Jumistik
E-ISSN: 2964-3953 P-ISSN: 2964-3888
Jurnal Manajemen Informatika, Sistem Informasi dan Teknologi Komputer

Vol.4 No.2 Desember 2025
Hal 664-672
Https://ojs.amiklps.ac.id

PENERAPAN *MACHINE LEARNING* UNTUK ANALISIS SENTIMEN AGODA DENGAN ALGORITMA *KNN, NAIVE BAYES* DAN *SVM*

Popi Rindiani[1*], Jeni Fatmawati[2], Sofian Wira Hadi[3], Agung Fazriansyah[4], Lady Agustin Fitriana[5*]
[1,2]Informatika, Fakultas Teknik dan Informatika, Universitas Bina Sarana Informatika
[3,4,5]Sistem Informasi, Fakultas Teknik dan Informatika, Universitas Bina Sarana Informatika
E-mail: 15225010@bsi.ac.id[1*], 15225008@bsi.ac.id[2], Sofian.sod@bsi.ac.id[3], Agung.fzr@bsi.ac.id[4], Lady.lag@bsi.ac.id[5*]

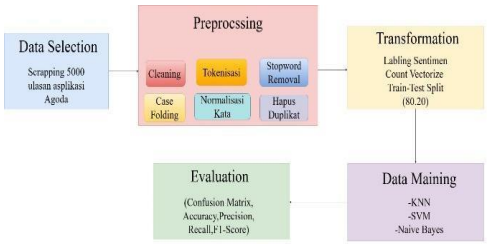
INFO ARTIKEL

Sejarah Artikel
Diterima : 30/10/2025
Direvisi : 10/11/2025
Diterbitkan : 01/12/2025

***Corresponding author**
15225010@bsi.ac.id

DOI: 10.70247/jumistik.v4i2.232
Https://ojs.amiklps.ac.id

GRAPHICAL ABSTRACT



ABSTRACT

The rapid advancement of digital technology has significantly transformed the tourism industry, particularly in online hotel booking services such as Agoda. The large volume of user reviews available on this platform serves as a valuable data source for analyzing customer satisfaction and perceptions. This study aims to conduct sentiment analysis on 5,000 Indonesian-language user reviews from the Agoda mobile application by comparing the performance of three machine learning algorithms: K-Nearest Neighbors (KNN), Naïve Bayes, and Support Vector Machine (SVM). Data were collected using a web scraping technique from Google Play Store and processed through several preprocessing stages, including cleaning, case folding, tokenization, word normalization, stopword removal, and stemming. Text representation was performed using the CountVectorizer method, with an 80:20 ratio of training and testing datasets. The experimental results show that the SVM algorithm achieved the highest performance with an accuracy of 84.1%, outperforming Naïve Bayes (65.3%) and KNN (61.7%). These findings indicate that SVM demonstrates superior capability in classifying positive, negative, and neutral sentiments in Indonesian text. The results of this research are expected to contribute to the development of sentiment analysis models and support service quality improvement based on user feedback.

***Keywords**: Agoda, Sentiment Analysis, Machine Learning, Support Vector Machine, User Reviews*

ABSTRAK

Perkembangan teknologi digital telah mendorong transformasi signifikan dalam industri pariwisata, termasuk pada layanan pemesanan hotel *daring* seperti Agoda. Banyaknya ulasan pengguna yang diunggah melalui *platform* ini menjadi sumber data potensial untuk menganalisis kepuasan dan persepsi pelanggan. Penelitian ini bertujuan melakukan analisis sentimen terhadap 5.000 ulasan pengguna aplikasi Agoda berbahasa Indonesia dengan membandingkan kinerja tiga algoritma *machine learning, yaitu K-Nearest Neighbors (KNN), Naïve Bayes, dan Support Vector Machine (SVM).* Data diperoleh melalui teknik *web scraping* dari *Google Play Store*, kemudian diproses melalui tahapan *preprocessing* yang mencakup *cleaning, case folding*, tokenisasi, normalisasi kata, *stopword removal*, dan *stemming.* Representasi teks dilakukan menggunakan metode *CountVectorizer* dengan pembagian data latih dan data uji sebesar 80:20. Hasil pengujian menunjukkan bahwa algoritma SVM menghasilkan *performa* terbaik dengan akurasi 84,1%, melampaui *Naïve Bayes* (65,3%) dan *KNN* (61,7%). Temuan ini

menunjukkan bahwa *SVM* lebih unggul dalam mengklasifikasikan sentimen positif, negatif, dan netral pada teks berbahasa Indonesia. Hasil penelitian ini diharapkan dapat menjadi referensi dalam pengembangan model analisis sentimen serta mendukung peningkatan kualitas layanan berbasis ulasan pengguna.

Kata kunci: Agoda, Analisis Sentimen, *Machine Learning, Support Vector Machine*, Ulasan Pengguna

PENDAHULUAN

Perkembangan teknologi digital telah membawa perubahan *signifikan* dalam berbagai aspek kehidupan manusia, termasuk bisnis pariwisata dan layanan pemesanan akomodasi online. *Platform* pemesanan hotel seperti Agoda memberikan kemudahan kepada pengguna dalam melakukan pencarian, pemesanan, dan ulasan tentang layanan akomodasi secara efisien dan cepat [1]. Volume besar ulasan yang dihasilkan menciptakan potensi data besar untuk diolah guna memahami sentimen dan persepsi pelanggan terhadap layanan yang disediakan.

Analisis sentimen telah menjadi pendekatan yang banyak digunakan dalam mengolah data ulasan pengguna untuk mengklasifikasikan opini mereka ke dalam kategori sentimen positif, negatif, atau netral. Metode *machine learning seperti K-Nearest Neighbor (KNN), Naïve Bayes, dan Support Vector Machine (SVM)* secara luas diaplikasikan untuk melakukan tugas klasifikasi ini dengan tingkat akurasi yang terus meningkat[2]. Dengan analisis sentimen ini, manajemen hotel dapat memperoleh wawasan yang lebih tajam untuk meningkatkan kualitas layanan dan menanggapi keluhan pelanggan secara tepat.

Meski kemudahan layanan sangat dirasakan, banyak pengguna Agoda mengeluhkan permasalahan kritis seperti hilangnya tiket secara tiba-tiba dan proses *refund* yang tidak transparan, yang menimbulkan ketidakpuasan pelanggan serta berdampak negatif terhadap *citra platform* tersebut[3]. Karena itu, penting untuk melakukan analisis sentimen ulasan pengguna guna mengungkap pola keluhan dan aspek sentimen negatif yang dominan agar bisa diambil langkah perbaikan strategis.

