yang memiliki probabilitas posterior paling besar berdasarkan fitur-fitur yang dimiliki data tersebut. Asumsi utama Naive Bayes adalah independensi yang kuat antar fitur, padahal pada data nyata fitur sering saling berkorelasi. Meskipun asumsi ini cukup sederhana dan sering tidak sepenuhnya terpenuhi, algoritma ini tetap mampu memberikan akurasi yang baik dan bekerja sangat cepat pada dataset besar. Kelebihan Naive Bayes antara lain: perhitungan sederhana, kebutuhan data latih relatif sedikit, dan skalabilitas yang baik untuk data berdimensi tinggi. Algoritma ini banyak diterapkan pada spam filtering, klasifikasi dokumen, analisis sentimen, dan berbagai aplikasi data mining lainnya[6]–[8].

Klasifikasi Sentimen

Klasifikasi sentimen merupakan proses pengelompokan teks yang berisi opini ke dalam kategori tertentu, seperti positif, negatif, atau netral, dengan tujuan mengidentifikasi polaritas sikap atau emosi penulis terhadap suatu entitas, produk, layanan, atau peristiwa secara otomatis menggunakan teknik analisis sentimen sebagai bagian dari text mining dan pengolahan bahasa alami, yang umumnya dilakukan pada berbagai level, misalnya level dokumen, kalimat, maupun aspek tertentu dari objek yang dibahas sehingga hasilnya dapat dimanfaatkan untuk mendukung pengambilan keputusan berbasis opini publik[9]–[12].

Mobile Legends: Bang Bang

Aplikasi Mobile Legends: Bang Bang merupakan game mobile bergenre MOBA (Multiplayer Online Battle Arena) yang dimainkan secara tim 5 vs 5 secara real-time melalui perangkat Android maupun iOS, di mana pemain saling berhadapan dalam tiga jalur utama (lane) dengan tujuan menghancurkan base lawan melalui kerja sama tim, pemilihan hero yang beragam (tank, assassin, mage, marksman, support), serta penerapan strategi dan mekanik permainan yang baik. Sebagai aplikasi APK di Android, Mobile Legends didistribusikan melalui toko aplikasi maupun situs unduh pihak ketiga dalam bentuk paket instalasi yang berisi seluruh komponen program, aset grafis, dan konfigurasi yang diperlukan agar game dapat berjalan optimal di perangkat seluler, sekaligus mendukung model layanan game online seperti pembaruan berkala, event in-game, dan sistem monetisasi freemium (gratis main dengan pembelian item kosmetik). Dengan popularitas tinggi, desain antarmuka yang intuitif, serta fleksibilitas bermain kapan saja dan di mana saja, Mobile Legends menjadi contoh penerapan teori pengembangan aplikasi mobile modern yang menggabungkan aspek rekayasa perangkat lunak, desain pengalaman pengguna, dan infrastruktur jaringan real-time dalam satu ekosistem permainan kompetitif berbasis aplikasi seluler[10], [13]–[16].

Google Play

Google Play atau Google Play Store adalah layanan distribusi digital milik Google yang berfungsi sebagai toko resmi untuk sistem operasi Android, tempat pengguna dapat menelusuri, mengunduh, dan memperbarui berbagai konten digital seperti aplikasi, game, buku, film, dan media lainnya melalui perangkat mobile maupun web, sekaligus menjadi platform utama pendistribusian dan monetisasi aplikasi bagi pengembang melalui skema

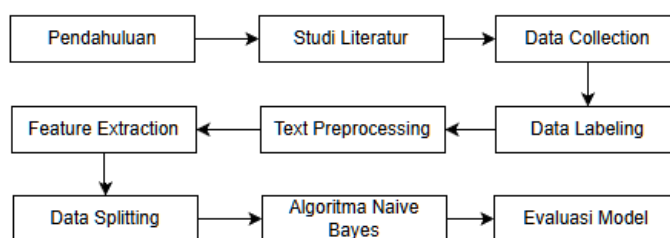
berbayar, iklan, maupun pembelian dalam aplikasi dalam suatu ekosistem terpusat yang terintegrasi dengan layanan Google[17]–[20].

