



## ***Sentiment Review Analysis on the BSB Mobile Application Using Convolutional Neural Networks and Support Vector Machine Algorithm***

### **Analisis Ulasan Sentimen pada Aplikasi Mobile BSB Menggunakan Algoritma Convolutional Neural Networks dan Support Vector Machine**

**Muhamamad Faza<sup>1</sup>, M. Rudi Sanjaya<sup>2\*</sup>, Ali Ibrahim<sup>3</sup>, M. Husni Syahbani<sup>4</sup>**

<sup>1,2,3,4</sup>Departemen Sistem Informasi, Fakultas Ilmu Komputer,  
Universitas Sriwijaya, Indonesia

E-Mail: <sup>1</sup>mf682478@gmail.com, <sup>2\*</sup>m.rudi.sjy@ilkom,unsri.ac.id

Received Mar 12th 2026; Revised Apr 06th 2026; Accepted Apr 21th 2026; Available Online Apr 26th 2026

Corresponding Author: M. Rudi Sanjaya

Copyright ©2026 by Authors, Published by Institut Riset dan Publikasi Indonesia (IRPI)

#### **Abstract**

*The growing number of user reviews on digital platforms such as the Google Play Store provides various opinions that reflect both satisfaction and dissatisfaction with the application's performance. Sentiment analysis is required to categorize these reviews into positive, negative, or neutral classes, enabling developers to evaluate and improve service quality. This study applies CNN and SVM methods to perform the Support Vector Machine (SVM) to classify sentiments in user-generated reviews of the BSB Handheld application. The research stages include data collection, text preprocessing, and model development for CNN and SVM. The proposed models were tested on the available dataset and evaluated using Accuracy, precision, Recall, and F1-score to assess their effectiveness. The results indicate that results show indicating that the CNN model achieves an Accuracy of 80,10%, while the SVM model reaches 80,70%. Based on the evaluation, SVM performs slightly better than CNN in classifying sentiment. The empirical results indicate that the employed machine-learning methods, particularly SVM, are effective for analyzing user opinions of the BSB Mobile application. The findings of this research are anticipated to provide significant input to developers in improving application features and user experience.*

**Keywords:** BSB Mobile, Convolutional Neural Networks, Sentiment Analysis, Support Vector Machine, Text Classification

#### **Abstrak**

Peningkatan jumlah ulasan yang diberikan pengguna pada platform digital seperti Google Play Store menghasilkan berbagai opini yang mencerminkan kepuasan maupun ketidakpuasan terhadap layanan aplikasi tersebut. Analisis sentimen diperlukan guna mengkategorikan opini pengguna dalam ulasan dari data teks menjadi informasi penting. Penelitian ini menggunakan pendekatan metode *Convolutional Neural Network* (CNN) serta algoritma pendekatan *Support Vector Machine* (SVM) dalam rangka melakukan pengelompokan data sentimen pada ulasan aplikasi BSB Mobile. Tahapan proses penelitian mencakup tahapan pengumpulan data ulasan, pra-pemrosesan teks, pembentukan algoritma CNN beserta SVM, serta penilaian performa memanfaatkan parameter evaluasi, yaitu tingkat akurasi, presisi, *recall*, dan *F1-Score*. Merujuk pada temuan tersebut, dapat disimpulkan dari analisis bahwa arsitektur CNN menghasilkan tingkat akurasi yang menyentuh angka 80,10%, sedangkan model SVM mencapai akurasi 80,70%. Berdasarkan hasil evaluasi, model SVM sedikit lebih unggul dibandingkan dengan CNN dalam mengklasifikasikan ulasan sentimen. Temuan ini menunjukkan bahwa metode *machine learning*, khususnya SVM, dapat menjadi alternatif yang efektif dalam menganalisis opini pengguna terhadap aplikasi BSB Mobile.

