JNATIA Volume 4, Nomor 2, Februari 2026 p-ISSN: 2986-3929
Jurnal Nasional Teknologi Informasi dan Aplikasinya e-ISSN: 3032-1948

Analisis Sentimen Gemini AI Menggunakan Multinomial Naïve Bayes dengan TF-IDF dan BoW

Yeremi Kornelius Purba[a1], I Gede Surya Rahayuda[a2]

Program Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam,
Universitas Udayana, Bali
Jalan Raya Kampus Udayana, Bukit Jimbaran, Kuta Selatan, Badung, Bali, Indonesia
[1]yeremikornelius2005@gmail.com
[2]igedesuryarahayud@unud.ac.id

Abstract

The development of artificial intelligence Large Language Models, such as Gemini AI, has attracted widespread public attention. The advanced development of Gemini AI is inseparable from valuable public reviews for product evaluation, but the massive amount of reviews makes manual analysis inefficient. This study aims to conduct sentiment analysis on Gemini AI application reviews using the Multinomial Naïve Bayes classification algorithm. The primary focus is to compare the performance of two feature extraction methods: Term Frequency-Inverse Document Frequency (TF-IDF) and Bag-of-Words (BoW). A total of 1,000 reviews were collected from the Google Play Store, which, after undergoing preprocessing and data labelling, resulted in 438 data points for analysis. The model evaluation results show that the Multinomial Naive Bayes model using TF-IDF feature extraction is more effective in analyzing the sentiment of Gemini AI app reviews with an Accuracy of 88%, Precision of 88%, Recall of 100%, and F1-Score of 93%.

Keywords: *Sentiment analysis, Gemini AI, Multinomial Naïve Bayes, TF-IDF, BoW*

1. Pendahuluan

Perkembangan teknologi dan kecerdasan buatan di era saat ini telah banyak mengubah cara manusia berinteraksi dengan informasi dan layanan digital. Salah satu kemajuan dari teknologi kecerdasan buatan AI saat ini adalah *Large Language Model* (LLM). *Large Language Model* (LLM) merupakan program kecerdasan buatan yang mampu memahami, menghasilkan, dan memprediksi konten baru serta mampu berinteraksi dengan menggunakan bahasa manusia. Salah satu produk dari program *Large Language Model* (LLM) adalah Gemini AI yang diluncurkan oleh perusahaan pemimpin inovasi teknologi, yakni Google. Kehadiran aplikasi Gemini AI ini banyak menarik perhatian seluruh masyarakat dunia, termasuk masyarakat Indonesia, sehingga pastinya aplikasi ini tidak terlepas dari kritik dan ulasan pengguna selama menggunakan aplikasi Gemini AI ini.

Ulasan aplikasi merupakan sumber data yang sangat berharga bagi pengembang aplikasi karena berisi ulasan, opini, atau kritik dari pengguna aplikasi yang bersifat positif atau negatif. Ulasan pengguna ini dapat membantu dalam mengevaluasi setiap fitur aplikasi, mengidentifikasi *bug*, dan memahami kebutuhan pengguna. Namun, jumlah ulasan pengguna aplikasi pastinya terus bertambah setiap harinya, sehingga analisis secara manual tidak lagi menjadi efisien dan akan banyak memakan waktu. Data yang tidak terstruktur ini memerlukan pendekatan komputasional untuk dapat diolah menjadi informasi yang bermakna. Di sinilah peran *text mining* dan analisis sentimen menjadi sangat relevan digunakan. *Text mining* merupakan proses mengekstrak kumpulan dokumen atau informasi menjadi informasi yang bermakna dengan cara mengidentifikasi atau menemukan pola antar dokumen, sehingga dapat diketahui hubungan antar polanya [1]. Sedangkan, analisis sentimen adalah cabang dari *Natural Language Processing* (NLP) dengan tujuan mengidentifikasi sentimen dari data tekstual, apakah data tekstualnya merupakan sentimen positif, negatif, atau netral.

