

Evaluasi Aplikasi Raileo Melalui Analisis Sentimen Ulasan Playstore Dengan Metode Naive Bayes

Haris Junianto^{1*}, Primandani Arsi², Bagus Adhi Kusuma³, Dhanar Intan Surya Saputra⁴

^{1,2,3,4}Informatika, Fakultas Ilmu Komputer, Universitas Amikom Purwokerto, Indonesia
Jl. Letjend. Pol. Soemarto, Watumas, Purwokerto, Jawa Tengah

Email: : ^{1*}21sa1002@mhs.amikompurwokerto.ac.id, ²ukhti.prima@amikompurwokerto.ac.id,
³bagus@amikompurwokerto.ac.id, ⁴dhanarsaputra@amikompurwokerto.ac.id

Received : January, 2024

Accepted : January, 2024

Published : April, 2024

Abstract

The Raileo application is a staffing platform owned by PT. KAI, functions as a personnel data management system. Effective application development requires data as a basis, and one source of data that can be utilized is user reviews. User reviews provide valuable information regarding application performance, user needs, and security aspects. However, challenges arise in managing review data which often contains sarcasm, creating ambiguous meaning and lowering accuracy levels. This research proposes a solution by applying sentiment analysis using Naive Bayes logarithms to 1047 Raileo review data. This method produces an accuracy rate of 94%, with positive and negative sentiment classification. The research results show the words that appear most frequently in Raileo reviews, such as "eror", "sulit", "titik presensi", "titik absen", "titik lokasi", "bug", "lemot," "gagal", "mantap", "bagus", "oke", "mudah", "mempermudah", "mantul", "lengkap", "keren", "ok", "inovatif", "inovasi", "semoga", "sukses", dan "membantu". These words can be used as a key to analyze all the sentiments contained in the review. In addition, this research identifies "presence point" as the highest negative sentiment word that needs attention in further development. From this sentiment analysis research, the Raileo application produces the highest sentiment value, namely positive sentiment.

Keywords: reviews, application, raileo, NB, sentiment

Abstrak

Aplikasi Raileo merupakan platform kepegawaian milik oleh PT. Kereta Api Indonesia (Persero), berfungsi sebagai sistem pengelolaan data kepegawaian. Pengembangan aplikasi yang efektif memerlukan data sebagai dasar, dan salah satu sumber data yang dapat dimanfaatkan adalah ulasan pengguna. Ulasan pengguna memberikan informasi berharga terkait performa aplikasi, kebutuhan pengguna, dan aspek keamanan. Namun, tantangan muncul dalam mengelola data ulasan yang sering kali mengandung sarkasme, menciptakan makna ambigu dan menurunkan tingkat akurasi. Penelitian ini mengusulkan solusi dengan menerapkan analisa sentimen menggunakan Algoritma Naive Bayes pada 1047 data yang diperoleh dari ulasan Raileo. Metode ini menghasilkan tingkat akurasi sebesar 90%, dengan klasifikasi sentimen positif dan negatif. Hasil penelitian memperoleh kata-kata yang paling banyak muncul dalam ulasan Raileo, seperti "eror", "sulit", "titik presensi", "titik absen", "titik lokasi", "bug", "lemot," "gagal", "mantap", "bagus", "oke", "mudah", "mempermudah", "mantul", "lengkap", "keren", "ok", "inovatif", "inovasi", "semoga", "sukses", dan "membantu". Kata-kata ini dapat dijadikan sebagai kunci untuk menganalisis seluruh sentimen yang terdapat dalam ulasan. Selain itu, penelitian ini mengidentifikasi "titik presensi" sebagai kata sentimen negatif tertinggi yang perlu mendapatkan perhatian dalam pengembangan selanjutnya. Dari penelitian sentimen analisis ini aplikasi Raileo menghasilkan nilai sentimen tertinggi yaitu sentimen positif.

Kata Kunci: ulasan, aplikasi, raileo, NB, sentimen

1. PENDAHULUAN

PT. Kereta Api Indonesia (KAI), sebagai perusahaan penyedia jasa transportasi darat di sektor perkeretaapian, dihadapkan pada tuntutan optimalisasi pelayanan terhadap masyarakat, terutama dalam menjalankan kebijakan pemerintah dalam menyediakan transportasi yang dapat meningkatkan minat masyarakat untuk beralih menggunakan transportasi umum. Masyarakat kini mengharapkan layanan transportasi yang cepat, aman, efisien, dan nyaman. Dalam mewujudkan itu tak terlepas dari peran serta kemajuan teknologi. Dalam menjawab tuntutan tersebut, PT. KAI merespon perkembangan teknologi dengan melakukan transformasi digital pada seluruh proses bisnisnya. Salah satu aspek yang mengalami transformasi adalah manajemen pegawai, salah satunya menghadirkan aplikasi berbasis mobile bernama "RAILEO." Aplikasi ini dirancang untuk mengakomodir segala kebutuhan pegawai, dan mengatasi kendala jarak dan waktu. Dengan tujuan meningkatkan produktivitas dan kinerja mereka yang tersebar di seluruh Indonesia. Selain itu, RAILEO hadir untuk mempermudah pengelolaan sistem informasi data pegawai secara real-time.

Meski sistem ini membawa berbagai manfaat, tidak terlepas dari berbagai kendala dalam pengembangannya agar dapat memenuhi kebutuhan sesuai harapan pegawai. Salah satu solusi untuk mengidentifikasi kendala ini, sebagai landasan pengembangan aplikasi selanjutnya, adalah melalui analisis data ulasan pengguna. Ulasan pengguna menjadi sumber informasi yang dapat memberikan gambaran real-time mengenai permasalahan yang terjadi pada aplikasi [1]. Pengguna aplikasi, ketika menghadapi kendala, cenderung menuangkannya dalam ulasan yang tersedia di platform Play Store. Dari data ulasan kita dapat menggunakan analisa sentimen dengan proses pengolahan teks agar dapat dianalisis. Sentimen analisis menghasilkan data dalam bentuk informasi yang dapat di hitung nilai akurasi, sehingga data yang dihasilkan tidak menghasilkan data yang ambigu atau data yang belum jelas tingkat akurasi [2].

Analisa Sentimen pada penelitian ini bertujuan untuk mengukur tingkat efektivitas fungsionalitas aplikasi dengan mengetahui gambaran mengenai persepsi, penilaian, dan

perasaan penggunanya ketika menggunakan aplikasi [3], [4]. Sehingga dapat diperoleh data sebagai bahan untuk mengidentifikasi kelebihan dan kekurangan, dengan memperhatikan fitur-fitur yang dianggap baik oleh pengguna dan area mana saja yang perlu ditingkatkan [5]. Oleh karena itu, dalam penelitian ini diharapkan dapat membantu pengembang dalam melakukan perbaikan atau pengembangan aplikasi lebih lanjut untuk meningkatkan kualitas aplikasi sesuai dengan kebutuhan dan harapan pengguna.