**Kata Kunci:** Analisis Sentimen, BSB Mobile, Convolutional Neural Network, Klasifikasi Teks, Support Vector Machine

#### **1. PENDAHULUAN**

Kemajuan teknologi informasi [1] telah menimbulkan perubahan yang sangat signifikan dalam berbagai aspek kehidupan industri perbankan. Layanan keuangan yang sebelumnya bergantung pada

interaksi tatap muka kini telah bertransformasi menjadi layanan digital yang dapat diakses melalui perangkat *Mobile*. *Mobile banking* menjadi salah satu inovasi utama karena memberikan kemudahan akses transaksi, efisiensi waktu, dan keamanan bagi nasabah. Bank Sumsel Babel (BSB) sebagai salah satu Bank Pembangunan Daerah (BPD) di Indonesia juga mengikuti perkembangan ini dengan meluncurkan aplikasi BSB *Mobile* sebagai sarana layanan digital bagi nasabahnya.

Data Google Play Store per Januari 2025 menunjukkan bahwa aplikasi BSB *Mobile* telah mendapatkan lebih dari 8.000 ulasan dengan rating rata-rata 3,9 dari 5. Variasi dalam penilaian ini mengisyaratkan adanya perbedaan pengalaman pengguna dalam menggunakan layanan aplikasi, sehingga menarik untuk dianalisis melalui pendekatan analisis sentimen. Ulasan-ulasan tersebut mencakup berbagai aspek penggunaan, mulai dari kenyamanan, keamanan, hingga kendala teknis yang dialami pengguna. Keluhan yang sering muncul antara lain akun yang terblokir tanpa pemberitahuan, saldo terpotong meskipun transaksi gagal, proses *login* yang lambat, hingga bug pada tampilan antarmuka aplikasi. Namun, jumlah ulasan yang sangat besar dan terus mengalami peningkatan setiap hari membuat pihak Bank Sumsel Babel sulit untuk melakukan analisis secara manual. Proses pembacaan ulasan proses manual tersebut memakan waktu yang cukup signifikan serta rentan menghasilkan interpretasi yang subjektif. Selain itu, pihak bank belum memiliki sistem yang mampu memetakan keluhan pengguna secara otomatis berdasarkan kategori sentimen, sehingga pengambilan keputusan untuk perbaikan aplikasi tidak dapat dilakukan secara optimal. Kondisi ini menunjukkan perlunya sebuah metode analisis yang dapat memproses data ulasan yang berjumlah besar secara cepat, secara tidak bias, serta akurat [2].

Analisis sentimen telah menjadi pendekatan yang umum digunakan untuk mengevaluasi opini interaksi pengguna dengan berbagai fasilitas berbasis teknologi. Sejumlah studi sebelumnya mengungkapkan bahwa metode ini efektif dalam mengidentifikasi kecenderungan opini serta isu utama yang muncul dalam ulasan. Misalnya, Maheri *et al.* [3] menganalisis sentimen pada aplikasi layanan publik menggunakan berbagai algoritma *machine learning* dan menemukan bahwa metode SVM mampu memberikan akurasi lebih stabil dibandingkan algoritma lain pada *dataset* kecil. Namun, penelitian tersebut memiliki keterbatasan karena hanya menggunakan satu teknik representasi fitur tanpa membandingkan pendekatan lain yang lebih kompleks. Penelitian lain oleh Setiawati *et al.* [4] mengevaluasi sentimen pada ulasan platform *e-commerce* dan melaporkan bahwa CNN menghasilkan akurasi tinggi karena kemampuan ekstraksi fiturnya yang lebih baik. Meskipun demikian, model CNN membutuhkan jumlah data yang besar agar dapat melakukan generalisasi secara optimal. Selanjutnya, mengevaluasi kinerja dari berbagai algoritma seperti metode pengklasifikasi yang menerapkan *Teorema Bayes*, SVM, dan algoritma berbasis *ensemble learning* guna klasifikasi sentimen produk skincare. Mereka menemukan bahwa SVM memiliki keunggulan dalam menangani fitur berdimensi tinggi, tetapi kinerjanya menurun ketika dihadapkan pada kelas data yang tidak seimbang [5]. Penelitian selanjutnya menunjukkan bahwa kombinasi metode deep learning dan *machine learning* dapat meningkatkan performa analisis sentimen, namun pendekatan tersebut memerlukan tuning parameter yang kompleks dan sumber daya komputasi yang lebih tinggi. Keempat studi tersebut menunjukkan bahwa tidak ada algoritma yang selalu menjadi yang terbaik dalam setiap kondisi kondisi, dan performa sangat dipengaruhi oleh karakteristik *dataset* yang dianalisis [6].

