

# Analisis Sentimen Ulasan Pengguna Aplikasi Sociolla Menggunakan Algoritma Support Vector Machine dengan Optimasi Grid Search

Suci Fitriani, Dinda Lestari<sup>\*</sup>, Iin Seprina

Fakultas Ilmu Komputer, Program Studi Sistem Informasi, Universitas Sriwijaya, Indralaya, Indonesia

Email: <sup>1</sup>suciftrni97@gmail.com, <sup>2,\*</sup>dinda.lestari@gmail.com, <sup>3</sup>aflasaja99@gmail.com

Email Penulis Korespondensi: dinda.lestari@gmail.com,

Submitted: 11/11/2025; Accepted: 16/12/2025; Published: 16/12/2025

**Abstrak**—Perkembangan teknologi digital menggerakkan inovasi di industri kecantikan, salah satunya melalui platform Soco by Sociolla yang menyediakan ulasan produk secara daring. Banyaknya ulasan pengguna membuka peluang untuk dilakukan analisis sentimen guna mengetahui persepsi pengguna terhadap kualitas layanan. Tantangan utama dalam pemodelan sentimen pada ulasan produk kecantikan adalah penggunaan bahasa yang variatif, subjektif, dan informal sehingga pola distribusi sangat beragam. Dengan demikian, penelitian ini tidak hanya menerapkan algoritma Support Vector Machine (SVM) untuk klasifikasi sentimen tetapi juga membandingkan dua kernel yaitu, kernel Linear dan Radial Basis Function (RBF) serta mengevaluasi pengaruh optimasi hyperparameter menggunakan Grid Search pada konteks data e-commerce kecantikan. Penelitian menggunakan 3.387 ulasan dari hasil pengumpulan data yang melalui tahapan preprocessing teks, pelabelan, ekstraksi fitur menggunakan TF-IDF, pembagian data, pelatihan model, dan evaluasi. Hasil menunjukkan kernel baseline RBF memberikan kinerja paling baik dengan akurasi 88,5%, sedangkan kernel baseline linear memiliki akurasi 87,76%. Sementara itu, hasil optimasi Grid Search memperoleh akurasi 86,22% yang mengindikasikan bahwa konfigurasi hyperparameter yang dieksplorasi belum mampu melampaui performa baseline RBF meskipun memberikan hasil yang stabil pada cross-validation. Temuan ini menunjukkan bahwa karakteristik linguistik pada ulasan kecantikan lebih tepat ditangani oleh kernel non-linear sehingga lebih efektif dibanding kernel linear dalam mengenali pola non-linear pada data ulasan pengguna. Hasil ini menunjukkan bahwa optimasi hyperparameter tidak selalu menghasilkan peningkatan akurasi model, terutama ketika konfigurasi parameter baseline SVM sudah berada pada performa yang mendekati optimal untuk karakteristik data yang digunakan.

**Kata Kunci:** Analisis Sentimen; Grid Search; Machine Learning; Sociolla; Support Vector Machine

**Abstract**—The rapid growth of digital technology has driven innovation in the beauty industry, one of which is the Soco by Sociolla platform that provides online product reviews. The increasing number of user reviews offers opportunities for conducting sentiment analysis to understand users' perceptions of service quality. The main challenge in modeling sentiment for beauty product reviews lies in the use of highly varied, subjective, and informal language, which results in diverse distribution patterns. Therefore, this study not only applies the Support Vector Machine (SVM) algorithm for sentiment classification but also compares two kernels—Linear and Radial Basis Function (RBF)—and evaluates the effect of hyperparameter optimization using Grid Search in the context of beauty e-commerce data. A total of 3,387 reviews were collected and processed through several stages, including text preprocessing, labeling, feature extraction using TF-IDF, data splitting, model training, and evaluation. The results show that the baseline RBF kernel provides the best performance with an accuracy of 88.5%, while the baseline Linear kernel achieves an accuracy of 87.76%. Meanwhile, Grid Search optimization produces an accuracy of 86.22%, indicating that the explored hyperparameter configurations were unable to exceed the performance of the RBF baseline despite delivering stable results during cross-validation. These findings suggest that the linguistic characteristics of beauty reviews are more effectively addressed by non-linear kernels, making them superior to Linear kernels in recognizing non-linear patterns within user review data. Furthermore, the results indicate that hyperparameter optimization does not always lead to increased model accuracy, particularly when the baseline SVM configuration is already performing near optimally for the characteristics of the dataset used.

**Keywords:** Sentiment Analysis; Grid Search; Machine Learning; Sociolla; Support Vector Machine

## 1. PENDAHULUAN

Perkembangan teknologi revolusi industri 4.0 dalam era digital mendorong berbagai sektor terus tumbuh. Seperti pertumbuhan aplikasi digital di sektor perdagangan online (*e-commerce*) yang memungkinkan pengguna melaksanakan transaksi jual beli barang dan jasa online dengan mudah dan efisien [1]. Inovasi ini tidak hanya memperluas aksesibilitas bagi konsumen tetapi juga menjadi peluang bagi platform untuk berfokus pada pengalaman pengguna secara personal. Salah satu *e-commerce* pertama di Indonesia yang berfokus pada personal di industri kecantikan dan perawatan tubuh adalah Sociolla sejak tahun 2015. Sociolla juga memperluas bisnis dengan merilis Super App kecantikan bernama Soco by Sociolla [2]. Aplikasi ini mencapai lebih dari 1 juta unduhan pengguna yang terus bertambah sejak rilis di Indonesia.

Pertumbuhan ini menghasilkan berbagai ulasan dari pengguna sebagai aset untuk evaluasi berkelanjutan. Ulasan tersebut merupakan sumber data yang menjadi indikator kepercayaan bagi pengembang aplikasi karena dapat mengungkap persepsi dan bahan evaluasi utama bagi platform. Melalui ulasan pengguna pengembang dapat menyesuaikan fitur, memperbaiki kesalahan teknis, dan meningkatkan kualitas. Maka diperlukan analisis sentimen terhadap ulasan pengguna aplikasi Soco sebagai dasar mendalami persepsi serta kepuasan pengguna untuk evaluasi serta peningkatan kualitas layanan. Analisis sentimen merupakan metode yang efektif dalam mengidentifikasi sikap, opini, dan emosi yang terkandung dalam teks [3][4]. Menggunakan teknik pengolahan data dalam menetapkan apakah sentimen pada teks tersebut sifatnya positif, netral, atau negatif [5].