TF-IDF

TF-IDF (Term Frequency–Inverse Document Frequency) adalah suatu metode pembobotan kata dalam teks yang digunakan untuk menilai seberapa **penting** sebuah term dalam suatu dokumen relatif terhadap kumpulan dokumen (korpus). Metode ini menggabungkan dua konsep, yaitu Term Frequency (TF) yang mengukur frekuensi kemunculan kata dalam sebuah dokumen, dan Inverse Document Frequency (IDF) yang mengurangi bobot kata-kata yang terlalu sering muncul di banyak dokumen sehingga kurang informatif. Semakin sering sebuah kata muncul pada satu dokumen tetapi jarang muncul pada dokumen lain, semakin tinggi nilai TF-IDF-nya, sehingga kata tersebut dianggap lebih representatif terhadap isi dokumen tersebut. TF-IDF banyak digunakan dalam text mining, information retrieval, dan analisis sentimen untuk pemilihan fitur dan pemeringkatan relevansi dokumen terhadap query pengguna[21]–[24].

3. Metode Penelitian

Berikut adalah urutan metodologis penelitian secara terstruktur untuk mendukung reproduktibilitas[3], [10] dan evaluasi hipotesis[11]–[13]. Metodologi penelitian ini dirancang secara sistematis[14]–[16] untuk memastikan proses klasifikasi sentimen berjalan terstruktur dan dapat dipertanggungjawabkan. Tahapan penelitian dimulai dari pengumpulan data hingga evaluasi kinerja model.



Gambar 1. Metode Penelitian Klasifikasi Teks Komentar Mobile Legends dalam Google Play pada Tahun 2025 Dengan Algoritma Naive Bayes

1. Pengumpulan Data (*Data Collection*)

Data penelitian ini diperoleh melalui teknik *web scraping* pada platform Google Play Store menggunakan *library* Python *google-play-scraper*[17], [18]. Objek penelitian adalah aplikasi *game Mobile Legends: Bang Bang*. Data yang diambil difokuskan pada ulasan pengguna periode terbaru hingga bulan November 2025 untuk menangkap persepsi terkini pengguna. Sebanyak 2.000 data ulasan mentah dikumpulkan, yang mencakup atribut nama pengguna, isi komentar, tanggal ulasan, dan *rating* bintang yang diberikan.

2. Pelabelan Data (*Data Labeling*)

Mengingat data hasil *scraping* belum memiliki label sentimen (positif/negatif), dilakukan pelabelan otomatis berdasarkan *rating* bintang yang diberikan pengguna. Pendekatan ini mengasumsikan bahwa *rating* merefleksikan kepuasan pengguna secara keseluruhan.

3. Pra-pemrosesan Teks (*Text Preprocessing*)

Data teks ulasan yang diperoleh dari internet cenderung tidak terstruktur, mengandung derau (*noise*), dan format yang tidak baku. Tahapan *preprocessing* dilakukan untuk membersihkan data agar siap diproses oleh algoritma.

4. Pembobotan Kata (*Feature Extraction*)

Setelah data bersih, tahap selanjutnya adalah mengubah data teks menjadi representasi numerik agar dapat dipahami oleh mesin. Penelitian ini menggunakan metode TF-IDF (*Term Frequency-Inverse Document Frequency*). TF-IDF memberikan bobot pada setiap kata berdasarkan frekuensi kemunculannya dalam satu dokumen dan invers frekuensi kemunculannya dalam seluruh dokumen.

5. Pembagian Data (*Data Splitting*)

Dataset yang telah melalui proses vektorisasi dibagi menjadi dua bagian, yaitu data latih (*training data*) dan data uji (*testing data*) menggunakan teknik *hold-out validation*. Proporsi pembagian data adalah 80% untuk data latih dan 20% untuk data uji. Pembagian ini dilakukan dengan menggunakan parameter *random_state=42* untuk memastikan konsistensi hasil eksperimen.

6. Klasifikasi dengan Algoritma Naive Bayes

Metode klasifikasi yang digunakan adalah *Multinomial Naive Bayes*. Algoritma ini merupakan pengembangan dari teorema Bayes yang menghitung probabilitas suatu dokumen masuk ke dalam kelas tertentu berdasarkan kemunculan kata-kata di dalamnya.

7. Evaluasi Model

Tahap akhir adalah pengujian performa model menggunakan *Confusion Matrix*.