Beberapa studi terkini menerapkan teknologi *machine learning* untuk analisis sentimen utamanya pada ulasan hotel dan layanan serupa. Prasetyaningrum (2025) menggunakan PCA untuk reduksi dimensi dan mengaplikasikan *Voting classifier* dalam *sentiment analysis* ulasan *hotel* dengan hasil akurasi sebesar 95,29%[1]. Di sisi lain, penelitian oleh Dhamma (2025) membandingkan tiga algoritma *Naïve Bayes, KNN* dan *Logistic Regression* untuk analisis sentimen ulasan *Google hotel Novotel,* dimana *Logistic Regression* berhasil mencapai akurasi terbaik sebesar 94,54%[2]. Suryadi et al. (2024) pada penelitiannya juga membandingkan *SVM* dan *Random Forest* untuk analisis ulasan beberapa aplikasi pemesanan hotel dan mencatat *SVM* unggul dari segi stabilitas dengan akurasi tinggi[3].

Penelitian ini bertujuan menerapkan dan membandingkan algoritma *KNN, Naïve Bayes*, dan *SVM* dalam analisis sentimen ulasan pengguna Agoda untuk menghasilkan model klasifikasi terbaik yang dapat memberikan gambaran menyeluruh mengenai sentimen pengguna. Diharapkan hasil dari penelitian ini turut mendukung pengambilan keputusan manajemen dalam meningkatkan kepuasan pelanggan dan memperbaiki permasalahan utama yang kerap dihadapi oleh pengguna.

TINJAUAN PUSTAKA

2.1 Sentimen Analisis

Analisis sentimen adalah teknik untuk mengidentifikasi, mengekstrak, dan mengklasifikasikan opini, emosi, atau sikap yang terkandung dalam teks ke dalam kategori positif, negatif, atau netral. Teknik ini berperan penting dalam memahami persepsi publik dari data teks berjumlah besar secara otomatis[4].

Sentiment analysis adalah proses *NLP* untuk mengidentifikasi, mengekstrak, dan menganalisis sentimen, opini, atau emosi dalam teks. Analisis ini bertujuan memahami orientasi sentimen (positif, negatif, atau netral) serta menggali informasi untuk mendukung pengambilan keputusan berbasis opini masyarakat di berbagai media, terutama media sosial[5].

Dalam konteks *machine learning,* algoritma *Naïve Bayes* sering digunakan karena kemampuannya dalam klasifikasi teks dengan tingkat *efisiensi* dan *performa* yang baik terhadap data besar dan beragam[6].

Selain itu, kalangan riset juga memanfaatkan algoritma lain seperti *K-Nearest Neighbor (KNN)* dan *Support Vector Machine (SVM)* untuk meningkatkan akurasi analisis sentimen, dengan prestasi masing-masing tergantung pada karakteristik dataset dan proses *preprocessing*[7].

Jadi analisis sentimen merupakan alat penting dalam pengolahan data teks untuk mendukung pengambilan keputusan berbasis opini masyarakat. Misalnya dalam industri perhotelan, analisis sentimen

membantu memetakan keluhan dan kepuasan pelanggan yang berdampak pada perbaikan layanan dan pengelolaan reputasi platform.

2.2. Agoda

Agoda adalah salah satu contoh *online travel agent (OTA)* yang menjadi bagian dari *e-commerce* di industri perhotelan. *OTA* seperti Agoda digunakan hotel untuk memperluas jangkauan pasar dan mempercepat proses *reservasi* kamar secara *online*, sehingga meningkatkan efisiensi bisnis dan kualitas layanan hotel di era *revolusi* industri 4.0[8].

Agoda merupakan *platform e-commerce* berbasis web dan aplikasi yang menyediakan layanan *reservasi* akomodasi secara *online*, dengan reputasi tinggi di industri perhotelan dan pariwisata berkat jumlah unduhan yang sangat besar serta kualitas layanannya yang teruji[9].

Jadi dapat disimpulkan bahwa Agoda merupakan *platform e-commerce* terkemuka di industri perhotelan yang tidak hanya mempermudah proses *reservasi* secara online tetapi juga menawarkan kemudahan akses *global* lewat dukungan multibahasa dan pilihan akomodasi yang sangat banyak, menjadikan Agoda pilihan utama bagi para pelaku bisnis perhotelan maupun wisatawan modern.

2.3 Algoritma K-Nearest Neighbor (KNN)

KNN adalah algoritma *supervised learning* yang sederhana namun efektif, yang berdasarkan pada konsep bahwa data dengan atribut yang mirip cenderung memiliki label kelas yang sama. Proses *KNN* dimulai dengan memilih sebuah titik data, lalu mencari K tetangga terdekat menggunakan metrik jarak seperti *Euclidean* atau *Manhattan,* dan menentukan kelas berdasarkan mayoritas label tetangga terdekat[10].

KNN merupakan metode klasifikasi sederhana namun efisien yang dapat diaplikasikan pada analisis sentimen teks terutama dengan pemilihan parameter yang tepat, menghasilkan akurasi yang baik dan mampu menangani isu data *noise* pada dataset beragam[11].

Jadi dapat disimpulkan bahwa *KNN* merupakan metode klasifikasi yang *intuitif, efisien,* dan *adaptif,* tepat untuk aplikasi seperti klasifikasi citra medis maupun analisis sentimen teks, dengan kemampuan mengatasi permasalahan data yang memiliki banyak variasi dan *noise* selama parameter disesuaikan dengan baik.

2.4. Algoritma Naive Bayes

Algoritma *Naive Bayes* adalah metode klasifikasi probabilistik yang mengasumsikan bahwa setiap fitur memiliki kontribusi *independen* terhadap kelas yang dianalisis[12].

Naive Bayes adalah *algoritma data mining* berbasis probabilitas yang menggunakan *teorema Bayes* dengan asumsi *independensi* antar atribut. Algoritma ini melakukan klasifikasi dengan menghitung probabilitas posterior suatu kelas berdasarkan frekuensi kelas pada data *training,* sehingga efektif untuk prediksi di berbagai bidang[13].

Naive Bayes adalah algoritma klasifikasi yang mengoptimalkan probabilitas pengawasan dan mampu melakukan klasifikasi secara efektif dalam berbagai kasus terutama data teks dan data berlabel lainnya. Kelebihan utama adalah kemudahan implementasi dan hasil yang baik meski terdapat ketergantungan antar fitur[14].