Penerapan pada penelitian analisis sentimen ini memanfaatkan algoritma *Support Vector Machine* (SVM) yang bekerja mencari *hyperplane* optimal guna membagi kelas sentimen melalui margin maksimum sehingga menghasilkan klasifikasi yang stabil [6]. Kinerja model SVM sangat dipengaruhi oleh pemilihan dan pengaturan *hyperparameter* seperti nilai C, gamma, serta jenis kernel yang digunakan. Untuk memperoleh konfigurasi terbaik teknik optimasi dapat menggunakan metode *grid search* yang secara sistematis mengevaluasi berbagai kombinasi parameter untuk menemukan model konfigurasi dengan performa paling optimal.

Dalam penelitian sebelumnya, sejumlah peneliti telah memanfaatkan metode SVM untuk menganalisis sentimen pada berbagai domain. Sebagai contoh, dalam analisis sentimen pengguna Twitter terhadap pelayanan Telkom dan Biznet, penerapan SVM menghasilkan akurasi 79,6% untuk Telkom dan 83,2% untuk Biznet, dengan *precision* masing-masing 76,5% dan 78,8% [7]. Selain itu, penelitian mengenai analisis sentimen review aplikasi Ruangguru dengan SVM dan optimasi *K-fold cross-validation* mendapatkan nilai akurasi terbaik 90,2% pada *K-fold* 10, dengan *precision* 90,3% [8]. Penelitian lain menunjukkan bahwa optimasi parameter SVM menggunakan algoritma *grid search* mampu meningkatkan kinerja klasifikasi secara signifikan di mana kombinasi parameter terbaik menghasilkan akurasi hingga 94% pada identifikasi citra biji kopi robusta [9].

Namun berbeda dari ulasan platform umum seperti Twitter maupun aplikasi pendidikan seperti Ruangguru, ulasan pada sektor industri kecantikan memiliki karakteristik linguistik yang lebih subjektif, emosional, dan kaya dengan istilah teknis kosmetik sehingga membentuk beberapa pola data non-linear [10]. Beberapa studi menemukan bahwa pengguna sering memadukan deskripsi teknis dengan penilaian emosional dalam satu ulasan, sehingga distribusi fitur sentimen sulit dipetakan secara linear dan memerlukan pendekatan pemodelan berbasis non-linear [11]. Hingga saat ini, belum banyak penelitian yang secara spesifik menganalisa sentimen pengguna pada platform kecantikan berbasis komunitas di Indonesia, padahal ulasan kecantikan memiliki pengaruh signifikan terhadap keputusan pembelian dibanding kategori produk lain.

Mengacu pada pernyataan tersebut, peneliti terdorong untuk mengeksplorasi lebih lanjut analisis sentimen dengan penerapan metode SVM terutama perbandingan performa SVM sebelum dan sesudah dioptimalkan *grid search* pada ulasan pengguna aplikasi Soco di Google Play Store. Oleh karena itu, penelitian ini berfokus kepada evaluasi kinerja kernel linear dan RBF serta menguji apakah optimasi *hyperparameter* melalui *grid search* dapat menghasilkan peningkatan akurasi dibanding kernel *baseline* tanpa optimasi. Melalui penelitian ini diharapkan dapat menghasilkan model klasifikasi yang lebih akurat dan memberikan rekomendasi konkret bagi pengembang untuk meningkatkan kualitas layanan *e-commerce* berbasis komunitas.

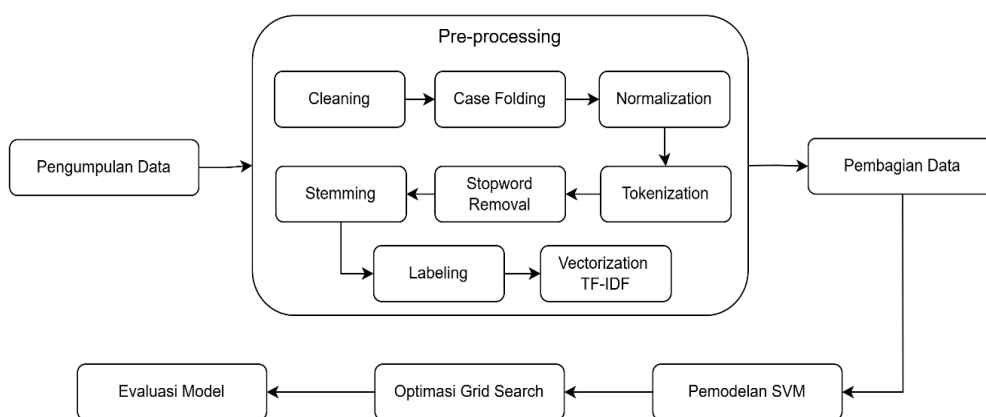
## 2. METODOLOGI PENELITIAN

### 2.1 Metodologi Penelitian

Penelitian ini menggunakan pendekatan kuantitatif berbasis komputasi dengan menerapkan algoritma SVM untuk klasifikasi sentimen ulasan pengguna aplikasi Soco. Seluruh proses pengembangan model dilakukan menggunakan platform Google *Colaboratory* sebagai lingkungan berbasis *cloud* untuk pemrograman *Phyton* yang memungkinkan integrasi eksekusi kode, teks, gambar, dan elemen multimedia yang memudahkan pengembangan model *machine learning* dalam satu *notebook* tanpa instalasi lokal.

### 2.2 Tahapan Penelitian

Tahapan penelitian dirancang secara sistematis untuk menghasilkan model analisis sentimen yang optimal dan dapat digeneralisasikan dengan baik. Gambar 1 berikut merupakan tahapan atau rangkaian kegiatan penelitian.