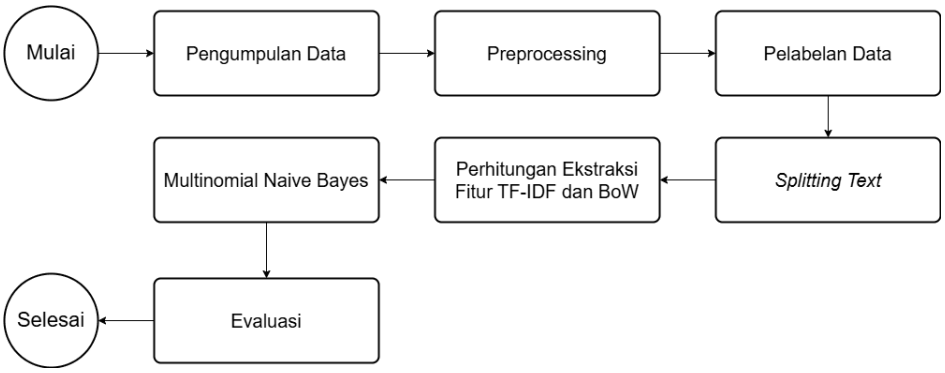
Sebelum mengklasifikasikan sentimen dari data tekstual ulasan aplikasi, diperlukan proses pelabelan untuk menentukan polaritas atau sentimen dari setiap ulasan. Pelabelan secara manual dianggap tidak efesien karena jumlah ulasan sebuah aplikasi terus bertambah setiap harinya sehingga akan banyak memakan waktu. Di sisi lain, pelabelan dengan metode *pre-trained classifier* sering dianggap banyak mengalami kendala dalam menangkap istilah spesifik yang sering muncul dalam aplikasi tertentu. Sebagai solusi, metode *Lexicon – based* hadir dengan pendekatan yang menggunakan kamus kata – kata dengan penanda polaritas untuk mengevaluasi sentimen dalam data tekstual [2].

Dalam melakukan klasifikasi sentimen, diperlukan algoritma *Machine Learning* untuk dapat memproses klasifikasi sentimen secara otomatis. Salah satu algoritma *Machine Learning* yang cocok digunakan untuk klasifikasi sentimen adalah *Multinomial Naïve Bayes*. Terdapat beberapa penelitian terdahulu yang melakukan analisis sentimen menggunakan algoritma *Multinomial Naïve Bayes*. Penelitian yang dilakukan oleh Wibawa (2023) yang membandingkan ekstraksi fitur TF-IDF dengan *Bag of Words* (BoW) terhadap "Teks Berbahasa Bali" dengan menggunakan algoritma *Multinomial Naïve Bayes* yang menghasilkan performa TF-IDF yang lebih baik daripada *Bag of Words* (BoW) dengan akurasi sebesar 91,5% [3]. Dan penelitian yang dilakukan Arminda dkk. (2023) juga menggunakan algoritma *Multinomial Naïve Bayes* terhadap ulasan aplikasi BRIMO, namun dalam membagi data *sampling*-nya, penulis menggunakan teknik *K-fold cross Validation* yang mendapatkan akurasi sebesar 98,02% [4].

Dalam penelitian ini, akan dilakukan analisis komparatif terhadap dua metode ekstraksi fitur yang sering diterapkan dalam algoritma *Multinomial Naïve Bayes*, yakni *Term Frequency-Inverse Document Frequency* (TF-IDF) dan *Bag of Words* (BoW). Tujuannya adalah mengevaluasi dan menentukan metode pembobotan kata yang menghasilkan akurasi yang tinggi dari *dataset* ulasan aplikasi Gemini AI.

2. Metode Penelitian

Penelitian menganalisis sentimen ulasan aplikasi Gemini AI ini dilakukan dengan melalui beberapa tahapan dari pengumpulan data hingga evaluasi model. Semua tahapan penelitian harus dilakukan agar mendapatkan hasil yang baik dan maksimal. Berikut diagram alur yang akan dilakukan pada penelitian ini seperti dapat dilihat pada Gambar 1.



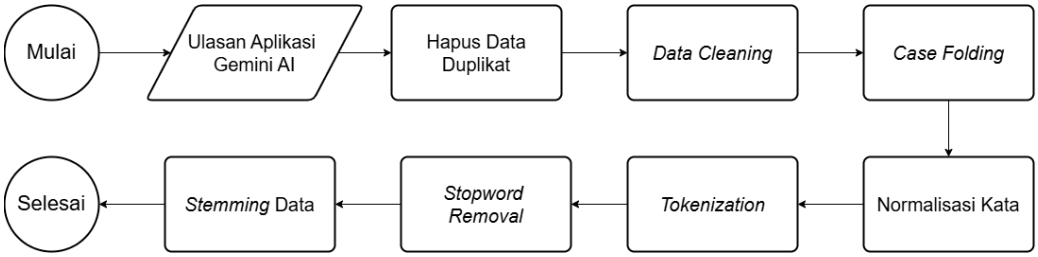
Gambar 1. Tahapan Penelitian

2.1. Pengumpulan Data

Dataset yang digunakan dalam peneltian merupakan data ulasan aplikasi Gemini AI yang diambil dari *website* Google Play Store. Metode pengambilan datanya menggunakan *web scrapping*. *Web scrapping* merupakan proses mengekstraksi informasi dari halaman web menggunakan sebuah perangkat lunak atau alat otomatis [5]. Salah satu dari alat dari *web scrapping* adalah modul atau *library* dari bahasa pemrograman Python. Dan alat ini yang digunakan mengambil data dalam penelitian ini, yakni modul atau *library google-play-scrapper*.

