

Deteksi Sentimen Komentar Aplikasi Gobis Suroboyo dengan Metode Naive Bayes dan Metode Regresi Logistik

Shifa Elmaliyasari¹, Muhammad Arsyad Alzam², Nanda Aulia Pratiwi³, Shindi Shella May Wara⁴, Kartika Maulida Hindrayani⁵

^{1,2,3,4,5}Program Studi Sains Data, Universitas Pembangunan Nasional “Veteran” Jawa Timur, Indonesia

Email: ¹23083010022@student.upnjatim.ac.id, ²23083010082@student.upnjatim.ac.id, ³23083010011@student.upnjatim.ac.id,

⁴shindi.shella.fasilkom@upnjatim.ac.id, ⁵kartika.maulida.ds@upnjatim.ac.id

ABSTRAK

Penelitian ini membahas analisis sentimen terhadap komentar pengguna aplikasi Gobis Suroboyo menggunakan algoritma Naive Bayes dan Regresi Logistik. Data diperoleh melalui metode *web scraping* dari Google Play Store, dengan total 1.015 komentar yang kemudian melalui proses pra-pemrosesan teks seperti pembersihan data, *case folding*, *stemming*, normalisasi, *filtering*, *tokenizing*, dan pemilihan fitur menggunakan TF-IDF. Label sentimen ditentukan berdasarkan rating pengguna, dengan rating di atas 3 sebagai positif dan 3 ke bawah sebagai negatif. Hasil penelitian menunjukkan bahwa algoritma Naive Bayes lebih baik dalam mengklasifikasikan sentimen positif dengan *precision* sebesar 81% dan *f1-score* sebesar 77%, sedangkan Regresi Logistik unggul pada sentimen negatif dengan *precision* sebesar 82% dan *f1-score* sebesar 82%. Visualisasi *WordCloud* memperlihatkan kata-kata dominan seperti “aplikasi”, “bagus”, dan “halte” yang mencerminkan perhatian pengguna terhadap fitur aplikasi dan layanan transportasi. Temuan ini menunjukkan bahwa kedua algoritma memiliki performa yang kompetitif dan dapat diandalkan untuk mengevaluasi opini publik terhadap layanan digital berbasis komentar. Penelitian ini diharapkan dapat menjadi acuan bagi pengembang aplikasi dan pemerintah daerah dalam meningkatkan kualitas layanan publik digital.

Kata Kunci: Analisis Sentimen, Naïve Bayes, Regresi Logistik, Gobis Suroboyo

ABSTRACT

This research discusses sentiment analysis of user comments on the Gobis Suroboyo application using the Naive Bayes algorithm and Logistic Regression. Data was obtained through web scraping method from Google Play Store, with a total of 1,015 comments which then went through text pre-processing such as data cleaning, case folding, stemming, normalisation, filtering, tokenizing, and feature selection using TF-IDF. Sentiment labels were determined based on user ratings, with ratings above 3 as positive and 3 and below as negative. The results show that the Naive Bayes algorithm is better at classifying positive sentiment with a precision of 81% and f1-score of 77%, while Logistic Regression excels at negative sentiment with a precision of 82% and f1-score of 82%. The WordCloud visualisation shows dominant words such as “app”, “good”, and “bus stop” that reflect users attention to the app features and transportation services. The findings show that both algorithms have competitive and reliable performance for evaluating public opinion on comment-based digital services. This research is expected to be a reference for app developers and local governments in improving the quality of digital public services.

Keywords: Sentiment Analysis, Naïve Bayes, Logistic Regression, Gobis Suroboyo

Penulis Korespondensi:

Shifa Elmaliyasari

Email: 23083010022@student.upnjatim.ac.id

Article Info

Diterima: 9 Juni 2025

Direvisi: 12 Juni 2025

Disetujui: 16 Juni 2025

This is an open access article under the [CC BY](https://creativecommons.org/licenses/by/4.0/) license.



1. PENDAHULUAN

Dalam era digital saat ini, layanan publik semakin banyak mengadopsi platform digital sebagai sarana utama untuk berinteraksi dengan masyarakat. Salah satu contohnya adalah aplikasi Gobis Suroboyo, yang dirancang untuk memberikan layanan informasi transportasi berbasis *real-time* kepada warga Surabaya. Aplikasi ini memungkinkan pengguna untuk mengakses jadwal keberangkatan, rute perjalanan, serta lokasi armada transportasi umum seperti bus secara lebih akurat. Namun demikian, efektivitas suatu aplikasi tidak hanya dapat diukur dari fungsionalitasnya saja, melainkan juga dari umpan balik yang diberikan oleh pengguna melalui platform seperti Google Play Store. Kolom komentar pada Google Play Store menjadi salah satu media penting untuk menangkap pengalaman pengguna secara langsung, termasuk permasalahan teknis, kepuasan layanan, maupun saran perbaikan. Komentar pengguna mencerminkan pengalaman nyata dan dapat dijadikan dasar evaluasi kualitas layanan secara objektif maupun pengambilan keputusan berbasis data oleh pengelola layanan.