Gambar 1. Alur Proses Penelitian

Alur penelitian ditunjukkan pada Gambar 1 yang terdiri dari pengumpulan data → *preprocessing* (*cleaning*, *case folding*, *normalization*, *tokenization*, *stopword removal*, *stemming*, *labeling*, dan *vectorization*) → pembagian



data → pemodelan SVM → optimasi *Grid Search* → evaluasi model. Setiap tahapan saling berkaitan dan membentuk suatu siklus yang di mulai dari tahap pengumpulan data hingga tahap evaluasi model. Proses mencakup beberapa tahapan berikut meliputi:

- a. Pengumpulan Data  
 Dalam penelitian ini data dikumpulkan dari berbagai ulasan pengguna aplikasi Soco dari Google Play Store via web *scraping* menggunakan *library google-play-scraper*. Kualitas data sangat menentukan keakuratan model sehingga tahap ini perlu memastikan data yang diambil relevan, representatif, dan bebas dari duplikasi. Sebanyak 3.387 ulasan dikumpulkan dalam format *Comma-Separated Values* (CSV) kemudian diubah menjadi dataset terstruktur.
- b. Data *Preprocessing*  
 Tahapan *preprocessing* merupakan rangkaian proses mengonversi data tidak terstruktur menjadi format terstruktur untuk meningkatkan kualitas serta konsistensi data untuk tahapan analisis [12]. Tahapan ini membersihkan dan mempersiapkan teks melalui *cleaning*, *normalization*, *case folding*, *stopword removal*, *tokenizing*, *stemming*, *vectorization* TF-IDF dan *labeling* otomatis berdasarkan rating [13].
- c. Pembagian Data  
 Tahapan memisahkan dataset menjadi 80:20 dengan metode *stratified split* untuk data latih dan data uji sebagai pelatihan dan validasi generalisasi. Satu subset digunakan sebagai data uji sementara subset lainnya digunakan sebagai data latih analisis model [14].
- d. Pemodelan Data SVM  
 Model analisis yang digunakan yakni algoritma SVM. Prinsip utama SVM adalah menemukan *hyperplane* yang memaksimalkan jarak antar kelas dalam ruang fitur sehingga solusi yang diperoleh bersifat unik dan optimal [15]. Penerapan SVM untuk klasifikasi sentimen mencari *hyperplane* optimal dengan margin maksimal dan kernel *trick* untuk data non-linear.
- e. Optimasi Grid search  
*Grid search* merupakan teknik pencarian terarah yang secara sistematis mengevaluasi berbagai kombinasi nilai *hyperparameter* dengan melakukan uji coba satu persatu kombinasi model dan melakukan validasi pada setiap kombinasi [16]. Kombinasi *hyperparameter* SVM terdiri dari ( $C = [0.1, 1, 10, 100]$ ,  $\gamma = [0.01, 0.1, 1, 10]$ , dan kernel = ['linear', 'rbf']) secara sistematis dengan teknik validasi *5-Fold-cross-validation* untuk mendapatkan konfigurasi terbaik. Kernel linear dipergunakan ketika data relatif mampu dipisah dengan linear, sedangkan kernel rbf diterapkan untuk memodelkan pola hubungan non-linear dengan cara memproyeksikan data ke ruang fitur dengan dimensi lebih tinggi supaya mampu dipisah dengan optimal.
- f. Evaluasi Model  
 Evaluasi kinerja algoritma klasifikasi yang telah digunakan dalam penelitian untuk menentukan seberapa baik kinerja model dalam memprediksi data baru. Performa yang di ukur diantaranya, *confusion matrix*, *precision*, *accuracy*, *recall*, dan *F1-score*.

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Pengumpulan Data

Data dikumpulkan dari ulasan pengguna aplikasi Soco melalui Google Play Store memanfaatkan teknik web *scraping* pada Google *Colaboratory*. Data didapat dari rentang tahun 2019 hingga 2025 meliputi *username*, *review*, *rating*, dan *date* lalu disimpan dalam format *Comma-Separated Values* (CSV). Hasil *scraping* mendapatkan data sebanyak 3.387 ulasan pengguna. Ulasan pengguna ditampilkan pada Tabel 1.

Tabel 1. Hasil Scraping Ulasan

Username	Review	Rating	Date
Kadek Agung	saya sangat suka aplikasi ini	5	8/22/2025 13:53
dewa ayu	Kenapa saya tidak bisa instal ya?	2	5/25/2024 1:31
El Mf	Gk bisa di download	1	8/25/2023 0:33
Syifa Nur Azizah	produknya ori semuaaaa	5	5/11/2020 14:02
Pengguna Google	ga bsa2 mau belanja	1	4/24/2019 15:52

Pada Tabel 1 menunjukkan beberapa contoh ulasan pengguna yang memiliki kalimat informal, penggunaan emotikon serta singkatan. Selain itu, terdapat keberagam rating yang memperlihatkan variasi sentimen positif dan negatif.

#### 3.2 Data Preprocessing

Tahap awal dalam menyiapkan data mentah agar memenuhi standar kualitas data yang dibutuhkan dalam proses analisis dan penambangan data melalui data *preprocessing*. Proses ini mencakup berbagai langkah seperti *cleaning*, *case folding*, *tokenizing*, *stemming*, *labeling*, dan *vectorization* yang bertujuan guna mengganti teks mentah jadi format lebih terstruktur serta mudah diproses oleh algoritma. Berikut tahapan dari *preprocessing* diantaranya:



### 3.2.1 Cleaning

Proses menghapus elemen tidak selaras berupa tanda baca, URL, angka, karakter khusus dari teks. Tujuannya meminimalkan gangguan semantik dalam proses ekstraksi fitur sehingga algoritma fokus pada konten yang relevan. Tahapan ini memanfaatkan library *re* (*regular expression*) untuk mengenali dan menghapus pola teks yang tidak diperlukan secara otomatis. Hasil tahapan ini ditampilkan pada Tabel 2.

**Tabel 2.** Hasil Cleaning

Review	Cleaned
saya sangat suka aplikasi inii	saya sangat suka aplikasi inii
Kenapa saya tidak bisa instal ya?	Kenapa saya tidak bisa instal ya
Gk bisa di download	Gk bisa di download
produknya ori semuaaaa	produknya ori semua
ga bsa2 mau belanja	ga bsa mau belanja

Hasil pada Tabel 2 menunjukkan proses *cleaning* yang berhasil menghapus emotikon, tanda baca, dan angka yang mengganggu struktur kalimat sembari mempertahankan makna inti dari setiap ulasan.

### 3.2.2 Case Folding

Proses mengganti seluruh karakter huruf jadi huruf kecil (*lowercase*) untuk menjaga konsistensi dan menghindari redundansi dalam kosakata. Tahap ini mengurangi kompleksitas data karena sistem memperlakukan kata dengan bentuk huruf berbeda sebagai satu entitas yang sama. Tahapan dilakukan dengan menggunakan fungsi *lower()* pada *code python* yang secara otomatis mengubah seluruh huruf dalam teks menjadi huruf kecil. Hasil tahapan ini ditampilkan pada Tabel 3.

**Tabel 3.** Hasil Case Folding

Case Folding
saya sangat suka aplikasi inii
kenapa saya tidak bisa instal ya
gk bisa di download
produknya ori semua
ga bsa mau belanja

Hasil pada Tabel 3 menunjukkan seluruh karakter huruf telah berhasil dikonversi ke huruf kecil.

