

Analisis Sentimen Naïve Bayes dengan TF-IDF dan 10-Fold pada Ulasan Aplikasi X

Cha Cha Kirana, Nabila Rizky Oktadini*

Sistem Informasi, Fakultas Ilmu Komputer, Universita Sriwijaya, Palembang, Indonesia

Email: ¹chachakirana04@gmail.com, ^{2,*}nabilarizky@unsri.ac.id

Email Penulis Korespondensi: nabilarizky@unsri.ac.id *

Submitted: 09/08/2025; Accepted: 11/12/2025; Published: 31/12/2025

Abstrak– Penelitian ini bertujuan untuk mengidentifikasi dan mengklasifikasikan sentimen pada ulasan pengguna aplikasi X (Twitter) ke dalam kategori positif dan negatif secara otomatis untuk memperoleh gambaran objektif mengenai persepsi pengguna. Kebaruan penelitian terletak pada penerapan rangkaian preprocessing teks secara lengkap berbasis TF-IDF yang dikombinasikan dengan evaluasi model menggunakan teknik 10-fold cross-validation pada dataset berskala besar berjumlah 5.000 ulasan dari Google Play Store, sehingga menghasilkan model yang lebih akurat dan stabil dibanding pendekatan terdahulu. Metode penelitian mencakup tahapan text preprocessing (case folding, stopword removal, stemming, tokenizing, normalisasi karakter dan pembersihan simbol), transformasi TF-IDF untuk representasi numerik, dan klasifikasi menggunakan algoritma Naïve Bayes serta Support Vector Machine (SVM). Hasil penelitian menunjukkan bahwa model SVM memberikan performa tertinggi dengan akurasi rata-rata 90,02%, presisi 89,97%, recall 89,23%, dan F1-score 89,53%, melampaui Naïve Bayes yang memperoleh akurasi 87,14%, presisi 88,13%, recall 85,29%, dan F1-score 86,19%. Visualisasi wordcloud mengonfirmasi perbedaan kosakata dominan antara sentimen positif dan negatif. Temuan ini menegaskan bahwa SVM lebih unggul untuk klasifikasi sentimen berbasis teks pada ulasan aplikasi. Implikasi penelitian menunjukkan bahwa hasil analisis sentimen dapat menjadi sumber informasi berbasis data bagi pengembang dalam memetakan masalah dominan, meningkatkan kualitas layanan, dan memaksimalkan pengalaman pengguna secara berkelanjutan.

Kata Kunci: Naïve Bayes; Analisis Sentimen; TF-IDF; Ulasan Pengguna; Klasifikasi Teks; Penambangan Data

Abstract– This study aims to automatically identify and classify user reviews of the X (Twitter) application into positive and negative sentiment categories to obtain an objective overview of user perceptions. The novelty of this research lies in the implementation of a complete text-preprocessing pipeline combined with TF-IDF-based feature weighting and comprehensive model evaluation using 10-fold cross-validation on a large-scale dataset of 5,000 Google Play Store reviews, resulting in a more accurate and stable sentiment classification model than previous approaches. The methodological stages include text preprocessing (case folding, stopword removal, stemming, tokenizing, character normalization, and special-character removal), numerical transformation using TF-IDF, and sentiment classification using the Naïve Bayes and Support Vector Machine (SVM) algorithms. The results indicate that the SVM model achieves the highest performance, with an average accuracy of 90.02%, precision 89.97%, recall 89.23%, and F1-score 89.53%, outperforming Naïve Bayes which obtained 87.14% accuracy, 88.13% precision, 85.29% recall, and 86.19% F1-score. Word-cloud visualization further demonstrates clear differences in dominant vocabulary between positive and negative sentiment classes. These findings confirm that SVM provides superior capability for text-based sentiment classification on application reviews. The implications of this research highlight that sentiment-analysis results can serve as valuable data-driven insights for developers to identify recurring issues, enhance service quality, and improve user experience sustainably.

Keywords: Naïve Baye; Sentiment Analysis; TF-IDF; User Reviews; Text Classification; Data Mining

1. PENDAHULUAN

Media sosial telah menjadi sarana penting bagi masyarakat untuk menyampaikan pendapat dan pengalaman secara terbuka. Aplikasi X (Twitter), sebagai salah satu platform yang paling populer di Indonesia, memiliki jumlah pengguna yang terus meningkat, yakni 13,8 juta pada tahun 2020 dan diprediksi mencapai 17,8 juta pada tahun 2025 [1]. Secara global, X menempati posisi teratas dalam jumlah unduhan, dengan Amerika Serikat mencapai 76,9 juta dan Jepang 58,95 juta, sedangkan Indonesia menempati urutan kelima dengan 18,45 juta unduhan [2]. Jumlah pengguna yang besar menjadikan X ruang interaksi digital yang aktif dan menjadi sumber opini publik terhadap berbagai isu [3]. Selain itu, X menyediakan akses tanpa pembatasan waktu maupun tempat, sehingga pengguna dapat secara kontinu menyampaikan opini mengenai berbagai hal, mulai dari kejadian sehari-hari hingga isu-isu serius, kapan saja dan di mana saja [4].

Namun, berdasarkan ulasan di Google Play Store, banyak pengguna menyampaikan keluhan terkait bug setelah pembaruan, penurunan kualitas tampilan antarmuka, serta menurunnya kenyamanan aplikasi, yang tercermin dari banyaknya rating bintang satu dan dua [5]. Ulasan tersebut menggambarkan persepsi nyata pengguna terhadap kualitas layanan aplikasi. Penelitian sebelumnya menunjukkan bahwa model Naïve Bayes yang diterapkan pada data ulasan aplikasi mampu mencapai akurasi 75,5% setelah menggunakan SMOTE [6] menandakan potensi metode ini dalam klasifikasi sentimen.

Rating dan ulasan pada aplikasi X berperan sebagai indikator utama tingkat kepuasan dan pengalaman pengguna. Melalui penilaian bintang dan komentar tertulis, pengguna menyampaikan persepsi mereka mengenai kinerja aplikasi, fitur, tampilan, serta perubahan yang terjadi. Informasi ini tidak hanya membantu pengguna lain dalam mempertimbangkan sebelum mengunduh aplikasi, tetapi juga menjadi masukan penting bagi pengembang

untuk melakukan perbaikan dan pengembangan layanan secara berkelanjutan. Data rating dan komentar yang tersedia di Google Play Store dapat dimanfaatkan sebagai sumber analisis sentimen untuk mengelompokkan opini pengguna ke dalam kategori positif dan negatif menggunakan algoritma Naïve Bayes, sehingga hasilnya dapat mendukung evaluasi dan peningkatan kualitas aplikasi secara lebih terarah.