Analisis sentimen merupakan metode dalam *text mining* yang digunakan untuk mengevaluasi opini pengguna berdasarkan teks ulasan [1]. Teknik ini memungkinkan sistem untuk mengklasifikasikan opini menjadi kategori positif dan negatif. Dalam konteks aplikasi layanan publik, pendekatan ini sangat penting karena dapat mengungkap kepuasan atau keluhan pengguna secara masif dan cepat tanpa harus survei secara langsung yang dapat memakan waktu dan biaya. Selain itu, analisis sentimen juga dapat digunakan untuk memantau persepsi publik secara berkala, mendeteksi isu-isu krusial yang sedang berkembang, serta merumuskan strategi peningkatan kualitas layanan berdasarkan data yang bersifat empiris.

Sejumlah penelitian sebelumnya telah dilakukan untuk mengkaji efektivitas algoritma klasifikasi dalam analisis sentimen pada berbagai jenis aplikasi. Nugraha dan Gustian pada penelitiannya menggunakan algoritma Naive Bayes Classifier untuk mengklasifikasikan komentar pengguna dari tiga aplikasi transportasi online (Gojek, Grab, dan Maxim), dan menunjukkan bahwa aplikasi Maxim memperoleh akurasi tertinggi yaitu 93% [2]. Dalam studi yang berbeda, Nugroho, Susilo, dan Retnoningsih membandingkan performa beberapa algoritma termasuk Naive Bayes, Random Forest, dan Regresi Logistik dalam menganalisis ulasan pengguna aplikasi "Access by KAI", dan menyimpulkan bahwa Regresi Logistik menghasilkan akurasi terbaik sebesar 84% [3]. Penelitian lain oleh Dinda membandingkan kinerja algoritma Naive Bayes dan Regresi Logistik dalam konteks prediksi penyakit jantung. Hasilnya menunjukkan bahwa Naive Bayes sedikit lebih unggul dengan akurasi 85,50% dibanding Regresi Logistik yang mencapai 83,19% [4]. Adapun Insan et al. yang menganalisis sentimen pengguna terhadap aplikasi BRImo dan menunjukkan bahwa algoritma Naive Bayes mampu mengklasifikasikan ulasan dengan akurasi sebesar 84,52%, yang memperkuat efektivitas metode ini dalam mengolah data teks dari ulasan pengguna di Google Play Store [5].

Temuan-temuan ini menunjukkan bahwa baik Naive Bayes maupun Regresi Logistik memiliki performa yang kompetitif dalam berbagai konteks klasifikasi teks, namun belum ada studi yang secara khusus membandingkannya pada aplikasi layanan publik lokal seperti Gobis Suroboyo. Padahal, karakteristik teks ulasan yang pendek, informal, dan dalam bahasa Indonesia memiliki tantangan tersendiri dalam proses klasifikasi otomatis. Penelitian ini bertujuan untuk mengisi kekosongan tersebut dengan membandingkan performa dua algoritma klasifikasi, yakni Naive Bayes dan Regresi Logistik, dalam menganalisis sentimen komentar pengguna aplikasi Gobis Suroboyo di Google Play Store. Penelitian ini diharapkan dapat memberikan insight kepada pengembang aplikasi dan pemerintah daerah mengenai persepsi publik terhadap layanan digital mereka.

2. METODE PENELITIAN

2.1 Teknik Pengambilan Data

2.1.1 Sumber Data

Dengan adanya media sosial, orang dapat secara bebas menyatakan pendapat mereka. Hal ini merupakan bentuk partisipasi pasif yang bermanfaat dalam perencanaan kota, karena membantu perancang dan penyedia layanan kota membuat keputusan yang lebih baik. Salah satu contohnya adalah transportasi umum perkotaan, di mana pengguna terus memberikan umpan balik, baik komentar positif maupun kritik [6]. Data yang digunakan dalam penelitian ini berasal dari ulasan pengguna aplikasi GoBis Suroboyo di Google Play Store. Platform ini dipilih sebagai sumber utama karena dua alasan. Pertama, Google Play Store memberikan akses publik ke berbagai ulasan pengguna, termasuk rating dan teks komentar. Kedua, data yang diperoleh secara *real-time* memungkinkan untuk menggambarkan pengalaman pengguna saat ini secara akurat. Penelitian ini menggunakan berbagai kriteria untuk menentukan sumber data.

2.1.2 Metode Pengumpulan Data

Pengambilan data dilakukan dengan metode *web scraping*, yang secara otomatis mengekstrak data ulasan dari Google Play Store. *Web Scraping* adalah proses mengekstrak data dari internet dengan cara apa pun. Ini dapat mengubah data tidak terstruktur menjadi data terstruktur yang dapat disimpan dan divalidasi di *database*. Metode ini dipilih karena mampu mengumpulkan data dalam jumlah besar dengan efisiensi sambil mengurangi kesalahan manusia yang mungkin terjadi saat mengumpulkan data secara manual. Beberapa tahapan penting diperlukan dalam proses ini, termasuk penyaringan ulasan berdasarkan kriteria khusus dan penerapan filter. Sebelum dianalisis, data melalui proses validasi untuk memastikan bahwa data sudah lengkap dan akurat. Meskipun Metode ini memiliki beberapa kelemahan, seperti keterbatasan untuk mengakses Google Play Store dan data historis yang terlalu lama.