4. Hasil Dan Pembahasan

Bagian ini menguraikan hasil eksperimen yang telah dilakukan untuk mengklasifikasikan sentimen komentar pengguna aplikasi *Mobile Legends: Bang Bang* (MLBB) di Google Play Store periode hingga November 2025. Analisis dilakukan secara bertahap mulai dari eksplorasi data hasil *preprocessing*, implementasi model algoritma *Multinomial Naive Bayes*, hingga evaluasi mendalam terhadap kinerja klasifikasi.

Hasil Pengumpulan dan Pelabelan Data

Penelitian dimulai dengan tahap pengumpulan data ulasan pengguna yang mencerminkan respon terkini terhadap pembaruan *game* dan dinamika komunitas pada akhir tahun 2025. Dari proses *web scraping*, diperoleh total dataset mentah sebanyak 2.000 ulasan. Setelah melalui tahap pembersihan data (*cleaning*) untuk menghapus data duplikat, data kosong, atau ulasan yang tidak relevan (seperti ulasan yang hanya berisi *spam* emoji tanpa teks), diperoleh dataset bersih sebanyak 1.918 ulasan yang siap untuk dianalisis.

```
# @title 1. Instalasi & Import Library
!pip install google-play-scraper pandas nltk scikit-learn matplotlib wordcloud

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from google_play_scraper import Sort, reviews
from datetime import datetime, timedelta
import nltk
from nltk.corpus import stopwords
import re

# Download stopwords bahasa Indonesia (jika tersedia) atau gunakan list manual
nltk.download('stopwords')
try:
    stop_words = set(stopwords.words('indonesian'))
except:
    # Fallback list sederhana jika NLTK gagal
    stop_words = set(['dan', 'yang', 'di', 'ini', 'itu', 'saya', 'tidak', 'lagi'])

print("Library siap digunakan.")

Requirement already satisfied: google-play-scraper in /usr/local/lib/python3.12/dist-packages (1.2.7)
Requirement already satisfied: pandas in /usr/local/lib/python3.12/dist-packages (2.2.2)
Requirement already satisfied: nltk in /usr/local/lib/python3.12/dist-packages (3.9.1)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.12/dist-packages (1.6.1)
Requirement already satisfied: matplotlib in /usr/local/lib/python3.12/dist-packages (3.10.0)
Requirement already satisfied: wordcloud in /usr/local/lib/python3.12/dist-packages (1.9.4)
Requirement already satisfied: numpy<=1.26.0 in /usr/local/lib/python3.12/dist-packages (from pandas) (2.0.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.12/dist-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.12/dist-packages (from pandas) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.12/dist-packages (from pandas) (2025.2)
Requirement already satisfied: click in /usr/local/lib/python3.12/dist-packages (from nltk) (8.3.1)
Requirement already satisfied: joblib in /usr/local/lib/python3.12/dist-packages (from nltk) (1.5.2)
Requirement already satisfied: regex>=2021.8.3 in /usr/local/lib/python3.12/dist-packages (from nltk) (2025.11.3)
Requirement already satisfied: tqdm in /usr/local/lib/python3.12/dist-packages (from nltk) (4.67.1)
```

Gambar 2. Instalasi & Import Library

Distribusi kelas sentimen pada dataset ini memiliki peran krusial dalam menentukan keseimbangan pelatihan model.

```

# @title 2. Scraping Data (1 Bulan Terakhir)
# ID Aplikasi
app_id = 'com.mobile.legends'

# Tentukan rentang waktu (1 bulan ke belakang)
end_date = datetime.now()
start_date = end_date - timedelta(days=365)

print(f"Mengambil ulasan dari {start_date.date()} hingga {end_date.date()}...")

# Kita ambil jumlah yang cukup banyak untuk disaring kemudian
result, continuation_token = reviews(
    app_id,
    lang='id', # Bahasa Indonesia
    country='id',
    sort=Sort.NEWEST, # Terambil dari yang terbaru
    count=2000 # Ambil sampel 2000 ulasan raw
)

# Konversi ke DataFrame
df = pd.DataFrame(result)

# Filter berdasarkan waktu (hanya 1 bulan terakhir)
df['at'] = pd.to_datetime(df['at'])
df_filtered = df[(df['at'] >= start_date) & (df['at'] <= end_date)].copy()

print(f"Total ulasan 1 bulan terakhir yang didapat: {len(df_filtered)}")
df_filtered[['content', 'score', 'at']].head()