Selain itu, penelitian terdahulu yang menggunakan pendekatan kualitatif menunjukkan bahwa persepsi publik terhadap konten digital dipengaruhi oleh jenis konten yang dikonsumsi. Konten visual, narasi pribadi, dan data berbasis fakta terbukti memberi pengaruh signifikan dalam membentuk opini positif, sedangkan konten sensasional dan provokatif sering memicu polarisasi dan penyebaran disinformasi [7]. Temuan tersebut menguatkan bahwa opini pengguna pada media sosial dan aplikasi digital mencerminkan pengalaman nyata mereka terhadap suatu layanan, termasuk persepsi terhadap perubahan maupun performa aplikasi.

Analisis sentimen menjadi salah satu pendekatan umum untuk menganalisis opini pengguna, dengan tujuan mengidentifikasi dan mengklasifikasikan opini dalam teks menjadi sentimen positif atau negatif. Dalam analisis sentimen, metode Naïve Bayes termasuk teknik klasifikasi yang populer karena kesederhanaannya, efektivitasnya dalam pengolahan teks, serta tingkat akurasi yang relatif tinggi ketika diaplikasikan pada dataset besar. Beberapa penelitian sebelumnya melaporkan akurasi Naïve Bayes mencapai 90% dengan pembagian data pelatihan dan pengujian 80:20 [8]. Penelitian sebelumnya menemukan bahwa penggunaan *paylater* dianggap kurang positif oleh masyarakat. Pada pengujian model menggunakan *confusion matrix*, algoritma *Naïve Bayes Classifier* menunjukkan tingkat akurasi yang lebih tinggi, yaitu sebesar 91%. Meskipun algoritma *Naïve Bayes* memiliki struktur yang sederhana, metode ini tetap efektif dalam melakukan klasifikasi teks, terutama untuk analisis sentimen. Metode ini sangat sesuai untuk digunakan pada dataset berukuran besar karena memberikan kinerja yang konsisten dan mampu mencapai tingkat akurasi yang cukup memuaskan. [9].

Klasifikasi ulasan sentimen membantu memahami persepsi pengguna terhadap aplikasi X secara lebih sistematis. Ulasan dari Google Play Store mencerminkan pengalaman pengguna serta menjadi indikator kualitas layanan, fitur, tampilan, dan pembaruan aplikasi. Dengan memisahkan ulasan ke dalam sentimen positif dan negatif, dapat diidentifikasi keluhan yang paling sering muncul serta aspek yang diapresiasi. Informasi ini berguna bagi pengembang dalam melakukan evaluasi dan perbaikan aplikasi, sekaligus menjadi referensi bagi calon pengguna sebelum mengunduh aplikasi tersebut.

Dalam penelitian ini, metode *Naïve Bayes* dipilih karena memiliki sejumlah keunggulan yang relevan untuk analisis teks. Algoritma ini dikenal sederhana namun efektif dalam menangani data dalam jumlah yang besar. Selain ringan secara komputasi, *Naïve Bayes* juga mampu memberikan performa yang cukup tinggi. Beberapa penelitian terdahulu menunjukkan bahwa metode ini dapat menghasilkan akurasi mencapai 82% dan presisi 93% [10] dan adapula yang menghasilkan angka 87% pada skenario pembagian data 80:20 dalam klasifikasi sentimen [11].

Berdasarkan kondisi tersebut, research gap muncul pada kurangnya penelitian yang secara khusus menganalisis sentimen pada ulasan aplikasi X dalam skala besar dengan pendekatan pemrosesan teks modern dan evaluasi model yang komprehensif melalui teknik validasi silang. Untuk mengatasi gap tersebut, penelitian ini bertujuan menerapkan algoritma Naïve Bayes untuk mengklasifikasikan ulasan pengguna aplikasi X menjadi sentimen positif dan negatif. Penelitian ini juga mengevaluasi performa model menggunakan akurasi, presisi, recall, dan F1-score setelah melalui tahap preprocessing data teks dan pembobotan TF-IDF.

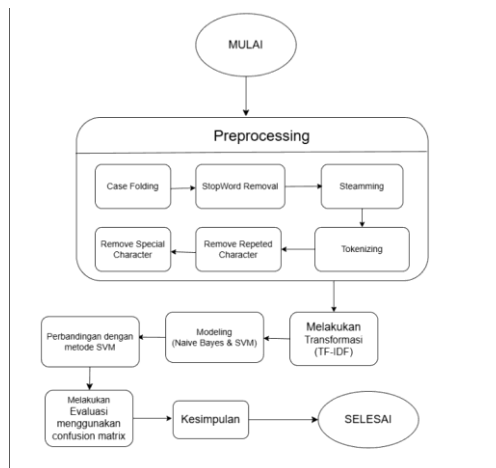
Kontribusi penelitian ini terletak pada penerapan metode Naïve Bayes dengan rangkaian preprocessing lengkap berbasis TF-IDF serta evaluasi model menggunakan teknik 10-fold cross-validation pada dataset besar berjumlah 5.000 ulasan aplikasi X dari Google Play Store. Pendekatan ini diharapkan mampu menyajikan gambaran yang lebih akurat dan terbaru mengenai persepsi pengguna serta menyediakan informasi berbasis data bagi pengembang aplikasi dalam meningkatkan kualitas layanan.

Tujuan penelitian ini adalah mengklasifikasikan ulasan pengguna aplikasi X ke dalam sentimen positif dan negatif menggunakan algoritma Naïve Bayes serta mengevaluasi performanya melalui akurasi, presisi, recall, dan F1-score. Penelitian ini diharapkan memberikan manfaat praktis berupa informasi berbasis data mengenai pandangan pengguna dan mendukung pengembang dalam meningkatkan kualitas layanan aplikasi.

2. METODE PENELITIAN

2.1 Tahapan Penelitian

Dalam penelitian ini, metode KDD (Knowledge Discovery in Database) dipilih oleh peneliti karena keunggulannya dalam mengidentifikasi pola secara sistematis dari data yang kompleks, sehingga mempermudah pemahaman terhadap informasi yang diperoleh [12]. Gambar 1 dibawah ini menunjukkan alur penelitian menggunakan metode KDD.



Gambar 1 Alur Penelitian

2.2 Data Selection

Tahap awal dalam proses KDD adalah pemilihan data, di mana dilakukan proses pengumpulan serta pemberian label pada data, data tersebut diperoleh dari situs Google Play Store.

2.3 Preprocessing

Pada tahap praproses, data mentah yang telah di kumpulkan akan diproses menjadi data yang siap digunakan pada tahap berikutnya. Tahapan ini sangat penting karena bertujuan untuk meningkatkan kualitas data sehingga analisis dapat dilakukan secara optimal. Tahap praproses mencakup enam langkah, yaitu:

2.3.1 Case folding

Merupakan tahapan pengolahan teks yang bertujuan untuk menyamakan bentuk huruf dalam sebuah teks menjadi satu format yang konsisten, biasanya dengan mengubah semua huruf menjadi huruf kecil (*lowercase*) [13]. Hal ini bertujuan agar model tidak menganggap kata yang sama dalam bentuk yang berbeda memiliki makna yang berbeda, seperti “Hari” dan “hari”.