2.2 Pra-pemrosesan Data

Tahap pra-pemrosesan data merupakan langkah yang sangat penting dalam melakukan analisis sentimen berbasis *text mining*, karena data mentah dari Google Play Store umumnya tidak terstruktur. Langkah-langkah yang dilakukan dalam pra-pemrosesan data adalah sebagai berikut :

1. Proses *Cleaning*
Proses *Cleaning* merupakan tahapan yang bertujuan untuk menghapus simbol-simbol atau karakter khusus, username, angka, serta tautan yang tidak relevan untuk proses analisis.
2. Proses *Case Folding*
Proses *Case Folding* mengubah seluruh huruf pada komentar menjadi huruf kecil atau *lowercase* untuk menyamakan format teks. Daftar kata yang diubah seperti “Aplikasi Ini bAgus” menjadi “aplikasi ini bagus”.
3. Proses *Stemming*
Proses *Stemming* bertujuan untuk mengubah kata berimbuhan menjadi bentuk baku. Anggap kata “menunggu” menjadi “tunggu”.
4. Proses Normalisasi Kata
Proses Normalisasi Kata akan mengubah kata-kata singkatan atau campuran bahasa menjadi kata dalam bentuk baku, seperti “tdk” menjadi “tidak”. Tahapan ini dilakukan agar model mengenali makna kata dengan benar.
5. Proses *Filtering*
Proses *Filtering* dilakukan dengan menghapus kata sambung dan tidak mengandung informasi yang spesifik, seperti “dan”, “dengan”, ataupun “yang”. *Filtering* dilakukan dengan menggunakan daftar *stopwords* yang telah didefinisikan.
6. Proses *Tokenizing*
Proses *Tokenizing* merupakan proses pemecahan kalimat menjadi unit-unit kata atau token. Sebagai contoh, kalimat “Aplikasi ini sangat berguna” akan diubah menjadi ["aplikasi", "ini", "sangat", "berguna"].
7. Proses Pemilihan Fitur
Proses Pemilihan Fitur bertujuan untuk menentukan kata-kata yang paling relevan agar dapat digunakan dalam klasifikasi sentimen. Menggunakan metode *Term Frequency-Inverse Document Frequency (TF-IDF)* untuk membantu menonjolkan kata-kata yang bersifat unik dan bermakna dalam analisis.
8. Proses *Labeling*
Proses *Labeling* akan mengkategorikan komentar berdasarkan nilai rating yang diberikan pengguna. Jika rating lebih dari 3, maka komentar diklasifikasikan sebagai sentimen ‘positif’ sedangkan jika rating 3 ke bawah, maka dikategorikan sebagai ‘negatif’.

2.3 Analisis Sentimen dengan Naive Bayes

Metode Naive Bayes merupakan algoritma klasifikasi berbasis probabilistik yang banyak digunakan dalam analisis sentimen karena kesederhanaannya dan efektivitasnya dalam menangani data teks berdimensi tinggi. Algoritma ini menghitung probabilitas sebuah teks termasuk dalam kelas tertentu (misalnya sentimen positif atau negatif) dengan mengasumsikan independensi antar kata. Penelitian oleh Hilmi dan Irwiensyah membuktikan efektivitas metode ini dalam menganalisis ulasan pengguna TikTok, dengan akurasi 83,66% [7]. Sementara itu, Khofifah dan Rahayu berhasil mengelompokkan sentimen masyarakat terhadap destinasi wisata menggunakan pendekatan serupa [8]. Rumus dasar Naive Bayes:

$$P(C_i | X) \propto P(C_i) \cdot \prod_{j=1}^{\{n\}} P(x_j | C_i) \quad (1)$$

Keterangan:

- $P(C_i | X)$: Probabilitas bahwa X dokumen termasuk ke dalam kelas C_i
- $P(C_i)$: Probabilitas awal (prior) dari kelas C_i
- $P(x_j | C_i)$: Probabilitas kata ke- j muncul dalam dokumen yang termasuk kelas C_i
- X : Dokumen atau ulasan yang dianalisis, terdiri dari beberapa kata fitur
- x_j : Kata ke- j dalam dokumen
- n : Jumlah total kata (fitur) dalam dokumen
- $\prod_{j=1}^{\{n\}}$: Simbol perkalian terhadap semua probabilitas $P(x_j | C_i)$ dari $j = 1$ hingga n
- \propto : Simbol proporsionalitas yang menyatakan bahwa hasilnya sebanding

Metode ini digunakan untuk mengklasifikasikan komentar pengguna aplikasi Gobis Suroboyo menjadi sentimen positif dan negatif.