2.2. Text Preprocessing

Preprocessing text merupakan proses yang dilakukan pertama kali kepada data tekstual dengan tujuan pembersihan dan penyusunan ulang data agar terstruktur sehingga data dapat lebih mudah diolah oleh model algoritma [6]. Tahapan dalam melakukan *preprocessing* dimulai dari penghapusan data duplikat dari *dataset*, pembersihan data dari karakter non-alfabet sekaligus membuat seluruh karakter alfabet dalam format huruf kecil. Kemudian, dilanjutkan dengan proses normalisasi kata slang yang terdapat di dalam *dataset*, menjadikan seluruh kata di dalam satu kalimat menjadi satu kata yang berdiri sendiri atau dapat disebut proses tokenisasi. Selanjutnya, *dataset* akan diproses ke tahap *stopword removal*, yakni menghapus seluruh kata yang tidak memiliki arti. Dan terakhir, setiap kata di dalam *dataset* diubah ke dalam bentuk kata dasar. Berikut diagram alur *preprocessing* yang dapat dilihat ada Gambar 2.



Gambar 2. Tahapan *Preprocessing*

2.3. Pelabelan Data

Pelabelan data merupakan proses mengidentifikasi data ulasan ke dalam sentimen positif, negatif, atau netral. Salah satu metode yang dapat digunakan untuk pelabelan data adalah dengan menggunakan metode *Lexicon - based*. Metode *Lexicon - based* merupakan metode yang menggunakan kamus sentimen yang berisi daftar kata yang diberi bobot untuk mencocokkan setiap kata di dalam teks kemudian dijumlahkan untuk dicari polaritas keseluruhan teks [7]. Kamus *Lexicon* yang digunakan dalam penelitian ini menggunakan *InSet* (*Indonesia Sentiment) Lexicon* yang terdiri dari 3.609 kata positif dan 6.609 kata negatif [8].

2.4. Splitting Text

Splitting text merupakan proses membagi *dataset* menjadi dua bagian, yakni data latih (*training*) dan data uji (*testing*). Data latih *(training*) digunakan untuk melatih model algoritma klasifikasi yang digunakan dalam penelitian. Sedangkan, data uji (*testing*) digunakan sebagai data untuk menguji model algoritma klasifikasi yang sudah dilatih sebelumnya menggunakan data latih (*training*).

2.5. Ekstraksi Fitur

Ektraksi fitur merupakan metode untuk mengubah data tekstual menjadi bentuk numerik agar dapat diolah ke dalam model algoritma pemrosesan data. Dalam penelitian ini, ekstraksi fitur yang digunakan adalah *Term Frequency-Invers Document Frequency* (TF-IDF) dan *Bag-of-Words* (BoW).

a. Term Frequency-Invers Document Frequency (TF-IDF)

TF-IDF merupakan algoritma untuk ekstraksi fitur dengan mengubah data tekstual menjadi angka yang digunakan untuk menentukan bobot dari sebuah kata di dalam setiap dokumen [9].Nilai TF-IDF didapatkan dengan mengalikan TF (*Term Frequency*) dengan IDF (*Invers Document Frequency*), di mana TF (*Term Frequenxy*) mengukur seberapa sering sebuah kata muncul di dalam sebuah dokumen dan IDF (*Invers Document Frequency*) mengukur

seberapa langka sebuah kata di seluruh dokumen. Berikut persamaan matematika dari perhitungan TF-IDF.

$$W(d,t) = TF(d,t) \times \log\left(\frac{n}{df(t)}\right) \quad (1)$$

Jadi dengan menggunakan TF-IDF, kata yang sering muncul akan memiliki nilai yang cenderung kecil dibandingkan dengan kata yang jarang muncul.

b. *Bag-of-Words* (BoW)

Bag-of-Words merupakan alogritma ekstraksi fitur yang mengubah data tekstual menjadi angka berbentuk vektor dengan menghitung frekuensi kemunculan kata pada seluruh dokumen [9]. Jadi dapat dikatakan, *Bag-of-Words* merupakan proses yang mempelajari seluruh kosakata atau *vocabulary* dari seluruh dokumen yang kemudian dimodelkan di tiap dokumennya jumlah kemunculan setiap katanya [3].