Berdasarkan penelitian yang dilakukan oleh Abdul Rahman Fauzi dkk., mereka menganalisa ulasan pengguna game Apex Mobile menggunakan metode algoritma Naïve Bayes menghasilkan nilai tingkat keakuratan sebesar 86%, dari sejumlah data uji sebanyak 9105 data review sehingga menghasilkan sentimen positif sebanyak 7279 review dan 1246 review sentimen negatif dari ulasan yang diberikan pengguna [6]. Selanjutnya pada penelitian Sendi Alpin Rizaldi dkk., dari hasil penelitiannya mereka memperoleh hasil bahwa ulasan Playstore pada Aplikasi Jamsostek Mobile dengan menggunakan Naïve Bayes memperoleh Sentimen negatif yang cukup tinggi dari pengguna aplikasi yang mengindikasikan bahwa pengguna memberikan komentar negatif paling banyak terhadap aplikasi tersebut dimana dari hasil jumlah prediksi diperoleh nilai sentimen negatif sebesar 3528 ulasan sedangkan sentimen positif diperoleh sebesar 1472 ulasan dari total data yang dilakukan pengujian sebanyak 5000 data ulasan dengan tingkat akurasi sebesar 95% [7]. Selain itu pada penelitian yang dilakukan oleh Java Khitamasari dkk., dalam penelitiannya bahwa Algoritma Naïve bayes sangat relevan digunakan untuk menganalisis sentimen berupa ulasan, dimana penelitian tersebut menganalisa sentimen ulasan user terhadap Aplikasi Tx-Id dengan total data uji sebanyak 1600 data, memperoleh hasil nilai persentase akurasi sebesar 71,77%, terbagi menjadi klasifikasi positif sebesar 1259 data dan klasifikasi negatif sebesar 334 data [8].

Selain itu berdasarkan penelitian Robiata Tsania Salsabila Aditya Putri dkk., terkait komparasi Naïve Bayes dan KNN (K-Nearest Neighbor), dalam menganalisa Sentimen Aplikasi Gapura UB dari Ulasan Pengguna pada Playstore, penelitian tersebut memperoleh

kesimpulan bahwa performa Naïve Bayes lebih baik dibandingkan dengan Algoritma KNN , dimana dari total data uji sebanyak 300 data diperoleh hasil KNN dengan persentase nilai akurasi sebesar 88.5%, nilai kepresisian 88.7%, nilai recall sebesar 88.2%, dan nilai f-measure 88.2%. Sedangkan K-Nearest Neighbor menghasilkan nilai keakuratan 84.8%, nilai tingkat kepresisian 85.4%, nilai tingkat recall 84.6% dan nilai tingkat f-measure 84.1% yang dihasilkan dari nilai k=5 [9]. Pada penelitian lain seperti pada penelitian yang dilakukan Hery Oktafiandi dkk., dalam penelitiannya menganalisis perbandingan Algoritma Menggunakan Machine Learning dalam Analisis Sentimen Terhadap Google Play Store, dimana terdapat tiga Algoritma yang dilakukan perbandingan antara lain : Naives Bayes (NB), K-Nearest Neighbors (KNN) dan Random Forest. Dari hasil perbandingan tersebut diperoleh hasil akurasi Algoritma Naïve bayes paling unggul dibandingkan dengan algoritma lain dengan persentase keakuratan Algoritma Naïve Bayes 79%, Algoritma KNN 77% dan Algoritma Random Forest 75% , dimana dalam penelitian tersebut melakukan analisa menggunakan jumlah data uji yang berbeda-beda yaitu pada data uji dengan jumlah 40 data, 100 data dan terakhir 1000 data [10].

Dari penelitian terdahulu seperti yang telah dijelaskan sebelumnya, Peneliti dalam melakukan penelitian ini memilih Algoritma Naïve Bayes yang bisa dimanfaatkan untuk mengklasifikasikan data mining. Selain itu Algoritma Naïve Bayes digunakan untuk menganalisa data dalam bentuk teks berupa ulasan sehingga relevan dengan topik penelitian ini. Dikarenakan ulasan pengguna mayoritas berisi pengalaman pengguna, dimana pada saat pengguna menggunakan suatu jasa, produk atau aplikasi, cenderung melibatkan psikologis atau emosional yang dituangkan ke dalam ulasan pengguna sehingga dalam analisa sentimen membutuhkan pengklasifikasian sentimen negatif dan sentimen positif, agar dapat digunakan untuk memperoleh data yang akurat [11], [12].

Teknik sentimen analisis yang diterapkan dalam penelitian ini melibatkan beberapa tahap, dimulai dari pengumpulan data menggunakan teknik scraping di Play Store. Pemilihan metode pengumpulan data ini karena memiliki keunggulan tersendiri, di mana Play

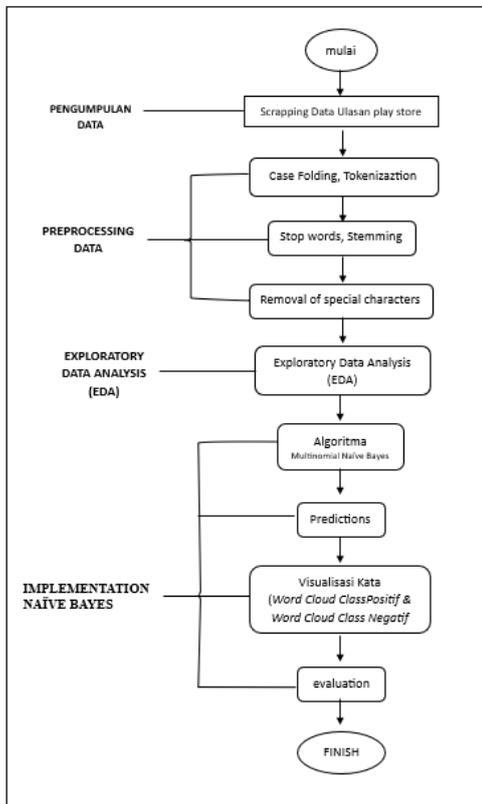
Store menyediakan platform terbuka untuk pengguna dalam mengungkapkan pandangan mereka tentang aplikasi secara bebas. Scrapping data di Play Store memungkinkan menulis ulasan yang bersifat objektif sesuai pengalaman yang dialami pengguna. Dengan demikian, data yang diperoleh melalui metode ini dianggap lebih representatif dan memberikan gambaran yang akurat terkait sentimen pengguna terhadap aplikasi yang diteliti.