### 3.2.3 Normalization

Proses menyelaraskan kata-kata yang tidak baku jadi bentuk yang selaras untuk memastikan semua kata sesuai kaidah ejaan Bahasa Indonesia yang benar [17]. Proses penting pada teks informal yang banyak mengandung singkatan atau kata slang agar mendapatkan format teks yang konsisten di seluruh dataset. Tahapan ini memanfaatkan kamus kata baku tersedia di *kaggle* [18]. Hasil tahapan dipaparkan dalam Tabel 4.

**Tabel 4.** Hasil Normalization

Normalization
saya sangat suka aplikasi ini
kenapa saya tidak bisa instal ya
tidak bisa di download
produknya orisinal semua
tidak bisa mau belanja

Hasil pada Tabel 4 menunjukkan bahwa beberapa kata slang seperti “inii”, “gk”, “ori”, dan “ga bsa” telah berhasil diubah menjadi bentuk baku Bahasa Indonesia (“ini”, “tidak”, “orisinal”, “tidak bisa”).

### 3.2.4 Tokenization

Proses membagi teks menjadi unit terkecil atau token. Tahapan ini sebagai dasar bagi pembentukan vektor fitur. Tahap ini membantu sistem mengenali batas antar kata serta memfasilitasi representasi teks dalam bentuk numerik sehingga setiap kata diidentifikasi secara akurat untuk kebutuhan pemodelan dan analisis [19]. Tahapan ini menggunakan library NLTK. Hasil tahapan dipaparkan dalam Tabel 5.

**Tabel 5.** Hasil Tokenization

Tokenization
['saya', 'sangat', 'suka', 'aplikasi', 'ini']
['kenapa', 'saya', 'tidak', 'bisa', 'instal', 'ya']
['tidak', 'bisa', 'di', 'download']

Tokenization
['produknya', 'orisinal', 'semua']
['tidak', 'bisa', 'mau', 'belanja']

Hasil pada Tabel 5 menunjukkan bahwa setiap ulasan telah dipisahkan ke dalam daftar token tanpa kehilangan struktur makna kalimat.

### 3.2.5 Stopword Removal

Proses menghapus kata-kata umum yang tidak menyumbang makna signifikan atas analisis teks. Langkah ini membantu menghilangkan noise serta merupakan komponen utama dalam *pipeline preprocessing* berfungsi mempertahankan kata-kata yang bermakna dan mempercepat proses ekstraksi fitur. Tahapan ini menggunakan *library Sastrawi* modul *StopwordRemovalFactory*. Hasil tahapan ini ditampilkan Tabel 6.

Tabel 6. Hasil Stopword Removal

Stopword Removal
['sangat', 'suka', 'aplikasi']
['instal']
['download']
['produknya', 'orisinal', 'semua']
['mau', 'belanja']

Hasil pada Tabel 6 menunjukkan bahwa kata-kata dengan frekuensi tinggi tetapi tidak membawa makna sentimen seperti (saya, tidak, di, dan ya) telah dihilangkan, sehingga hanya terdapat kata-kata bermuatan informasi.

### 3.2.6 Stemming

Proses mengganti variasi kata jadi bentuk dasar yang sama supaya variasi morfologis seperti “bermain”, “memainkan”, dan “bermainan” dianggap sebagai satu entitas semantik yang sama, yaitu “main”. Proses ini menyatukan berbagai macam kata turunan ke dalam satu bentuk akar yang sama [20]. Proses *stemming* menekan redundansi kata dan memperkuat representasi fitur semantik pada model pembelajaran mesin. Tahapan ini menggunakan *library Sastrawi* modul *StemmerFactory*. Hasil tahapan ini dipaparkan dalam Tabel 7.

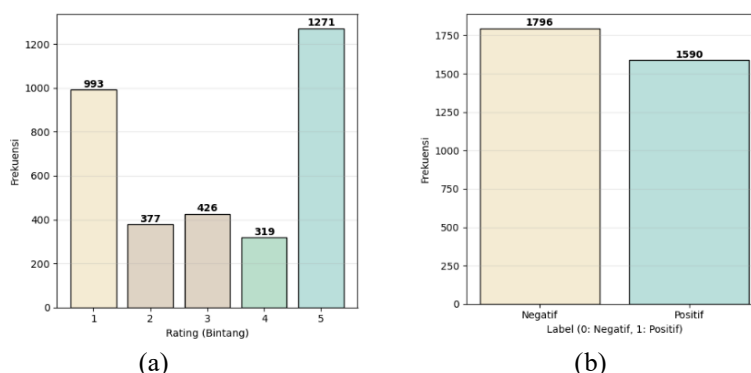
Tabel 7. Hasil Stemming

Stemming
['sangat', 'suka', 'aplikasi']
['instal']
['download']
['produk', 'orisinal', 'semua']
['mau', 'belanja']

Hasil pada Tabel 7 menunjukkan bahwa kata-kata yang mengalami perubahan bentuk telah dikembalikan ke bentuk dasar sehingga tidak lagi muncul sebagai fitur yang berbeda dalam pembobotan TF-IDF.

### 3.2.7 Labeling

Tahap *labeling* merupakan tahap penting dalam analisis sentimen yang bertujuan untuk menentukan kelas opini setiap ulasan umumnya dipecah jadi tiga kategori utama yakni positif, negatif, netral. Distribusi pelabelan sentimen aplikasi Soco dibagi menjadi dua kelas yaitu, untuk ulasan dengan rating 4–5 menjadi kelas positif sebanyak 1.796 dan untuk rating 1–3 menjadi kelompok kelas negatif sebanyak 1590. Hasil distribusi ditampilkan pada Gambar 2(a) dan hasil setelah pelabelan ditampilkan pada Gambar 2(b).



Gambar 2. Distribusi *labeling* dari (a) rating pengguna dan setelah pengelompokan (b) *labeling*

Gambar 2(a) menunjukkan distribusi rating bintang yang diberikan pengguna aplikasi Soco pada Google Play Store. Gambar 2(b) menyajikan hasil distribusi ulasan setelah rating dikonversi menjadi dua kelas sentimen, yaitu positif (rating 4–5) dan negatif (rating 1–3).