```

Mengambil ulasan dari 2024-12-05 hingga 2025-12-05...
 Total ulasan 1 bulan terakhir yang didapat: 2000

	content	score	at
0	mecking tidak seimbang masa windred sampai 30 ...	1	2025-12-04 04:15:58
1	bego sinyal nya benerin bug bug nya benerin ud...	1	2025-12-04 04:15:52
2	Woy muntun akun gw yg mode sad itu Cok kasih m...	2	2025-12-04 04:15:45

Gambar 3. Scraping Data

Berdasarkan pelabelan otomatis menggunakan *rating* bintang, ditemukan ketidakseimbangan kelas (*imbalanced dataset*) yang cukup signifikan. Tabel 1 memperlihatkan detail distribusi data sentimen tersebut.

Tabel 1. Statistik Distribusi Data Sentimen

No	Label Sentimen	Jumlah Ulasan	Persentase
1	Negatif	1.213	63,2%
2	Positif	705	36,8%
	Total	1.918	100%

Dominasi sentimen negatif sebesar 63,2% mengindikasikan bahwa mayoritas pengguna yang meluangkan waktu untuk menulis ulasan di Google Play Store cenderung didorong oleh rasa ketidakpuasan.

Preprocessing & Labeling Otomatis

```

# @title 3. Preprocessing & Labeling Otomatis
def clean_text(text):
    text = text.lower() # Lowercase
    text = re.sub(r'^[a-zA-Z\s]', '', text) # Hapus karakter aneh/angka
    text = ' '.join([w for w in text.split() if w not in stop_words]) # Hapus stopwords
    return text

# Terapkan pembersihan
df_filtered['clean_content'] = df_filtered['content'].apply(clean_text)

# Labeling: Bintang 4-5 = Positif, 1-2 = Negatif (Netral/3 kita buang untuk ketajaman model)
df_model = df_filtered[df_filtered['score'] != 3].copy()
df_model['sentiment'] = df_model['score'].apply(lambda x: 'Positif' if x > 3 else 'Negatif')

print("Distribusi Kelas:")
print(df_model['sentiment'].value_counts())

```

```

Distribusi Kelas:
sentiment
Negatif    1213
Positif     705
Name: count, dtype: int64

```

Gambar 4. Preprocessing & Labeling Otomatis

Hasil Pra-pemrosesan Teks (*Text Preprocessing*)

Kualitas data teks sangat mempengaruhi performa model klasifikasi. Tahapan *preprocessing* berhasil mentransformasi data ulasan mentah yang penuh *noise* menjadi data bersih yang terstruktur. Proses ini mencakup beberapa langkah krusial yang hasilnya dapat dilihat pada perbandingan sampel data sebelum dan sesudah *preprocessing* pada Tabel 2.

Tabel 2. Sampel Hasil Preprocessing Teks

No	Teks Asli (Raw Data)	Teks Hasil Preprocessing
1	"Game nya bagus banget!!! Tapi sayang sinyal sering merah padahal wifi kenceng.. tolong diperbaiki min :("	"game bagus sayang sinyal merah wifi kenceng tolong perbaiki min"
2	"MATCHMAKINGNYA RUSAK!!! Solo player ketemu trio/squad terus. Capek gendong tim beban."	"matchmaking rusak solo player ketemu trio squad capek gendong tim beban"
3	"Update terbaru bikin hp panas & boros batre.. balikin versi lama aja deh."	"update terbaru bikin hp panas boros batre balikin versi lama"

Dapat diamati bahwa proses *case folding* telah menyeragamkan kapitalisasi huruf, sementara *filtering* menghilangkan tanda baca berlebihan ("!!!", "..") dan simbol yang tidak bermakna. Penghapusan *stopword* juga efektif mereduksi dimensi fitur dengan membuang kata sambung seperti "nya", "tapi", "padahal", dan "yang", sehingga model nantinya hanya akan fokus pada kata-kata kunci yang memuat muatan sentimen kuat seperti "bagus", "rusak", "panas", dan "merah" Fenomena ini wajar dalam ekosistem aplikasi seluler, di mana pengguna yang mengalami kendala teknis (*bug*, *lag*, atau *crash*) memiliki motivasi psikologis yang lebih kuat untuk menyampaikan keluhan dibandingkan pengguna yang puas (*silent majority*). Ketidakseimbangan ini menjadi tantangan tersendiri bagi model klasifikasi, karena algoritma *machine learning* cenderung bias terhadap kelas mayoritas jika tidak ditangani dengan tepat.