Jadi dapat disimpulkan *Algoritma Naive Bayes* merupakan metode klasifikasi probabilistik yang sederhana dan efektif, menggunakan *teorema Bayes* dengan asumsi setiap fitur memberikan kontribusi *independen* terhadap kelas target. Algoritma ini menghitung probabilitas setiap kelas untuk data baru dan memasukkannya ke kelas dengan nilai probabilitas tertinggi.

2.5 Algoritma Support Vector Machine (SVM)

Algoritma *Support Vector Machine (SVM)* adalah metode klasifikasi yang mencari *hyperplane* optimal untuk memisahkan kelas data dengan *margin* terbesar[15]. *Support Vector Machine (SVM)* adalah algoritma yang mencari *hyperplane* optimal untuk memisahkan dua kelas dengan *margin maksimum*. SVM efektif pada data kecil maupun besar, mampu menangani banyak atribut, dan mudah diimplementasikan. Awalnya untuk klasifikasi *biner*, *SVM* kini juga dapat digunakan untuk multi-kelas, regresi, dan deteksi *outlier*[16].

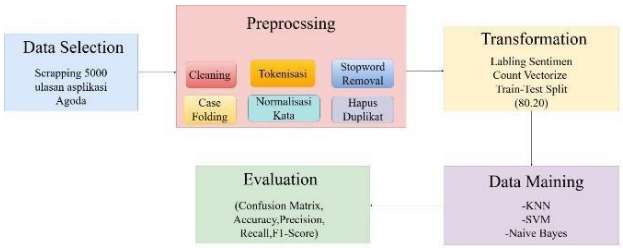
Support Vector Machine (SVM) adalah algoritma klasifikasi yang mencari *hyperplane* optimal untuk memisahkan dua kelas dengan *margin* terbesar. Algoritma ini efektif untuk data besar dan berdimensi tinggi, serta dapat menggunakan fungsi kernel untuk klasifikasi *non-linear*. Pada penelitian ini, *SVM* diterapkan dengan seleksi fitur menggunakan *mutual information* untuk meningkatkan akurasi klasifikasi teks berita [17].

Jadi dapat di simpulkan *Support Vector Machine (SVM)* adalah algoritma pembelajaran mesin yang berfungsi untuk memisahkan data ke dalam kelas tertentu dengan menentukan batas pemisah paling optimal di antara data tersebut. Metode ini berprinsip mencari *margin* terbesar agar hasil klasifikasi lebih akurat. *SVM* memiliki keunggulan dalam mengolah data berdimensi tinggi dan dapat menggunakan fungsi *kernel* untuk menangani pola data yang tidak *linear*.

METODOLOGI PENELITIAN

3.1 Tahapan Penelitian

Metode penelitian ini mengadopsi tahapan *Knowledge Discovery in Database (KDD)* dengan langkah-langkah utama: seleksi data, *preprocessing*, transformasi data, *data mining*, dan evaluasi model.
Seperti yang ditunjukkan gambar 1, Diagram alir Proses Penelitian:



Gambar 1. Tahapan Penelitian
Sumber : Hasil Olah Data

Gambar 1 menggambarkan alur proses analisis

```
from google_play_scraper import reviews, Sort

app_id = 'com.agoda.mobile.consumer'

def get_reviews(app_id, lang='id', count=5000, sort=Sort.NEWEST, filter_score
    try:
        result, continuation_token = reviews(
            app_id,
            lang=lang,
            country='id',
            sort=sort,
            count=count,
            filter_score_with=filter_score_with,
            filter_device_with=filter_device_with,
            continuation_token=continuation_token
        )
```

sentimen ulasan aplikasi Agoda. Tahapan dimulai dari pemilihan data hasil *scrapping*, dilanjutkan preprocessing hingga di uji dengan algoritma.

3.2 *Pemilihan Data*

Penelitian ini dimulai dengan pengumpulan data dengan cara *scraping* menggunakan pustaka *google-play-scraper* untuk mengekstrak ulasan pengguna dari aplikasi *Agoda* di *Google Play Store* secara otomatis. Data diambil berdasarkan ID aplikasi yaitu agoda, dengan filter bahasa Indonesia dan wilayah Indonesia, serta disusun berdasarkan ulasan terbaru. Hasilnya ditemukan 5.000 ulasan yang kemudian dikonversi ke bentuk *DataFrame Pandas* dan disimpan dalam format *CSV*. Gambar 2 berikut memperlihatkan Proses *Scraping.*

Gambar 2. Proses *Scraping*
Sumber : Hasil Olah Data

Tahap *preprocessing* dilakukan secara bertahap untuk memastikan data benar-benar siap diolah. Proses pembersihan dimulai dengan menghilangkan gangguan pada teks seperti *URL, emoji,* simbol, angka, *HTML,* dan *duplikasi* ulasan. Selanjutnya, seluruh teks diubah ke bentuk huruf kecil agar format penulisan lebih konsisten dan memudahkan analisis. Tokenisasi diterapkan untuk memisahkan kata per kata dalam setiap kalimat pada ulasan. Upaya normalisasi kata dilakukan dengan mengganti kata tidak baku atau slang menjadi kata baku menggunakan kamus khusus, sehingga makna dan analisis dapat lebih akurat. Selain itu, kata-kata umum yang tidak memiliki kontribusi signifikan terhadap analisis *(stopword)* dihilangkan dari data. Sebagai langkah akhir, data yang memiliki duplikasi ulasan serta data kosong atau *null* diidentifikasi dan dihapus, sehingga hanya data ulasan yang *relevan* dan bersih dipakai pada tahap selanjutnya.

a. Cleaning Data

Proses *cleaning* atau pembersihan teks merupakan tahap awal dalam *text preprocessing* yang bertujuan untuk menghilangkan elemen-elemen yang tidak *relevan* dan berpotensi mengganggu hasil analisis sentimen. Elemen-elemen tersebut meliputi tautan *URL*, *tag HTML, emoji,* simbol, angka, dan *username* pengguna.

b. Case folding

Tahap ini berfungsi untuk menormalkan teks ulasan dengan beberapa langkah *preprocessing*, seperti menghapus *mention, hashtag*, tautan *URL*, angka, tanda baca, serta *emoji* yang tidak *relevan*. Selain itu, karakter yang berulang lebih dari dua kali juga dihilangkan agar teks menjadi lebih rapi dan seragam. Selanjutnya, seluruh teks diubah menjadi huruf kecil (*lowercase*) guna menjaga konsistensi penulisan.