2.3.2 StopWord Removal

Merupakan tahapan untuk menghapus kata-kata yang tidak bermakna atau kata penghubung yang tidak memengaruhi konteks kalimat [13]. Tujuannya untuk mengurangi *noise* dalam data dan fokus pada kata yang lebih bermakna.

2.3.3 Stemming

Merupakan tahapan untuk mengubah kata-kata modern atau singkatan menjadi bentuk dasar [13]. Proses ini dilakukan untuk mengurangi variasi kata yang berasal dari akar kata yang sama, sehingga kata-kata tetap dikenal sebagai entitas yang sama, seperti “menyukai” dan “disukai” akan dikembalikan menjadi kata suka.

2.3.4 Tokenizing

Merupakan tahapan memecah teks menjadi unit-unit yang lebih kecil, biasanya berupa kata atau frasa yang disebut token [13]. Proses ini memungkinkan model untuk menghitung frekuensi kemunculan kata-kata tertentu dalam teks, yang akan digunakan dalam langkah selanjutnya, seperti “aplikasi ini bagus” akan di ubah menjadi [“aplikasi”, “ini”, “bagus”].

2.3.5 Remove Repeted Character

Tahap ini merupakan tahapan untuk mendeteksi karakter berulang, Contoh, kata “siiiaaappppp” akan berubah menjadi “siap”. “Waaaahhh” menjadi “wah”, dan “maantuuulll” menjadi “mantul” [14]. Bertujuan untuk menormalisasikan bentuk kata agar konsisten.

2.3.6 Removed Special Character

Proses Menghapus Karakter Khusus merupakan salah satu tahap dalam praproses yang bertujuan untuk membersihkan data teks dari tanda baca, emotikon, dan angka. Tujuannya agar teks menjadi lebih bersih dan mudah diolah dalam proses analisis selanjutnya [15].

2.4 Transformasi (TF-IDF)

Term frequency-inverse document frequency (TF-IDF) adalah teknik yang menggabungkan *term frequency* (TF), yang menghitung kemunculan kata dalam satu dokumen, dan *inverse document frequency* (IDF), yang menilai pentingnya kata berdasarkan frekuensinya di seluruh dokumen [16]. Setiap kata dalam dokumen memiliki bobot lokal (TF) dan global (IDF). Bobot lokal menunjukkan seberapa sering kata muncul dalam satu dokumen, sedangkan bobot global menunjukkan tingkat kepentingannya di seluruh dokumen. Kedua bobot ini dikalikan untuk menghasilkan nilai “TFIDF” yang lebih representatif [17]. Perhitungan metode TF-IDF dijelaskan melalui rumus berikut.

$$TF - IDF_{t,d} = 1 + \log (FT_{t,d}) * \log \frac{N}{DF_T} \quad (1)$$

Dengan penjelasan sebagai berikut:

$TF - IDF_{t,d}$ = Bobot kata t dalam ulasan d pengguna terhadap aplikasi X .

$FT_{t,d}$ (*Term Frequency*) = Banyaknya kata t muncul dalam satu *review* pengguna terhadap aplikasi X .

N = Jumlah total ulasan yang dianalisis.

DF_T (*Document Frequency*) = Banyaknya ulasan yang mengandung kata t (kata kata umum seperti “aplikasi”, “twitter”, “fitur” akan memiliki nilai IDF lebih kecil).

2.5 Naïve Bayes

Naïve Bayes termasuk metode yang dapat diaplikasikan dalam proses data mining melalui pengelompokan data ke dalam kelas-kelas yang relevan. Perbedaan pada data dalam suatu pola atau model mampu diidentifikasi oleh *Naïve Bayes* secara efektif [18]. Sebagai salah satu teknik klasifikasi statistik, *Naïve Bayes* digunakan untuk memperkirakan probabilitas keanggotaan dalam suatu *class* tertentu [19]. Selain itu, *Naïve Bayes* juga berperan dalam pengelompokan opini individu terhadap suatu isu, dengan membedakan apakah opini tersebut bernada positif atau negatif [20]. Perhitungan algoritma *Naïve Bayes* dapat dilihat melalui rumus berikut:

$$P(H|X) = \frac{P(X|H).P(H)}{P(X)} \quad (2)$$

Keterangan:

X = Data dengan class yang belum diketahui

H = Hipotesis data X merupakan suatu class spesifik

(H/X) = Probabilitas hipotesis H berdasarkan kondisi X (*Posteriori probability*)

H = Probabilitas hipotesis H (*prior probability*)

(X/H) = Probabilitas X berdasarkan kondisi pada hipotesis H

X = Probabilitas X

2.6 Support Vector Machine

Support Vector Machine (SVM) adalah algoritma pembelajaran mesin yang digunakan untuk klasifikasi, deteksi anomali, dan analisis regresi. Klasifikasi dilakukan dengan menggunakan hyperplane yang memisahkan kelas positif dan negatif [21].

$$f(x) = \sum_{i=1}^n \alpha_i y_i K(x_i, x) + b \quad (3)$$

Keterangan:

α_i = bobot support vector

y_i = label kelas (+1 atau -1)

$K(x_i, x)$ = Fungsi kernal yang mengukur kemiripan data

b = bias

2.7 Confusion Matrix

Confusion matrix merupakan sebuah tabel yang memperlihatkan hasil klasifikasi yang dihasilkan oleh suatu model. Pada konteks analisis sentimen, confusion matrix menjelaskan jumlah prediksi yang tepat maupun yang keliru dari model tersebut. Dengan menggunakan confusion matrix, evaluasi kinerja model dapat dilakukan secara lebih komprehensif melalui perhitungan metrik seperti akurasi, presisi, *recall*, dan *F1-score* [22].

Tabel 1. Confusion Matrix

	Actual True	Actual False
True Predicated	True Positive (TP)	False Negative (FP)
False Predicated	False Negative (FN)	True Negative (TN)

Tabel 1 memperlihatkan bahwa *confusion matrix* terdiri atas empat kategori, yaitu True Positive (TP) yang merujuk pada data yang diprediksi positif dan prediksi tersebut tepat; False Positive (FP), yaitu data yang diperkirakan positif namun prediksinya keliru; True Negative (TN), yakni data yang diprediksi negatif dengan hasil yang akurat; serta False Negative (FN), yang merupakan data diprediksi negatif tetapi prediksinya salah. Setiap nilai dari kategori-kategori ini akan digunakan untuk menghitung metrik evaluasi selanjutnya.