Setelah data terkumpul, langkah selanjutnya adalah melakukan preprocessing data. Proses ini meliputi beberapa langkah seperti case folding untuk menyamakan format teks, tokenisasi untuk memecah teks menjadi kata-kata atau frasa, stopwords untuk menghilangkan kata-kata umum yang tidak memberikan kontribusi signifikan, stemming atau lemmatization untuk mereduksi kata-kata ke bentuk dasar, dan penghapusan karakter khusus. Selanjutnya, analisis Exploratory Data Analysis (EDA), di mana penelitian dilakukan untuk memahami distribusi dan tren dalam data. Proses vektorisasi TF-ID digunakan untuk mengonversi data teks menjadi representasi numerik [13], memfasilitasi pemrosesan lebih lanjut. Untuk mengklasifikasikan data sentimen, diterapkan algoritma Naïve Bayes sebagai metode klasifikasi yang efisien. Hasil dari analisis sentimen di visualisasi dalam bentuk "word cloud" untuk memberikan gambaran informasi tentang kata-kata yang paling sering muncul dalam setiap kategori sentimen. Terakhir, dilakukan evaluasi performa dari algoritma dengan menerapkan teknik K-Fold Cross Validation, yang membagi data menjadi beberapa subset untuk melatih dan menguji model secara iteratif.

Hasil analisis sentimen ini diharapkan dapat memberikan pemahaman mendalam mengenai persepsi dan kepuasan pengguna, identifikasi area perbaikan, serta dukungan pengembangan aplikasi sesuai dengan kebutuhan dan harapan pengguna. Melibatkan teknik K-Fold Cross Validation, penelitian ini memperoleh inspirasi dari penelitian sebelumnya yang berhasil menerapkan metode ini dengan tingkat akurasi yang signifikan. Dengan demikian, penelitian ini tidak hanya mendukung pengembangan aplikasi RAILEO, tetapi juga memberikan kontribusi dalam meningkatkan kualitas layanan PT. KAI kepada masyarakat.

2. METODE PENELITIAN

Dalam rangkaian penelitian ini, menggunakan beberapa tahapan metodologi dimana proses dimulai dengan alur penelitian sebagai berikut: Pengumpulan Data, dilanjutkan dengan Preprocessing Data, dan dilanjutkan Exploratory Data Analysis (EDA) untuk menganalisis data secara rinci. Kemudian Word Cloud digunakan sebagai metode visualisasi untuk mengidentifikasi kata-kata kunci yang muncul secara signifikan. Selanjutnya dilakukan implementasi Algoritma Naïve Bayes sebagai metode analisis sentimen. Langkah tersebut diintegrasikan dalam suatu diagram alur tahapan penelitian yang secara jelas digambarkan pada gambar 1. Flowchart Alur Penelitian dibawah ini:



Gambar 1. Flowchart Alur Penelitian

2.1. Pengumpulan Data



Gambar 2. Proses Scraping Dataset

Analisis sentimen sering digunakan untuk menganalisa pendapat seseorang yang dituangkan dalam bentuk teks yang mencerminkan ekspresi dari pengalaman yang diterima [14]. Sentimen analisis diproses dengan menggunakan komputasi untuk mengolah data yang terstruktur maupun tidak struktur dengan mengekstrak dan memahaminya, sehingga menghasilkan sentimen dari beberapa informasi yang diperoleh melalui pendapat, emosi atau komentar dalam dataset. *Scraping* merupakan teknik pengumpulan data yang menggunakan suatu platform media sosial maupun layanan digital seperti Google Play Store. *Data scraping* umumnya dilakukan dengan mengekstrak seluruh atau sebagian data yang bersifat semi-terstruktur [15], [16]. Dalam penelitian ini, pada tahapan pengumpulan data diperoleh dengan cara melakukan *scraping* data pada ulasan aplikasi Raileo di Play Store. Sejumlah 1047 data berhasil dikumpulkan dari total 2000 ulasan yang terdapat di Play Store. Adapun sisanya merupakan data yang hanya berisi penilaian (*rating*) tanpa disertai pernyataan ulasan. Scraping dilakukan dengan menggunakan platform google colab dan menghasilkan data dalam format CSV yang dapat kita dapatkan dengan mendownloadnya.

Table 1: Sampel Data Hasil Scraping

<i>username</i>	<i>score</i>	<i>content</i>
#makdhell movie	5	Oke
Tri Apriliono muslimin	5	Masih sering error
nurrachman	5	Bagus
jaya di	5	Good lah
Sandi Arika Rojak	5	Ok
Agus wng22	5	selama ini bagus
Hilmy Hiba	5	Tampilan UI bagus dan mudah di gunakan
Rio Suhadi	4	Buka aplikasi sinyal malah hilang
Kemas Fauzi	5	Sangat bermanfaat

2.2. Preprocessing Data

Preprocessing data merupakan tahap pemrosesan awal sebelum dilakukan proses analisa dengan menggunakan algoritma Naïve Bayes dimana tahapan ini data mentah dilakukan *CaseFolding*, *Tokenization*, *Stopwords*, *Stemming* atau *Lemmatization*, dan *Removal of special of character* [8].

2.3. Case Folding

Tahapan ini yaitu tahap memodifikasi semua huruf kapital di dalam teks menjadi huruf-huruf kecil unuk menghindari perbedaan dalam mengenali kata yang sebenarnya. Berikut ini contoh gambar data yang telah dilakukan *Case Folding* [17] , [18] .

Tabel 2: Contoh data yang sudah di *Case Folding*

<i>username</i>	<i>score</i>	<i>content</i>	<i>cleaned_text</i>
Tri Apriliono	5	Masih sering error Sangat	masih sering error sangat
Rizky Hermansyah	5	membantu dan mempermudah mencari sesuat...	membantu dan mempermudah mencari sesuat...
muslimin nurrachman	5	Bagus	bagus
Hena Ruswana	5	ðŸ–	
jaya di	5	Good lah	good lah

2.4. Tokenization

Pada tahap ini adalah tahapan dimana teks dipecah menjadi unit kata yang lebih kecil bertujuan untuk mengelompokan kata yang diperlakukan sebagai pemisah maupun bukan pemisah [19]. Berikut ini perbandingan gambar data yang telah di lakukan tokenization dan belum dilakukan tokenization.

Tabel 3: Contoh data yang sudah di *Tokenization*

<i>username</i>	<i>content</i>	<i>tokens</i>
Tri Apriliono	Masih sering error	[masih, sering, error]
Rizky Hermansyah	Sangat membantu dan mempermudah mencari sesuat...	[sangat, membantu, dan, mempermudah, mencari, ...]
muslimin nurrachman	Bagus	[bagus]
Hena Ruswana	ðŸ–	[]
jaya di	Good lah	[good, lah]

2.5. Stopwords

Stopwords merupakan tahap menghapus kata-kata yang tidak memberikan makna yang berarti pada analisis. Contoh: *stopwords* melibatkan kata-kata seperti "yang", "atau", "dan", dan sebagainya. Penerapan teknik stopwords bertujuan untuk meningkatkan kemampuan sistem dalam mengklasifikasikan sentimen pada setiap teks dengan lebih akurat [20].