2.4 Analisis Sentimen dengan Regresi Logistik

Metode Regresi Logistik merupakan algoritma klasifikasi yang digunakan untuk memprediksi probabilitas suatu data termasuk ke dalam kelas tertentu, seperti sentimen positif atau negatif. Dalam penelitian ini, algoritma ini dipilih karena mampu

menangani data teks berdimensi tinggi secara efisien. Penelitian oleh Averina et al. menunjukkan bahwa Regresi Logistik dapat mengklasifikasikan sentimen ulasan film dengan akurasi 83% untuk dua kelas menggunakan fitur *CountVectorizer* dan teknik *dimensionality reduction* [9]. Temuan ini menjadi dasar pemilihan metode Regresi Logistik dalam penelitian ini untuk mengklasifikasikan komentar aplikasi Gobis Suroboyo. Secara matematis, fungsi logistik dalam klasifikasi biner didefinisikan sebagai:

$$P(y = 1 | x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n)}} \quad (2)$$

Keterangan:

- $P(y = 1 | x)$: Probabilitas bahwa input x termasuk dalam kelas 1 (misalnya: sentimen positif)
- e : Bilangan eksponensial natural (Euler), dengan nilai mendekati 2,718
- β_0 : Intersep atau konstanta model (bias)
- $\beta_1, \beta_2, \dots, \beta_n$: Koefisien regresi untuk masing-masing fitur
- x_1, x_2, \dots, x_n : Nilai fitur atau variabel independen dalam data teks
- n : Jumlah total fitur yang digunakan dalam model klasifikasi

3. HASIL DAN PEMBAHASAN

3.1. Data Scraping

Data scraping pada penelitian ini dilakukan pada tanggal 12 Mei 2025 menggunakan tools Google Collaboratory. Proses ini bertujuan untuk mengambil komentar atau ulasan pengguna terhadap aplikasi GoBis yang tersedia pada platform Google Play Store, dengan rentang skor atau rating dari 1 hingga 5. Berdasarkan data yang tersedia saat itu, terdapat total 2.604 komentar, tetapi hanya 1.015 komentar yang berhasil di *scraping* karena adanya keterbatasan teknis atau kebijakan akses data. Hasil *scraping* tersebut kemudian disimpan dalam bentuk file Excel (.csv) dan dilanjutkan ke tahap pra-pemrosesan data untuk analisis lebih lanjut.

Tabel 1. Hasil *scraping* data

Waktu	Komentar	Skor
0 5/1/2025 12:20	Aplikasi sebetulnya bagus untuk penggunaan transportasi umum dlm mengurangi pemakaian kendaraan pribadi, Akan tetapi tampilannya perlu diperbaiki. Karena sangat...	3
1 5/7/2025 0:53	Dulu udah install ini dan uninstall krn app ini sdh nggak berguna lagi semenjak ada mitra darat. saya kira itu...	1
2 4/28/2025 6:45	Dengan adanya aplikasi ini sangat menguntungkan bagi kita pengguna transfort dgn bus, bnyk arah...	5
...
1012 10/26/2024 4:37	sangat senang	5
1013 5/18/2022 0:02	Cakep bgt	5
1014 4/28/2019 9:39	👍👍👍	5

3.2. Data Pem-prosesan

Pra-pemrosesan merupakan langkah penting sebelum mengimplementasikan algoritma klasifikasi pada data. Tahap-tahap pada langkah pra-pemrosesan data adalah sebagai berikut:

a. *Cleaning*

Langkah awal yang dilakukan adalah membersihkan data dari karakter atau elemen yang tidak diperlukan seperti tanda baca, angka, atau simbol khusus. Bertujuan untuk menghilangkan elemen-elemen yang tidak relevan agar teks menjadi lebih bersih dan terstruktur untuk dianalisis.

Tabel 2. Proses *cleaning* data

Teks	Cleaning
Aplikasi sebetulnya bagus untuk penggunaan transportasi umum dlm mengurangi pemakaian kendaraan pribadi, Akan tetapi tampilannya perlu diperbaiki.	Aplikasi sebetulnya bagus untuk penggunaan transportasi umum dlm mengurangi pemakaian kendaraan pribadi Akan tetapi tampilannya perlu diperbaiki

b. *Case Folding*

Proses ini mengubah seluruh huruf dalam teks menjadi huruf kecil. Hal ini penting agar sistem tidak membedakan antara kata yang seharusnya sama akan tetapi ditulis dengan perbedaan kapitalisasi. Langkah ini bertujuan untuk menyamakan format teks dengan mengubah seluruh huruf menjadi huruf kecil demi konsistensi dalam pengolahan.

Tabel 3. Proses *case folding*

Teks	Case Folding
Aplikasi sebetulnya bagus untuk penggunaan transportasi umum dlm mengurangi pemakaian kendaraan pribadi Akan tetapi tampilannya perlu diperbaiki	aplikasi sebetulnya bagus untuk penggunaan transportasi umum dlm mengurangi pemakaian kendaraan pribadi akan tetapi tampilannya perlu diperbaiki

c. *Stemming*

Stemming dilakukan untuk mengembalikan setiap kata ke bentuk dasarnya. Hal ini membantu mengurangi jumlah variasi kata dan menyatukan kata-kata yang berasal dari akar yang sama.