2.6. Multinomial Naïve Bayes

Algoritma *Multinomial Naïve Bayes* merupakan salah satu metode pembelajaran probabilistik dan variasi dari algoritma *Naïve Bayes* yang digunakan dalam *Natural Language Processing* (NLP) yang bekerja pada konsep *term frequency* yang memiliki arti seberapa sering sebuah kata muncul dalam dokumen yang diberikan dan berapa jumlah kemunculan katanya [10]. Berikut bentuk matematis dari Multinomial Naïve Bayes.

$$P(p|n) \propto P(p) \prod_{1 \le k \le nd} P(t_k|p) \quad (2)$$

Dimana P(tk|p) merupakan probabilitas munculnya dokumen *text* (tk), n adalah jumlah dokumen, dan p adalah polaritas.

2.7. Evaluasi

Tahap evaluasi merupakan tahap menguji hasil penelitian penerapan data ulasan aplikasi ke dalam algoritma pemrosesan data menggunakan *confussion matrix*. Pada tahap ini akan dilakukan pengukuran ketepatan model dalam memprediksi kelas sentimen pada dataset yang digunakan. Dan juga terdapat empat parameter dalam mengukur ketepatan model, yakni *accuracy, precision, recall,* dan *F1-score* [11].

3. Hasil dan Pembahasan

3.1. Pengumpulan Data

Dataset yang digunakan dalam penelitian ini merupakan data ulasan aplikasi Gemini AI yang didapatkan dari pihak ketiga, yakni *website* Google Play Store sebanyak 1000 data ulasan dari berbagai *rating* pengguna. Berikut tampilan *dataset* yang dapat dilihat pada Tabel 1.

Tabel 1. *Dataset* Ulasan Aplikasi Gemini AI

userName	score	content
Putra Shyputrah	5	ok
Samudra Aidan	5	bagus bangeeeeeet
David Oenshao	5	sangat bagus di digunakan untuk menanyakan pertanyan yang saya tidak tau menjadi tau terutama saat menanyakan tentang pelajaran jawabanya sangat mudah di mengerti

p-ISSN: 2986-3929
e-ISSN: 3032-1948

userName	score	content
Mia Mekii	3	sebenernya bagus cuman agak berat dan gue baru tau kalok ai punya kelemahannya masing2
ade elfani	5	Bagus banget! GONG BANGET GILA NIH AI UDAH BISA VC AJE

3.2. Text Preprocessing

Dataset yang dikumpulkan sebanyak 1000 data, selanjutnya dilakukan proses preprocessing, dimulai dari penghapusan data duplikat hingga mengubah setiap kata di dalam dataset ke dalam bentuk kata dasar. Setelah melewati tahap *preprocessing*, *dataset* berkurang menjadi 575 data. Berikut *dataset* setelah melewati tahapan *preprocessing* yang dapat dilihat pada Tabel 2.

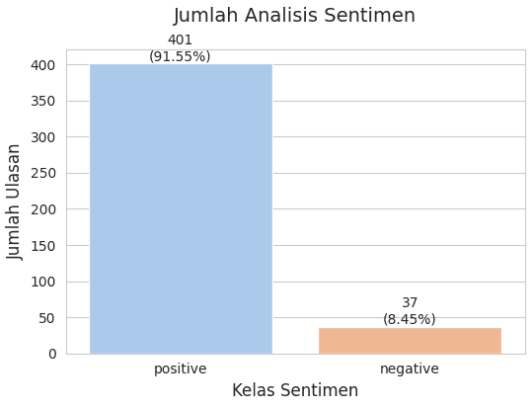
Tabel 2. Hasil *Preprocessing*

No.	Proses Preprocessing	Hasil
1	*content*	sangat bagus di digunakan untuk menanyakan pertanyan yang saya tidak tau menjadi tau terutama saat menanyakan tentang pelajaran jawabanya sangat mudah di mengerti
2	*Data Cleaning*	sangat bagus di digunakan untuk menanyakan pertanyan yang saya tidak tau menjadi tau terutama saat menanyakan tentang pelajaran jawabanya sangat mudah di mengerti
3	*Case Folding*	sangat bagus di digunakan untuk menanyakan pertanyan yang saya tidak tau menjadi tau terutama saat menanyakan tentang pelajaran jawabanya sangat mudah di mengerti
4	Normalisasi Kata	sangat bagus di digunakan untuk menanyakan pertanyan yang saya tidak tau menjadi tau terutama saat menanyakan tentang pelajaran jawabanya sangat mudah di mengerti
5	*Tokenization*	[sangat,bagus,di,digunakan,untuk,menanyakan,pertanyan, yang,saya,tidak,tau,menjadi,tau,terutama,saat,menanyaka n,tentang,pelajaran,jawabanya,sangat,mudah,di,mengerti]
6	*Stopword Removal*	[bagus,pertanyan,tau,tau,pelajaran,jawabanya,mudah,men gerti]
7	*Stemming*	[bagus,pertanyan,tau,tau,ajar,jawabanya,mudah,erti]