2.6. Stemming atau Lemmatization

Stemming atau *lemmatization* merupakan teknik untuk mengubah dan mengembalikan kata kedalam bentuk semula sehingga analisis tidak terganggu oleh varian kata [21]. Gambar data yang telah dilakukan stemming sebagai berikut :

Tabel 4: Contoh data yang telah di *Stemming* atau *Lemmatization*

<i>Username</i>	<i>Content</i>	<i>lemmatized_review</i>
Tri Apriliono	Masih sering error Sangat	error
Rizky Hermansyah	membantu dan mempermudah mencari sesuat...	membantu dan mempermudah mencari yg perlukan
muslimin nurrachman	Bagus	bagus
Hena Ruswana	ðŸ–	
jaya di	Good lah	good

3. HASIL DAN PEMBAHASAN

3.1. Data Labeling

Pada tahap pelabelan data, peneliti menggunakan fungsi "map" untuk memberikan label berdasarkan Rating dalam kolom 'score'. Dimana pada Nilai-nilai skor 1.0, 2.0, dan 3.0 di konversikan ke label 'Negative' di berikan variabel 0, sementara skor 4.0 dan 5.0 dikonversikan ke label 'Positive' diberikan variabel 1. Dalam proses ini bertujuan untuk memudahkan interpretasi dan analisis data, dengan setiap entri dalam kolom 'score' diubah sesuai aturan pemetaan. Hasilnya, kolom 'label' mencerminkan kategori positif atau negatif berdasarkan skor. Selanjutnya hasil tersebut dilakukan verifikasi oleh ahli bahasa Indonesia untuk memastikan keakuratan labeling data tersebut dengan pencocokan kalimat ulasan yang ditulis pengguna untuk menghindari kata

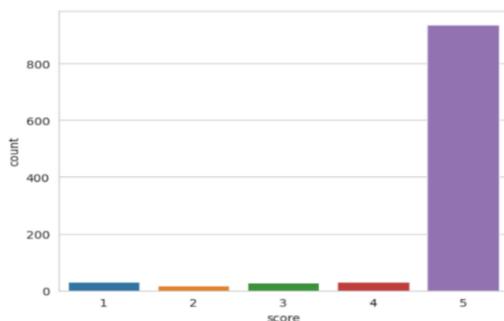
yang mengandung sarkasme ataupun ulasan yang tidak memrepresentasikan rating. Berikut tabel sampel data hasil labeling.

Tabel 5. Sampel data hasil Labeling

content	label
Sangat membantu dan mempermudah mencari sesuatu yg kita perlukan	1
Bagus	1
Good lah	1
Ok	1
Saran bagi pengembang , agar bisa melakukan pengajuan pengajuan seperti JPK agar bisa dilakukan lewat online/raileo agar lebih efisien.	0
selama ini bagus	1
Sangat bermanfaat	1
ok bae lah	1
Mantap...	1
Aplikasi sering error absen gak bisa2 alhasil absensi jadi telat merah terus padahal sudah datang dari 20mnt sebelum jam masuk pas pulang absen juga sama error terus parah tolong benahi dulu sistemnya jadi banyak yg di rugikan absensi jadi jelek gara2 susah absen error terus	0

3.2. Exploratory Data Analysis (EDA)

Tahapan mengidentifikasi agar sistem dapat mengenali pola, memeriksa asumsi, menemukan anomali dan menguji hipotesis dari suatu data agar dapat meningkatkan nilai akurasi [22], Berikut ini gambar data hasil proses EDA.



Gambar 3. data grafik hasil identifikasi jumlah data dan score (rating) aplikasi Raileo

```

Input data has 1047 rows and 10 columns
score 1.0 = 31 rows
score 2.0 = 18 rows
score 3.0 = 29 rows
score 4.0 = 32 rows
score 5.0 = 937 rows

```

Gambar 4. data hasil identifikasi jumlah data dan score (rating) aplikasi Raileo

3.3. Vektorisasi TF-ID

Tahap ini menjadi langkah persiapan sebelum melaksanakan implementasi Naive Bayes karena Naive Bayes hanya mampu menghitung data jika data tersebut direpresentasikan dalam bentuk vektor numerik. Oleh karena itu, TF-IDF berfungsi sebagai alat untuk mengubah dokumen berupa teks menjadi representasi numerik [23], sesuai dengan format yang diharapkan oleh algoritma Naive Bayes.

Berikut ini alur proses dalam melakukan vektorisasi TF-IDF

- Membuat kode program seperti pada gambar 7 ini, disini peneliti melakukan pemilihan fitur dan label dari suatu DataFrame. Fitur yang dipilih adalah kolom 'lemmatized_review', 'content_len', dan 'punct', yang masing-masing mewakili teks yang telah di-lematisasi, panjang teks tanpa spasi, dan persentase tanda baca dalam teks. Label diambil dari kolom 'label'. Setelah pemilihan fitur dan label, kita menggunakan perintah print untuk menampilkan dimensi atau bentuk dari fitur (X) dan label (y). Ini membantu untuk memastikan bahwa data telah dipilih dan diatur dengan benar sebelum proses analisis atau pemodelan dilakukan.

```

# Memilih fitur dan label dari dataframe
X = df[['lemmatized_review', 'content_len', 'punct']]
y = df['label']

# Menampilkan dimensi (bentuk) dari fitur (X) dan label (y)
print(X.shape)
print(y.shape)

```

(1047, 3)
(1047,)

Gambar 5. Listing program pemilihan fitur dan label Data Frame

- Membuat kode program seperti pada gambar 6, Dalam kode program tersebut, peneliti menggunakan fungsi train_test_split dari pustaka scikit-learn

untuk memecah dataset menjadi 2 bagian: data latih dan data uji. Variabel X dan y merepresentasikan fitur dan label. Dengan menentukan `test_size=0.3`, kita menetapkan bahwa 30% dari data akan digunakan sebagai data pengujian, sementara 70% yang sisanya akan digunakan sebagai data pelatihan. Parameter `random_state=0` memastikan bahwa pembagian dilakukan dengan cara yang sama setiap kali kode dijalankan untuk reproduktibilitas. Setelah pemisahan, kita menggunakan perintah `print` untuk menampilkan dimensi atau bentuk dari data pelatihan dan data pengujian, termasuk jumlah baris dan kolom untuk fitur serta jumlah baris untuk label. Hal ini membantu untuk memastikan bahwa pembagian data telah berhasil dan sesuai dengan kebutuhan model pembelajaran mesin yang akan digunakan.

```
# Memisahkan data menjadi data pelatihan dan data pengujian menggunakan train_test_split
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=0)

# Menampilkan dimensi (bentuk) data pelatihan dan data pengujian
print(X_train.shape) # Menampilkan dimensi data pelatihan fitur (jumlah baris, jumlah kolom)
print(X_test.shape) # Menampilkan dimensi data pengujian fitur (jumlah baris, jumlah kolom)
print(y_train.shape) # Menampilkan dimensi data pelatihan label (jumlah baris)
print(y_test.shape) # Menampilkan dimensi data pengujian label (jumlah baris)
```

(732, 3)
(315, 3)
(732,)
(315,)