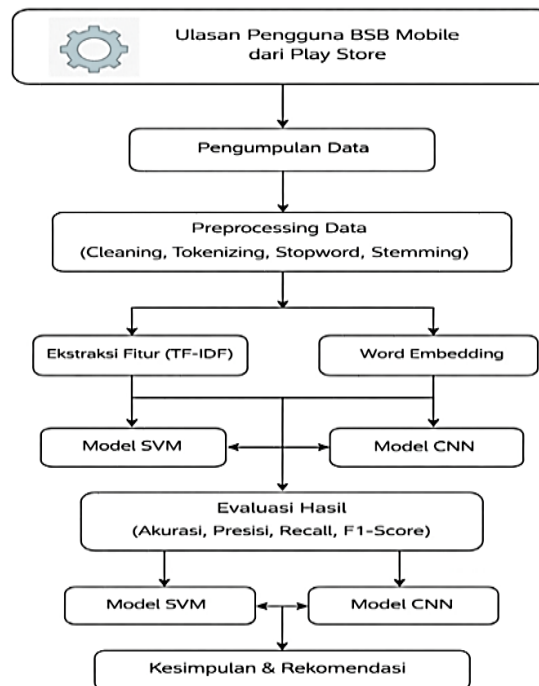
Berdasarkan temuan tersebut, dapat disimpulkan bahwa penelitian mengenai analisis sentimen pada domain *mobile banking*, khususnya aplikasi BSB *Mobile*, masih jarang dilakukan. Selain itu, sebagian besar studi sebelumnya hanya memanfaatkan satu metode klasifikasi sehingga belum memberikan pemahaman komparatif mengenai algoritma yang memiliki efikasi lebih tinggi dalam memproses rating aplikasi perbankan. Oleh karena itu, penelitian ini penting dilakukan untuk mengisi kesenjangan tersebut melalui evaluasi dua pendekatan populer dalam Pemrosesan Bahasa Alami (PBA), yaitu *Convolutional Neural Network* (CNN) dan *Support Vector Machine* (SVM) [7]. CNN digunakan karena kemampuannya dalam mengekstraksi fitur secara otomatis dan mendeteksi pola lokal antar kata, sedangkan SVM dipilih sebagai pembanding karena terbukti efektif pada data berdimensi tinggi dengan representasi TF-IDF [8]. Perbandingan kedua teknik ini bertujuan untuk mengidentifikasi model yang paling sesuai dalam menganalisis kecenderungan emosi pengguna aplikasi BSB *Mobile* [9].

Dengan demikian, tujuan utama penelitian ini adalah (1) mengklasifikasikan sentimen ulasan pengguna BSB *Mobile* secara otomatis menggunakan CNN dan SVM, (2) membandingkan performa kedua algoritma berdasarkan metrik evaluasi standar, serta (3) memberikan rekomendasi ilmiah bagi pihak bank dalam meningkatkan kualitas layanan digital. Penelitian ini diharapkan dapat berkontribusi pada perluasan kajian analisis sentimen pada sektor *mobile banking* yang masih kurang dieksplorasi serta memberikan solusi berbasis data untuk pengambilan keputusan strategis.