3.3. Pelabelan Data

Tahap pelabelan data dengan menggunakan metode *Lexicon - based* bekerja dengan cara setiap kata di dalam *dataset*-nya dicocokkan dengan kamus *Lexicon - based*. Sentimen yang digunakan dalam penelitian ini adalah sentimen positif dan negatif. Jadi setiap kata di dalam satu baris *dataset* dikelompokkan ke dalam kedua sentimen tersebut, kemudian sentimen yang memiliki kata terbanyak di dalamnya akan mewakili sentimen di satu baris *dataset* ulasan tersebut. Distribusi hasil pelabelan dapat dilihat pada Gambar 3.

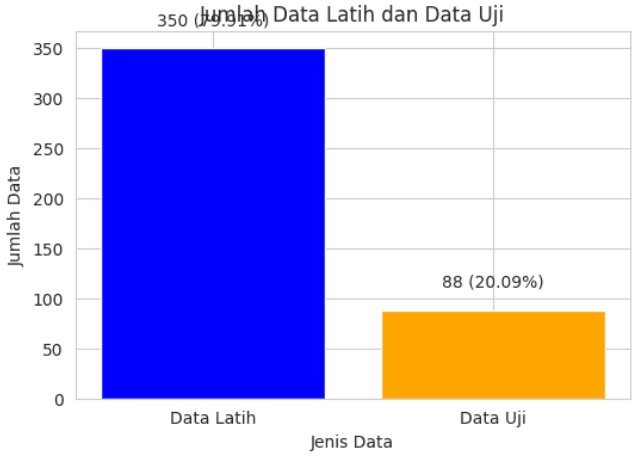
p-ISSN: 2986-3929
e-ISSN: 3032-1948



Gambar 3. Frekuensi Data Setiap Label

3.4. Splitting Text

Pembagian data menjadi data latih (*training*) dan data uji (*testing*) dalam penelitian ini menggunakan modul atau *library* dari Python, yakni *scikit-learn*. *Scikit-learn* merupakan modul atau *library Machine Learning* yang menyediakan banyak modul untuk mengakses data, persiapan data, dan pembuatan model statistik [12]. Dalam penelitian ini *dataset* dibagi menjadi data latih sebanyak 80% dan data uji sebanyak 20%. Berikut hasil distribusi pembagian data latih dan data uji yang dapat dilihat pada Gambar 4



Gambar 4. Frekuensi Data Uji dan Data Latih

3.5. Ekstraksi Fitur

Dalam penelitian ini, metode ekstraksi fitur yang digunakan untuk mengubah data tekstual menjadi bentuk numerik, yakni *Term Frequency-Invers Document Frequency* (TF-IDF) dan *Bag-of-Words* (BoW).

a. Term Frequency-Invers Document Frequency (TF-IDF)

Dalam penelitian ini, ekstraksi fitur dengan menggunakan TF-IDF dibantu dengan modul *scikit-learn* Python, yakni TfidfVectorizer. Cara kerjanya adalah semakin jarang sebuah kata itu muncul di dalam dokumen, maka nilai dari kata tersebut akan semakin besar, begitu juga sebaliknya. Jumlah fitur yang digunakan dalam modul ini adalah sebanyak 400 fitur dan *ngram* yang digunakan adalah *unigram* dan *bigram*. *Unigram* dan *bigram*

p-ISSN: 2986-3929
e-ISSN: 3032-1948

digunakan agar kombinasi kata terhitung menjadi satu kesatuan. Berikut tampilan kode dari ekstraksi fitur dengan TF-IDF yang dapat dilihat pada Gambar 5.

```
# --- EKSTRAKSI FITUR: TF-IDF ---
tfidf_vectorizer = TfidfVectorizer(max_features = 400, ngram_range=(1, 2))
X_tfidf = tfidf_vectorizer.fit_transform(X_text)
```

Gambar 5. Kode Ekstraksi Fitur TF-IDF

c. *Bag-of-Words* (BoW)