Gambar 6. Listing Program untuk membagi Dataset

- c. Dalam kode program ini, peneliti menggunakan `TfidfVectorizer` dari `scikit-learn` untuk mengubah teks ulasan yang telah dilamatisasi menjadi representasi vektor TF-IDF. Parameter `max_df` dan `min_df` digunakan untuk mengabaikan kata-kata yang muncul terlalu sering atau terlalu jarang dalam dokumen. Setelah menerapkan vektorisasi pada data pelatihan (`X_train['lemmatized_review']`), peneliti mendapatkan matriks TF-IDF untuk data pelatihan (`tfidf_train`). Proses yang sama kemudian diulang untuk data pengujian (`X_test['lemmatized_review']`), menghasilkan matriks TF-IDF untuk data pengujian (`tfidf_test`). Selanjutnya, peneliti menggabungkan matriks TF-IDF dengan fitur tambahan, yaitu 'content_len' dan 'punct', menggunakan `pd.concat`. Ini menciptakan `DataFrame` baru (`X_train_vect` dan `X_test_vect`) yang mencakup fitur-fitur tersebut bersama dengan matriks TF-IDF yang sesuai. Hasilnya adalah representasi vektor yang lengkap dari setiap dokumen, yang siap digunakan sebagai input untuk

model klasifikasi atau analisis selanjutnya. Dengan menampilkan beberapa baris pertama dari `X_train_vect` menggunakan `head()`, peneliti dapat memastikan bahwa proses vektorisasi telah berjalan dengan sukses dan mendapatkan gambaran awal dari representasi data yang akan digunakan dalam model.

```
from sklearn.feature_extraction.text import TfidfVectorizer
tfidf = TfidfVectorizer(max_df = 0.5, min_df = 2)
# ignore terms that occur in more than 50% documents and the ones that occur in less than 2
tfidf_train = tfidf.fit_transform(X_train['lemmatized_review'])
tfidf_test = tfidf.transform(X_test['lemmatized_review'])

X_train_vect = pd.concat([X_train[['content_len', 'punct']],
                          pd.DataFrame(tfidf_train.toarray()),
                          axis=1)
X_test_vect = pd.concat([X_test[['content_len', 'punct']],
                        pd.DataFrame(tfidf_test.toarray()),
                        axis=1)

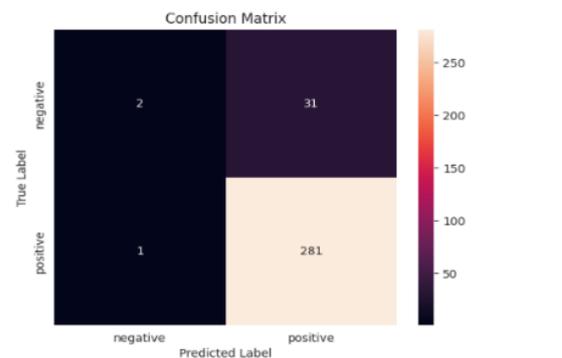
X_train_vect.head()
```

content_len punct 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29
0 0.0
1 0.0
2 0.0
3 0.0
4 0.0

Gambar 7. Listing program dan Output Convert kata menjadi vektor

3.4. Pengklasifikasian Data Menggunakan Algoritma Naïve Bayes.

Proses Klasifikasi Naïve Bayes melibatkan perhitungan kelas target berdasarkan fitur-fitur probabilitas yang diamati dalam data latih [24]. Keunggulan naïve bayes relatif mudah di implementasikan, memiliki kecepatan komputasi yang tinggi dan kinerja yang baik pada data set dengan dimensi tinggi, Berikut ini hasil pengklasifikasian dengan menggunakan algoritma naïve bayes [25].



Gambar 8. Prediction label

	precision	recall	f1-score	support
0	0.67	0.06	0.11	33
1	0.90	1.00	0.95	282
accuracy			0.90	315
macro avg	0.78	0.53	0.53	315
weighted avg	0.88	0.90	0.86	315

Gambar 9. Gambar Output hasil perhitungan Sistem

3.5. Diskusi

3.5.1 Perhitungan Manual

Berikut ini peneliti melakukan penghitungan manual dengan menggunakan rumus seperti ditunjukkan pada penjelasan dibawah ini:

a. Precision

Diketahui:

TP = 28, FP = 33

1. Untuk kelas 0

$$\begin{aligned} \text{Precision} &= \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positive (FP)}} \\ &= \frac{0}{0+0} = \text{Undefined} \end{aligned} \quad (1)$$

2. Untuk Kelas 1

$$\begin{aligned} \text{Precision} &= \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positive (FP)}} \\ &= \frac{282}{282 + 33} \\ &= 0,895 \\ &= 0,9 \end{aligned} \quad (2)$$

b. Recall

Diketahui:

TP = 2, FN = 31

1. Untuk Kelas 0 :

$$\begin{aligned} \text{Recall} &= \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negative (FN)}} \\ &= \frac{282}{282 + 31} = 0,061 \\ &= 0,06 \end{aligned} \quad (3)$$

2. Untuk kelas 1:

$$\begin{aligned} \text{Recall} &= \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negative (FN)}} \\ &= \frac{282}{282 + 0} \\ &= 1 \end{aligned} \quad (4)$$

c. F1-Score

1. Untuk kelas 0

Diketahui:

Precision = undefined,

recall = 0,061

F1 Score

$$\begin{aligned} &= \frac{2 \times \text{precision} \times \text{recall}}{\text{Precision} + \text{Recall}} \\ &= \frac{\text{Undefined} \times 0,061}{\text{Undefined} + 0,061} \\ &= \text{Undefined} \end{aligned} \quad (5)$$

2. Untuk kelas 1 :

F1 Score

$$\begin{aligned} &= \frac{2 \times 0,895 \times 1}{\text{Precision} + \text{Recall}} \\ &= \frac{2 \times \text{undefined} \times 0,061}{0,895 + 1} \\ &= 0,945 \end{aligned} \quad (6)$$

a. Support

1. Suport adalah jumkah ibservasu aktual dalam setiap kelas.

2. Untuk kelas 0 : Support = 33

3. Untuk kelas 1 : support = 282

b. Acuraccy

1. Accuracy

$$\begin{aligned} \text{Accuracy} &= \frac{\text{Correct Prediction}}{\text{Total Prediction}} \\ &= \frac{282}{315} \\ &= 0,895 \\ &= 0,9 \end{aligned} \quad (7)$$

2. Akurasi sekitar 0,895 atau jika dibulatkan menjadi 0,9

c. Macro AVG:

1. Macro Avg =

$$\begin{aligned} &= \frac{\text{precision (kelas 0)} + \text{precision (kelas 1)}}{2} \\ &= \frac{\text{Undefined} + 0,895}{2} \\ &= \text{Undefined} \end{aligned} \quad (8)$$