## 2. METODOLOGI PENELITIAN

Penelitian ini mengadopsi pendekatan kuantitatif guna mengukur dan mengklasifikasikan sentimen secara sistematis pada ulasan pengguna aplikasi BSB *Mobile* yang diambil dari Google Play Store melalui web scraping. Data diproses melalui tahap *case folding*, pembersihan teks, tokenisasi, *stopword removal*, dan stemming sebelum diberi label positif, negatif, atau netral. Pemodelan dilakukan menggunakan metode

vektor pendukung yang membentuk *hyperplane optimal Machines* sebuah algoritma pembelajaran mesin yang berbasis pada pencarian *hyperplane* optimal yang mengintegrasikan teknik pembobotan kata TF-IDF untuk ekstraksi fitur, serta arsitektur *Convolutional Neural Network* (CNN) untuk proses klasifikasi dengan *embedding*, *convolutional*, dan *dense layer*. Kedua model dievaluasi menggunakan akurasi, *precision*, *recall*, *F1-score*, dan *confusion matrix* untuk menentukan metode yang paling efektif dalam mengklasifikasikan sentimen ulasan pengguna. Kerangka penelitian ditunjukkan pada Gambar 1.



**Gambar 1.** Kerangka Pemikiran Penelitian

## 2.1 Pengambilan Data

Data riset diperoleh dari kumpulan testimoni pengguna perangkat lunak BSB *Mobile* yang diperoleh melalui sarana pengambilan data ulasan dari Google Play Store dengan memanfaatkan pustaka Google Play Scraper dan selanjutnya diarsipkan ke dalam bentuk berkas *Comma Separated Values* (CSV). Beberapa kolom penting digunakan, meliputi *score* (rating 1–5), *content* (isi ulasan), *at* (tanggal ulasan), dan *userName* (nama pengguna). Jumlah keseluruhan data awal dicatat terlebih dahulu untuk mengetahui total ulasan sebelum proses pembersihan data dilakukan.

## 2.2 Preprocessing Data

Tahap *processing* pengolahan data dilakukan guna menjamin bahwa ulasan teks yang diterapkan pada penelitian dalam keadaan telah rapi dan siap diproses menggunakan algoritma CNN dan SVM [10]. Proses ini diawali dengan *case folding*, yaitu mengubah seluruh huruf dalam teks menjadi huruf kecil agar tidak terjadi perbedaan makna akibat penggunaan huruf besar. Selanjutnya dilakukan pembersihan teks dengan menghapus komponen noninformatif yang mencakup karakter numerik serta simbol tanda baca, karakter khusus, URL, emotikon, dan spasi berlebih, sehingga teks menjadi lebih terstruktur. Tahap berikutnya adalah tokenisasi, yaitu membagi teks ulasan ke dalam unit kata yang lebih sederhana sehingga dapat diproses sebagai unit analisis. Setelah itu, dilakukan proses penghilangan kata-kata umum untuk menghapus kalimat yang sering muncul dan memberikan nilai penting bagi arti kalimat, misalnya kata “yang”, “dan”, “atau”, beserta data tekstural lainnya yang dinilai kurang relevan untuk analisis sentimen. Proses *stemming* tahap berikutnya diaplikasikan menggunakan algoritma proses reduksi kata ke bentuk dasar dalam bahasa Indonesia seperti Sastrawi guna mentransformasi setiap token menjadi kata dasar guna meminimalisir ambiguitas. Fitur berbeda tetapi memiliki bentuk asal yang sama mampu dikenali oleh model.

Usai tahapan pra-pemrosesan data tekstual selesai dilakukan dan distandardisasi, data siap digunakan untuk proses ekstraksi fitur pada SVM melalui metode TF-IDF serta diubah menjadi *embedding* untuk pemodelan CNN. Seluruh tahap *processing data* ini bertujuan meningkatkan kualitas *dataset* dan memastikan bahwa model dapat belajar dari representasi teks yang lebih informatif dan konsisten. Dengan *preprocessing* yang tepat, akurasi sistem pengklasifikasi guna membedakan polaritas sentimen dapat meningkat secara signifikan dan data perolehan penelitian menjadi lebih valid.