Ekstraksi fitur dengan BoW juga menggunakan modul *scikit-learn*, yakni CountVectorizer. Cara kerjanya adalah dengan menghitung jumlah kemunculan kata di seluruh dokumen. Sama seperti ekstraksi fitur TF-IDF, BoW juga menggunakan jumlah maksimal fitur sebanyak 400 fitur dan *ngram*, yakni *unigram* dan *bigram*. Berikut tampilan kode dari ekstraksi fitur BoW yang dapat dilihat pada Gambar 6.

```
# --- EKSTRAKSI FITUR: BOW ---
bow_vectorizer = CountVectorizer(max_features = 400, ngram_range=(1, 2))
X_bow = bow_vectorizer.fit_transform(X_text)
```

Gambar 6. Kode Ektraksi Fitur BoW

3.6. Model *Multinomial Naïve Bayes*

Model algoritma *Multinomial Naïve Bayes* dibangun menggunakan modul *scikit-learn* dari Python, yakni MultinomialNB. Data yang digunakan untuk pemodelan algoritma *Multinomial Naïve Bayes* adalah data latih yang sudah melewati tahap ekstraksi fitur TF-IDF dan BoW. Berikut tampilan kode dari model *Multinomial Naïve Bayes* yang dapat dilihat pada Gambar 7.

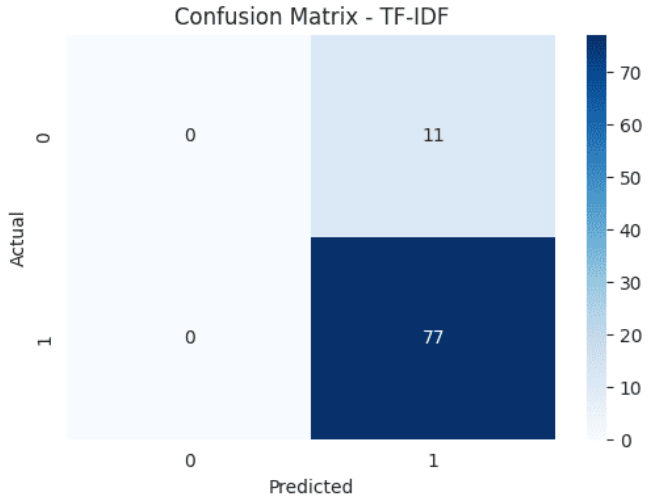
```
# Model MultinomialNB
model = MultinomialNB()
model.fit(X_train, y_train)
```

Gambar 7. Kode Program MultinomialNB

3.7. Evaluasi Model

Setelah dilakukan pemodelan algoritma klasifikasi, yakni *Multinomial Naïve Bayes* dengan menggunakan data latih, selanjutnya model akan dilakukan pengujian dan evaluasi dengan menggunakan data uji. Berikut hasil evaluasi model dengan menggunakan ekstraksi fitur TF-IDF yang divisualisasikan ke dalam bentuk *confussion matrix* yang dapat dilihat pada Gambar 8 dan Gambar 9.

p-ISSN: 2986-3929
e-ISSN: 3032-1948



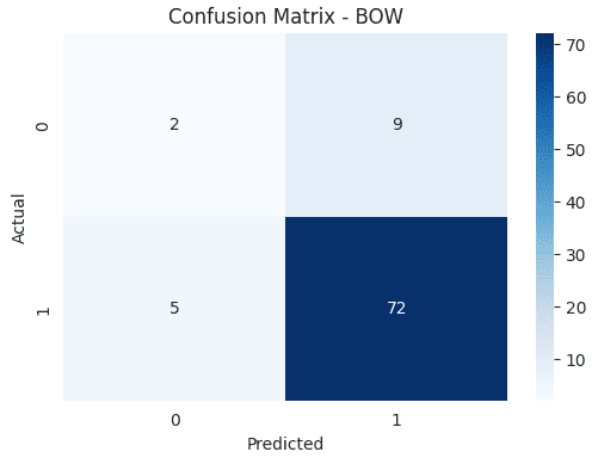
Gambar 8. *Confussion Matrix* pada Ekstraksi Fitur TF-IDF

Hasil *confussion matrix* seperti yang ditunjukkan Gambar 8, dapat disimpulkan bahwa model *Multinomial Naïve Bayes* berhasil memprediksi secara benar data bersentimen positif sebanyak 77 data. Namun, model juga gagal memprediksi data yang seharusnya bersentimen negatif menjadi positif (*False Positive*) sebanyak 11 data. Berikut beberapa kasus salah klasifikasi yang terdapat pada model *Multinomial Naïve Bayes* dengan ekstraksi fitur TF-IDF yang dapat dilihat pada Tabel 3.