- a) Akurasi, adalah ukuran yang menunjukkan tingkat kecocokan antara nilai aktual dengan hasil prediksi klasifikasi, baik pada kelas positif maupun negatif.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

- b) Presisi, adalah kemampuan model dalam memprediksi nilai positif yang benar

$$Precision = \frac{TP}{TP + FP} \quad (5)$$

- c) *Recall*, adalah ketepatan model untuk memprediksi nilai positif dari semua nilai actual yang benar benar positif

$$Recall = \frac{TP}{TP + FN} \quad (6)$$

- d) Nilai *F1-Score* adalah rata-rata harmonis dari presisi dan recall yang menunjukkan bahwa performa model dalam melakukan klasifikasi akan semakin baik apabila nilai tersebut mendekati angka satu.

$$F1 - Score = 2 \times \frac{Precision + Recall}{Precision \times Recall} \quad (7)$$

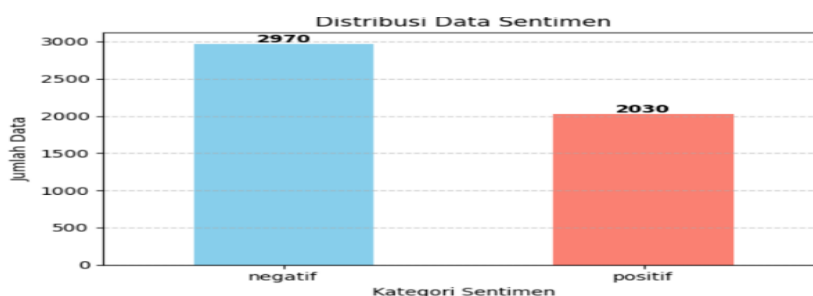
3. HASIL DAN PEMBAHASAN

3.1 Data selection

Pada proses pemilihan data kali ini, berhasil dikumpulkan sebanyak 5000 data. Data yang digunakan dalam penelitian ini berupa ulasan dan *rating*, dengan rentang *rating* 1-3 maka ulasan akan mendapat label negatif. Jika *rating* 4-5 maka akan mendapat label positif. Proses pelabelan dilakukan secara otomatis menggunakan fungsi pada python, pada proses ini didapatkan 2030 ulasan positif dan 2970 ulasan negatif. Hasil sampel pemilihan data akan di jabarkan dalam table 2.

Tabel 2. Sampel hasil tahap *Data Selection*

<i>Username</i>	<i>Score</i>	<i>Content</i>	<i>Sentimen</i>
Pengguna Google	1	1. Akun tiba-tiba gak bisa DM karena aKun dibatasi. Apa yang menjadi penyebab dibatasi?. 2. X setelah diupdate dmnya error, bug, ilang, 3.Akun tiba2 keluar. Dibatasi untuk follow. 4. Telpon ga berfungsi	Negatif
...
Pengguna Google	5	Media penyeimbang media massa konvensional, penyalur info netizen, semoga lebih independen lagi ke depan	Positif



Gambar 2. Distribusi Data Sentimen

Diagram batang yang digunakan memperjelas perbedaan jumlah antara kedua kategori sentimen, dengan batang berwarna biru muda untuk sentimen negatif dan merah muda untuk sentimen positif

3.2 Preprocessing

Setelah data diperoleh, maka selanjutnya dilakukan tahap praproses untuk mempermudah proses analisis. Tahap ini menggunakan beberapa library seperti sastrawi, NLTK dan sebagainya. Praproses pada penelitian ini meliputi penyeragaman huruf, penghapusan kata umum, pengembalian kata ke bentuk dasar, pemotongan kalimat menjadi kata-kata (token), menghapus karakter berulang, menghapus karakter khusus.

a) Case Folding

Pada tahap *case folding* ini bertujuan untuk mengubah semua huruf menjadi huruf kecil (*lowercase*) dengan menggunakan fungsi *str.lower()* untuk mengubah semua huruf menjadi huruf kecil. Hasilnya akan disimpan dalam *DataFrame* yang telah di perbaharui bernama *text_lower*. Hasil dari *case folding* akan di jelaskan pada tabel 3.

Tabel 3. Sampel hasil tahap *Case Folding*

<i>Content</i>	<i>Case Folding</i>
1. Akun tiba-tiba gak bisa DM karena aKun dibatasi. Apa yang menjadi penyebab dibatasi?. 2. X setelah diupdate dmnya error, bug, ilang, 3.Akun tiba2 keluar. Dibatasi untuk follow. 4. Telpn ga berfungsi	1. akun tiba-tiba gak bisa dm karena akun dibatasi. apa yang menjadi penyebab dibatasi?. 2. x setelah diupdate dmnya error, bug, ilang, 3.akun tiba2 keluar. dibatasi untuk follow. 4. telpon ga berfungsi
...	...
Media penyeimbang media massa konvensional, penyalur info netizen, semoga lebih independen lagi ke depan	media penyeimbang media massa konvensional, penyalur info netizen, semoga lebih independen lagi ke depan

b) StopWord Removal

Tahap ini bertujuan untuk menghapus kata kata umum atau *stopwords* yang tidak memiliki makna penting dalam analisis dengan mengunduh *library* nltk lalu mengunduh daftar *stopword* serta mengambil *stopword* bahasa indonesia dari nltk dan disimpan dalam bentuk set untuk efisiensi pencarian. Hasilnya akan disimpan dalam kolom baru yaitu *content_no_stopwords*.

Tabel 4. Sampel hasil tahap *StopWord Removal*

<i>Case Folding</i>	<i>StopWord Removal</i>
1. akun tiba-tiba gak bisa dm karena akun dibatasi. apa yang menjadi penyebab dibatasi?. 2. x setelah diupdate dmnya error, bug, ilang, 3.akun tiba2 keluar. dibatasi untuk follow. 4. telpon ga berfungsi	1. akun gak dm akun dibatasi. penyebab dibatasi?. 2. x diupdate dmnya error, bug, ilang, 3.akun tiba2 keluar. dibatasi follow. 4. telpon ga berfungsi
...	...
media penyeimbang media massa konvensional, penyalur info netizen, semoga lebih independen lagi ke depan	media penyeimbang media massa konvensional, penyalur info netizen, semoga independen

c) *Steaming*

Pada tahap ini mengubah kata turunan menjadi bentuk dasarnya dengan menghilangkan imbuhan yang ada sesuai dengan aturan tertentu. Proses ini dilakukan untuk mengurangi variasi kata yang berasal dari akar kata yang sama. Hasilnya akan disimpan ke dalam *DataFrame* dengan nama baru yaitu *content_stemmed*.

Tabel 5. Sampel hasil tahap *Steaming*

<i>StopWord Removal</i>	<i>Steaming</i>
1. akun gak dm akun dibatasi. penyebab dibatasi?. 2. x diupdate dmnya error, bug, ilang, 3.akun tiba2 keluar. dibatasi follow. 4. telpon ga berfungsi	1 akun gak dm akun batas sebab batas 2 x diupdate dmnya error bug ilang 3 akun tiba2 keluar batas follow 4 telpon ga fungsi
...	...
media penyeimbang media massa konvensional, penyalur info netizen, semoga independen	media imbang media massa konvensional salur info netizen semoga independen

d) *Tokenizing*

Pada tahap ini merupakan proses memecah teks menjadi unit-unit kecil yang disebut token yang biasanya berupa kata atau frasa. Hasilnya akan disimpan dalam *DataFrame* yang telah diperbaharui dengan kolom baru bernama *content_tokenized*.

