ANALISIS SENTIMEN ULASAN PENGGUNA APLIKASI SPOTIFY DI GOOGLE PLAY STORE MENGGUNAKAN ALGORITMA *NAIVE BAYES*

Wahyu Ramadlan As'ari, Muhammad Arifin, Diana Laily Fithri, Pratomo Setiaji
Sistem Informasi, Universitas Muria Kudus
Jl. Lkr. Utara, Kayuapu Kulon, Gondangmanis, Kec.Bae, Kudus
202153094@std.umk.ac.id

ABSTRAK

Perkembangan teknologi yang pesat telah mengubah cara masyarakat dalam mendengarkan musik. Platform *streaming* musik menjadi pilihan utama bagi banyak orang karena fleksibilitasnya dalam menyediakan berbagai genre musik yang dapat diakses kapan saja dan di mana saja. Spotify merupakan platform *streaming* musik terbesar di dunia dengan lebih dari 500 juta pengguna aktif bulanan, namun memiliki rating lebih rendah dibandingkan pesaingnya di Google Play Store. Hal ini menunjukkan adanya ketidakpuasan pengguna. Penelitian ini bertujuan untuk melakukan analisis sentimen ulasan pengguna Spotify menggunakan algoritma *Naive Bayes* guna mengidentifikasi kecenderungan sentimen positif dan negatif. Pada Penelitian ini, data ulasan dikumpulkan melalui teknik *web scraping* menggunakan Python dan Google Colaboratory. Setelah data terkumpul, dilakukan tahap *pre-processing* yang mencakup *lemmatization*, *stopwords removal*, *tokenization and padding*, serta *label binarization*. Sentimen dikategorikan berdasarkan skor ulasan pengguna, di mana skor 1-3 diklasifikasikan sebagai negatif, sedangkan skor 4-5 sebagai positif. Model kemudian dilatih menggunakan algoritma *Naive Bayes* dan dievaluasi menggunakan metrik akurasi, presisi, recall, dan F1-score. Hasil evaluasi menunjukkan bahwa model *Naive Bayes* memiliki akurasi sebesar 86,5%, presisi 86,47%, *recall* 86,5%, dan F1-*score* 84,9%. yang mengindikasikan bahwa model mampu mengklasifikasikan sentimen dengan baik. Hasil penelitian ini diharapkan dapat memberikan wawasan bagi pengembang Spotify untuk meningkatkan kualitas layanan berdasarkan ulasan pengguna.

Kata kunci : *Spotify, Analisis Sentimen, Naïve Bayes, Machine Learning*

1. PENDAHULUAN

Transformasi digital telah merevolusi industri musik dengan munculnya layanan streaming yang menawarkan akses instan ke jutaan lagu. Salah satu platform terbesar, Spotify, telah menguasai pasar global dengan lebih dari 500 juta pengguna aktif bulanan. Namun, meskipun memiliki dominasi pasar yang kuat, Spotify menghadapi tantangan serius dalam mempertahankan kepuasan pengguna. Rating aplikasi ini di Google Play Store hanya 4.2, lebih rendah dibandingkan pesaingnya seperti Joox dan YouTube Music (4.4) serta SoundCloud (4.7). Fenomena ini menimbulkan pertanyaan mengapa platform streaming musik terbesar justru memiliki tingkat kepuasan yang lebih rendah dibandingkan pesaingnya [1].

Ulasan pengguna mengungkap berbagai keluhan yang berulang, seperti iklan yang mengganggu pada versi gratis, kualitas rekomendasi lagu yang kurang akurat, keterbatasan fitur dalam mode gratis, serta bug teknis yang sering terjadi. Banyaknya ulasan negatif menunjukkan adanya kesenjangan antara ekspektasi pengguna dan kualitas layanan yang diberikan Spotify. Hal ini dapat berdampak pada tingkat kepuasan dan loyalitas pengguna yang pada akhirnya memengaruhi retensi pelanggan serta citra platform di pasar layanan streaming musik. Jika masalah-masalah ini tidak segera diatasi, potensi penurunan jumlah pengguna aktif dan peningkatan churn rate dapat menjadi tantangan bagi Spotify dalam mempertahankan dominasinya di industri [2].

Dengan jumlah ulasan yang mencapai puluhan juta, perusahaan menghadapi tantangan besar dalam memahami pola kepuasan dan ketidakpuasan pengguna secara manual. Analisis manual terhadap data dalam skala besar tidak hanya memakan waktu, tetapi juga rentan terhadap bias dan inkonsistens. Oleh karena itu, diperlukan pendekatan berbasis kecerdasan buatan untuk menggali pola dari data ulasan tersebut secara sistematis [3].

Analisis sentimen merupakan teknik berbasis pemrosesan bahasa alami (*Natural Language Processing*/NLP) yang memungkinkan ekstraksi opini pengguna dari teks. Dengan mengklasifikasikan ulasan menjadi positif dan negatif, analisis ini dapat membantu mengidentifikasi aspek-aspek spesifik yang disukai dan tidak disukai pengguna. Namun, analisis sentimen bukan tanpa tantangan. Variasi bahasa, penggunaan slang, kontekstualisasi opini, serta bias data dapat memengaruhi akurasi model. Oleh karena itu, pemilihan algoritma yang tepat menjadi faktor krusial dalam penelitian ini [4] [5].