c. Normalisasi Kata

Tahap *normalisasi kata* dilakukan untuk mengganti kata tidak baku menjadi kata baku menggunakan kamus kata baku yang diambil dari file kamuskatabaku.xlsx di *GitHub*. Proses ini dijalankan menggunakan fungsi *replace_taboo_words,* yang mencocokkan setiap kata dalam teks dengan entri pada kamus. Jika ditemukan padanan kata baku, sistem akan menggantinya secara otomatis.Sebagai contoh, kata seperti "*gk*" diubah menjadi "*tidak*" dan "*bgt*" menjadi "*banget*". Hasil dari tahap ini membuat teks menjadi lebih seragam dan sesuai dengan kaidah bahasa Indonesia sehingga meningkatkan akurasi pada tahap analisis selanjutnya.

d. Tokenisasi dan Pelabelan

Proses *tokenisasi* dilakukan untuk memecah setiap teks ulasan menjadi satuan kata yang lebih kecil menggunakan fungsi *tokenize*. Setiap kalimat diubah menjadi daftar kata berdasarkan spasi agar dapat dianalisis secara terpisah. Selanjutnya, dilakukan tahap pelabelan sentimen secara otomatis menggunakan kamus *leksikon InSet (Indonesian Sentiment Lexicon),* yang terdiri dari daftar kata positif dan negatif. Fungsi *determine_sentiment(text)* digunakan untuk menghitung jumlah kata positif dan negatif pada setiap ulasan. Berdasarkan hasil perhitungan tersebut, setiap ulasan diklasifikasikan ke dalam tiga kategori sentimen, yaitu positif, negatif, atau netral.

e. Stopword Removal

Tahap *stopword removal* bertujuan untuk menghapus kata-kata umum yang tidak memiliki makna penting dalam analisis sentimen, seperti "dan", "yang", "di", "ke", serta kata lain yang sering muncul namun tidak berkontribusi terhadap konteks opini pengguna. Dalam kodingan ini, proses dilakukan menggunakan daftar *stopword* bahasa Indonesia dari pustaka NLTK melalui fungsi *stopwords.words('indonesian')*. Setiap kata dalam teks dibandingkan dengan daftar *stopword*, lalu dihapus jika termasuk di dalamnya. Hasilnya adalah teks yang lebih bersih dan fokus pada kata-kata bermakna yang *relevan* dengan

sentimen ulasan.

f. Stemming Data

Tahap *stemming* dilakukan untuk mengembalikan setiap kata ke bentuk dasarnya (*root word*). Proses ini menggunakan *library Sastrawi* melalui fungsi stem_text, yang menerapkan algoritma *Nazief–Adriani* untuk Bahasa Indonesia. Contohnya, kata "*menginap*" diubah menjadi "*inap*" dan "*bagusan*" menjadi "*bagus*". Proses ini membantu mengurangi variasi kata dengan makna serupa sehingga model dapat mengenali kata secara konsisten. Hasil dari tahap *stemming* disimpan dalam kolom *steming_data,* yang kemudian digunakan sebagai input pada tahap ekstraksi fitur dan analisis sentimen selanjutnya.

3.3. Transformasi

Pada tahap ini data teks diubah menjadi bentuk numerik menggunakan teknik *CountVectorizer,* sehingga *algoritma machine learning* dapat mengolah data tersebut secara optimal. Dataset dibagi menjadi data latih dan data uji dengan rasio 80:20 agar proses *evaluasi* model lebih *objektif* dan *valid*.

3.4. Data Mining

Klasifikasi sentimen dilakukan dengan menerapkan tiga algoritma *yaitu Support Vector Machine (SVM), K-Nearest Neighbor (KNN), dan Naïve Bayes Multinomial*. Setiap model dilatih menggunakan data latih dan kemudian diuji pada data uji untuk menghasilkan prediksi sentimen positif, negatif, dan netral.

3.5. Evaluasi

Model yang telah dihasilkan dievaluasi menggunakan *metrik confusion matrix, akurasi, precision, recall, dan F1-score*. Hasil evaluasi disajikan dalam bentuk grafik *performa* dan *confusion matrix*, sehingga *interpretasi* serta perbandingan antar model dapat dilakukan dengan jelas.

HASIL DAN PEMBAHASAN

1. Pengumpulan data

penelitian ini menggunakan pendekatan *web scraping* untuk memperoleh data ulasan pengguna aplikasi *Agoda* secara otomatis *dari platform Google Play Store*. Proses ini dilakukan dengan memanfaatkan pustaka *google_play_scraper,* yang memungkinkan pengambilan hingga 5000 ulasan terbaru berbahasa Indonesia berdasarkan *App ID com.agoda.mobile.consumer*. Sebagaimana ditunjukan pada gambar 3, Setiap ulasan yang berhasil dikumpulkan berisi atribut penting seperti *Review ID*, *Username*, *Rating*, *Review Text*, dan *Date* yang menggambarkan opini serta pengalaman pengguna terhadap aplikasi Agoda. Berdasarkan hasil pemuatan, diperoleh 5000 entri data ulasan yang *valid* dengan lima kolom utama yang siap digunakan dalam proses *preprocessing* dan analisis sentimen menggunakan algoritma *Machine Learning*.

```
data.head(4)
```

	Review ID	Username	Rating	Review Text	Date
0	deb7747a-4f15-42b2-8b58-ec987626b973	Pengguna Google	1	aplikasinya ngga bisa di buka, padahal jaringa...	2025-10-20 23:54:27
1	3b7d8d1b-2f38-4690-a42c-5a70c62e899e	Pengguna Google	1	buat temen2 klo mau psen hotel jgn di aplikasi...	2025-10-20 22:52:01
2	a28cbd48-806e-44e1-8707-fac1896b460e	Pengguna Google	1	banyak hotel yg tidak menggunakan/melayani pak...	2025-10-20 09:27:27
3	cbe14f5e-5ad2-4e36-8524-666fd87fe6c5	Pengguna Google	1	saya pesan hotel disini, pas mau check-in id p...	2025-10-20 08:53:47

Gambar 3. hasil ulasan
Sumber : Hasil Olah Data

2. Preprocessing Data

Tahap *preprocessing* dilakukan untuk membersihkan dan menyiapkan data teks sebelum dilakukan analisis sentimen. Tujuannya agar data menjadi seragam, bebas dari *noise,* dan dapat diproses oleh algoritma *machine learning*. Tahapan ini meliputi beberapa proses berikut:

A. Cleaning

Proses ini dilakukan melalui serangkaian fungsi pembersihan seperti *remove_URL()* untuk menghapus tautan atau *URL, remove_html()* untuk menghilangkan tag *HTML, remove_emoji()* untuk menghapus simbol atau emotikon yang tidak memiliki makna tekstual, *remove_symbols()* untuk menghapus karakter *non-alfanumerik* seperti tanda baca dan simbol khusus, serta *remove_numbers()* untuk menghapus angka yang tidak berkontribusi terhadap analisis sentimen.