Tabel 4. Proses *stemming*

Teks	Steeming
aplikasi sebetulnya bagus untuk penggunaan transportasi umum dlm mengurangi pemakaian kendaraan pribadi akan tetapi tampilannya perlu diperbaiki	aplikasi betul bagus untuk guna transportasi umum dlm kurang pakai kendara pribadi akan tetapi tampil perlu baik

d. Normalisasi Kata

Tahapan ini bertugas menyesuaikan kata tidak baku atau kata gaul menjadi bentuk yang standar dan sesuai dengan kamus.

Tabel 5. Proses normalisasi kata

Teks	Normalisasi Data
aplikasi betul bagus untuk guna transportasi umum dlm kurang pakai kendara pribadi akan tetapi tampil perlu baik	aplikasi betul bagus untuk guna transportasi umum dalam kurang pakai kendara pribadi akan tetapi tampil perlu baik

e. *Filtering*

Filtering digunakan untuk menyaring kata-kata yang dianggap tidak membawa informasi penting dalam analisis, seperti kata sambung atau kata umum.

Tabel 6. Proses *filtering* data

Teks	Filtering
aplikasi betul bagus untuk guna transportasi umum dalam kurang pakai kendara pribadi akan tetapi tampil perlu baik	aplikasi bagus transportasi pakai kendara pribadi tampil

f. *Tokenizing*

Tokenizing memecah teks menjadi bagian-bagian kecil seperti kata, frasa, atau kalimat. Proses ini penting untuk mengidentifikasi unit dasar dari teks yang akan dianalisis dalam tahapan selanjutnya.

Tabel 7. Proses *Tokenizing*

Teks	Tokenizing
aplikasi bagus transportasi pakai kendara pribadi tampil	'aplikasi', 'bagus', 'transportasi', 'pakai', 'kendara', 'pribadi', 'tampil'

g. Pemilihan Fitur

Hasil data yang dihasilkan pada tahapan sebelumnya akan dilakukan tahap pemilihan fitur menggunakan metode TF-IDF. Metode ini bertujuan mengubah kumpulan dokumen teks menjadi bentuk numerik agar dapat dianalisis oleh model Naive Bayes dan Regresi Logistik.

Tabel 8. Proses pemilihan fitur

Dokumen	aplikasi	bagus	bantu	...	tolong	transportasi	tunggu
0	0.154514	0.179958	0.000000	...	0.196603	0.555644	0.000000
1	0.373359	0.000000	0.000000	...	0.000000	0.671313	0.000000
2	0.513341	0.000000	0.000000	...	0.000000	0.000000	0.000000

...
1013	0.000000	0.000000	0.000000	...	0.000000	0.000000	0.000000
1014	0.000000	0.000000	0.000000	...	0.000000	0.000000	0.000000
1015	0.000000	0.000000	0.000000	...	0.000000	0.000000	0.000000

h. *Labeling*

Proses *labeling* untuk mengategorikan komentar berdasarkan *rating* yang diberikan pengguna. Komentar dengan *rating* di atas 3 diberi label sentimen positif, sedangkan *rating* 3 ke bawah diberi label negatif. Berikut ditampilkan tabel yang menunjukkan proporsi jumlah data berlabel positif dan negatif yang telah diperoleh.

Tabel 9. Proses *labeling*

Label	Jumlah	Proporsi(%)
Positif	555	54.7337
Negatif	460	45.2663

3.3. Evaluasi Model

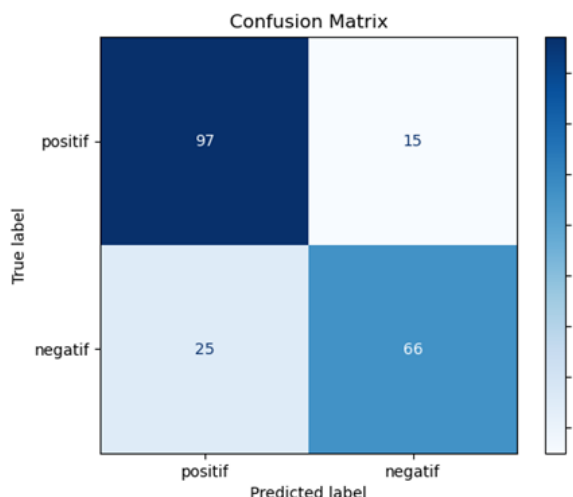
Evaluasi performa merupakan tahap penting untuk menilai seberapa baik model dalam mengelompokkan data ke dalam kelas yang sesuai. Evaluasi ini biasanya dilakukan dengan menggunakan metrik seperti akurasi, *precision*, *recall*, dan *f1-score*. Akurasi mengukur proporsi keseluruhan prediksi yang benar, *precision* menunjukkan ketepatan prediksi positif, *recall* mengukur kemampuan model dalam menangkap data positif secara benar, dan *f1-score* memberikan gambaran keseimbangan antara *precision* dan *recall*.