Salah satu algoritma yang sering digunakan dalam analisis sentimen adalah *Naive Bayes*, yang berbasis *Teorema Bayes*. Algoritma ini populer karena kesederhanaannya, efisiensi dalam menangani dataset besar, serta performa yang kompetitif dalam klasifikasi teks. Namun, asumsi independensi antarfitur dalam *Naive Bayes* dapat menjadi kelemahan, terutama dalam menangani kalimat kompleks dengan makna kontekstual yang kuat. Oleh karena itu, penelitian ini akan melakukan pra-pemrosesan data yang ketat untuk meningkatkan akurasi model, termasuk *lemmatization*, *stopwords removal*, *tokenization*, serta *label binarization* [6].

Penelitian ini bertujuan untuk mengidentifikasi faktor-faktor utama yang memengaruhi sentimen pengguna terhadap Spotify berdasarkan ulasan di Google Play Store dengan menggunakan algoritma Naive Bayes. Hasil penelitian ini diharapkan dapat memberikan wawasan strategis bagi pengembang Spotify dalam merancang kebijakan perbaikan layanan berbasis data, seperti peningkatan kualitas rekomendasi lagu, pengelolaan iklan, serta optimalisasi fitur pada versi gratis dan premium. Selain itu, penelitian ini juga berkontribusi dalam ranah akademik dengan memperkaya eksplorasi penerapan *machine learning*, khususnya algoritma *Naive Bayes* [7].

2. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Penelitian analisis sentimen terkait penggunaan aplikasi Kinemaster menggunakan metode *Naive Bayes* melakukan pengumpulan dataset berupa ulasan pengguna dari berbagai platform online. Proses *preprocessing* yang diterapkan meliputi tokenisasi, *stopword removal*, dan *stemming* untuk membersihkan serta mempersiapkan data sebelum dilakukan klasifikasi. Data dibagi menjadi dua bagian, yaitu data latih dan data uji dengan perbandingan 80:20. Hasil eksekusi klasifikasi menggunakan *Naive Bayes Classifier* menunjukkan bahwa dari data yang dianalisis, sentimen positif mendominasi dengan akurasi sebesar 85%, presisi 82%, *recall* 74%, dan f1-*score* 78% [8].

Hasil analisis sentimen pengguna aplikasi CapCut berdasarkan ulasan di Google Play Store dengan dataset sebanyak 880 ulasan menunjukkan bahwa mayoritas ulasan bersentimen negatif. Evaluasi menggunakan *confusion matrix* menghasilkan akurasi sebesar 84,09%, presisi 91,91%, dan *recall* 73,53%. Nilai akurasi yang cukup tinggi menunjukkan bahwa model mampu mengklasifikasikan sentimen dengan baik, sementara presisi yang tinggi menandakan bahwa sebagian besar prediksi positif memang benar-benar positif [9].

Pengujian analisis sentimen ulasan pengguna produk kesehatan di Tokopedia menggunakan algoritma *Naive Bayes* terdiri dari beberapa tahapan, yaitu *crawling*, *pre-processing*, pembobotan kata, dan klasifikasi sentimen. Hasil evaluasi menggunakan uji *Confusion Matrix* menunjukkan bahwa model memiliki akurasi sebesar 88%. Penelitian ini menegaskan bahwa pendekatan NLP yang digunakan dalam klasifikasi dua kelas, yaitu positif dan negatif, efektif dalam menangani informasi tekstual dalam skala besar [10].

2.2. Spotify

Spotify adalah layanan streaming musik digital yang didirikan pada tahun 2006 di Swedia. Layanan ini memungkinkan pengguna mendengarkan musik secara gratis dengan iklan atau berlangganan untuk mendapatkan pengalaman bebas iklan dan fitur tambahan seperti pemutaran offline [11].

2.3. *Data Mining*

Data mining adalah proses menggali informasi dari kumpulan data besar dan kompleks untuk mengidentifikasi pola atau tren yang berguna dalam pengambilan keputusan. Prosesnya mencakup pengumpulan data dari berbagai sumber, pembersihan data (*data cleaning*) untuk menghilangkan noise atau duplikasi, pemodelan data (*data modeling*) dengan menerapkan teknik seperti klasifikasi atau clustering, serta evaluasi model untuk menilai akurasi dan efektivitasnya sebelum digunakan dalam analisis lebih lanjut [12].

2.4. *Machine Learning*

Machine learning adalah cabang kecerdasan buatan yang memungkinkan sistem belajar dari data tanpa pemrograman eksplisit. Metode pembelajarannya terbagi menjadi tiga jenis utama, yaitu *supervised learning*, di mana model dilatih menggunakan data berlabel untuk memprediksi output berdasarkan pola yang telah dipelajari, *unsupervised learning*, di mana model mengidentifikasi pola tersembunyi dalam data tanpa label, dan *reinforcement learning*, di mana model belajar melalui interaksi dengan lingkungan dengan mendapatkan *reward* atau penalti untuk meningkatkan kinerjanya [13].

2.5. Analisis Sentimen

Analisis sentimen adalah proses identifikasi dan ekstraksi opini dari teks untuk menentukan apakah suatu ulasan bersifat positif, negatif, atau netral. Teknik ini memanfaatkan pemrosesan bahasa alami (*Natural Language Processing*/NLP) dan algoritma *machine learning* untuk memahami konteks serta pola dalam teks, memungkinkan sistem secara otomatis mengklasifikasikan sentimen berdasarkan kata-kata, frasa, atau struktur kalimat yang digunakan oleh pengguna [14].