Tabel 6. Sampel hasil tahap *Tokenizing*

<i>Steaming</i>	<i>Tokenizing</i>
1 akun gak dm akun batas sebab batas 2 x diupdate dmnya error bug ilang 3 akun tiba2 keluar batas follow 4 telpon ga fungsi	['1', 'akun', 'gak', 'dm', 'akun', 'batas', 'sebab', 'batas', '2', 'x', 'diupdate', 'dmnya', 'error', 'bug', 'ilang', '3', 'akun', 'tiba2', 'keluar', 'batas', 'follow', '4', 'telpon', 'ga', 'fungsi']
...	...
media imbang media massa konvensional salur info netizen semoga independen	['media', 'imbang', 'media', 'massa', 'konvensional', 'salur', 'info', 'netizen', 'semoga', 'independen']

e) *Remove Repeted Character*

Pada tahap ini akan dilakukan menghapus karakter yang di tulis berulang-ulang yang tidak di perlukan dalam proses. Hasilnya akan disimpan pada kolom baru yaitu *content_no_repeat*.

Tabel 7. Sampel hasil tahap *Remove Repeted Character*

<i>Tokenizing</i>	<i>Remove Repeted Character</i>
['1', 'akun', 'gak', 'dm', 'akun', 'batas', 'sebab', 'batas', '2', 'x', 'diupdate', 'dmnya', 'error', 'bug', 'ilang', '3', 'akun', 'tiba2', 'keluar', 'batas', 'follow', '4', 'telpon', 'ga', 'fungsi']	['1', 'akun', 'gak', 'dm', 'akun', 'batas', 'sebab', 'batas', '2', 'x', 'diupdate', 'dmnya', 'eror', 'bug', 'ilang', '3', 'akun', 'tiba2', 'keluar', 'batas', 'folow', '4', 'telpon', 'ga', 'fungsi']
...	...
['media', 'imbang', 'media', 'massa', 'konvensional', 'salur', 'info', 'netizen', 'semoga', 'independen']	['media', 'imbang', 'media', 'masa', 'konvensional', 'salur', 'info', 'netizen', 'semoga', 'independen']

f) Remove Special Character

Tahapan ini bertujuan untuk menghapus karakter *non-alfabet* dan simbol simbol lainnya akan dihapus karena tidak memiliki nilai dan dapat mengganggu proses selanjutnya. Hasilnya akan disimpan dalam kolom baru yaitu *content_cleaned*.

Tabel 8. Sampel hasil tahap *Remove Special Character*

<i>Remove Repeted Character</i>	<i>Remove Special Character</i>
['1', 'akun', 'gak', 'dm', 'akun', 'batas', 'sebab', 'batas', '2', 'x', 'diupdate', 'dmnya', 'eror', 'bug', 'ilang', '3', 'akun', 'tiba2', 'keluar', 'batas', 'folow', '4', 'telpon', 'ga', 'fungsi']	['akun', 'gak', 'dm', 'akun', 'batas', 'sebab', 'batas', 'x', 'diupdate', 'dmnya', 'eror', 'bug', 'ilang', 'akun', 'tiba', 'keluar', 'batas', 'folow', 'telpon', 'ga', 'fungsi']
...	...
['media', 'imbang', 'media', 'masa', 'konvensional', 'salur', 'info', 'netizen', 'semoga', 'independen']	['media', 'imbang', 'media', 'masa', 'konvensional', 'salur', 'info', 'netizen', 'semoga', 'independen']

3.3 Transformasi (TF-IDF)

Tahap ini mengubah data teks menjadi bentuk numerik menggunakan TF-IDF. Prosesnya mencakup instalasi *pandas* dan *scikit-learn*, memuat dataset, serta menerapkan *TfidfVectorizer* pada kolom *“text_token_joined”* untuk menghasilkan *tfidf_matrix*. Fitur kata diperoleh melalui *get_feature_names_out* dan disimpan dalam *DataFrame* berisi nilai TF, DF, IDF, dan TF-IDF. Ekstraksi fitur menggunakan parameter *ngram_range=(1,1)*, *use_idf=True*, *smooth_idf=True*, dan pola tokenisasi *(?u)\b\w+\b*. Pemodelan klasifikasi dilakukan dengan *Multinomial Naïve Bayes* menggunakan *alpha=1.0*, *fit_prior=True*, dan *class_prior=None*.

Tabel 9. Tranformasi TF-IDF

Kata	TF	DF	IDF	TF-IDF
aplikasi	0.142857	662	3.020618	0.184871
bagus	0.142857	580	3.152642	0.192952
bantu	0.142857	63	5.358510	0.327958
cari	0.142857	25	6.259297	0.383089
info	0.142857	27	6.185189	0.378553

3.4 Modeling

a) Naïve Bayes

Tahapan ini melibatkan model *Naïve Bayes* yang akan divalidasi menggunakan teknik *10-fold cross-validation*, dengan cara mengimpor *library pandas* dan *scikit-learn* untuk analisis data, *TfidfVectorizer*, *Kfold* dan *cross_validate* untuk pengujian, *MultinomialNB* untuk klasifikasi *Naïve Bayes* serta *tabulate* untuk tabel hasil. Memuat dataset dan menerapkan *TfidfVectorizer* pada kolom *“content_cleaned”*, selanjutnya menerapkan *MultinomialNB* dan melakukan *10-fold cross-validation*. Hasil evaluasi berupa akurasi, presisi, *recall* dan *f1-score* disimpan dalam *DataFrame* baru, kemudian ditampilkan dalam format tabel.

Tabel 10. Hasil model Naive Bayes

Fold	Accuracy	Precision	Recall	F1-Score
Fold- 1	86.8%	87.54%	84.76%	85.71%
Fold- 2	89%	89.71%	86.47%	87.69%
Fold- 3	87.8%	89.05%	85.86%	86.87%
Fold- 4	87.2%	88.31%	85.43%	86.33%
Fold- 5	88.2%	88.84%	86.92%	87.59%
Fold- 6	86%	86.99%	84.6%	85.29%
Fold- 7	86.4%	88.09%	85.13%	85.79%
Fold- 8	88.8%	89.8%	87.14%	88.04%
Fold- 9	87%	88.29%	85.43%	86.25%
Fold- 10	84.2%	84.71%	81.14%	82.32%
Mean	87.14%	88.13%	85.29%	86.19%

Tabel 10 memperlihatkan hasil pengujian, dimana nilai akurasi tertinggi dicapai pada fold-2 sebesar 89%. Rata-rata performa dari semua fold menunjukkan akurasi sebesar 87,14%, presisi 88,13%, recall 85,29%, dan F1-Score 86,19%. Kemampuan *Naïve Bayes* dalam mengklasifikasikan data terbukti cukup efektif melalui hasil tersebut.