Tabel 3. Kasus Salah Klasifiikasi Metode TF-IDF

content	Actual_label	Predict_label
aplikasi bagus buat candaan	negative	positive
baru saja saya hapus aplikasi ini dari HP saya karena tidak bisa membuat gambar yang saya minta	negative	positive
kesalahan fatal dari Gemini yaitu tidak bisa undo/menghapus/mengulang pesan yang sudah lewat beberapa kali,sedangkan chatgpt bisa,itu saja	negative	positive
tidak semudah dulu, sekarang minta buat prompt dari gambar susah, edit prompt susah, lebih tidak mampu memahami teks dari sebelumnya	negative	positive
tolol buat prompt aja malah kaya kena batasan umur!	negative	positive

p-ISSN: 2986-3929
e-ISSN: 3032-1948



Gambar 9. *Confussion Matrix* pada Ekstraksi Fitur BoW

Hasil *confussion matrix* seperti yang ditunjukkan Gambar 9, dapat disimpulkan bahwa dengan ekstraksi fitur BoW model dapat memprediksi secara benar data bersentimen positif sebanyak 72 data, dan data bersentimen negatif sebanyak 2 data. Namun, model juga gagal memprediksi data seharunya bersentimen negatif menjadi positif (*False Positive*) sebanyak 9 data, dan data bersentimen positif menjadi negatif (*False Negative*) sebanyak 5 data. Berikut beberapa kasus salah klasifikasi yang terdapat pada model *Multinomial Naïve Bayes* dengan ekstraksi fitur BoW yang dapat dilihat pada Tabel 4.

Tabel 4. Kasus Salah Klasifiikasi Metode BoW

content	Actual_label	Predict_label
AI nya sangat bagus, responnya detail, jelas,kontekstual,minusnya seringkali terkena limit berkali-kali pada sebuah percakapan(tapi bisa bikin prompt di percakapan baru) mungkin karena prompt yang detail sempe harus nunggu sekitar 24jam buat bisa dipakai lagi, memangsih ini karena beban server yang terlalu tinggi karena prompt detail dan AI harus menganalisis history disebuah percakapan agar respon sesuai konteks dan relavan, AI ini bagus tapi hal yang mengganggu hanyalah limit	positive	negative
baru saja saya hapus aplikasi ini dari HP saya karena tidak bisa membuat gambar yang saya minta	negative	positive
kadang benar kadang ngawur gimana si nyuruh buat ini malah buat itu. di update lagi lah ai nya	positive	negative
aplikasi bagus buat candaan	negative	positive
Syukurlah masalah lag ketika ganti jendela layar HP udah diatasi. Tapi muncul masalah baru sekarang, setelah update dan opsi "penalaran" dipecah menjadi "penalaran" dan "pro", setiap kali menggunakan yang perminataan seperti buat dokumen di canvas, itu selalu "eror" atau "gagal" padahal hasilnya di layar udah jadi dan BAGUS, tapi pas mau finishing, tiba-tiba gagal dan eror, akhirnya KOSONG ga ada apa-apa yang muncul, kayak sia-sia nunggu, padahal sempat liat sekilas hasilnya, sayang sekali.	positive	negative

Berdasarkan kedua *confussion matrix* di atas, maka dapat didapatkan nilai perbandingan *accuracy, precision, recall,* dan *F1-score* sebagai berikut.

Tabel 5. Perbandingan evaluasi TF-IDF dan BoW

	Accuracy	Precision	Recall	F1-score
TF-IDF	88%	88%	100%	93%
BoW	84%	89%	94%	91%

Berdasarkan hasil perbandingan pada Tabel 5, terlihat bahwa metode ekstraksi fitur TF-IDF secara keseluruhan memberikan performa yang lebih baik dibandingkan metode BoW. Keunggulan ini disebabkan oleh kemampuan TF-IDF dalam memberikan bobot yang lebih proporsional terhadap kata – kata yang lebih spesifik.

Dalam *dataset* ulasan aplikasi Gemini AI ini, sering kali muncul kata – kata umum yang memiliki frekuensi tinggi namun tidak membawa informasi sentimen yang kuat. Pada metode BoW, kata – kata yang umum ini akan mendapatkan bobot nilai yang tinggi karena frekuensi kemunculannya yang tinggi. Sebaliknya metode TF-IDF akan memberikan bobot nilai yang rendah pada kata – kata umum ini.