Tabel 10. Hasil evaluasi model

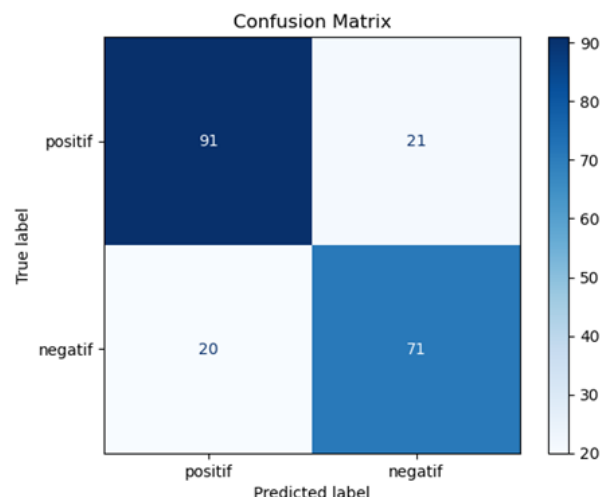
Modeling	Accuracy	Precision	Recall	F1-score
Naive Bayes	80.2%	Positif 81%	73%	77%
		Negatif 80%	87%	83%
Regresi Log	79.8%	Positif 77%	78%	78%
		Negatif 82%	81%	82%

Berdasarkan tabel yang menunjukkan bahwa Naive Bayes lebih unggul dalam memprediksi kelas positif dengan nilai *precision* sebesar 81% dibanding Regresi Logistik dengan nilai *precision* sebesar 77%, disisi lain Regresi Logistik lebih baik untuk memprediksi kelas negatif dengan nilai *precision* sebesar 82% dibanding Naive Bayes dengan nilai sebesar 80%. Nilai *recall* Naive Bayes lebih tinggi untuk kelas negatif sebesar 87%, sementara Regresi Logistik menunjukkan performa yang lebih seimbang sebesar 78% untuk kelas positif dan 81% untuk kelas negatif. *F1-score* Regresi Logistik juga lebih konsisten di kedua kelas, yakni sebesar 78% untuk kelas positif dan 82% untuk kelas negatif, sedangkan Naive Bayes unggul pada kelas negatif dengan nilai 83% dibanding dengan kelas positif yaitu 77%. Hal ini menunjukkan bahwa model Regresi Logistik memiliki performa yang lebih seimbang. Keseimbangan ini penting karena menunjukkan bahwa model tidak condong atau bias terhadap salah satu kelas saja. Model yang seimbang lebih andal karena mampu memprediksi berbagai sentimen secara konsisten, meski akurasi per kelas sedikit lebih rendah dibanding model yang bias ke satu kelas.

Untuk memahami performa model klasifikasi secara lebih mendalam, selain menggunakan matrik evaluasi seperti *precision*, *recall*, dan *f1-score*, analisis juga dilakukan melalui *confusion matrix*. *Confusion matrix* memberikan gambaran rinci mengenai jumlah prediksi benar dan salah untuk masing-masing kelas, termasuk *true positive*, *false positive*, *true negative*, dan *false negative*. Pendekatan ini penting untuk menilai efektivitas model, terutama ketika terdapat ketidakseimbangan kelas atau ketika kesalahan klasifikasi pada satu kelas memiliki dampak yang lebih besar.



Gambar 1. *Confusion matriks* Naive Bayes



Gambar 2. *Confusion matriks* Regresi Logistik

Berdasarkan *confusion matrix* dari kedua model klasifikasi, terlihat bahwa Naive Bayes menghasilkan 97 prediksi benar untuk kelas positif dan 66 untuk kelas negatif, dengan kesalahan masing-masing 15 dan 25. Di sisi lain, Regresi Logistik memprediksi 91 data positif dan 71 data negatif dengan benar, serta menghasilkan kesalahan klasifikasi sebanyak 21 dan 20. Naive Bayes memperoleh akurasi sebesar 80.2%, sedangkan Regresi Logistik menghasilkan akurasi sebesar 79.8%. Hal ini menunjukkan bahwa Naive Bayes lebih akurat dalam mendeteksi kelas positif, sementara Regresi Logistik sedikit lebih baik dalam mengenali kelas negatif. Meskipun selisih akurasinya kecil, perbedaan distribusi prediksi benar antar kelas ini bisa menjadi pertimbangan penting tergantung pada prioritas klasifikasi dalam konteks penggunaan model.

3.4. Visualisasi Data

Penelitian ini juga dilengkapi dengan visualisasi data dalam bentuk *WordCloud* untuk memberikan gambaran umum mengenai kata-kata yang paling sering muncul dalam kumpulan komentar pengguna. Visualisasi ini bertujuan untuk memudahkan identifikasi topik atau istilah yang dominan secara intuitif. Pada penelitian ini hasil dari *WordCloud* untuk seluruh data komentar aplikasi Gobis dapat dilihat pada gambar sebagai berikut:



Gambar 3. *WordCloud* seluruh komentar

Berdasarkan visualisasi *WordCloud* dari seluruh data komentar, terlihat bahwa kata-kata seperti “aplikasi”, “bus”, “bagus”, “tolong”, dan “halte” merupakan yang paling dominan muncul. Hal ini menunjukkan bahwa pengguna banyak memberikan perhatian terhadap aspek teknis dari aplikasi serta layanan transportasi yang disediakan. Selain itu, kemunculan kata seperti “bantu”, “rute”, “menunggu”, dan “surabaya” juga mengindikasikan adanya interaksi pengguna terkait informasi rute, lokasi, dan pengalaman penggunaan layanan di kota tersebut. Dominasi kata-kata tersebut mencerminkan fokus utama komentar, baik berupa pujian, masukan, maupun keluhan terhadap layanan transportasi yang sedang dievaluasi. Berikut ini wordcloud dari data komentar positif dan negatif mengenai data komentar aplikasi Gobis yang dapat dilihat pada gambar sebagai berikut :