2. Macro Avg

$$\begin{aligned} &= \frac{\text{Recall (kelas 0)} + \text{REcall (kelas 1)}}{2} \\ &= \frac{0,061 + 1}{2} \\ &= 0,5305 \end{aligned} \quad (9)$$

3. Macro Avg

$$\begin{aligned} &= \frac{\text{F1 - Score (kelas 0)} + \text{F1 - Score(kelas 1)}}{2} \\ &= \frac{\text{Undefined} + 0,945}{2} \\ &= \text{Undefined} \end{aligned} \quad (10)$$

d. Weight Avg

$$\begin{aligned} \text{Weight Avg} &= \frac{\text{precision (kelas0)} \times \text{Support (kelas0)} + \text{Precison (kelas0)} \times \text{Support (kelas1)}}{\text{Total Support}} \end{aligned}$$

$$= \frac{UndefinedX33 + 0,895X282}{33 + 282}$$

$$= Undefined \quad (11)$$

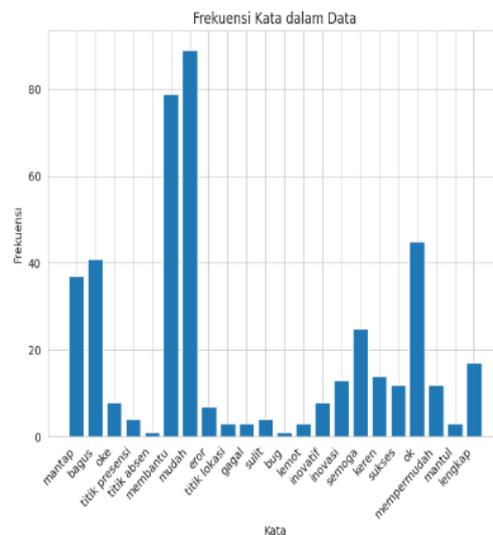
Dari hasil analisis *Confusion Matrix* yang dihasilkan, dapat disimpulkan bahwa model klasifikasi ini memiliki kecenderungan untuk memprediksi dengan akurasi tinggi pada sampel-sampel yang sebenarnya termasuk dalam kelas "positif," seperti terlihat dari jumlah True Positives yang signifikan sebanyak 281. Dengan nilai precision 90%, nilai recall 100% f1-score 95%. Meskipun demikian, hasil itu menunjukkan ada beberapa kendala dalam mengklasifikasikan. Sampel-sampel yang seharusnya masuk ke dalam kelas "negatif," seperti yang terlihat pada jumlah True Negatives yang relatif rendah sebanyak 2, Kesulitan terbesar terletak pada tingginya jumlah False Positives, di mana sistem melakukan kesalahan dengan memprediksi sejumlah 31 sampel sebagai kelas "positif" padahal seharusnya termasuk dalam kelas "negatif." Kesimpulan ini memberikan gambaran tentang area di mana model memerlukan peningkatan, khususnya dalam mengurangi kesalahan dalam memprediksi kelas "positif" yang sebenarnya "negatif." Evaluasi lanjutan dengan menggunakan metrik seperti precision, recall, dan f1-dengan menggunakan metode *Performing K-Fold Cross Validation* akan membantu memberikan gambaran yang lebih rinci.

3.5.2. Analisa Frekuensi kata yang mengandung Sentimen dalam dataset.

Dari hasil analisis kata berdasarkan hasil klasifikasi kata sentimen positif dan negatif peneliti menghitung jumlah kata yang muncul dan masuk dalam klasifikasi sentimen. Hasil perhitungan Frekuensi dari kata yang berada di dataset menunjukkan bahwa kata "mudah" menempati kata paling banyak pada ulasan pengguna aplikasi yaitu sebanyak 89 kata sementara kata "bug" menempati kata paling sedikit yaitu 1 kata.

Tabel 6: Frekuensi Kata dalam data

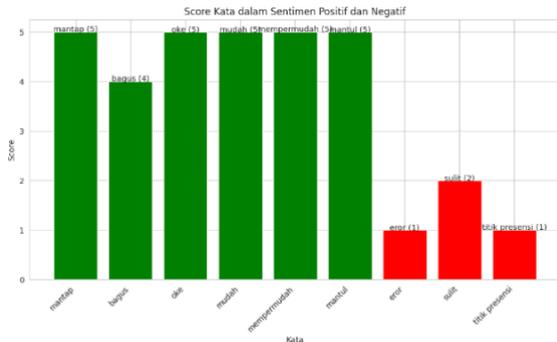
Item Kata	Frekuensi
bagus	41
bug	1
eror	7
gagal	3
inovasi	13
inovatif	8
keren	14
lemot	3
lengkap	17
mantul	3
membantu	79
mempermudah	12
mudah	89
ok	45
oke	8
semoga	25
sukses	12
sulit	4
titik absen	1
titik lokasi	3
titik presensi	4



Gambar 10. Grafik Frekuensi Kata

3.5.3. Analisa Score (Rating) berdasarkan kata Sentimen Positif dan Negatif

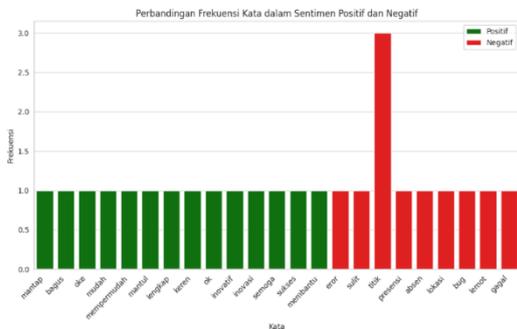
Pada tahap ini peneliti menghitung jumlah score (rating) pada setiap kata yang menempati score 1-5, seperti ditunjukkan pada gambar 11.



Gambar 11. Diagram Batang Score kata dalam sentimen positif

3.5.4. Analisa Perbandingan Frekuensi Sentimen Positif dan Negatif dalam kata

Pada tahap ini peneliti melakukan perbandingan kata sentimen positif dan negatif berdasarkan masing-masing kata, didalam data diagram batang menunjukan kata “titik presensi” menempati kata yang paling tinggi nilai sentimennya.