2.6. *Naive Bayes Classifier*

Naive Bayes adalah algoritma klasifikasi berbasis probabilitas yang sederhana, mengacu pada *Teorema Bayes* dengan asumsi bahwa setiap fitur dalam data bersifat independen satu sama lain. Proses kerja algoritma *Naive Bayes* dimulai dengan menghitung probabilitas prior untuk setiap kelas, yaitu peluang awal sebelum melihat data. Selanjutnya, dihitung kemungkinan fitur muncul dalam setiap kelas. Kombinasi dari probabilitas prior dan kemungkinan fitur ini digunakan untuk menentukan probabilitas posterior, di mana kelas dengan probabilitas tertinggi dipilih sebagai hasil klasifikasi [15].

Persamaan model *Naive Bayes* adalah sebagai berikut:

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)} \quad (1)$$

Dimana:

- $P(A|B)$ adalah probabilitas A terjadi mengingat bahwa B telah terjadi.
- $P(B|A)$ adalah probabilitas B terjadi jika A telah terjadi.

- $P(A)$ adalah probabilitas terjadinya A.
- $P(B)$ adalah probabilitas terjadinya B.

2.7. *Confusion Matrix*

Confusion matrix merupakan alat evaluasi dalam *machine learning* yang digunakan untuk menilai kinerja model klasifikasi. Matriks ini menyajikan informasi mengenai hasil prediksi model dibandingkan dengan nilai aktual dalam dataset uji. Tujuan utama penggunaan *confusion matrix* adalah memberikan pemahaman yang lebih mendalam tentang performa model, dibandingkan hanya mengandalkan metrik akurasi. *Confusion matrix* membantu dalam mengevaluasi efektivitas algoritma machine learning dengan lebih komprehensif [16].

Tabel 1. *Confusion matrix*

	True	***False***
True (Positive)	*TP (True Positive) Correct result*	*FP (False Positive) Unexpected result*
False (Negative)	*FN (False Negative) Missing result*	*TN (True Negative) Correct absence of result*

Dari Tabel 1 berikut adalah metrik evaluasi yang digunakan:

2.7.1. *Accuracy*

Accuracy mengukur persentase prediksi yang benar terhadap keseluruhan prediksi yang dilakukan oleh model. Metrik ini memberikan gambaran umum tentang seberapa baik model melakukan klasifikasi. Rumusnya adalah sebagai berikut:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \quad (2)$$

2.7.2. *Precision*

Precision mengukur tingkat akurasi model dalam memprediksi setiap kelas (positif, netral, negatif). Precision berfokus pada seberapa banyak prediksi positif yang benar dari semua prediksi positif. Rumusnya sebagai berikut:

$$Precision = \frac{TP}{TP+FP} \times 100\% \quad (3)$$

2.7.3. *Recall*

Recall mengukur kemampuan model dalam mendeteksi seluruh data yang benar-benar termasuk dalam suatu kelas. Rumusnya adalah sebagai berikut:

$$Recall = \frac{TP}{TP+FN} \times 100\% \quad (4)$$

2.7.4. F1-*Score*

F1-*Score* merupakan rata-rata harmonis dari Precision dan Recall, yang memberikan keseimbangan antara kedua metrik tersebut. Rumusnya adalah sebagai berikut:

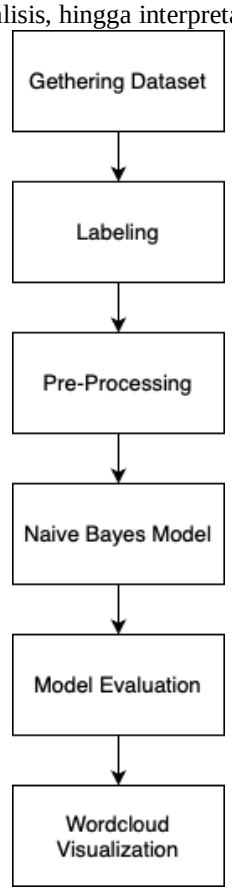
$$\text{F1} - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \times 100\% \quad (5)$$

2.8. Google Colaboratory

Google Colaboratory adalah platform berbasis *cloud* yang memungkinkan pengguna menjalankan kode Python secara interaktif di lingkungan notebook tanpa perlu menginstal perangkat lunak apapun di komputer lokal. Platform ini digunakan dalam pengolahan data, analisis statistik, serta implementasi *machine learning* dan *deep learning* [17].

3. METODE PENELITIAN

Dalam penelitian ini, setiap tahapan dirancang secara berurutan agar proses penelitian dapat berjalan dengan baik, mulai dari tahap pengumpulan data, pengolahan, analisis, hingga interpretasi hasil.



Gambar 1. Metodologi penelitian

Berikut merupakan tahapan yang dilakukan pada penelitian ini yang terdapat pada Gambar 1 diatas:

3.1. *Gethering Dataset*

Tahap pertama dalam penelitian ini adalah *Gathering Dataset*, yaitu mengumpulkan data ulasan pengguna dari sumber yang relevan. Data diperoleh melalui proses *scraping* dari Google Play Store atau sumber lain yang menyediakan ulasan pengguna terkait aplikasi yang dianalisis.