Gambar 4. WordCloud komentar positif



Gambar 5. WordCloud komentar negatif

Berdasarkan hasil visualisasi *WordCloud* dari komentar dengan sentimen positif, terlihat bahwa kata-kata seperti “bagus,” “bantu,” “terima kasih,” “surabaya,” “nyaman,” dan “bersih” mendominasi. Hal ini mencerminkan bahwa pengguna merasa puas terhadap layanan, terutama dalam hal kenyamanan, kebersihan, serta kemudahan penggunaan aplikasi. Kemunculan kata “terima kasih” dan “bantu” menunjukkan adanya apresiasi dari pengguna terhadap manfaat layanan yang mereka rasakan. Sementara itu, pada *WordCloud* komentar dengan sentimen negatif, kata-kata seperti “tolong,” “masuk,” “error,” “penuh,” “daftar,” dan “jadwal” tampak lebih menonjol. Ini mengindikasikan adanya berbagai keluhan pengguna terkait kendala teknis maupun operasional, seperti kesulitan login, aplikasi error, serta masalah kapasitas dan ketepatan jadwal. Jika dibandingkan, perbedaan kosakata yang dominan antara komentar positif dan negatif dapat memberikan gambaran menyeluruh mengenai aspek layanan yang perlu dipertahankan maupun yang perlu ditingkatkan.

Hasil tersebut menunjukkan bagaimana model klasifikasi memproses dan memahami opini pengguna secara efektif. Sebagai contoh, komentar “Dulu udah install ini dan uninstall... Sungguh, sebuah kecanggihan gimmick yang memper ruwet keadaan saja” berhasil diklasifikasikan sebagai negatif, yang mencerminkan ketidakpuasan pengguna terhadap perubahan kebijakan aplikasi. Kata-kata seperti “hapus”, “gimmick”, “memper”, dan “ruwet” turut memperkuat identifikasi sentimen tersebut. Sebaliknya, komentar “Dengan adanya aplikasi ini sangat menguntungkan... keliling kota Sby mengasyikkan dan murah meriah” diklasifikasikan sebagai positif, ditandai dengan kata-kata seperti “untung”, “asyik”, dan “murah”. Temuan ini memperlihatkan bahwa model tidak hanya mampu bekerja pada data yang telah diproses, tetapi juga memahami konteks opini dalam komentar asli secara akurat.

4. KESIMPULAN

Penelitian ini bertujuan untuk mengevaluasi dan membandingkan performa algoritma Naive Bayes dan Regresi Logistik dalam mengklasifikasikan sentimen komentar pengguna terhadap aplikasi Gobis Suroboyo yang diambil dari Google Play Store. Berdasarkan hasil evaluasi, dapat disimpulkan bahwa algoritma Naive Bayes memiliki keunggulan dalam mengidentifikasi komentar dengan sentimen positif dengan nilai *precision* sebesar 81% dan *f1-score* sebesar 77%, sementara Regresi Logistik menunjukkan performa yang lebih seimbang untuk kedua kelas dengan *f1-score* sebesar 78% untuk sentimen positif dan 82% untuk sentimen negatif. Dari segi akurasi keseluruhan, kedua model memiliki perbedaan yang sangat tipis, yaitu 80.2% untuk Naive Bayes dan 79.8% untuk Regresi Logistik.

Analisis *confusion matrix* memperkuat temuan ini, di mana Naive Bayes lebih tepat dalam mengklasifikasikan komentar positif, sementara Regresi Logistik lebih tepat pada komentar negatif. Bukti klasifikasi pada komentar asli menunjukkan bahwa model mampu mengidentifikasi sentimen secara akurat dan memahami konteks opini pengguna, sehingga berdampak langsung pada kualitas interpretasi data. Visualisasi dalam bentuk *WordCloud* juga mengungkapkan bahwa kata-kata seperti “aplikasi”, “bus”, “bagus”, “halte”, dan “tolong” mendominasi komentar pengguna, mencerminkan fokus utama terhadap aspek teknis dan kualitas layanan transportasi. Penelitian ini memberikan kontribusi terhadap pengembangan sistem evaluasi layanan publik berbasis data ulasan, serta memberikan masukan yang berharga bagi pengembang aplikasi dan pemerintah daerah dalam meningkatkan layanan. Penelitian ini dapat dikembangkan lebih lanjut dengan menerapkan teknik *ensemble learning*, memperluas cakupan data, serta menambahkan klasifikasi multikategori sentimen atau analisis emosi untuk memperkaya interpretasi opini publik secara *real-time*.