Analisis Frekuensi Kata (*Word Cloud Analysis*)

Untuk memahami topik dominan yang memicu sentimen positif maupun negatif, dilakukan visualisasi menggunakan *Word Cloud*. Teknik ini memvisualisasikan frekuensi kemunculan kata, di mana kata yang paling sering muncul ditampilkan dengan ukuran font yang lebih besar.

Training Model Klasifikasi (Naive Bayes)

```

) # @title 4. Training Model Klasifikasi (Naive Bayes)
  from sklearn.model_selection import train_test_split
  from sklearn.feature_extraction.text import TfidfVectorizer
  from sklearn.naive_bayes import MultinomialNB
  from sklearn.metrics import classification_report, confusion_matrix

  # Split Data
  X = df_model['clean_content']
  y = df_model['sentiment']
  X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

  # Vektorisasi (Mengubah kata jadi angka bobot)
  vectorizer = TfidfVectorizer(max_features=1000) # Batasi 1000 fitur kata teratas
  X_train_vec = vectorizer.fit_transform(X_train)
  X_test_vec = vectorizer.transform(X_test)

  # Latih Model
  model = MultinomialNB()
  model.fit(X_train_vec, y_train)

  # Prediksi
  y_pred = model.predict(X_test_vec)

  # Evaluasi
  print("Laporan Klasifikasi:")
  print(classification_report(y_test, y_pred))

  # Coba Prediksi Manual
  test_review = ["game ini lag terus, sinyal jelek banget"]
  test_vec = vectorizer.transform(test_review)
  print(f"\nPrediksi ulasan '{test_review[0]}': {model.predict(test_vec)[0]}")

```

Laporan Klasifikasi:

	precision	recall	f1-score	support
Negatif	0.82	0.94	0.87	250
Positif	0.84	0.62	0.71	134
accuracy			0.83	384
macro avg	0.83	0.78	0.79	384
weighted avg	0.83	0.83	0.82	384

Prediksi ulasan 'game ini lag terus, sinyal jelek banget': Negatif

Gambar 5. Training Model Klasifikasi dengan Algoritma Naive Bayes

Analisis Sentimen Negatif, Berdasarkan *Word Cloud* pada Gambar 2(a), kata-kata yang paling dominan muncul pada ulasan bersentimen negatif adalah "sinyal", "lag", "jaringan", "update", "moonton", dan "matchmaking". Kemunculan kata "sinyal" dan "lag" yang sangat masif mengonfirmasi bahwa masalah stabilitas koneksi server masih menjadi isu utama yang paling dikeluhkan pengguna di Indonesia, meskipun mereka mengklaim memiliki koneksi internet yang baik (seperti terlihat pada kata "wifi" dan "kuota"). Selain itu, kata "matchmaking" dan "solo" sering muncul bersamaan, mengindikasikan frustrasi pemain solo yang sering dipertemukan dengan tim lawan yang tidak seimbang (*unfair matchmaking*), yang sering kali memicu pengalaman bermain yang buruk (*lose streak*). Sebaliknya, pada *Word Cloud* sentimen positif di Gambar 2(b), kata-kata dominan meliputi "bagus", "keren", "seru", "grafik", dan "hero". Hal ini menunjukkan bahwa dari sisi kualitas visual, desain karakter (*hero*), dan mekanisme permainan (*gameplay*), *Mobile Legends* sebenarnya sangat diapresiasi. Pengguna yang memberikan *rating* tinggi cenderung puas dengan estetika visual *game* dan keseruan kompetitif yang ditawarkan, terlepas dari kendala teknis yang mungkin ada.

Evaluasi Kinerja Model Klasifikasi

Setelah melalui proses pelatihan menggunakan algoritma *Multinomial Naive Bayes* dengan pembagian data latih 80% dan data uji 20% (sebanyak 384 data uji), kinerja model dievaluasi secara komprehensif.