3.2. *Labeling*

Setelah data terkumpul, langkah berikutnya adalah memberi label *(labeling)* pada data. *Labeling* adalah proses pemberian label atau kategori

pada data mentah untuk membedakan kelas atau kategori tertentu.

3.3. *Preprocessing Data*

Preprocessing Data merupakan tahap yang sangat penting dalam analisis sentimen karena akan dilakukan proses text cleaning, data akan diolah dan dibersihkan untuk digunakan dalam model machine learning. Langkah-langkah *Preprocessing Data* adalah sebagai berikut:

3.3.1. *Lemmatization*

Lemmatization adalah proses mengubah kata-kata ke bentuk dasarnya. *Lemmatization* dilakukan untuk menyederhanakan variasi kata yang memiliki makna sama agar model lebih efisien dalam mengenali pola.

3.3.2. *Stopwords Removal*

Stopwords Removal adalah proses menghapus kata-kata yang tidak memberikan informasi penting atau tidak signifikan dalam analisis teks

3.3.3. *Tokenization and Padding*

Tokenization adalah proses memecah teks menjadi unit kecil yang disebut token. *Padding* adalah proses menambahkan elemen kosong (biasanya berupa angka nol) ke data teks yang telah ditokenisasi sehingga semua data memiliki panjang yang sama.

3.3.4. *Label Binarization*

Label Binarization adalah proses mengubah label kategori menjadi format numerik yang dapat diproses oleh algoritma machine learning. Dalam analisis sentimen, proses ini biasanya mengonversi label seperti "Positif" dan "Negatif" menjadi angka.

3.4. *Naive Bayes Model*

Data yang telah siap akan diproses menggunakan model *Naive Bayes*, salah satu metode berbasis probabilitas dan statistik dalam *machine learning* yang bekerja dengan memprediksi probabilitas kemiripan data sebelumnya terhadap data baru.

3.5. *Model Evaluation*

Tahap selanjutnya adalah mengevaluasi kinerja model menggunakan *Confusion Matrix*. Evaluasi dilakukan dengan menghitung akurasi, presisi, *recall*, dan F1-*score* untuk mengetahui sejauh mana model dapat mengklasifikasikan ulasan dengan baik.

3.6. *Wolrdcloud Visualization*

Tahap terakhir adalah menampilkan visualisasi kata menggunakan *Wordcloud*, di mana kata-kata yang sering muncul ditampilkan dengan ukuran lebih besar, sedangkan kata-kata yang jarang muncul memiliki ukuran lebih kecil. Teknik ini membantu dalam memahami pola kata yang dominan dalam ulasan pengguna, sehingga dapat memberikan gambaran awal mengenai aspek-aspek yang paling sering dibahas, baik dalam konteks positif maupun negatif.

4. HASIL DAN PEMBAHASAN

4.1. *Gethering Dataset*

Data dikumpulkan melalui *web scraping* menggunakan Python dan Google Colaboratory dari ulasan pengguna Spotify di Google Play Store. Data yang diambil mencakup username, rating, tanggal, dan ulasan.

```
[ ] import csv
    from google_play_scraper import reviews, Sort

    # ID aplikasi dari Google Play Store
    app_id = 'com.spotify.music'

    # Mengambil ulasan
    result, continuation_token = reviews(
        app_id,
        lang='id',  # Bahasa Indonesia
        country='id',  # Negara Indonesia
        sort=Sort.NEWEST,  # Sortir berdasarkan ulasan terbaru
        count=10000  # Jumlah ulasan yang ingin diambil
    )

    # Membuat atau membuka file CSV untuk menyimpan ulasan
    with open('spotify_newest.csv', mode='w', newline='', encoding='utf-8') as file:
        writer = csv.writer(file)
        # Menulis header CSV sesuai dengan atribut yang diminta
        writer.writerow(['userName', 'score', 'at', 'content'])

        # Menyimpan ulasan ke file CSV
        for review in result:
            writer.writerow([review['userName'], review['score'], review['at'], review['content']])

    print("Ulasan telah disimpan ke spotify_newest.csv")
```

Gambar 2. Prosess *scraping* dataset di google colaboratory

Dari Gambar 2 di atas data dikumpulkan menggunakan pustaka google play scraper dengan tautan aplikasi com.spotify.music. Proses ini berhasil mengambil 10.000 ulasan dengan kriteria pemilihan komentar berbahasa Indonesia, berasal dari negara Indonesia, serta disortir berdasarkan ulasan terbaru yang paling relevan. Dataset yang telah diperoleh kemudian disimpan dalam format CSV untuk tahap analisis lebih lanjut.

Tabel 2. Sampel ulasan

Ulasan	Rating
Aplikasi sangat bagus, fitur-fiturnya lengkap dan mudah digunakan.	5
Saya suka aplikasi ini, sangat bagus.	4
Mantap tapi banyak iklan yang mengganggu.	3
Kenapa tiba-tiba lagu suka berhenti?	2
Aplikasi sering force close, sangat mengecewakan	1

Tabel 2 berisi data sampel yang diambil dari hasil web scraping dan mencerminkan berbagai tingkat kepuasan pengguna terhadap aplikasi Spotify.