### 2.3 Pembobotan Fitur

Pembobotan fitur dilakukan untuk mengonversi data tekstual yang sudah melalui tahapan diproses membentuk numerik yang telah diproses menjadi bentuk angka yang bisa dibaca melalui model pembelajaran mesin (*machine learning*) [11]. Studi ini mengimplementasikan skema TF-IDF guna melakukan pembobotan berdasarkan frekuensi kemunculan kalimat pada suatu ulasan serta tingkat kelangkaannya dalam keseluruhan dokumen. Bobot tersebut menghasilkan matriks fitur yang kemudian digunakan sebagai input utama pada model SVM untuk membantu mengidentifikasi pola kata yang berpengaruh terhadap sentimen.

### 2.4 Klasifikasi Sentimen

Klasifikasi sentimen merupakan proses pengelompokan ulasan pengguna ke dalam tiga kategori, yaitu positif, negatif, dan netral [12]. Setelah melalui tahap *preprocessing* dan pembobotan fitur, dua algoritma digunakan untuk melakukan proses kategorisasi dengan mengadopsi pendekatan SVM sekaligus CNN. Algoritma ruang hiperplana (*hyperplane*) memanfaatkan representasi TF-IDF untuk menentukan batas pemisah antarkelas, sedangkan CNN mempelajari pola bahasa melalui lapisan *embedding* dan konvolusi. Hasil dari kedua model kemudian dibandingkan untuk mengetahui algoritma yang memberikan performa terbaik dalam mengidentifikasi sentimen pengguna aplikasi BSB *Mobile*.

#### 2.4.1 Convolutional Neural Network (CNN)

CNN (*Convolutional Neural Network*) adalah arsitektur salah satu metode deep learning yang awalnya dimanfaatkan dalam pengolahan citra, namun kini banyak diterapkan pada pemrosesan teks [13]. CNN bekerja dengan mengekstraksi pola lokal melalui operasi konvolusi sehingga mampu mengenali susunan kata (n-gram) yang relevan dalam penentuan sentimen. Dalam penelitian ini, CNN digunakan untuk mempelajari struktur dan hubungan antar kata pada ulasan pengguna BSB *Mobile*. Proses pemodelannya meliputi beberapa tahapan, yaitu:

1. *Embedding*, di mana setiap kata direpresentasikan sebagai vektor berdimensi tertentu;
2. Convolution, yang menerapkan filter untuk menghasilkan *feature map* berdasarkan pola kata;
3. Max-pooling, yang memilih nilai fitur paling dominan untuk mereduksi dimensi;
4. Fully connected layer dan softmax mengubah hasil pooling menjadi probabilitas untuk setiap kelas sentimen.

#### 2.4.2 Support Vector Machine (SVM)

*Support Vector Machine* (SVM) adalah algoritma klasifikasi yang bertujuan menentukan *hyperplane* terbaik untuk memisahkan data pada ruang berdimensi tinggi [14]. Pendekatan ini secara luas diimplementasikan pada klasifikasi teks berkat kapasitasnya dalam mengolah *dataset* yang bersifat sparse serta memiliki performa yang konsisten pada *dataset* berukuran kecil hingga menengah.

Secara prinsip, SVM bekerja dengan mencari batas pemisah yang memberikan jarak maksimum antarkelas. Model menghitung kombinasi antara bobot dan fitur pada setiap data, kemudian menambahkan nilai bias untuk menentukan posisi data terhadap *hyperplane*. Nilai positif menunjukkan bahwa data berada pada satu kelas, sedangkan nilai negatif menunjukkan kelas yang lain. Untuk kasus di mana *dataset* memiliki karakteristik ketidakterpisahan secara linear, algoritma SVM mengimplementasikan transformasi kernel guna memproyeksikan *dataset* menuju ruang fitur berdimensi tinggi (*high-dimensional space*), yang memungkinkan proses pemisahan kelas proses partisi data menjadi lebih efisien serta lebih optimal [15].

### 2.5 Pengujian Performa Model

Pengujian performa model digunakan untuk mengevaluasi efektivitas metode perhitungan CNN dan SVM untuk mengklasifikasikan sentimen ulasan pengguna. Evaluasi ini menggunakan sejumlah metrik yang umum digunakan pada tugas klasifikasi, yaitu *Accuracy*, *Precision*, *Recall*, dan *F1-Score*, yang sudah luas digunakan dalam riset tentang analisis sentimen sebelumnya [16]. Seluruh metrik tersebut dihitung berdasarkan evaluasi kinerja model dipetakan melalui empat parameter fundamental dalam *confusion matrix*, yakni *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), serta *False Negative* (FN) [17].