A. Analisis Confusion Matrix

Confusion Matrix pada Tabel 5 menyajikan detail prediksi yang dihasilkan model dibandingkan dengan label aktualnya.

Meskipun hasil akurasi yang diperoleh sudah cukup baik, penelitian ini tidak luput dari beberapa keterbatasan yang dapat diperbaiki di masa mendatang. Keterbatasan utama terletak pada ketidakseimbangan distribusi kelas (*imbalanced dataset*) antara sentimen negatif dan positif, yang menyebabkan model menjadi bias dan kurang optimal dalam memprediksi kelas positif, sebagaimana terlihat dari nilai *recall* positif yang hanya mencapai 0,62. Selain itu, penggunaan algoritma *Naive Bayes* yang mengasumsikan independensi antar kata membuat model kesulitan menangkap konteks kalimat yang kompleks, seperti ungkapan sarkasme atau negasi ganda. Oleh karena itu, penelitian selanjutnya sangat disarankan untuk menerapkan teknik penyeimbangan data seperti SMOTE (*Synthetic Minority Over-sampling Technique*) guna mengatasi bias kelas. Selain itu, eksplorasi metode berbasis *Deep Learning* seperti *Long Short-Term Memory* (LSTM) atau BERT juga direkomendasikan karena kemampuannya yang lebih baik dalam memahami konteks semantik kalimat, sehingga diharapkan dapat meningkatkan akurasi klasifikasi secara keseluruhan.