Review Text	cleaning
sudah pakai aplikasi ini puluhan tahun	sudah pakai aplikasi ini puluhan tahun
proses refund sulit, tidak ad layanan hotline....	proses refund sulit tidak ad layanan hotline d...
pesawat hilang 1	pesawat hilang

Gambar 4. Proses *Cleaning*
Sumber : Hasil Olah Data

Gambar 4 menunjukkan hasil tahap *cleaning*, di mana teks pada kolom *Review Text* telah dibersihkan dan disimpan dalam kolom *Cleaning*. Proses ini menghapus elemen tidak *relevan* seperti angka, simbol, *emoji*, dan *URL*, sehingga menghasilkan teks yang lebih rapi dan siap digunakan untuk tahap

preprocessing berikutnya seperti *case folding* dan *tokenization*.

B. Case Folding

Fungsi *case_folding(text)* memeriksa setiap data pada kolom *cleaning* jika data berupa teks (*string*), maka akan dikonversi menjadi huruf kecil menggunakan perintah *text.lower()*. Proses ini kemudian diterapkan ke seluruh baris menggunakan *df['cleaning'].apply(case_folding)* dan hasilnya disimpan dalam kolom baru bernama *case_folding*. Dengan langkah ini, kata seperti "Hotel", "HOTEL", dan "hotel" akan dianggap sama oleh sistem, sehingga mengurangi redundansi dan meningkatkan akurasi dalam proses analisis sentimen. Terlihat pada gambar 5, seluruh teks berhasil dikonversi menjadi huruf kecil sebagai hasil dari proses *case folding* untuk menjaga konsistensi penulisan data.

cleaning	case_folding
sudah pakai aplikasi ini puluhan tahun	sudah pakai aplikasi ini puluhan tahun
proses refund sulit tidak ad layanan hotline d...	proses refund sulit tidak ad layanan hotline d...
pesawat hilang	pesawat hilang
kocak sih hotel penu tapi masih ada sisa di ap...	kocak sih hotel penu tapi masih ada sisa di ap...
pesanan hotel bermasalah tidak bisa membantu s...	pesanan hotel bermasalah tidak bisa membantu s...

Gambar 5. hasil *case folding*
Sumber : Hasil Olah Data

C. Normalisasi Kata

Kode ini melakukan normalisasi kata tidak baku dengan mengganti istilah *slang* di kolom *case_folding* menggunakan *replace_taboo_words()*, hasilnya disimpan di kolom normalisasi beserta informasi kata baku, kata tidak baku, dan *hash* kata, seperti yang di tunjukan pada gambar 6. Dataset kemudian disederhanakan dengan hanya mengambil kolom *relevan* (Rating, *Review Text, cleaning, case_folding*, normalisasi).

case_folding	normalisasi
sudah pakai aplikasi ini puluhan tahun	sudah pakai aplikasi ini puluhan tahun
proses refund sulit tidak ad layanan hotline d...	proses refund sulit tidak ada layanan hotline ...
pesawat hilang	pesawat hilang
kocak sih hotel penu tapi masih ada sisa di ap...	kocak sih hotel penu tapi masih ada sisa di ap...
pesanan hotel bermasalah tidak bisa membantu s...	pesanan hotel bermasalah tidak bisa membantu s...

Gambar 6. Hasil Normalisasi kata
Sumber **:** Hasil Olah Data

D. Stopword Removal

Tahap *stopword removal* bertujuan untuk menghapus kata-kata umum yang tidak memiliki makna penting dalam analisis sentimen, seperti "dan", "yang", "di", "ke", serta kata lain yang sering muncul namun tidak berkontribusi terhadap konteks opini pengguna. Gambar 7 menunjukan kata-kata umum seperti "dan", "yang", serta "di" telah dihapus untuk memfokuskan teks hanya pada kata bermakna penting.

tokenize	stopword removal
[sudah, pakai, aplikasi, ini, puluhan, tahun]	[pakai, aplikasi, puluhan]
[proses, refund, sulit, tidak, ad, layanan, ho...	[proses, refund, sulit, ad, layanan, hotline, ...
[pesawat, hilang]	[pesawat, hilang]

Gambar 7 . Hasil *Stopword removal*
Sumber : Hasil Olah Data

E. Stemming Data

Tahap *stemming* dilakukan untuk mengembalikan setiap kata ke bentuk dasarnya (*root word*). Proses ini menggunakan *library Sastrawi* melalui fungsi *stem_text,* yang menerapkan algoritma *Nazief–Adriani* untuk Bahasa Indonesia. Contohnya, kata *"menginap"* diubah menjadi *"inap"* dan *"bagusan"* menjadi *"bagus"*. Seperti yang di tunjukan pada gambar 8 berikut.

stopword removal	steming_data
[pakai, aplikasi, puluhan]	pakai aplikasi puluh
[proses, refund, sulit, ad, layanan, hotline, ...	proses refund sulit ad layan hotline duit nyan...
[pesawat, hilang]	pesawat hilang
[kocak, sih, hotel, penu, sisa, aplikasi, udh,...	kocak sih hotel penu sisa aplikasi udh tf refu...
[pesanan, hotel, bermasalah, membantu, pesan, ...	pesan hotel masalah bantu pesan cek in tolak h...