Berdasarkan hasilnya adapun beberapa kelebihan dari model Naive Bayes ini yaitu, model ini memiliki metode yang sederhana dan efektif dalam pengolahan teks, ringan secara komputasi dan memberikan kinerja konsisten pada dataset berukuran besar. Namun selain memiliki kelebihan model Naive Bayes juga memiliki kelemahan yaitu, mengansumsi bahwa setiap fitur (kata) saling indenpenden, yang tidak selalu realistis dalam data teks.

b) Support Vector Machine

Tahapan ini melibatkan model SVM linear yang divalidasi menggunakan 10-fold cross-validation, dengan cara mengimpor library pandas dan scikit-learn untuk analisis data, *TfidfVectorizer*, *KFold*, dan *cross_validate* untuk pengujian, SVC untuk klasifikasi SVM, serta tabulate untuk tabel hasil. Dataset dimuat dan kolom “*content_cleaned*” diubah menjadi representasi TF-IDF, selanjutnya model SVM diterapkan dan dievaluasi menggunakan 10-fold cross-validation. Hasil evaluasi berupa akurasi, presisi, recall, dan F1-score disimpan dalam *DataFrame* baru, kemudian ditampilkan dalam format tabel.

Tabel 11. Hasil model *Support Vector Machine*

Fold	Accuracy	Precision	Recall	F1-Score
Fold- 1	92.6%	92.42%	92.03%	92.22%
Fold- 2	91.6%	90.88%	91.07%	90.97%
Fold- 3	90.2%	90.34%	89.23%	89.69%
Fold- 4	90%	90.5%	88.85%	89.47%
Fold- 5	90%	89.75%	89.64%	89.7%
Fold- 6	88.2%	88.32%	87.48%	87.81%
Fold- 7	90.2%	90.5%	89.67%	89.97%
Fold- 8	91%	90.88%	90.4%	90.62%
Fold- 9	90.2%	90.4%	89.48%	89.85%
Fold- 10	86.2%	85.75%	84.46%	85.01%
Mean	90.02%	89.97%	89.23%	89.53%

Tabel 11 memperlihatkan hasil pengujian, dimana nilai akurasi tertinggi dicapai pada fold-1 sebesar 92.42%. Rata-rata performa dari semua fold menunjukkan akurasi sebesar 92.6%, presisi 92.42%, recall 92.03%, dan F1-Score 92.22%. Kemampuan *Naïve Bayes* dalam mengklasifikasikan data terbukti cukup efektif melalui hasil tersebut.

Berdasarkan hasil pengujian, model SVM ini memiliki beberapa kelebihan yaitu, lebih efektif menangani data yang berdimensi tinggi seperti TF-IDF dan lebih kuat terhadap data yang tidak seimbang dan fitur sparse, sehingga performanya stabil. Selain itu model ini juga memiliki kekurangannya yaitu sensitif terhadap pemilihan parameter (kernel) yang tepat serta memakan waktu penelitian komputasi yang lebih lama terutama ketika menggunakan kernel non-linear.

3.5 Keunggulan SVM dibandingkan dengan Naive Bayes

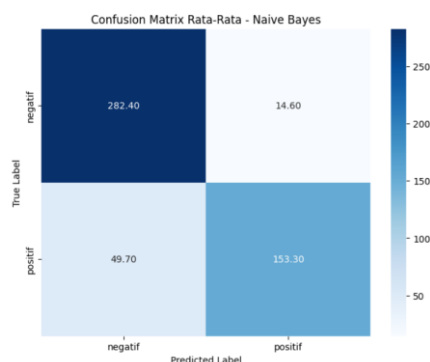
Berdasarkan hasil 10-Fold Cross-Validation, model Support Vector Machine (SVM) Linear terbukti unggul secara signifikan dibandingkan dengan *Naïve Bayes* dalam tugas klasifikasi sentimen. Keunggulan SVM terlihat

pada semua metrik rata-rata, di mana Akurasi rata-ratanya mencapai 90.02% (melampaui Naïve Bayes 87.14%), dan F1-Score rata-ratanya sebesar 89.53% (lebih tinggi dari Naïve Bayes 86.19%). Selain itu, kinerja maksimal SVM mencapai Akurasi tertinggi 92.60% pada Fold-1, jauh melampaui puncak kinerja Naïve Bayes 89.00%. Data ini secara konsisten menegaskan bahwa SVM lebih efektif menangani data teks berdimensi tinggi seperti TF-IDF, sehingga mampu memisahkan kelas positif dan negatif secara lebih akurat. Tidak seperti Naïve Bayes yang mengasumsikan setiap fitur saling independen, SVM dapat menangkap hubungan antar kata yang memengaruhi konteks sentimen. Selain itu, SVM lebih kuat terhadap data yang tidak seimbang dan fitur yang bersifat sparse, sehingga performanya lebih stabil.

3.6 Evaluasi

a) Evaluasi Model Naive Bayes

Tahap ini merupakan proses evaluasi kinerja model Naive Bayes dalam melakukan klasifikasi data, yang dilakukan menggunakan *confusion matrix* dengan cara mengimpor *library pandas* dan *scikit-learn* untuk analisis data, *TfidfVectorizer* untuk representasi numerik, *KFold* dan *cross_validate* untuk pengujian, serta *MultinomialNB* untuk klasifikasi *Naïve Bayes*. Mengimpor *library* untuk tahap evaluasi, *confusion matrix* untuk membuat matriks, *matplotlib.pyplot* untuk visualisasi plot, dan *seaborn* untuk menampilkan *heatmap*. Menjalankan semua fungsi dan memperoleh hasil evaluasi model *Naïve Bayes*.

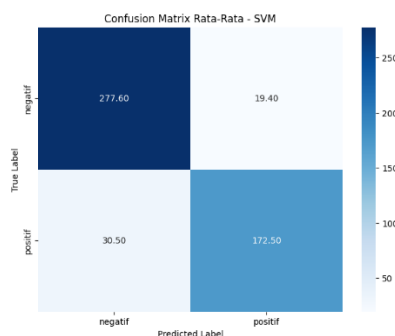


Gambar 3. Naive Bayes

Hasil dari *confusion matrix* itu menunjukkan performa model klasifikasi *Naïve Bayes* dalam membedakan antara data berlabel positif dan negatif. Dari matriks tersebut, terlihat bahwa model mampu mengklasifikasikan 282,40 data negatif dengan benar (*True Negative*) dan hanya salah memprediksi 14,60 data negatif sebagai positif (*False Positive*). Sementara itu, model juga berhasil memprediksi 153,30 data positif secara benar (*True Positive*), namun masih terdapat 49,70 data positif yang diprediksi sebagai negatif (*False Negative*).

b) Evaluasi Model SVM

Tahap ini merupakan proses evaluasi kinerja model SVM dalam melakukan klasifikasi data, yang dilakukan menggunakan *confusion matrix*, dengan cara mengimpor *library pandas* dan *scikit-learn* untuk analisis data, *TfidfVectorizer* untuk representasi numerik, *KFold* untuk pembagian data, serta *SVC* untuk klasifikasi SVM. Library tambahan *confusion_matrix* digunakan untuk membuat matriks evaluasi, *matplotlib.pyplot* untuk visualisasi plot, dan *seaborn* untuk menampilkan *heatmap*. Seluruh fungsi dijalankan untuk menghitung dan menampilkan *confusion matrix* rata-rata, sehingga performa model SVM dapat dievaluasi secara menyeluruh.