### 3.2.8 Vectorization

Tahap mengubah teks hasil preprocessing jadi representasi numerik memanfaatkan metode TF-IDF (*Term Frequency–Inverse Document Frequency*) supaya mampu diproses algoritma pembelajaran mesin [21]. Proses ini dilakukan menggunakan *library scikit-learn* dengan konfigurasi unigram. Hasilnya mencakup matriks mencerminkan tiap ulasan berbentuk angka dan menjadi dasar utama untuk analisis sentimen serta klasifikasi. Berikut hasil kata dengan nilai TF-IDF muncul pada tabel 8.

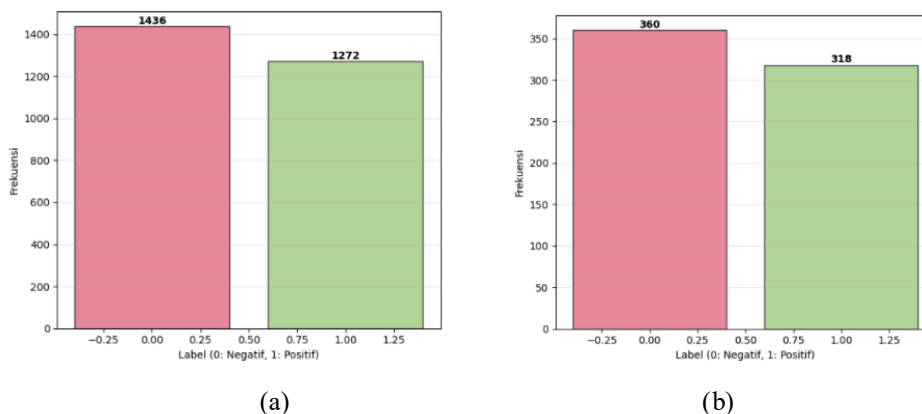
**Tabel 8.** Daftar Hasil Vectorization

Kata	Rata-rata	Frekuensi
aplikasi	0.028386	711
banget	0.027319	654
bagus	0.04241	433
belanja	0.016122	391
produk	0.014382	332
sociolla	0.013632	325
suka	0.018869	323
baik	0.016073	309
barang	0.012485	307

Tabel 8 menampilkan beberapa contoh proses *vectorization* TF-IDF yang mengubah token kata hasil preprocessing menjadi representasi numerik sehingga dapat diproses oleh algoritma klasifikasi.

### 3.3 Pembagian Data

Dataset dibagi menjadi dua subset, yaitu data uji (*training set*) dan data latih (*testing set*). Pembagian data secara proporsional memungkinkan model belajar pola dari data latih dan diuji pada data yang tidak dikenal sebelumnya. Proporsi pembagian yang digunakan pada penelitian ini ialah 80% untuk data latih dan 20% untuk data uji seperti yang tertera pada Gambar 3.



**Gambar 3.** Pembagian data (a) *training set* dan (b) *testing set*

Gambar 3 menunjukkan proses pembagian dataset menjadi dua kelompok data, yaitu Gambar 3(a) *training set* sebanyak 1.436 data negatif dan 1272 data positif dan Gambar 3(b) *testing set* sebanyak 360 data negatif dan 318 data positif.

### 3.4 Pemodelan Data

Penelitian ini menggunakan SVM dua jenis kernel, yakni linear dan *Radial Basis Function* (RBF) sebagai model *baseline*. Kernel linear dipergunakan data secara alami mampu dipisahkan dengan linear sedangkan kernel RBF digunakan untuk menangani pola non-linear yang kompleks dengan mempertahankan fleksibilitas dalam representasi fitur [22]. Parameter awal yang digunakan pada kedua model ditampilkan pada Tabel 9.

**Tabel 9.** Parameter SVM

Model SVM	Parameter		
	C	Gamma	Random State
SVM Linear	1.0	Scale	42

Model SVM	Parameter		
	C	Gamma	Random State
SVM RBF	1.0	Scale	42

Parameter pada Tabel 9 untuk model SVM linear dan RBF menggunakan *baseline* model dengan C diatur pada 1.0, Gamma diatur ke *scale* dan random state bernilai 42.

### 3.5 Optimasi Grid Search

Pada model SVM parameter yang dioptimalkan meliputi C, gamma, dan jenis kernel. Parameter C mengatur keseimbangan antara memperlebar margin pemisahan dan meminimalkan kesalahan klasifikasi sedangkan  $\gamma$  (gamma) menentukan sejauh mana tiap titik data memengaruhi pembentukan batas keputusan. Kombinasi parameter yang diuji dipaparkan di Tabel 10.

Tabel 10. Parameter Optimasi

	Parameter Grid		
	C	Gamma	Kernel
Optimasi	0,1	0,001	linear
	1	0,01	
	10	0,1	rbf
	100	1	

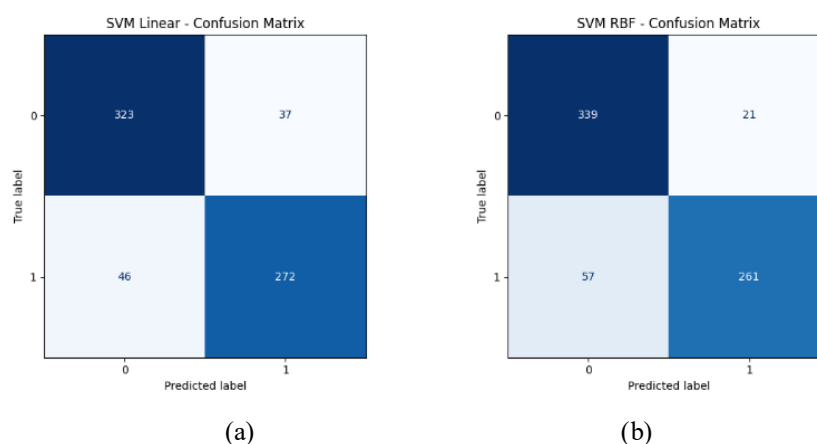
Tabel 10 menunjukkan konfigurasi parameter yang dieksplorasi dalam proses *grid search* untuk mengidentifikasi *hyperparameter* pada model SVM.

### 3.6 Evaluasi Model

Evaluasi dilaksanakan memanfaatkan metrik *accuracy*, *recall*, *precision*, dan *F1-score* guna menilai performa model mengklasifikasikan sentimen. Penggunaan keempat metrik memberi deskripsi keseluruhan mengenai kompetensi model mengetahui kelas secara benar, khususnya data tidak seimbang. Analisis hasil evaluasi ini menentukan efektivitas penerapan SVM dan kontribusi metode *grid search* terhadap peningkatan performa model.