Gambar 8. hasil *steming data*
Sumber : Hasil Olah Data

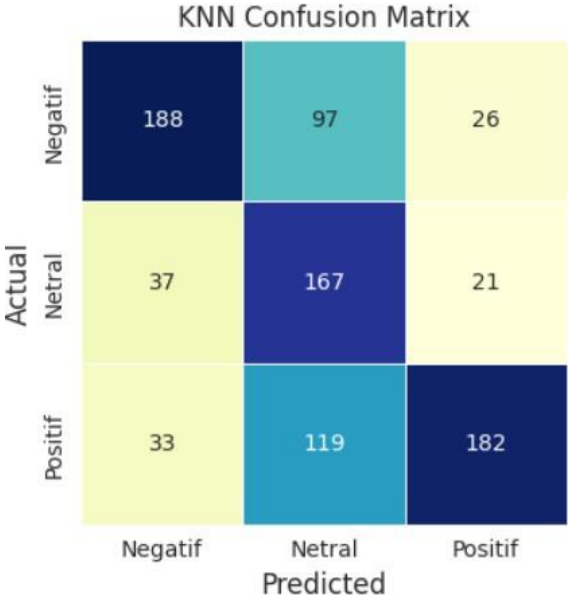
3. Ekstraksi Fitur

Tahap *ekstraksi fitur* bertujuan untuk mengubah teks hasil *preprocessing* menjadi bentuk *numerik* agar dapat diproses oleh algoritma klasifikasi. Dalam penelitian ini, digunakan metode *CountVectorizer* dari *library Scikit-learn*, yang bekerja dengan menghitung frekuensi kemunculan setiap kata dalam teks ulasan. Setiap kata yang muncul dalam kumpulan dokumen direpresentasikan sebagai fitur dalam bentuk *vektor numerik*.

Hasil *transformasi* ini menghasilkan *matriks fitur* dengan dimensi sesuai jumlah kata unik yang ada dalam dataset. Selanjutnya, data dibagi menjadi dua bagian menggunakan fungsi *train_test_split,* yaitu 80% data latih untuk proses pelatihan model dan 20% data uji untuk pengujian *performa* model. Pembagian ini dilakukan secara acak namun tetap seimbang agar model dapat belajar secara optimal dan diuji secara adil terhadap data yang belum pernah dilihat sebelumnya.

4. *Algoritma K-Nearest Neighbors (KNN)*

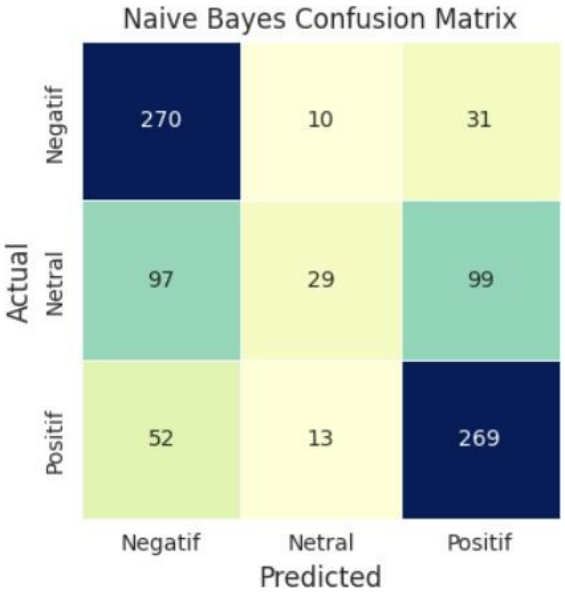
hasil klasifikasi menggunakan algoritma KNN, diperoleh akurasi keseluruhan sebesar 61,7%. Hasil Klasifikasi yang ditunjukan pada gambar 9, yang menunjukkan bahwa model mampu memprediksi sebagian besar label sentimen dengan tingkat keberhasilan sedang. Analisis lebih lanjut menunjukkan bahwa model memiliki *precision* tinggi pada kelas Positif (0,795) dan Negatif (0,729), artinya sebagian besar prediksi Positif dan Negatif benar. Sebaliknya, kelas Netral memiliki *precision* lebih rendah (0,436), meskipun *recall*-nya relatif tinggi (0,742), yang menunjukkan model mampu menangkap sebagian besar data Netral namun sering salah mengklasifikasikannya. *F1-score* untuk masing-masing kelas menegaskan ketidakseimbangan *performa*: Negatif (0,661), Netral (0,549), dan Positif (0,647).



Gambar 9. *Confusion Matrix for KNN*
Sumber : Hasil Olah Data

5. Algoritma *Naïve Bayes*

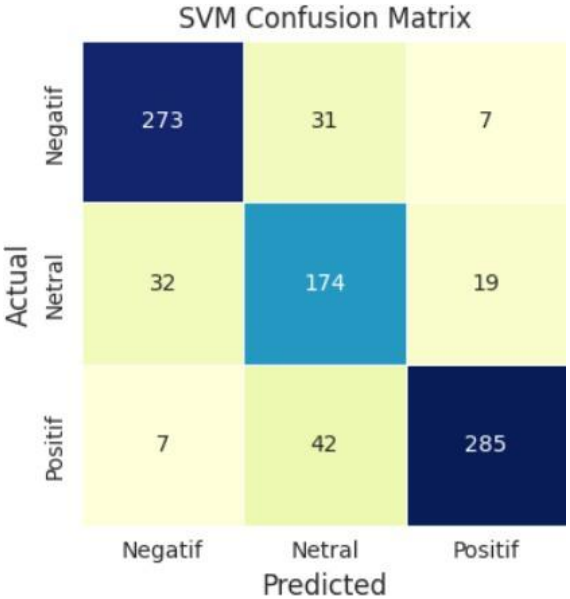
Hasil pengujian algoritma *Naive Bayes* untuk analisis sentimen Agoda menunjukkan akurasi 65,3%. Gambar 10 menunjukan bahwa *Confusion matrix for Naïve Bayes* Kelas *Negatif* memperoleh *precision* 0,644, *recall* 0,868, dan *f1-score* 0,740, sedangkan kelas *Positif* memiliki *precision* 0,674, *recall* 0,805, dan *f1-score* 0,734. Kelas *Netral* menunjukkan *performa* terendah dengan *precision* 0,558, *recall* 0,129, dan *f1-score* 0,209, mengindikasikan kesulitan model dalam mengenali sentimen netral. Rata-rata *f1-score macro* dan *weighted* masing- masing sebesar 0,561 dan 0,600, menunjukkan *performa* model yang moderat dan dipengaruhi ketidakseimbangan kelas.



Gambar 10. *Confusion Matrix for Naïve Bayes*
Sumber : Hasil Olah Data

6. *Algoritma Support Vector Machine (SVM)*

Pengujian algoritma *SVM* pada analisis sentimen *Agoda* menunjukkan akurasi 84,1%, Seperti yang terlihat pada gambar 11. Kelas *Negatif* memiliki *precision* 0,875, *recall* 0,878, dan *f1-score* 0,876, kelas *Positif* memperoleh *precision* 0,916, *recall* 0,853, dan *f1-score* 0,884, sedangkan kelas *Netral* memiliki *precision* 0,704, *recall* 0,773, dan f1-*score* 0,737. Rata- rata *f1-score macro* dan *weighted* masing-masing 0,832 dan 0,843, menunjukkan SVM mampu memberikan klasifikasi yang konsisten dan seimbang antar kelas.