4. Kesimpulan

Berdasarkan penelitian yang dilakukan dengan membandingkan dua ekstraksi fitur, yakni TF-IDF dan BoW pada data ulasan aplikasi Gemini AI dengan model algoritma *Multinomial Naïve Bayes* didapatkan TF-IDF lebih baik daripada BoW. Model yang menggunakan ekstraksi fitur TF-IDF memberikan hasil *accuracy* sebesar 88%, *precision* sebesar 88%, *recall* sebesar 100%, dan *F1-score* sebesar 93% dibandingkan dengan ekstraksi fitur BoW dengan nilai *accuracy, precision, recall,* dan *F1-score* secara berurut sebesar 84%, 89%, 94%, dan 91%. Dari penelitian ini dapat lebih ditingkatkan dengan mencoba dengan metode pelabelan data yang lain, menerapkan seleksi fitur, dan dataset yang lebih besar agar fitur maksimal saat ekstraksi fitur lebih banyak.

Daftar Pustaka

[1] I. D. Kurniati *et al.*, *Buku Ajar Text Mining*. 2015.
[2] S. Ratnaswari, N. C. Wibowo, D. Satria, and Y. Kartika, “ANALISIS SENTIMEN MENGGUNAKAN METODE LEXICON-BASED DAN SUPPORT VECTOR MACHINE PADA PRESIDEN DAN WAKIL PRESIDEN INDONESIA,” vol. 13, no. 1, pp. 362–368, 2025.
[3] P. Widyantara *et al.*, “Analisis Sentimen pada Teks Berbahasa Bali Menggunakan Metode Multinomial Naive Bayes dengan TF-IDF dan BoW,” *Jnatia*, vol. 2, no. 1, pp. 37–46, 2023.
[4] N. Fibriyanti Arminda, N. Sulistiyowati, and T. Nur Padilah, “Implementasi Algoritma Multinomial Naive Bayes Pada Analisis Sentimen Terhadap Ulasan Pengguna Aplikasi Brimo,” *JATI (Jurnal Mhs. Tek. Inform.*, vol. 7, no. 3, pp. 1817–1822, 2023, doi: 10.36040/jati.v7i3.7012.
[5] S. A. H. Bahtiar, C. K. Dewa, and A. Luthfi, “Comparison of Naïve Bayes and Logistic Regression in Sentiment Analysis on Marketplace Reviews Using Rating-Based Labeling,” *J. Inf. Syst. Informatics*, vol. 5, no. 3, pp. 915–927, 2023, doi: 10.51519/journalisi.v5i3.539.
[6] K. Yuni and I. G. Santi, “Analisis Sentimen Ulasan Traveloka Menggunakan Metode Naïve Bayes Classifier dan Information Gain,” vol. 3, no. November, pp. 63–70, 2024.
[7] S. A. Nugraha, “PENERAPAN LEXICON BASED UNTUK ANALISIS SENTIMEN MASYARAKAT INDONESIA TERHADAP DANANTARA,” vol. 9, no. 3, pp. 4949–4957, 2025.
[8] F. Koto, “InSet Lexicon : Evaluation of a Word List for Indonesian Sentiment Analysis in Microblogs,” pp. 391–394, 2017.
[9] K. Tri Putra, M. Amin Hariyadi, and C. Crysdian, “Perbandingan Feature Extraction Tf-Idf Dan Bow Untuk Analisis Sentimen Berbasis Svm,” *J. Cahaya MAndalika*, p. 1449, 2023.
[10] Yuyun, Nurul Hidayah, and Supriadi Sahibu, “Algoritma Multinomial Naïve Bayes Untuk Klasifikasi Sentimen Pemerintah Terhadap Penanganan Covid-19 Menggunakan Data Twitter,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 4, pp. 820–826, 2021,

doi: 10.29207/resti.v5i4.3146.
[11] S. Anggina, N. Y. Setiawan, and F. A. Bachtiar, “Analisis Ulasan Pelanggan Menggunakan Multinomial Naïve Bayes Classifier dengan Lexicon-Based dan TF-IDF Pada Formaggio Coffee and Resto,” *is Best Account. Inf. Syst. Inf. Technol. Bus. Enterp. this is link OJS us*, vol. 7, no. 1, pp. 76–90, 2022, doi: 10.34010/aisthebest.v7i1.7072.
[12] S. Pierre, “A Comprehensive Guide to Scikit-Learn (Sklearn).” [Online]. Available: https://builtin.com/machine-learning/scikit-learn-guide

p-ISSN: 2986-3929
e-ISSN: 3032-1948

Halaman ini sengaja dibiarkan kosong