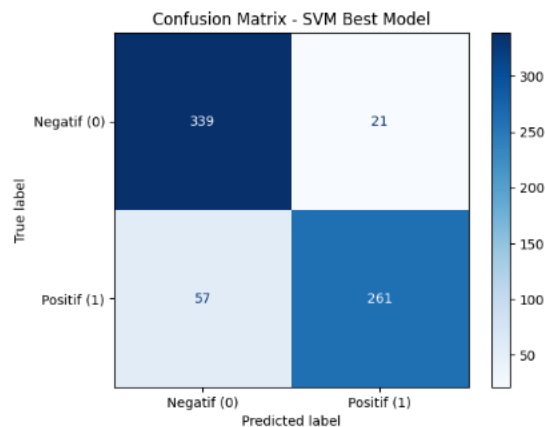
#### 3.6.1 Confusing Matrix

*Confusion matrix* termasuk evaluasi kerja dalam klasifikasi data mining yang menampilkan gambaran menyeluruh mengenai hasil prediksi model. Matriks ini memberikan representasi menyeluruh mengenai hubungan antara nilai aktual dan hasil prediksi model yang disusun dalam bentuk tabel dua dimensi. Hasil prediksi dari *confusion matrix* ditampilkan pada Gambar 4.



Gambar 4. Diagram *confusion matrix* svm (a) kernel linear dan (b) kernel rbf

Dari gambar 4(a), SVM linear menunjukkan *True Negative* (TN) sebesar 323, *False Positive* (FP) 37, *False Negative* (FN) 46, *True Positive* (TP) 272. Hal ini menandakan bahwa kernel linear sedikit lebih baik dalam mendeteksi data positif karena jumlah FN relatif lebih rendah dibanding RBF. Namun kernel linear kurang optimal dalam memprediksi data negatif terlihat dari TN yang lebih rendah dan FP yang lebih tinggi. Sebaliknya pada gambar 4(b), SVM rbf memiliki TN sebesar 339 dan FP 21 sementara FN 57 dan TP 261 menunjukkan bahwa kernel RBF lebih akurat dalam mengklasifikasikan data negatif meskipun sedikit kurang sensitif terhadap data positif. Perbandingan ini menunjukkan adanya *trade-off* antara kemampuan mendeteksi data positif dan negatif antara kedua kernel. Setelah dilakukan proses optimasi menggunakan metode *grid search* dengan parameter yang telah ditentukan diperoleh hasil prediksi model yang divisualisasikan dalam bentuk *confusion matrix* sebagaimana dilhatkan Gambar 8.



Gambar 5. Diagram confusion matrix best model

Confusion matrix hasil best model ditunjukkan pada Gambar 5 dengan TN 339, FP 21, FN 57, dan TP 261. Hasil ini identik dengan performa SVM RBF sebelum optimasi yang mengonfirmasi bahwa kernel RBF dengan parameter terbaik merupakan model paling optimal dalam penelitian ini. Model ini menunjukkan keseimbangan yang baik prediksi positif negatif maka mampu memberikan pengelompokkan lebih stabil dan mampu diandalkan.

### 3.6.2 Accuracy

Accuracy merupakan metrik yang mengukur tingkat kedekatan hasil prediksi model dengan nilai aktual. Nilai ini menunjukkan proporsi prediksi benar atas keseluruhan total prediksi yang dilaksanakan. Makin tinggi nilai akurasi makin baik kemampuan model memprediksi kelas dengan benar. Pada penelitian ini didapatkan nilai accuracy SVM yang ditampilkan pada Tabel 11.

Tabel 11. Nilai Accuracy

	Linear	Rbf	Best Model
Accuracy	0.8776	0.8850	0.8622

Berdasarkan hasil pada Tabel 11, model SVM dengan baseline RBF menghasilkan accuracy tertinggi sebesar 0,8850, diikuti kernel baseline linear dengan accuracy 0,8776. Sementara itu, best model hasil grid search menghasilkan accuracy 0,8622 dimana nilai ini lebih rendah dibandingkan performa baseline RBF. Berikut rata-rata nilai accuracy dari seluruh kombinasi hyperparameter yang dipaparkan Tabel 12.

Tabel 12. Nilai Accuracy Grid search

C	Gamma	Kernel	Accuracy
1.0	1.000	Rbf	0.862258
1.0	0.010	Linear	0.859676
1.0	0.001	Linear	0.859676
1.0	0.100	Linear	0.859676
1.0	1.000	Linear	0.859676
10.0	0.010	Rbf	0.858940
10.0	1.000	Rbf	0.857088
1.0	0.100	Rbf	0.856723
100.0	1.000	Rbf	0.856719
10.0	0.100	Rbf	0.854500

Hasil optimasi model menggunakan metode grid search menunjukkan bahwa kombinasi parameter terbaik algoritma SVM diperoleh nilai C = 1, gamma = 1, kernel = RBF. Konfigurasi ini memungkinkan model untuk memetakan pola non-linear pada data secara efektif sambil mempertahankan tingkat regularisasi yang seimbang. Namun, accuracy yang dihasilkan model optimasi 0,862258 tetap berada di bawah baseline RBF 0,8850.

### 3.6.3 Classification Report

Classification report merupakan bentuk ringkasan hasil evaluasi yang berisi sejumlah metrik utama seperti precision, F1-score, dan recall. Metrik precision menunjukkan tingkat ketepatan model memperkirakan kelas positif dengan benar sedangkan recall mengukur sejauh apa model berhasil menemukan seluruh sampel positif di dalam dataset. F1-score kemudian digunakan untuk menyeimbangkan kedua aspek tersebut sebagai rerata harmonis precision serta recall. Pada penelitian ini didapatkan nilai klasifikasi report SVM yang berisi precision, recall, f-1 score yang ditunjukkan pada Tabel 13.