Gambar 11. *Confusion Matrix for SVM*
Sumber : Hasil Olah Data

7. Evaluasi Model

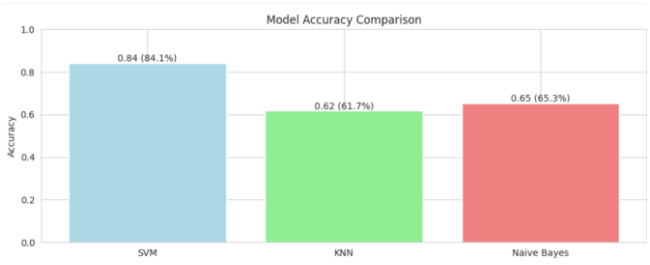
Hasil pengujian ketiga *algoritma machine learning* pada tabel 1 analisis sentimen Agoda menunjukkan perbedaan *performa* yang *signifikan*. *Algoritma SVM* mencapai akurasi tertinggi 84,1%, dengan kelas *Negatif* memiliki *precision* 0,875, *recall* 0,878, dan *f1-score* 0,876, kelas *Positif precision* 0,916, *recall* 0,853, *f1-score* 0,884, dan kelas *Netral precision* 0,704, *recall* 0,773, *f1-score* 0,737, menunjukkan kemampuan *SVM* dalam menangani ketiga kelas secara seimbang.

Tabel 1. Perbandingan Kinerja Model Sentimen Analisis

Metrik	*SVM*	*KNN*	*NBC*
Accuracy	*0.841*	*0.617*	*0.653*
Macro Precision	*0.832*	*0.653*	*0.653*
Macro Recall	*0.835*	*0.619*	*0.561*
Macro F1	*0.832*	*0.619*	*0.561*
Weighted F1	*0.843*	*0.627*	*0.600*

Sumber : Hasil Olah Data

Algoritma *Naïve Bayes* memperoleh akurasi 65,3%, dengan performa terbaik pada kelas *Negatif* (*precision* 0,644, *recall* 0,868, *f1-score* 0,740) dan kesulitan pada kelas *Netral* (*precision* 0,558, *recall* 0,129, *f1-score* 0,209), menandakan model kurang optimal mengenali sentimen netral. Sedangkan *KNN* memiliki akurasi 61,7%, dengan kelas *Positif* (*precision* 0,795, *recall* 0,545, *f1-score* 0,647) dan kelas *Negatif* (*precision* 0,729, *recall* 0,605, *f1-score* 0,661) yang cukup baik, tetapi *performa* kelas *Netral* (*precision* 0,436, *recall* 0,742, *f1-score* 0,549) menunjukkan ketidakseimbangan dalam prediksi. Secara keseluruhan, *SVM* menunjukkan *performa* paling konsisten dan seimbang antar kelas dibanding *KNN* dan *Naïve Bayes*, yang dipengaruhi oleh distribusi data dan kompleksitas kata dalam *review*. Hasil perbandingan dapat dilihat pada gambar 12 berikut.



Gambar 12. Perbandingan *Akurasi* Model
Sumber : Hasil Olah Data

8. *Visualisasi Data (NLP)*

Visualisasi ini menggunakan *WordCloud* yang menampilkan kata dengan ukuran proporsional terhadap frekuensi kemunculannya.

Gambar 13. *WordCloud* Sebelum dan Sesudah *Preprocessing*

Sumber : Hasil Olah Data

Gambar 13 tersebut menunjukkan perbandingan teks ulasan sebelum dan sesudah proses *cleaning*, *case folding*, *tokenisasi*, *normalisasi kata*, *stopword removal*, dan *stemming*. Pada *WordCloud* sebelum *preprocessing*, masih terdapat banyak kata yang tidak relevan seperti "yg", "nya", dan "ini". Setelah melalui proses *preprocessing*, kata-kata yang muncul menjadi lebih bermakna dan *relevan* seperti "hotel", "aplikasi", "pesan", dan "*refund*".Hasil ini menunjukkan bahwa proses *preprocessing* berhasil membersihkan data dan memfokuskan teks pada kata-kata yang mendukung analisis sentimen secara lebih akurat.

5. KESIMPULAN DAN SARAN

Penelitian ini berhasil menerapkan algoritma *machine learning K-Nearest Neighbors (KNN), Naïve Bayes,* dan *Support Vector Machine (SVM)* untuk melakukan analisis sentimen terhadap 5.000 ulasan pengguna aplikasi Agoda yang diperoleh dari *Google Play Store*. Melalui tahapan *preprocessing* yang meliputi *cleaning, case folding, tokenisasi, normalisasi kata, stopword removal,* dan *stemming*, data berhasil diproses menjadi bentuk yang lebih bersih dan terstandarisasi. Hasil evaluasi menunjukkan bahwa algoritma *SVM* dengan *kernel linear* memberikan performa terbaik dibandingkan *KNN* dan *Naïve Bayes* berdasarkan nilai *accuracy, precision, recall,* dan *F1-score*. Model *SVM* dinilai paling optimal dalam mengklasifikasikan sentimen positif, negatif, dan netral, sedangkan *Naïve Bayes* unggul dalam efisiensi komputasi, dan *KNN* menunjukkan *performa* yang stabil meskipun sensitif terhadap jarak antar data.

Sebagai pengembangan, penelitian selanjutnya disarankan untuk menerapkan metode *feature selection* seperti *TF-IDF* atau *Word2Vec* guna meningkatkan representasi fitur teks, serta mengeksplorasi model *deep learning* seperti *LSTM* atau *BERT* untuk memperoleh hasil klasifikasi yang lebih akurat. Selain itu, perluasan sumber data dari berbagai *platform* ulasan pengguna juga dapat dilakukan untuk memberikan gambaran yang lebih *komprehensif* terhadap persepsi masyarakat terhadap layanan Agoda.

DAFTAR PUSTAKA

[1] P. T. Prasetyaningrum, N. Ibrahim, dan O. Suria, "*Optimizing Sentiment Analysis of Hotel Reviews Using PCA and Machine Learning for Tourism Business Decision Support,*" vol. 8, no. 1, 2025.