4.2. *Labeling*

Setelah data dikumpulkan, setiap ulasan akan dikategorikan ke dalam dua kelas sentiment, yaitu sentimen positif dan sentiment negatif. Proses ini dilakukan secara otomatis menggunakan fungsi berikut.

```
# Define a function to map 'score' to sentiment categories
def map_score_to_sentiment(score):
    if score in [1, 2, 3]:
        return 'negative'
    else:
        return 'positive'
```

Gambar 3. Prosess *labeling* di google colaboratory

Dari Gambar 3 di atas ulasan dengan skor 1-3 dikategorikan sebagai negatif, sedangkan skor 4-5 dikategorikan sebagai positif.

Tabel 3. Contoh hasil *labeling*

Ulasan	Label
Aplikasi sangat bagus, fitur-fiturnya lengkap dan mudah digunakan.	Positif
Saya suka aplikasi ini, sangat bagus.	Positif
Mantap tapi banyak iklan yang mengganggu.	Positif
Kenapa tiba-tiba lagu suka berhenti?	Negatif
Aplikasi sering force close, sangat mengecewakan	Negatif

Tabel 3 menampilkan hasil pelabelan data ulasan dengan rating 1-2 dikategorikan sebagai sentimen negatif, sedangkan ulasan dengan rating 3-5 dikategorikan sebagai sentimen positif.

4.3. *Preprocessing Data*

Tahap *preprocessing* dilakukan untuk mempersiapkan data sebelum digunakan dalam analisis sentimen. Langkah-langkah yang diterapkan meliputi *lemmatization*, *stopwords removal*, *tokenization and padding*, serta *label binarization*.

```
# Function for preprocessing text
def preprocess_text(text):
    text = str(text)
    text = re.sub(r"\bga\b", "tidak", text)
    text = re.sub(r"\bgk\b", "tidak", text)
    text = re.sub(r"\bgak\b", "tidak", text)
    text = re.sub(r"\btdk\b", "tidak", text)
    text = re.sub(r"\bgbs\b", "tidak bisa", text)
    text = re.sub(r"\bgabisa\b", "tidak bisa", text)
    text = re.sub(r"\bknp\b", "kenapa", text)
    text = re.sub(r"\bsmp\b", "sampai", text)
    text = re.sub(r"\bdr\b", "dari", text)
    text = re.sub(r"\bkl\b", "kalau", text)
    text = re.sub(r"\bsngt\b", "sangat", text)
    text = re.sub(r"\bsgt\b", "sangat", text)
    text = re.sub(r"\bdngn\b", "dengan", text)
    text = re.sub(r"\bdmn\b", "dimana", text)
    text = re.sub(r"\bdonlot\b", "download", text)
    text = re.sub(r"\bplylst\b", "playlist", text)

    lemmatizer = WordNetLemmatizer()
    stop_words = set(stopwords.words('indonesian'))
    tokens = word_tokenize(text)
```

Gambar 4. Proses *preprocessing* data di google colaboratory

Gambar 4 menunjukkan proses *preprocessing data* yang dilakukan di Google Colaboratory untuk membersihkan dan mempersiapkan ulasan sebelum dianalisis lebih lanjut.

4.4. *Lemmatization*

Lemmatization bertujuan untuk mengubah kata-kata dalam teks ulasan ke bentuk dasar agar lebih konsisten. Misalnya, kata "mendengarkan" diubah menjadi "dengar" dan "lagunya" menjadi "lagu". Proses ini membantu mengurangi jumlah variasi kata yang memiliki makna serupa.

Tabel 4. Contoh hasil *lemmatization*

Ulasan asli	Hasil *lemmatization*
Aplikasi sangat bagus, fitur-fiturnya lengkap dan mudah digunakan.	aplikasi sangat bagus, fitur lengkap dan mudah guna
Saya suka aplikasi ini, sangat bagus.	saya suka aplikasi ini, sangat bagus
Mantap tapi banyak iklan yang mengganggu.	mantap tapi banyak iklan ganggu
Kenapa tiba-tiba lagu suka berhenti?	kenapa tiba-tiba lagu suka berhenti
Aplikasi sering force close, sangat mengecewakan	aplikasi sering force close, sangat mengecewakan

Tabel 4 merupakan contoh hasil dari *lemmatization*, di mana kolom kiri berisi ulasan asli sebelum diproses, sementara kolom kanan menampilkan ulasan yang telah melewati proses *lemmatization*.

4.5. *Stopwords Removal*

Stopwords removal dilakukan dengan menghapus kata-kata umum yang tidak memiliki makna signifikan dalam analisis sentimen, seperti "dan", "di", atau "ke" dan lain sebagainya.

Tabel 5. Contoh hasil *stopwords removal*

Ulasan asli	Hasil *stopwords removal*
Aplikasi sangat bagus, fitur-fiturnya lengkap dan mudah digunakan.	aplikasi bagus fitur lengkap mudah guna
Saya suka aplikasi ini, sangat bagus.	suka aplikasi bagus
Mantap tapi banyak iklan yang mengganggu.	mantap banyak iklan ganggu
Kenapa tiba-tiba lagu suka berhenti?	kenapa lagu suka berhenti
Aplikasi sering force close, sangat mengecewakan	aplikasi sering force close mengecewakan

Tabel 5 merupakan contoh hasil dari *stopwords removal*, di mana kolom kiri berisi ulasan asli sebelum diproses, sementara kolom kanan menampilkan ulasan yang telah melewati proses *stopwords removal*.

4.6. *Tokenization and Padding*

Tokenization memecah teks ulasan menjadi kata-kata atau token yang lebih kecil agar dapat diproses oleh model. *Padding* diterapkan untuk menyamakan panjang input sehingga setiap ulasan memiliki jumlah token yang seragam.