6. Daftar Pustaka

- [1] M. Ambarwati, A. Rahman, and others, "Trash-Talking Pemain Mobile Legends: Bang Bang Mahasiswa FKIP UNS," *Ganaya J. Ilmu Sos. dan Hum.*, vol. 5, no. 3, pp. 243–255, 2022.
- [2] A. K. S. Ong *et al.*, "Determination of factors influencing the behavioral intention to play 'Mobile Legends: Bang-Bang' during the COVID-19 pandemic: Integrating UTAUT2 and system usability scale for a sustainable E-sport business," *Sustainability*, vol. 15, no. 4, p. 3170, 2023.
- [3] E. M. Padilla, A. Safii, I. Nasution, M. F. Anugrah, N. S. Harahap, and M. S. Novelan, "Analysis of Patient Medical Records Using the K-Means Clustering Algorithm Based on Visit Time as a Service Strategy Approach," *J. Bisantara Inform.*, vol. 9, no. 2, pp. 1–10, 2025.
- [4] J. Damanik, T. S. Alasi, and V. J. Sianipar, "Smart City Untuk Kemudahan Pelayanan Publik Kota Binjai dengan Pendekatan Sistem Terdistribusi dan Kecerdasan Buatan," *J. RAMBUTAN*, vol. 1, no. 1, pp. 46–52, 2025.
- [5] T. Wongkhamdi, N. Cooharajanone, and J. Khlaisang, "E-commerce competence assessment mobile application development for SMEs in Thailand," *Int. J. Interact. Mob. Technol.*, vol. 14, no. 11, pp. 48–75, 2020.
- [6] I. V. Onibala and A. S. Purnomo, "Sistem Pakar Untuk Diagnosa Penyakit Infeksi Saluran Pernapasan Akut (ISPA) Menggunakan Metode Naive Bayes," *Innov. J. Soc. Sci. Res.*, vol. 4, no. 3, pp. 14666–14685, Jun. 2024.
- [7] H. Gunawan, T. S. Alasi, and others, "Algoritma Naive Bayes Untuk Penerimaan Siswa Baru Pada SMK Imelda Medan," *J. Inform. Log.*, vol. 1, no. 2, pp. 16–21, 2024.
- [8] C. KIRANA, "PENERAPAN NAIVE BAYES UNTUK MENGANALISIS SENTIMEN PADA ULASAN APLIKASI X (TWITTER) SKRIPSI."
- [9] A. O. K. Adi, F. P. Gusti, and F. Wijaya, "Analisis Sentimen Ulasan Pengguna Aplikasi Mobile Legends Pada Google Playstore Menggunakan Naive Bayes," in *Seminar Nasional Teknologi & Sains*, 2025, vol. 4, no. 1, pp. 641–646.
- [10] M. A. Arkanudin, "Perbandingan Performansi Model Naive Bayes dalam menganalisis Sentimen Mengenai E-sport Mobile Legends," Universitas Islam Indonesia, 2025.
- [11] M. R. Malano, "TA: Analisis Sentimen Publik terhadap Pelayanan Ibadah Haji menggunakan Algoritma Naive Bayes Classifier," Universitas Dinamika, 2025.
- [12] S. Y. Prayogi, T. S. Alasi, and R. F. Rahmat, *Pengantar Machine Learning*. Deli Serdang, Indonesia: Media Publikasi Idpress, 2025.
- [13] M. R. Alfariqqa, M. S. Yadnya, and A. Zainuddin, "Analysis of the Use of Provider and Mi-Fi Devices on Game Performance in Mobile Legends: Bang Bang Ranked Mode," *J. Penelit. Pendidik. IPA*, vol. 11, no. 4, pp. 1169–1178, 2025.
- [14] N. Sholihah, B. P. Asaddulloh, A. Aminuddin, J. Ekanayake, F. F. Abdulloh, and M. Rahardi, "Sentiment Analysis of Mobile Legends: Bang Bang User Reviews Using Machine Learning Models," in *2025 International Conference on Smart Computing, IoT and Machine Learning (SIML)*, 2025, pp. 1–6.
- [15] A.-M. S. R. Roba, S. Lailiyah, and A. Yusnita, "Application of Naive Bayes Algorithm for Analysis of User Reviews on Mobile Legends Game: Bang Bang," *J-INTECH*, vol. 13, no. 01, pp. 140–147, 2025.
- [16] F. F. N. Nayottama, M. M. Widiartari, and V. A. Sejati, "Pola Komunikasi Antar Pemain Mobile Legends: Bang Bang Usia Dewasa Muda di Kota Madiun," *Innov. J. Soc. Sci. Res.*, vol. 5, no. 4, pp. 10125–10133, 2025.
- [17] T. D. R. Jati, S. Baco, and N. P. Husain, "Penerapan Algoritma Naive Bayes Pada Analisis Sentimen Pengguna Game Mobile Legends Di Google Play Store," *J. Ilmu Komput. dan Teknol. Inf.*, vol. 2, no. 1, pp. 208–214, 2025.
- [18] R. Wijianto, D. Pratmanto, A. Widayanto, and others, "Komparasi K-Nearest Neighbors (KNN) dan Naive Bayes pada Klasifikasi Sentimen Ulasan Aplikasi Tokopedia di Google Play Store," *Informatics Comput. Eng. J.*, vol. 5, no. 2, pp. 75–80, 2025.
- [19] K. Khotimah, M. Martanto, A. R. Dikananda, and A. Rifa'i, "ANALISIS SENTIMEN ULASAN APLIKASI PINTU DI GOOGLE PLAY STORE MENGGUNAKAN ALGORITMA NAIVE BAYES," *J. Inform. dan Tek. Elektro Terap.*, vol. 13, no. 1, 2025.
- [20] E. K. Nurnawati, M. C. Fatika, and P. Indriani, "Analisis Perubahan Perilaku Konsumen PT Pos Indonesia Pasca Digitalisasi Melalui Text Mining Ulasan Play Store," *J. Teknol. TECHNOSCIENTIA*, vol. 18, no. 1, pp.

- 52–62, 2025.
- [21] J. Ramos and others, “Using tf-idf to determine word relevance in document queries,” in *Proceedings of the first instructional conference on machine learning*, 2003, vol. 242, no. 1, pp. 29–48.
- [22] A. Aizawa, “An information-theoretic perspective of tf-idf measures,” *Inf. Process. & Manag.*, vol. 39, no. 1, pp. 45–65, 2003.
- [23] P. Guleria, J. Frnda, and P. N. Srinivasu, “NLP based text classification using TF-IDF enabled fine-tuned long short-term memory: An empirical analysis,” *Array*, p. 100467, 2025.
- [24] F. Rifaldy, Y. Sibaroni, and S. S. Prasetyowati, “Effectiveness of Word2VEC and TF-IDF in sentiment classification on online investment platforms using support Vector Machine,” *JUPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 10, no. 2, pp. 863–874, 2025.