[2] Y. A. Dhamma dan S. P. Barus, "*Sentiment Analysis on Google Reviews Using NaÃ¯ve Bayes, K-Nearest Neighbors, and Logistic Regression to Improve Novotel Services,*" *J. Appl. Informatics Comput.*, vol. 9, no. 1, hal. 106–114, 2025, doi: 10.30871/jaic.v9i1.8923.

[3] S. Suryadi, D. Syahputra, N. Astrianda, R. A. Syahputra, dan R. Suhendra, "*Leveraging Machine Learning for Sentiment Analysis in Hotel Applications: A Comparative Study of Support Vector Machine and Random Forest Algorithms,*" *Brill. Res. Artif. Intell.*, vol. 4, no. 2, hal. 567–576, 2024, doi: 10.47709/brilliance.v4i2.4877.

[4] S. N. Sofyan dan M. Iqbal, "Analisis Sentimen Terhadap Dampak Inflasi Menggunakan *Naive Bayes,*" *Bull. Inf. Technol.*, vol. 6, no. 1, hal. 1–8, 2025, doi: 10.47065/bit.v5i2.1796.

[5] U. Faddillah, I. Sugiyarto, dan Lady Agustin Fitriana, "Analisis Sentimen Publik Pengguna Twitter Terhadap Kanker Serviks Di Indonesia Menggunakan *Algoritma Naive Bayes Classifier,*" *JATI (Jurnal Mhs. Tek. Inform.*, vol. 9, no. 3, hal. 4113–4121, 2025, doi: 10.36040/jati.v9i3.13460.s

[6] A. C. Wibisono, T. S. Nadira, dan T. Sutabri, "Analisis Sentimen Pelanggan pada *Platform Shopee* Menggunakan Metode *Naive Bayes,*" *Nusant. J. Multidiscip. Sci.*, vol. 2, no. 6, hal. 1259–1266, 2025, [Daring]. Tersedia pada: https://jurnal.intekom.id/index.php/njms

[7] D. Purnomo, W. Firgiawan, dan N. Nur, "Komparasi *Algoritma Random Forest, Naïve Bayes, dan SVM* pada Sentimen Kebijakan PPN 12%," *J. Tekno Kompak*, vol. 19, no. 2, hal. 155–167, 2025, doi: 10.33365/jtk.v19i2.122.

[8] E. Y. Sutrisno, A. C. Hidayat, dan A. Sutanto, "Pemanfaatan *E-Commerce dan Property Management System* Dalam Kegiatan Bisnis Perhotelan di Era *Revolusi Industri* 4.0," *J. Kepariwisataan Indones. J. Penelit. dan Pengemb. Kepariwisataan Indones.*, vol. 17, no. 1, hal. 85–98, 2023, doi: 10.47608/jki.v17i12023.85-98.

[9] S. Nuraeni, S. Putri, A. Syam, M. F. Wajdi, dan M. Malkan, "G-Tech : Jurnal Teknologi Terapan Implementasi Metode K-NN Untuk Menentukan Jurusan Siswa di SMAN 02 Manokwari," vol. 7, no. 1, hal. 89–95, 2023.

[10] I. H. Kusuma dan N. Cahyono, "Analisis Sentimen Masyarakat Terhadap Penggunaan *E-Commerce* Menggunakan *Algoritma K-Nearest Neighbor*," *J. Inform. J. Pengemb. IT*, vol. 8, no. 3, hal. 302–307, 2023, doi: 10.30591/jpit.v8i3.5734.

[11] A. Y. Permana dan M. Makmun, "Analisis Sentimen pada Teks Opini Penilaian Kinerja Dosen dengan Pendekatan Algoritma KNN," *J. Ilm. Komputasi*, vol. 19, no. 1, hal. 39–50, 2020, doi: 10.32409/jikstik.19.1.154.

[12] A. N. Huzna, I. Nurhayati, A. E. Saputri, dan M. Qomarul Huda, "Analisis Sentimen Terhadap Mobil Listrik di Indonesia Pada Twitter: Penerapan *Naïve Bayes Classifier* Untuk Memahami Opini Publik," *J. Sist. Informasi, Teknol. Inf. dan Komput.*, vol. 14, no. 2, hal. 80–149, 2024, [Daring]. Tersedia pada: https://jurnal.umj.ac.id/index.php/just-it/index

[13] M. Nizam Fadli, I. Sudahri Damanik, E. Irawan, S. Tunas Bangsa, dan S. Utara, "KLIK: Kajian Ilmiah Informatika dan Komputer Penerapan Metode *Naive Bayes* Dalam Menentukan Tingkat Kenyamanan Pada Rumah Sakit Terhadap Pasien," *Media Online*, vol. 2, no. 3, hal. 117–122, 2021, [Daring]. Tersedia pada: https://djournals.com/klik

[14] Juwita, M. Safii, dan B. Efendi Damanik, "*Algoritma Naïve Bayes* Untuk Memprediksi Penjualan Pada Toko *VJCakes* Pematang Siantar," *J. Mach. Learn. Artif. Intell.*, vol. 1, no. 4, hal. 337–346, 2022, doi: 10.55123/jomlai.v1i4.1674.

[15] M. A. Syahira dan R. Kurniawan, "*Analisis Sentimen Cyberbullying* Pada Media Sosial X Menggunakan *Metode Support Vector Machine,*" *J. Media Inform. Budidarma*, vol. 8, no. 3, hal. 1724, 2024, doi: 10.30865/mib.v8i3.7926.

[16] F. Abdusyukur, "KOMPUTA : Jurnal Ilmiah Komputer dan Informatika PENERAPAN ALGORITMA *SUPPORT VECTOR MACHINE (SVM)* UNTUK KLASIFIKASI PENCEMARAN NAMA BAIK KOMPUTA : Jurnal Ilmiah Komputer dan Informatika," *Komputa J. Ilm. Komput. dan Inform.*, vol. 12, no. 1, hal. 73–82, 2023.

[17] I Putu Gede Hendra Suputra, Linawati, I. G. Sukadarmika, dan N. P. Sastra, "Klasifikasi Judul Berita Bahasa Indonesia Menggunakan *Support Vector Machine Dan Seleksi Fitur Mutual Information,*" *J. Pendidik. Teknol. dan Kejuru.*, vol. 22, no. 1, hal. 69–79, 2025, doi: 10.23887/jptkundiksha.v22i1.89158.