Tabel 6. Contoh hasil *tokenization and padding*

Ulasan asli	Hasil *tokenization and padding*
Aplikasi sangat bagus, fitur-fiturnya lengkap dan mudah digunakan.	['aplikasi', 'bagus', 'fitur', 'lengkap', 'mudah', 'guna']
Saya suka aplikasi ini, sangat bagus.	['suka', 'aplikasi', 'bagus']

Ulasan asli	Hasil *tokenization and padding*
Mantap tapi banyak iklan yang mengganggu.	['mantap', 'banyak', 'iklan', 'ganggu']
Kenapa tiba-tiba lagu suka berhenti?	['kenapa', 'lagu', 'suka', 'berhenti']
Aplikasi sering force close, sangat mengecewakan	['aplikasi', 'sering', 'force', 'close', 'mengecewakan']

Tabel 6 merupakan contoh hasil dari *tokenization and padding*, di mana kolom kiri berisi ulasan asli sebelum diproses, sementara kolom kanan menampilkan ulasan yang telah melewati proses *tokenization and padding*.

4.7. *Label Binarization*

Label binarization dilakukan untuk mengubah kategori sentimen menjadi format numerik. Ulasan positif dikonversi menjadi nilai 1 dan ulasan negatif menjadi nilai 0. Konversi ini memungkinkan model memahami dan memproses sentimen dengan lebih efektif.

Tabel 7. Contoh hasil *label binarization*

Ulasan asli	Hasil *label binarization*
Aplikasi sangat bagus, fitur-fiturnya lengkap dan mudah digunakan.	1
Saya suka aplikasi ini, sangat bagus.	1
Mantap tapi banyak iklan yang mengganggu.	1
Kenapa tiba-tiba lagu suka berhenti?	0
Aplikasi sering force close, sangat mengecewakan	0

Tabel 7 merupakan contoh hasil dari *tokenization and padding*, di mana kolom kiri berisi ulasan asli sebelum diproses, sementara kolom kanan menampilkan ulasan yang telah melewati proses *tokenization and padding*.

4.8. *Naive Bayes Model*

Setelah data ulasan melewati tahap *preprocessing*, dilakukan proses pelatihan model klasifikasi menggunakan algoritma *Naive Bayes*, khususnya Multinomial *Naive Bayes*. Model ini digunakan karena kemampuannya dalam menangani data teks dengan menghitung probabilitas kemunculan kata dalam setiap kategori sentimen.

```
[ ]  # Train Naive Bayes model
     nb_model = MultinomialNB()
     nb_model.fit(X_train, y_train)

     # Predict sentiment labels for the test set
     y_pred_nb = nb_model.predict(X_test)
```

Gambar 5. Proses pembangunan model *naive bayes*

Gambar 5 menunjukkan model dilatih menggunakan data latih (X_train, y_train) dan diuji dengan data uji (X_test, y_test). Model ini bekerja dengan menghitung probabilitas kata-kata dalam dataset dan menentukan sentimen berdasarkan nilai probabilitas tertinggi.

4.9. *Model Evaluation*

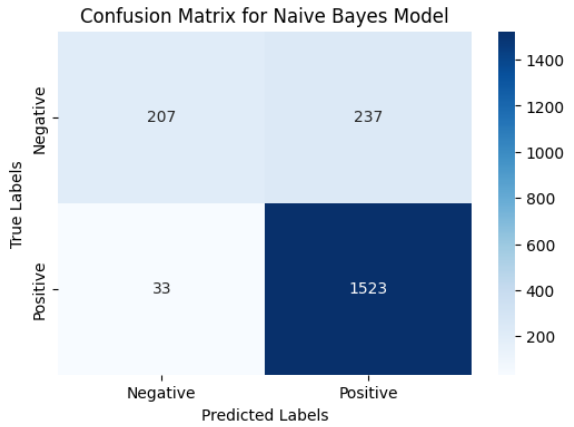
Setelah model dilatih, dilakukan evaluasi untuk mengetahui seberapa baik model dalam mengklasifikasikan sentimen ulasan pengguna.

menggunakan beberapa metrik untuk mengukur kinerja model *Naive Bayes*.

```
# Function to evaluate model and return metrics
def evaluate_model(y_true, y_pred):
    accuracy = accuracy_score(y_true, y_pred)
    precision = precision_score(y_true, y_pred, average='weighted', zero_division=0)
    recall = recall_score(y_true, y_pred, average='weighted', zero_division=0)
    f1 = f1_score(y_true, y_pred, average='weighted', zero_division=0)
    return accuracy, precision, recall, f1
```

Gambar 6. Prosess pembangunan *model evaluation*

Gambar 6 menunjukkan evaluasi dilakukan menggunakan beberapa metrik untuk mengukur kinerja model *Naive Bayes*. Fungsi ini menerima nilai sebenarnya (y_true) dan nilai prediksi (y_pred), kemudian menghitung masing-masing metrik dengan pendekatan berbobot (*weighted*).