Gambar 12. Grafik Frekuensi kata dalam sentimen Positif dan Negatif

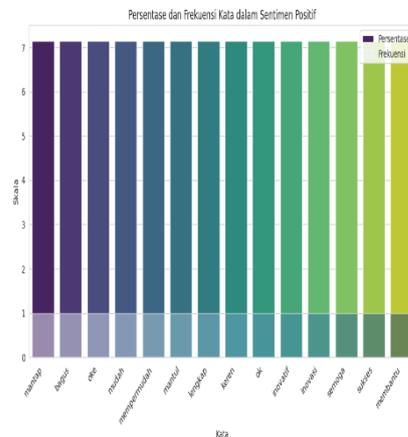
3.5.5. Analisa Persentase Sentimen Positif berdasarkan kata

Berdasarkan perhitungan sentimen positif peneliti menemukan adanya kestabilan kata dalam sentimen positif yaitu sebesar 7,14% pada sekuruh masing- masing kata sentimen positif

Tabel 7: Persentase kata Sentimen Positif

Kata	Frekuensi	persentase
mantap	1	7,14%
bagus	1	7,14%
oke	1	7,14%
mudah	1	7,14%
mempermudah	1	7,14%
mantul	1	7,14%
lengkap	1	7,14%
keren	1	7,14%

ok	1	7,14%
novatif	1	7,14%
inovasi	1	7,14%
semoga	1	7,14%
sukses	1	7,14%
membantu	1	7,14%



Gambar 13. Persentase Sentimen Positif berdasarkan kata

3.5.6. Analisa Persentase Sentimen Negatif berdasarkan kata

Dari hasil perhitungan persentase negatif berdasarkan kata, terdapat kata pada sentimen negatif tertinggi yaitu pada kata “titik presensi” mendapat nilai persentase sebanyak “27%”. Dapat dilihat pada gambar 15 diagram batang.

Tabel 8: Persentase Kata sentimen Negatif

Kata	Frekuensi	Persentase
titik	3	27%
eror	1	9%
sulit	1	9%
presensi	1	9%
absen	1	9%
lokasi	1	9%
bug	1	9%
lemot	1	9%
gagal	1	9%

evaluasi yang lebih komprehensif, sementara data yang dihasilkan langsung (naïve bayes) sebagai gambaran awal untuk di evaluasi. Setelah melakukan perhitungan peneliti mendapatkan tingkat nilai akurasi pada algoritma Naïve bayes sebesar 94%, Nilai Presisi 95 %, Nilai *Recall* 100% dan Nilai *F1-Score* 97%. berikut ini penjelasan tabel hasil perhitungan.

Tabel 9: Hasil K-Performing Validation

Akurasi	Presisi	recall	F1-Score
94%	95%	100%	97%

4. KESIMPULAN

Aplikasi Raileo sebagai aplikasi sistem management kepegawaian yang memiliki peran sebagai sistem pengolahan data, dan sistem informasi kepegawaian. Untuk melihat tingkat efektivitas dari aplikasi diperlukan data sebagai bahan acuan untuk mengembangkan aplikasi agar sesuai dengan yang di harapkan, teknik cara untuk memperoleh data tersebut adalah dengan mengambil data ulasan. Data ulasan menjadi acuan penting karena umumnya mencakup sentimen negatif dan positif yang mencerminkan pengalaman pengguna saat menggunakan aplikasi. Namun, ketika mengambil data dari ulasan, seperti melalui ulasan Google play, terjadi kendala terkait validitas data akibat adanya pengguna yang menuliskan ulasandengan sarkasme, menciptakan persepsi ambigu. Oleh sebab itu, di dalam penelitian ini, peneliti menganalisis setiap ulasan untuk memastikan data yang didapatkan valid. Hasil dari penelitian ini, peneliti menyimpulkan sebagai berikut: berdasarkan data uji sebanyak 1047 data dari total 2000 data hasil ulasan aplikasi Raileo di Play Store, teridentifikasi kata-kata yang sering muncul dalam ulasan, seperti "eror," "sulit," "titik presensi," "titik absen," "titik lokasi," "bug," "lemot," "gagal," "mantap," "bagus," "oke," "mudah," "mempermudah," "mantul," "lengkap," "keren," "ok," "inovatif," "inovasi," "semoga," "sukses," dan "membantu." Dalam analisis lebih lanjut, kata yang mencakup sentimen negatif tertinggi adalah "Titik," yang mencapai persentase sebanyak 27% dari total sentimen negatif. Hal tersebut mempresentasikan bahwa pengguna mengeluhkan kesulitan dalam mendeteksi titik lokasi pada saat melakukan presensi, yang

disuarakan melalui ulasan aplikasi raileo di google play. Hasil akhir analisis sentimen menunjukkan bahwa aplikasi Raileo mendapatkan sentimen positif sebesar 94% dan sentimen negatif sebesar 6%. Hal ini mengindikasikan bahwa tingkat performa aplikasi sangat baik dan cukup efektif dalam mengolah data kepegawaian. tingkat akurasi algoritma Naïve Bayes, berdasarkan hasil perhitungan menggunakan metode K-Performing Validation, mencapai tingkat keakuratan sebesar 94%, tingkat nilai kepresisian sebesar 95% dan tingkat nilai recall sebesar 100% dalam menganalisis data.

DAFTAR PUSTAKA

- [1] J. E. Reyes Arias, K. Kurtzhall, D. Pham, M. W. Mkaouer, and Y. N. Elglaly, "Accessibility Feedback in Mobile Application Reviews: A Dataset of Reviews and Accessibility Guidelines," in *CHI Conference on Human Factors in Computing Systems Extended Abstracts*, New York, NY, USA: ACM, Apr. 2022, pp. 1–7. doi: 10.1145/3491101.3519625.
- [2] A. Mohamed Mostafa, "Enhanced Sentiment Analysis Algorithms for Multi-Weight Polarity Selection on Twitter Dataset," *Intelligent Automation & Soft Computing*, vol. 35, no. 1, pp. 1015–1034, 2023, doi: 10.32604/iasc.2023.028041.
- [3] J. Fiaidhi and S. Mohammed, "Thick Data: A New Qualitative Analytics for Identifying Customer Insights," *IT Prof*, vol. 21, no. 3, pp. 4–13, May 2019, doi: 10.1109/MITP.2019.2910982.
- [4] S. Gupta and R. Sandhane, "Use of sentiment analysis in social media campaign design and analysis," *CARDIOMETRY*, no. 22, pp. 351–363, May 2022, doi: 10.18137/cardiometry.2022.22.351363.
- [5] E. Jalali, Z. Zojaji, and M. Soleimani, "A Sentiment Analysis Dataset Preparation Framework: A Case Study on User Reviews Dataset on Iranian Tourist Destinations," in *2023 9th International Conference on Web Research (ICWR)*, IEEE, May 2023, pp. 330–334. doi: 10.1109/ICWR57742.2023.10139217.
- [6] Abdul Rahman Fauzi, Ananda Hadi Saputra, and Muhammad Hasbi