Gambar 4. Support Vector Machine

Hasil dari *confusion matrix* itu menunjukkan performa model klasifikasi *Naïve Bayes* dalam membedakan antara data berlabel positif dan negatif. Dari matriks tersebut, terlihat bahwa model mampu mengklasifikasikan

277.60 data negatif dengan benar (*True Negative*) dan hanya salah memprediksi 19.40 data negatif sebagai positif (*False Positive*). Sementara itu, model juga berhasil memprediksi 172.50 data positif secara benar (*True Positive*), namun masih terdapat 30.50 data positif yang diprediksi sebagai negatif (*False Negative*). Hal ini menunjukkan bahwa model klasifikasi sentimen cukup konsisten dalam memprediksi sentimen berdasarkan skor yang diberikan pengguna pada data yang telah diproses. Ketika pengguna memberikan skor tinggi (positif), model juga cenderung mengklasifikasikannya sebagai sentimen positif, sehingga tidak terjadi error analisis.

3.7 Perbandingan Hasil Dengan Penelitian Sebelumnya

Hasil penelitian menunjukkan bahwa SVM mencapai akurasi 90.02% dan F1-score 89.53%, lebih tinggi dibandingkan Naïve Bayes yang memperoleh 87.14% dan 86.19%. Kinerja ini sejalan dengan temuan studi sebelumnya yang menempatkan SVM sebagai model yang unggul untuk data teks berdimensi tinggi. Performa Naïve Bayes dalam penelitian ini masih kompetitif. Hasilnya lebih baik dari temuan Eskiyaturofikh dan Suryono (75.5%) [6]. Kinerja SVM pada penelitian ini juga konsisten dengan hasil Thoriq *et al.* dan Rismanah *et al.* [1], [21]. Dengan dataset besar (5.000 ulasan) dan evaluasi 10-fold cross-validation, penelitian ini memberikan bukti empiris yang lebih stabil dibandingkan pendekatan pembagian data sederhana. Secara keseluruhan, hasil ini menegaskan keunggulan SVM dan menunjukkan keterbatasan Naïve Bayes dalam menangani kompleksitas fitur linguistik pada ulasan aplikasi.

3.8 Implikasi hasil penelitian

Penelitian ini menunjukkan bahwa model SVM efektif dalam memonitor persepsi pengguna terhadap aplikasi X secara real-time, dengan akurasi yang tinggi. Analisis sentimen mengidentifikasi kata-kata negatif seperti “error”, “bug”, “login”, dan “update” sebagai indikator area fungsional yang perlu diperbaiki, sementara kata-kata positif seperti “bagus”, “mantap”, dan “informasi” menandakan fitur yang diterima baik oleh pengguna dan dapat dipertahankan atau dikembangkan lebih lanjut. Temuan ini memberikan panduan praktis bagi pengembang untuk meningkatkan kualitas aplikasi berdasarkan umpan balik pengguna.

Secara akademis, penelitian ini menegaskan bahwa penerapan pipeline preprocessing yang lengkap, pembobotan TF-IDF, dan evaluasi 10-fold cross-validation dapat meningkatkan stabilitas dan akurasi model analisis sentimen. Temuan ini juga menunjukkan perlunya eksplorasi metode yang lebih canggih, seperti Random Forest, LSTM, atau BERT, serta penerapan teknik penyeimbangan data seperti SMOTE atau ADASYN untuk menangani ketidakseimbangan data pada penelitian selanjutnya.



Gambar 5. *WordCloud* Sentimen Positif

Gambar 5. menunjukkan *word cloud* untuk kelas sentimen positif, di mana kata-kata seperti “bagus”, “good”, “mantap”, “suka”, “oke”, “keren”, dan “nice” tampak mendominasi. Kemunculan kata-kata ini mengindikasikan bahwa mayoritas pengguna memberikan ulasan yang memuji kualitas aplikasi, mengapresiasi fitur yang disediakan, serta merasa puas dan terbantu dengan layanan yang diberikan. Selain itu, kata seperti “informasi”, “akun”, dan “update” juga muncul dalam konteks positif, menunjukkan bahwa fitur-fitur tersebut dianggap bermanfaat.



Gambar 6. *WordCloud* Sentimen Negatif

Gambar 6. Menunjukkan word cloud untuk kelas sentimen negatif. Dalam visualisasi ini, kata-kata seperti “gak”, “akun”, “login”, “aplikasi”, “error”, “bug”, dan “update” muncul dengan ukuran besar, menandakan

frekuensi kemunculan yang tinggi. Ini mencerminkan banyaknya keluhan pengguna yang berkaitan dengan masalah teknis, terutama dalam hal login akun dan kesalahan sistem. Kata seperti “tolong”, “teknis”, “salah”, dan “jelek” mengindikasikan adanya ketidakpuasan pengguna terhadap performa aplikasi.

4. KESIMPULAN

Penelitian ini berhasil menjawab masalah utama dengan mengidentifikasi dan mengklasifikasikan sentimen secara otomatis pada 5.000 ulasan pengguna aplikasi X di Google Play Store ke dalam kategori positif dan negatif. Proses ini didukung oleh model yang dikembangkan melalui tahapan preprocessing dan pembobotan fitur TF-IDF yang ketat. Penelitian ini lebih unggul dari studi terdahulu karena menggunakan metodologi evaluasi yang lebih ketat, yaitu 10-Fold Cross-Validation, untuk menjamin akurasi dan stabilitas model, berbeda dengan studi sebelumnya yang umumnya menggunakan pembagian data sederhana (misalnya 80:20). Hasil utama menunjukkan bahwa model Support Vector Machine (SVM) Linear terbukti unggul secara signifikan dibandingkan Naïve Bayes, dengan Akurasi rata-rata 90,02% dan F1-score 89,53%, melampaui Naïve Bayes (Akurasi 87,14%). Keunggulan ini menegaskan efektivitas SVM dalam menangani data teks berdimensi tinggi dan kemampuannya memisahkan kelas sentimen secara lebih akurat, memberikan gambaran objektif mengenai persepsi pengguna. Meskipun demikian, penelitian ini memuat keterbatasan, terutama terkait ketidakseimbangan jumlah data sentimen (data negatif lebih banyak) dan penggunaan satu domain dataset saja. Oleh karena itu, arah penelitian lanjutan disarankan untuk mengatasi keterbatasan ini dengan menerapkan teknik balancing data seperti SMOTE guna meningkatkan stabilitas model. Selain itu, direkomendasikan untuk membandingkan model dengan algoritma yang lebih canggih (seperti Random Forest, LSTM, atau BERT), serta menambahkan analisis lanjutan seperti Topic Modeling untuk mengidentifikasi faktor dominan yang lebih mendalam yang memengaruhi sentimen pengguna.