Gambar 7. Hasil *naive bayes classification matrics*

Gambar 7 menunjukkan hasil *Naive Bayes Classification Metrics* berhasil memprediksi 1.523 sampel positif dengan benar (*True Positive*/TP) dan 207 sampel negatif dengan benar (*True Negative*/TN). Model juga menghasilkan 237 prediksi positif yang sebenarnya negatif (*False Positive*/FP) dan 33 prediksi negatif yang sebenarnya positif (*False Negative*/FN). Selanjutnya model akan dihitung dengan berbagai matriks evaluasi dan didapat hasil sebagai berikut.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \times 100\%$$

$$Accuracy = \frac{1523 + 207}{1523 + 207 + 237 + 33} = \frac{1730}{2000} = 86{,}5\%$$

$$Precision = \frac{TP}{TP + FP} \times 100\%$$

$$Precision = \frac{1523}{1523 + 237} = \frac{1523}{1760} = 86{,}47\%$$

$$Recall = \frac{TP}{TP + FN} \times 100\%$$

$$Recall = \frac{1523}{1523 + 33} = \frac{1523}{1556} = 86{,}5\%$$

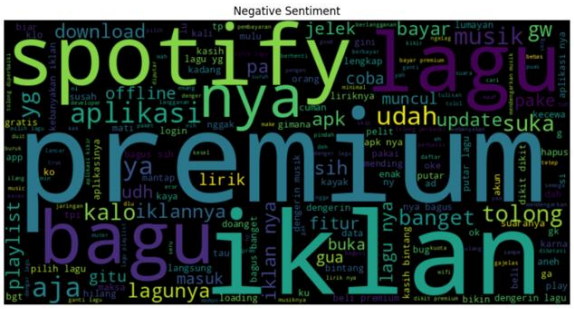
$$F1\text{-Score} = 2 \times \frac{Precision \times Recall}{Recall + Precision} \times 100\%$$

$$F1\text{-Score} = 2 \times \frac{0.8647 \times 0.8650}{0.8650 + 0.8647} = 84.9\%$$

Model ini mencapai akurasi sebesar 86,5%, dengan 1.730 dari 2.000 sampel berhasil diprediksi dengan benar. Presisi sebesar 86,47% menunjukkan bahwa dari semua prediksi positif 86,47% benar. Sementara itu, *recall* sebesar 86,5% mengindikasikan model mampu mendeteksi 86,5% dari total sampel positif. Kombinasi keduanya menghasilkan F1-*score* sebesar 84,9%, mencerminkan keseimbangan antara identifikasi positif yang benar dan kesalahan prediksi positif. Secara keseluruhan, model *Naive Bayes* menunjukkan performa yang baik dalam klasifikasi data sesuai tujuan penelitian.

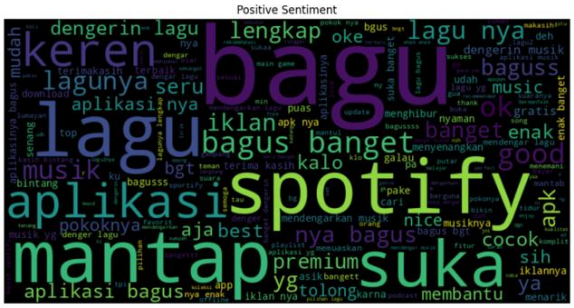
4.10. *Wordcloud Visualization*

Tahap akhir dalam analisis ini adalah *Wordcloud Visualization*, yang bertujuan untuk menampilkan kata-kata yang paling sering muncul dalam ulasan pengguna aplikasi Spotify. *Wordcloud* merupakan teknik visualisasi yang menyoroti kata-kata yang paling sering muncul dengan ukuran huruf yang lebih besar, semakin sering kata muncul dalam dataset, semakin besar ukurannya dalam visualisasi.



Gambar 8. Hasil *wordcloud* sentimen negatif

Gambar 8 menunjukkan hasil kata yang sering muncul dalam ulasan sentiment negatif, diantaranya "spotify", "premium", "iklan", "lagu", "ap likasi", "offline", "download", "jelek", dan "tolong". Kata-kata ini menunjukkan bahwa banyak pengguna mengeluhkan fitur premium, iklan yang mengganggu, serta masalah dalam mendownload lagu dan penggunaan *offline.*



Gambar 9. Hasil *wordcloud* sentimen positif

Gambar 9 menunjukkan hasil kata yang sering muncul dalam ulasan sentiment positif, diantaranya "bagus", "spotify", "mantap", "lagu", "aplikasi", "musik", "keren", "seru", "gratis", dan "suka". Kata-kata ini mencerminkan bahwa pengguna puas dengan kualitas musik, pengalaman mendengarkan yang menyenangkan, serta fitur gratis yang ditawarkan oleh Spotify.

5. KESIMPULAN DAN SARAN

Berdasarkan hasil analisis sentimen terhadap 10.000 ulasan pengguna Spotify menggunakan algoritma *Naive Bayes*, penelitian ini menunjukkan bahwa model yang diterapkan mampu mengklasifikasikan sentimen dengan akurasi sebesar 86,5%, presisi 86,47%, *recall* 86,5%, dan F1-*score* 84,9%. *Wordcloud visualization* mengungkapkan bahwa pengguna dengan sentimen negatif cenderung mengeluhkan iklan, fitur premium, serta keterbatasan akses *offline*, sementara pengguna dengan sentimen positif lebih banyak memuji kualitas musik, kemudahan penggunaan, dan pengalaman mendengarkan yang menyenangkan. Untuk penelitian selanjutnya, disarankan untuk membandingkan performa *Naive Bayes* dengan algoritma lain seperti *Random Forest* atau *Support Vector Machine* guna memperoleh hasil klasifikasi yang lebih optimal, serta meningkatkan analisis dengan mempertimbangkan aspek temporal atau tren sentimen seiring waktu.