- Abdillah, "Analisis Tingkat Kepuasan Pengguna Apex Mobile Berdasarkan Rating Dan Ulasan Google Play Store Menggunakan Naïve Bayes," vol. 1, pp. 17–21, Aug. 2023.
- [7] S. A. R. Rizaldi, S. Alam, and I. Kurniawan, "Analisis Sentimen Pengguna Aplikasi JMO (Jamsostek Mobile) Pada Google Play Store Menggunakan Metode Naïve Bayes," *STORAGE: Jurnal Ilmiah Teknik dan Ilmu Komputer*, vol. 2, no. 3, pp. 109–117, Aug. 2023, doi: 10.55123/storage.v2i3.2334.
- [8] Java Khitamasari and Isa Faqihuddin Hanif, "Analisis Sentimen Pengguna Terhadap Aplikasi Tix-Id Menggunakan Metode Naïve Bayes Pada Google Play Store," *Jurnal ICT: Information Communication & Technology*, vol. 23, pp. 81–83, Jul. 2023.
- [9] Robiata Tsania Salsabila Aditya Putri, Dian Eka Ratnawati, and Dwija Wisnu Brata, "Perbandingan Naïve Bayes dan K-Nearest Neighbor untuk Analisis Sentimen Aplikasi Gapura UB Berdasarkan Ulasan Pengguna pada Playstore," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 7, pp. 229–236, Jan. 2023.
- [10] Hery Oktafiandi, Winarnie, and Sayyid M. Raziq Olajuwon, "Perbandingan Algoritma untuk Analisis Sentimen Terhadap Google Play Store Menggunakan Machine Learning," *JURNAL EKONOMI DAN TEKNIK INFORMATIKA*, vol. 11, pp. 16–21, Jul. 2023.
- [11] Sambodo Rio Sasongko, "FAKTOR-FAKTOR KEPUASAN PELANGGAN DAN LOYALITAS PELANGGAN (LITERATURE REVIEW MANAJEMEN PEMASARAN)," *Jurnal Ilmu Manajemen Terapan*, vol. 3, no. 1, pp. 104–114, Oct. 2021, doi: 10.31933/jimt.v3i1.707.
- [12] M. I. P. Hant and H. Hendry, "DATA MINING TECHNIQUE USING NAÏVE BAYES ALGORITHM TO PREDICT SHOPEE CONSUMER SATISFACTION AMONG MILLENNIAL GENERATION," *Jurnal Teknik Informatika (Jutif)*, vol. 3, no. 4, pp. 829–838, Aug. 2022, doi: 10.20884/1.jutif.2022.3.4.295.
- [13] A. Bhansali, A. Chandravadiya, B. Y. Panchal, M. H. Bohara, and A. Ganatra, "Language Identification Using Combination of Machine Learning Algorithms and Vectorization Techniques," in *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, IEEE, Apr. 2022, pp. 1329–1334. doi: 10.1109/ICACITE53722.2022.9823628.
- [14] Mahardika Tania Nitami and Herny Februariyanti, "ANALISIS SENTIMEN ULASAN EKSPEDISI J&T EXPRESS MENGGUNAKAN ALGORITMA NAÏVE BAYES," *Jurnal Manajemen Informatika & Sistem Informasi (MISI)*, vol. 5, pp. 20–29, Jan. 2022.
- [15] R. Wahyudi and G. Kusumawardana, "Analisis Sentimen pada Aplikasi Grab di Google Play Store Menggunakan Support Vector Machine," *Jurnal Informatika*, vol. 8, no. 2, pp. 200–207, Sep. 2021, doi: 10.31294/ji.v8i2.9681.
- [16] Fauzan Setya Ananto and Firman Noor Hasan, "Implementasi Algoritma Naïve Bayes Terhadap Analisis Sentimen Ulasan Aplikasi MyPertamina pada Google Play Store," *Jurnal ICT: Information Communication & Technology*, vol. 1, pp. 75–80, Jul. 2023.
- [17] R. Wati and S. Ernawati, "Analisis Sentimen Persepsi Publik Mengenai PPKM Pada Twitter Berbasis SVM Menggunakan Python," *Jurnal Teknik Informatika UNIKA Santo Thomas*, pp. 240–247, Nov. 2021, doi: 10.54367/jtiust.v6i2.1465.
- [18] M. I. Petiwi, A. Triayudi, and I. D. Sholihati, "Analisis Sentimen Gofood Berdasarkan Twitter Menggunakan Metode Naïve Bayes dan Support Vector Machine," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 6, no. 1, p. 542, Jan. 2022, doi: 10.30865/mib.v6i1.3530.
- [19] S. M. Permataning Tyas, B. S. Rintyarna, and W. Suharso, "The Impact of Feature Extraction to Naïve Bayes Based Sentiment Analysis on Review Dataset of Indihome Services," *Digital Zone: Jurnal Teknologi Informasi dan Komunikasi*, vol. 13, no. 1, pp. 1–10, Apr. 2022, doi: 10.31849/digitalzone.v13i1.9158.
- [20] F. Rahutomo and A. R. T. H. Ririd, "Evaluasi Daftar Stopword Bahasa Indonesia," *Jurnal Teknologi Informasi*

- dan Ilmu Komputer*, vol. 6, no. 1, pp. 41–48, Jan. 2019, doi: 10.25126/jtiik.2019611226.
- [21] F. Sidik, I. Suhada, A. H. Anwar, and F. N. Hasan, “Analisis Sentimen Terhadap Pembelajaran Daring Dengan Algoritma Naive Bayes Classifier,” *Jurnal Linguistik Komputasional (JLK)*, vol. 5, no. 1, p. 34, Apr. 2022, doi: 10.26418/jlk.v5i1.79.
- [22] Abi Vegari and Setia Budi, “Implementasi Exploratory Data Analysis Pada Dataset Video Trending Harian YouTube,” *Jurnal Strategi*, vol. 2, pp. 397–411, Nov. 2020.
- [23] A. Halimi, K. Kusriani, and M. R. Arief, “ANALISIS SENTIMEN MASYARAKAT INDONESIA TERHADAP PEMBELAJARAN ONLINE DARI DI MEDIA SOSIAL TWITTER MENGGUNAKAN LEXICON DAN K-NEAREST NEIGHBOR,” *COREAI: Jurnal Kecerdasan Buatan, Komputasi dan Teknologi Informasi*, vol. 2, no. 1, pp. 18–28, Aug. 2021, doi: 10.33650/coreai.v2i1.2283.
- [24] . M., T. I. Fajri, and N. Hasdyna, “Classification of Graduation Students of the Faculty of Computers and Multimedia, Universitas Islam Kebangsaan Indonesia Using the Naïve Bayes Classifier Algorithm,” *International Journal of Research and Review*, vol. 9, no. 12, pp. 77–83, Dec. 2022, doi: 10.52403/ijrr.20221209.
- [25] B. Bakiyev, “Method for Determining the Similarity of Text Documents for the Kazakh language, Taking Into Account Synonyms: Extension to TF-IDF,” in *2022 International Conference on Smart Information Systems and Technologies (SIST)*, IEEE, Apr. 2022, pp. 1–6. doi: 10.1109/SIST54437.2022.9945747.
- [26] U. Bhimavarapu, G. Battineni, and N. Chintalapudi, “Automatic Classification of Hypertensive Retinopathy by Gray Wolf Optimization Algorithm and Naïve Bayes Classification,” *Axioms*, vol. 12, no. 7, p. 625, Jun. 2023, doi: 10.3390/axioms12070625.
- [27] K. Suzuki and T. Matsuzawa, “Model Soups for Various Training and Validation Data,” *AI*, vol. 3, no. 4, pp. 796–808, Sep. 2022, doi: 10.3390/ai3040048.