REFERENCES

- [1] E. M. Thoriq, D. E. Ratnawati, and B. Rahayudi, “Analisis Sentimen Opini Publik pada Media Sosial Twitter terhadap Vaksin Covid-19 menggunakan Algoritma Support Vector Machine dan Term Frequency-Inverse Document Frequency,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 12, pp. 5349–5355, 2021, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [2] P. Rainer, “Negara Dengan Pengguna Twitter Terbesar Dunia,” [data.goodstats.id](https://data.goodstats.id/statistic/negara-dengan-pengguna-twitter-terbesar-dunia-NcHsT?utm_source=chatgpt.com). [Online]. Available: https://data.goodstats.id/statistic/negara-dengan-pengguna-twitter-terbesar-dunia-NcHsT?utm_source=chatgpt.com
- [3] B. J. Manurung, B. Parga Zen, Y. Setiya Rafika Nur, R. Claudio Felle, and E. Ahmad Firdaus, “Analisis Sentimen Pengguna Twitter terhadap Konflik Rusia-Ukraina Menggunakan Naïve Bayes dan Lexicon Based Features,” *J. Inform.*, vol. 4, no. 1, pp. 8–16, 2025, doi: 10.57094/ji.v4i1.2574.
- [4] F. Herlando, A. R. Dzikrillah, F. Nufairi, E. Sinduningrum, and M. Sholeh, “Analisis Perbandingan Sentiment Dan Perbincangan Netizen Terhadap Twitter Pasca Pergantian Nama,” *JUPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 9, no. 1, pp. 360–367, 2024, doi: 10.29100/jupi.v9i1.4934.
- [5] E. Aulia, “Analisis User Experience Aplikasi Twitter Menggunakan Metode User Experience Questionnaire (Ueq),” *J. Tek. dan Sci.*, vol. 3, no. 1, pp. 31–39, 2024, doi: 10.56127/jts.v3i1.907.
- [6] E. Eskiyaturrofikoh and R. R. Suryono, “Analisis Sentimen Aplikasi X Pada Google Play Store Menggunakan Algoritma Naïve Bayes Dan Support Vector Machine (Svm),” *JUPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 9, no. 3, pp. 1408–1419, 2024, doi: 10.29100/jupi.v9i3.5392.
- [7] A. Murodi, Y. Ari, and N. Setiyoko, “The Role of Social Media in Shaping Public Perception of Politics,” *Formosa J. Appl. Sci.*, vol. 3, no. 12, pp. 4799–4812, 2024, [Online]. Available: <https://journal.formosapublisher.org/index.php/fjas>
- [8] T. Dzulkarnain, D. E. Ratnawati, and B. Rahayudi, “Penggunaan Metode Naïve Bayes Classifier pada Analisis Sentimen Penilaian Masyarakat Terhadap Pelayanan Rumah Sakit di Malang,” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 11, no. 5, pp. 993–1000, 2024, doi: 10.25126/jtiik.2024117979.
- [9] Alfandi Safira and F. N. Hasan, “Analisis Sentimen Masyarakat Terhadap Paylater Menggunakan Metode Naive Bayes Classifier,” *Zo. J. Sist. Inf.*, vol. 5, no. 1, pp. 59–70, 2023, doi: 10.31849/zn.v5i1.12856.
- [10] D. Normawati and S. A. Prayogi, “Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter,” *J. Sains Komput. Inform. (J-SAKTI)*, vol. 5, no. 2, pp. 697–711, 2021.
- [11] A. E. Perkasa and A. N. Putri, “Penerapan Naïve Bayes Untuk Analisis Sentimen Pada Ulasan Aplikasi Mobile Legends,” *BUILD. Informatics, Technol. Sci.*, vol. 6, no. 4, p. 2152–2164, 2025, doi: 10.47065/bits.v6i4.6507.
- [12] S. Ramos, J. Soares, S. Cembranel, I. Tavares, Z. Foroozandeh, and Z. Vale, “ScienceDirect ScienceDirect Data Data mining mining techniques techniques for for electricity electricity customer customer characterization characterization,” *Procedia Comput. Sci.*, vol. 186, pp. 475–488, 2021, doi: 10.1016/j.procs.2021.04.168.
- [13] S. Andini, R. Kurniawan, S. Anwar, and K. Cirebon, “ANALISIS SENTIMEN PENGGUNA X MENGENAI OPINI,” vol. 13, no. 2, pp. 665–671, 2025.
- [14] S. Khomsah, “Model Text-Preprocessing Komentar Youtube Dalam Bahasa Indonesia,” no. August 2020, 2021, doi: 10.29207/resti.v4i4.2035.
- [15] M. Alfonso, D. B. Rarasati, P. Studi, T. Informatika, F. Teknologi, and U. B. Mulia, “Sentiment Analysis of 2024 Presidential Candidates Election Using SVM Algorithm,” vol. 06, no. 02, pp. 110–115, 2024.

- [16] P. A. Henríquez and F. Alessandri, "Heliyon Analyzing digital societal interactions and sentiment classification in Twitter (X) during critical events in Chile," *Heliyon*, vol. 10, no. 12, p. e32572, 2024, doi: 10.1016/j.heliyon.2024.e32572.
- [17] A. Afda, "Analisa Dokumen Menggunakan Metode TF-IDF," vol. 5, no. 5, pp. 466–470, 2024.
- [18] T. Tinaliah and T. Elizabeth, "Analisis Sentimen Ulasan Aplikasi PrimaKu Menggunakan Metode Support Vector Machine," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 9, no. 4, pp. 3436–3442, 2022, doi: 10.35957/jatisi.v9i4.3586.
- [19] E. Fitri, "Analisis Sentimen Terhadap Aplikasi Ruangguru Menggunakan Algoritma Naive Bayes, Random Forest Dan Support Vector Machine," *J. Transform.*, vol. 18, no. 1, pp. 71–80, 2020, doi: 10.26623/transformatika.v18i1.2317.
- [20] A. A. Permana, W. A. Noviyanto, and D. A. Kristiyanti, "Sentimen Analisis Opini Masyarakat Terhadap UMKM Pada Media Sosial Twitter Dengan Metode Naïve Bayes Classifier," *J. Minfo Polgan*, vol. 12, no. 1, pp. 163–170, 2023, doi: 10.33395/jmp.v12i1.12337.
- [21] S. N. Rismanah *et al.*, "PENERAPAN ALGORITMA SUPPORT VECTOR MACHINE DALAM MENGANALISIS SENTIMEN ULASAN PELANGGAN," vol. 8, no. 1, pp. 406–412, 2024.
- [22] M. Rahardi, A. Aminuddin, F. F. Abdulloh, and R. A. Nugroho, "Sentiment Analysis of Covid-19 Vaccination using Support Vector Machine in Indonesia," no. January, 2022, doi: 10.14569/IJACSA.2022.0130665.