**Tabel 13.** Classification Report

Label	Matrix					
	Linear		Rbf		Best Model	
Precision	0.88	0.88	0.86	0.93	0.86	0.93
Recall	0.90	0.86	0.94	0.82	0.94	0.82
F-1 Score	0.89	0.87	0.90	0.87	0.90	0.87

Berdasarkan Tabel 13, model SVM dengan kernel Linear menunjukkan kinerja prediksi yang relatif seimbang antara kedua kelas, dengan *recall* kelas 0 sebesar 0,90 dan *recall* kelas 1 sebesar 0,86. Nilai *recall* kelas 0 yang lebih tinggi mengindikasikan bahwa kernel Linear lebih sensitif dalam mendeteksi ulasan negatif, namun sedikit kurang responsif terhadap ulasan positif. Model SVM kernel RBF menunjukkan pola yang berbeda. *Recall* kelas 0 mencapai 0,94 menandakan kemampuan model dalam mengenali ulasan negatif secara lebih konsisten dibanding kernel Linear. Namun, *recall* kelas 1 menurun menjadi 0,82 sehingga deteksi terhadap ulasan positif relatif lebih rendah. Meskipun demikian, nilai *precision* dan *F1-score* kelas 1 tetap tinggi (0,93 dan 0,87) yang mencerminkan keseimbangan prediksi dan tingkat kesalahan yang rendah pada kelas positif. Hasil *best model grid search* memiliki nilai metrik yang identik dengan kernel RBF sebelum optimasi. Hal ini mengonfirmasi bahwa kernel RBF tetap menjadi model terbaik secara keseluruhan meskipun hasil optimasi tidak meningkatkan kinerja di atas performa *baseline*. Hasil ini menunjukkan kemampuan kernel RBF dalam menangkap pola non-linear dan adaptif terhadap pola linguistik pada data ulasan pengguna aplikasi Soco terutama pada kelas negatif tanpa mengorbankan stabilitas performa pada kelas positif.

**3.6.4 Distribusi Sentimen**

Tahap terakhir dalam evaluasi adalah visualisasi distribusi sentimen yang menggambarkan proporsi antara kelas sentimen positif dan negatif. Berdasarkan hasil klasifikasi diperoleh 1.7960 ulasan positif (53%) dan 1.590 ulasan negatif (47%). Visualisasi ini membantu peneliti memahami kecenderungan umum dari opini pengguna terhadap aplikasi serta memberikan wawasan mengenai persebaran persepsi pengguna. Visualisasi *wordcloud* pada penelitian ini menampilkan distribusi kata paling sering timbul diulasan pengguna berdasarkan klasifikasi sentimen positif serta negatif. Penyebaran kata pada kedua kategori sentimen memberikan gambaran umum mengenai pola persepsi pengguna.



**Gambar 6.** Wordcloud (a) sentimen positif dan (b) sentimen negatif

Hasil penyebaran *wordcloud* positif dipaparkan pada Gambar 6(a) dan penyebaran *wordcloud* negatif dipaparkan pada Gambar 6(b). *Wordcloud* sentimen positif pengguna sering mengombinasikan evaluasi emosional (“bagus”, “bantu”, “diskon”, “good”) dengan istilah teknis produk (“skincare”, “orisional”) yang saling tumpang-tindih. Pola serupa juga terlihat pada sentimen negatif, di mana keluhan terhadap performa aplikasi (“lama”, “update”) disampaikan bersamaan dengan penilaian emosional seperti “kecewa” atau “keluar”. Keberagaman struktur leksikal dan sintaksis tersebut menyebabkan distribusi fitur sentimen tidak membentuk pemisahan linear sederhana. Kondisi ini menghasilkan *overlapping* antar fitur teks dan membentuk pola distribusi non-linear di ruang fitur. Distribusi *wordcloud* menunjukkan kernel RBF lebih efektif dibandingkan kernel linear dalam mempelajari pola data seperti ini. Kernel RBF dapat memetakan relasi kompleks antar fitur emosional dan teknis sehingga mampu menangkap pola *latent semantics* yang tidak dapat dijelaskan hanya melalui pemetaan linier.

**4. KESIMPULAN**

Riset penelitian telah menerapkan algoritma *Support Vector Machine* (SVM) dalam analisis sentimen ulasan pengguna aplikasi Soco dengan hasil yang akurat dan stabil. Hasil penelitian menunjukkan bahwa SVM dengan kernel RBF menghasilkan performa paling optimal dibandingkan kernel linear. Akurasi tertinggi dicapai oleh *baseline* RBF 0,885 diikuti kernel linear 0,8776 sedangkan model hasil optimasi *grid search* dengan kombinasi parameter paling baik (C = 1, gamma = 1, kernel = RBF) justru berada di bawah keduanya 0,8622. Penelitian ini menunjukkan bahwa kernel RBF tetap menjadi konfigurasi paling unggul untuk klasifikasi sentimen pada ulasan aplikasi Soco selaras dengan penelitian sebelumnya pada domain layanan digital seperti Telkom–Biznet dan Ruangguru yang sama-sama menunjukkan performa kuat metode SVM. Namun, berbeda dengan penelitian sebelumnya yang mengalami



peningkatan akurasi setelah optimasi *hyperparameter* pada penelitian ini justru menemukan pola sebaliknya parameter *baseline* SVM RBF sudah berada pada titik optimal sehingga proses *grid search* tidak menghasilkan peningkatan akurasi dan justru sedikit menurunkannya. Dengan demikian, penelitian ini menunjukkan bahwa optimasi *hyperparameter* tidak selalu menjamin peningkatan performa model terutama ketika *baseline* telah bekerja optimal untuk melakukan klasifikasi sentimen pada teks ulasan yang memiliki pola linguistik non-linear dan bersifat subjektif dan emosional. Hasil penelitian dapat dimanfaatkan oleh pengembang aplikasi sebagai dasar pengambilan keputusan dalam meningkatkan kualitas fitur, memperbaiki kendala teknis, dan memperkuat pengalaman belanja pengguna berbasis komunitas digital.