DAFTAR PUSTAKA

[1] A. S. Rahayu, A. Fauzi, and R. Rahmat, "Komparasi Algoritma Naïve Bayes Dan Support Vector Machine (SVM) Pada Analisis Sentimen Spotify," *J. Sist. Komput. dan Inform.*, vol. 4, no. 2, p. 349, 2022.

[2] F. Odelia, "Analisis Kepuasan Pengguna Aplikasi Spotify," *Jma)*, vol. 2, no. 2, pp. 3031–5220, 2024.

[3] S. Nafisyah and R. Sulistiyowati, "Analisis Sentimen Ulasan Produk Toko Online Esrocte untuk Peningkatan Pelayanan Menggunakan Algoritma Naïve Bayer," *Blantika Multidiscip. J.*, vol. 2, no. 8, pp. 667–673, 2024.

[4] E. F. Baharsyah, A. Armanto, T. H. B. Aviani, and C. Wulandari, "Analisis Sentimen Pengguna Terhadap Aplikasi Belajar Online Ruang Guru Pada Ulasan Google Play Store Menggunakan Algoritma Naïve Bayes dan Support Vector Machine," *Innov. J. Soc. Sci. Res.*, vol. 4, no. 3, pp. 2965–2979, 2024.

[5] M. Y. Siregar, A. Davy Wiranata, and R. A. Saputra, "KLIK: Kajian Ilmiah Informatika dan Komputer Analisis Sentimen Pada Ulasan Pengguna Aplikasi Streaming Vidio Menggunakan Metode Naïve Bayes," *Media Online*, vol. 4, no. 5, pp. 2419–2429, 2024.

[6] A. Bagas Pranata, A. R. Abdillah, and F. Irwiensyah, "KLIK: Kajian Ilmiah Informatika dan Komputer Analisis Sentimen Ulasan Pengguna Aplikasi Netflix Pada Google Play

Menggunakan Algoritma Naïve Bayes," *Media Online)*, vol. 4, no. 6, pp. 3091–3098, 2024.

[7] M. Gamma, A. Hakim, and F. Irwiensyah, "Analisis Sentimen Terhadap Ulasan Pengguna Pada Aplikasi BCA Mobile Menggunakan Metode Naïve Bayes," *J. Inf. Syst. Res.*, vol. 5, no. 4, pp. 911–921, 2024.

[8] K. Kevin, M. Enjeli, and A. Wijaya, "Analisis Sentimen Pengunaaan Aplikasi Kinemaster Menggunakan Metode Naive Bayes," *J. Ilm. Comput. Sci.*, vol. 2, no. 2, pp. 89–98, 2024.

[9] Meliyawati and F. N. Hasan, "Analisis Sentimen Pengguna Aplikasi CapCut Pada Ulasan di Play Store Menggunakan Metode Naïve Bayes," *Media Online*, vol. 4, no. 4, pp. 2272–2280, 2024.

[10] A. Ernawati, A. O. Sari, S. N. Sofyan, M. Iqbal, and R. F. W. Wijaya, "Implementasi Algoritma Naïve Bayes dalam Menganalisis Sentimen Review Pengguna Tokopedia pada Produk Kesehatan," *Bull. Inf. Technol.*, vol. 4, no. 4, pp. 533–543, 2023.

[11] S. Navisa, Luqman Hakim, and Aulia Nabilah, "Komparasi Algoritma Klasifikasi Genre Musik pada Spotify Menggunakan CRISP-DM," *J. Sist. Cerdas*, vol. 4, no. 2, pp. 114–125, 2021.

[12] H. Hendri, "Implementasi Data Mining Dengan Metode C4.5 Untuk Prediksi Mahasiswa Penerima Beasiswa," *Indones. J. Comput. Sci.*, vol. 10, no. 2, pp. 312–321, 2021.

[13] M. R. S. Alfarizi, M. Z. Al-farish, M. Taufiqurrahman, G. Ardiansah, and M. Elgar, "Penggunaan Python Sebagai Bahasa Pemrograman untuk Machine Learning dan Deep Learning," *Karya Ilm. Mhs. Bertauhid (KARIMAH TAUHID)*, vol. 2, no. 1, pp. 1–6, 2023.

[14] M. Arsadhana, B. Efendi, and M. Trihudiyatmanto, "ANALISIS KEPUASAN PELANGGAN MELALUI SENTIMEN ULASAN MENGGUNAKAN ALGORITMA NAIVE BAYE S," vol. XIII, no. 1, pp. 1–8, 2025.

[15] 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, 2024.

[16] E. Fauziningrum, M. Pd and M. P. Encis Indah Suryaningsih, S.T., "Evaluasi Dan Prediksi Penguasaan Bahasa Inggris Maritim Menggunakan Metode Decision Tree Dan Confusion Matrix (Studi Kasus Di Universitas Maritim Amni)," *Angew. Chemie Int. Ed. 6(11), 951–952.*, pp. 5–24, 2021.

[17] R. Gelar Guntara, "Pemanfaatan Google Colab Untuk Aplikasi Pendeteksian Masker Wajah Menggunakan Algoritma Deep Learning YOLOv7," *J. Teknol. Dan Sist. Inf. Bisnis*, vol. 5, no. 1, pp. 55–60, 2023.