REFERENSI

- [1] S. Rahayu, Y. MZ, J. E. Bororing, dan R. Hadiyat, “Implementasi Metode K-Nearest Neighbor (K-NN) untuk Analisis Sentimen Kepuasan Pengguna Aplikasi Teknologi Finansial FLIP,” *Edumatic: Jurnal Pendidikan Informatika*, vol. 6, no. 1, hlm. 98–106, Jun 2022, doi: 10.29408/edumatic.v6i1.5433.
- [2] D. Nugraha dan D. Gustian, “Analisis Sentimen Penggunaan Aplikasi Transportasi Online Pada Ulasan Google Play Store dengan Metode Naive Bayes Classifier,” 2024.
- [3] M. A. S. Nugroho, D. Susilo, dan D. Retnoningsih, “ANALISIS SENTIMEN ULASAN APLIKASI “ACCESS BY KAI” MENGGUNAKAN ALGORITMA MACHINE LEARNING,” *Jurnal Teknik Informasi dan Komputer (Tekinkom)*, vol. 7, no. 2, hlm. 820, Des 2024, doi: 10.37600/tekinkom.v7i2.1854.

- [4] D. A. Nugroho, "Perbandingan Kinerja Algoritma Logistic Regression dan Naïve Bayes dalam Prediksi Penyakit Jantung," Laporan Skripsi, Fakultas Ilmu Komputer, Universitas Mercu Buana, Jakarta, 2023.
- [5] M. Khoirul, U. Hayati, dan O. Nurdyawan, "ANALISIS SENTIMEN APLIKASI BRIMO PADA ULASAN PENGGUNA DI GOOGLE PLAY MENGGUNAKAN ALGORITMA NAIVE BAYES," 2023.
- [6] O. Lock dan C. Pettit, "Social media as passive geo-participation in transportation planning—how effective are topic modeling & sentiment analysis in comparison with citizen surveys?," *Geo-Spatial Information Science*, vol. 23, no. 4, hlm. 275–292, 2020, doi: 10.1080/10095020.2020.1815596.
- [7] N. F. Hilmi dan F. Irwiensyah, "Analisis Sentimen Terhadap Aplikasi Tiktok Dari Ulasan Pada Google Playstore Menggunakan Metode Naïve Bayes," *SMATIKA JURNAL*, vol. 14, no. 01, hlm. 146–156, Jul 2024, doi: 10.32664/smatika.v14i01.1210.
- [8] W. Khofifah, D. N. Rahayu, dan A. M. Yusuf, "Analisis Sentimen Menggunakan Naive Bayes Untuk Melihat Review Masyarakat Terhadap Tempat Wisata Pantai Di Kabupaten Karawang Pada Ulasan Google Maps," *Jurnal Interkom: Jurnal Publikasi Ilmiah Bidang Teknologi Informasi dan Komunikasi*, vol. 16, no. 4, hlm. 28–38, Jan 2022, doi: 10.35969/interkom.v16i4.192.
- [9] A. Averina, H. Hadi, dan J. Siswantoro, "Analisis Sentimen Multi-Kelas Untuk Film Berbasis Teks Ulasan Menggunakan Model Regresi Logistik," *Teknika*, vol. 11, no. 2, hlm. 123–128, Jun 2022, doi: 10.34148/teknika.v11i2.461.
- [10] S. S. M. Wara, A. F. Adziima, M. Nasrudin and A. R. Pratama, "Predictive Analysis of Government Application Comment on Playstore with Clustered Support Vector Machine," *2024 IEEE 10th Information Technology International Seminar (ITIS)*, Surabaya, Indonesia, 2024, pp. 84-88, doi: 10.1109/ITIS64716.2024.10845453.
- [11] B. Ramadhani dan R. R. Suryono, "Komparasi Algoritma Naïve Bayes dan Logistic Regression Untuk Analisis Sentimen Metaverse," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 8, no. 2, hlm. 714, Apr 2024, doi: 10.30865/mib.v8i2.7458.
- [12] C. Zai dan A. Rahman Isnain, "Komparasi Algoritma Naïve Bayes dan Support Vector Machine (SVM) pada Analisis Sentimen Capcut," vol. 9, no. 1, hlm. 2024.
- [13] H. Parasian Doloksaribu dan Y. T. Samuel, "KOMPARASI ALGORITMA DATA MINING UNTUK ANALISIS SENTIMEN APLIKASI PEDULILINDUNGI," vol. 16, no. 1, 2022, doi: 10.47111/JTI.
- [14] Friska Aditia Indriyani, Ahmad Fauzi, dan Sutan Faisal, "Analisis sentimen aplikasi tiktok menggunakan algoritma naïve bayes dan support vector machine," *TEKNOSAINS: Jurnal Sains, Teknologi dan Informatika*, vol. 10, no. 2, hlm. 176–184, Jul 2023, doi: 10.37373/tekno.v10i2.419.
- [15] J. Homepage, N. C. Agustina, D. Herlina Citra, W. Purnama, C. Nisa, dan A. Rozi Kurnia, "MALCOM: Indonesian Journal of Machine Learning and Computer Science The Implementation of Naïve Bayes Algorithm for Sentiment Analysis of Shopee Reviews on Google Play Store Implementasi Algoritma Naive Bayes untuk Analisis Sentimen Ulasan Shopee pada Google Play Store," vol. 2, hlm. 47–54, 2022.