## REFERENCES

- [1] A. Royan, S. S. Tyas, and M. Ridwan, "Analisis User Interface & User Experience Aplikasi Toco Menggunakan Metode Design Thinking," *J. Inform. dan Tek. Elektro Terap.*, vol. 13, no. 3, pp. 1060–1069, 2024, doi: <http://dx.doi.org/10.23960/jitet.v13i3.7150>.
- [2] E. V. Sitompul, "Pengaruh Komunikasi Pemasaran Digital dan Persepsi Kemudahan Penggunaan Aplikasi Belanja Online terhadap Keputusan Pembelian Produk Skincare: Studi pada Pengguna Aplikasi SOCO by Sociolla," *Arus J. Sos. dan Hum.*, vol. 4, no. 1, pp. 31–43, 2024.
- [3] J. Mansur, V. C. Mawardi, and T. Sutrisno, "Analisis Pendapat Publik Terhadap Public Figure dengan Menggunakan Metode Naive Bayes," *J. Ilmu Komput. dan Sist. Inf.*, vol. 7, no. 2, pp. 185–192, 2020.
- [4] I. Zulfahmi, "Analisis Sentimen Aplikasi PLN Mobile Menggunakan Metode Decision Tree," *J. Penelit. Rumpun Ilmu Tek.*, vol. 3, no. 1, pp. 11–21, 2024, doi: <https://doi.org/10.55606/juprit.v3i1.3096>.
- [5] R. Merdiansah and A. Ali Ridha, "Analisis Sentimen Pengguna X Indonesia Terkait Kendaraan Listrik Menggunakan IndoBERT," *J. Ilmu Komput. dan Sist. Inf.*, vol. 7, no. 1, pp. 221–228, 2024, doi: <https://doi.org/10.55338/jikomsi.v7i1.2895>.
- [6] I. N. Hassanah, S. Faisal, and A. M. Siregar, "Perbandingan algoritma Support Vector Machine dengan Decision Tree pada aplikasi Ruang Guru," *Kumpul. J. Ilmu Komput.*, vol. 10, no. 1, pp. 39–50, 2023, doi: [10.20527/klik.v10i1.602](https://doi.org/10.20527/klik.v10i1.602).
- [7] B. W. Sari and F. F. Haranto, "Implementasi Support Vector Machine Untuk Analisis Sentimen Pengguna Twitter Terhadap Pelayanan Telkom Dan Biznet," *J. Pilar Nusa Mandiri*, vol. 15, no. 2, pp. 171–176, 2019, doi: <https://doi.org/10.33480/pilar.v15i2.699>.
- [8] F. F. Irfani, M. Triyanto, and A. D. H. Hartanto, "Analisis Sentimen Review Aplikasi Ruangguru Menggunakan Algoritma Support Vector Machine," *J. Bisnis, Manajemen, dan Inform.*, vol. 16, no. 3, pp. 258–266, 2020, doi: <https://doi.org/10.26487/jbmi.v16i3.8607>.
- [9] H. Apriani, J. H. Jaman, and R. I. Adam, "Optimasi SVM Menggunakan Algoritme Grid Search untuk Identifikasi Citra Biji Kopi Robusta Berdasarkan Circularity dan Eccentricity," *J. Teknol. dan Sist. Komput.*, vol. 10, no. 1, pp. 12–19, 2022, doi: <https://doi.org/10.14710/jtsiskom.2022.13807>.
- [10] L. S. Lestari, T. Sutrisno, and I. Lewenusu, "Sentiment Analysis on Skincare Product Reviews Using Lexicon-Based and Comparison of SVM Kernel," *Indones. J. Soc. Technol.*, vol. 5, no. 11, pp. 5250–5259, 2024, doi: <https://doi.org/10.59141/jist.v5i11.8769>.
- [11] K. D. Dolar, R. Shilu, and T. M. Bhalodia, "Aspect Based Sentiment Analysis of Beauty Product Review using Machine Learning Algorithm," *Int. J. Innov. Res. Technol.*, vol. 11, no. 12, pp. 5149–5153, 2025, doi: [ijirt.org/article?manuscript=178784](https://doi.org/10.26487/jbmi.v16i3.8607).
- [12] C. Nurita, K. Sari, and R. R. Suryono, "Perbandingan Algoritma Naïve Bayes , Random Forest , dan SVM Untuk Analisis Sentiment Aplikasi PLN Mobile Pada Google Play Store," *Build. Informatics, Technol. Sci.*, vol. 7, no. 1, 2025, doi: <https://doi.org/10.47065/bits.v7i1.7164>.
- [13] N. Dhanur, A. Putra, D. Juardi, A. Sholehudin, and S. Abadi, "Penerapan Support Vector Machine pada Analisis Sentimen Kebijakan Pembelajaran Tatap Muka Saat Pandemi 2021," *J. Inform. Teknol.*, vol. 4, no. 2, pp. 190–199, 2023, doi: [10.37373/infotech.v4i2.790](https://doi.org/10.37373/infotech.v4i2.790).
- [14] A. Fullah *et al.*, "Analisis Sentimen Isu Penyalahgunaan Data Pada Layanan Pinjaman Online Menggunakan Support Vector Machine Di Platform X," *JITET (Jurnal Inform. dan Tek. Elektro Terap.)*, vol. 13, no. 3, pp. 1038–1047, 2025, doi: [10.23960/jitet.v13i3S1.7976](https://doi.org/10.23960/jitet.v13i3S1.7976).
- [15] S. Andrean, "Review Aplikasi Kredivo Menggunakan Analisis Sentimen dengan Algoritma Support Vector Machine," *J. Teknol. dan Sist. Inf. Univrab*, vol. 9, no. 1, pp. 39–49, 2024, doi: <https://doi.org/10.36341/rabit.v9i1.4107>.
- [16] A. Toha, P. Purwono, W. Gata, and A. Toha, "Model Prediksi Kualitas Udara dengan Support Vector Machines dengan Optimasi Hyperparameter GridSearch CV," *Bul. Ilm. Sarj. Tek. Elektro*, vol. 4, no. 1, pp. 12–21, 2022, doi: <https://doi.org/10.12928/biste.v4i1.6079>.
- [17] H. Tuhuteru, "Analisis Sentimen Masyarakat Terhadap Pembatasan Sosial Berksala Besar Menggunakan Algoritma Support Vector Machine," *Inf. Syst. Dev.*, vol. 5, no. 2, pp. 7–14, 2020.
- [18] Fornigulo, "Kamus Slag." Accessed: May 20, 2025. [Online]. Available: <https://www.kaggle.com/datasets/fornigulo/kamus-slag>
- [19] D. Jurafsky and J. H. Martin, *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models*, 3rd ed. Upper Saddle River, New Jersey: Prentice Hall, 2025.
- [20] S. Mujilawati, "Analisis sentimen pengguna aplikasi ChatGPT berdasarkan rating menggunakan metode leksikon," *J. Teknol. dan Sist. Inf. Univrab*, vol. 9, no. 1, pp. 131–137, 2024, doi: <https://doi.org/10.36341/rabit.v9i1.3845>.
- [21] Yusril, W. Fuadi, and Y. Afrillia, "Analisis Sentimen Review Aplikasi Stockbit di Google Play Store dan X Menggunakan Support Vector Machine," *J. Teknol. dan Sist. Inf. Univrab*, vol. 10, no. 2, pp. 1050–1062, 2025, doi: <https://doi.org/10.36341/rabit.v10i2.6446>.
- [22] K. Du, B. Jiang, J. Lu, and J. Hua, "Exploring Kernel Machines and Support Vector Machines : Principles , Techniques , and Future Directions," *Mathematics*, vol. 12, no. 3935, pp. 1–58, 2024, doi: [10.3390/math12243935](https://doi.org/10.3390/math12243935).