*Accuracy* digunakan untuk mengukur proporsi prediksi model yang benar terhadap keseluruhan data uji. *Precision* menilai ketepatan model dalam memprediksi kelas tertentu dengan membandingkan nilai TP terhadap FP. *Recall* mengukur kemampuan model dalam mengenali seluruh sampel yang benar-benar relevan pada suatu kelas dengan melihat perbandingan TP terhadap FN. Sementara itu, *F1-Score* merupakan harmonisasi keseimbangan harmonik antara nilai presisi (*precision*) serta perolehan (*Recall*) yang sangat efektif terutama ketika kondisi data menunjukkan adanya asimetri distribusi label, sehingga dapat memberikan evaluasi yang lebih stabil dibandingkan hanya mengandalkan satu metrik saja [18].

Hubungan antara keempat metrik tersebut divisualisasikan melalui *confusion matrix* yang menunjukkan distribusi prediksi benar dan salah untuk tiap kelas. Dengan menggunakan kombinasi metrik ini, performa CNN dan SVM dapat dianalisis secara menyeluruh sehingga menyajikan pemahaman yang

lebih menyeluruh mengenai kemampuan kedua model untuk melakukan klasifikasi sentimen ulasan pengguna aplikasi BSB *Mobile* [3].

**3. HASIL DAN PEMBAHASAN**

**3.1 Tahap *Preprocessing* Informasi**

Langkah *preprocessing* pengolahan data dilakukan guna membersihkan dan mempersiapkan data teks ulasan sebelum masuk ke proses pemodelan CNN dan SVM. Tahapan awal dari proses ini diimplementasikan melalui *case folding*, sebuah teknik yang bertujuan untuk mengubah semua huruf menjadi huruf kecil guna menjamin standardisasi karakter format teks yang lebih seragam. Selanjutnya dilakukan pembersihan teks dengan menghapus angka, tanda baca, emotikon, URL, serta karakter yang tidak relevan. Setelah itu, teks dipecah menjadi kata-kata melalui proses tokenisasi. Proses berikutnya merupakan tahapan *stopword removal*, yakni menyingkirkan himpunan kata-kata fungsional yang kurang berpengaruh besar terhadap substansi informasi kalimat, misalnya kata “yaitu”, dan, “dan”. Tahapan terakhir yaitu *stemming* menggunakan *library* Sastrawi untuk mengembalikan kata ke bentuk dasar sehingga keberagaman leksikal yang memiliki esensi semantik serupa mampu diproses secara koheren. Seluruh tahapan *preprocessing* ini bertujuan menghasilkan teks yang bersih, terstruktur, dan siap digunakan untuk proses pembobotan fitur maupun pelatihan model.

**Tabel 1.** Tahapan *Preprocessing* Data

| Tahapan Pra-pemrosesan Data               | Penjelasan                                                                                                                                                                                |
|-------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Penyeragaman Format Huruf                 | Melakukan penyeragaman seluruh karakter teks ke dalam bentuk huruf kecil (lowercasing) agar format karakter seragam.                                                                      |
| <i>Cleaning Text</i>                      | Membuang bagian-bagian yang tidak diperlukan misalnya berupa angka maupun simbol penulisan, URL, ikon ekspresi, dan karakter konten tulisan lain yang tidak memberikan informasi penting. |
| <i>Tokenizing</i>                         | Menguraikan kalimat ulasan ke dalam bagian-bagian kata (token) agar dapat diproses pada tahap berikutnya.                                                                                 |
| Penyingkiran kata-kata yang sering muncul | Membuang istilah-istilah yang sering muncul ( <i>stopwords</i> ) yang bukan berpengaruh signifikan terhadap arti sentimen, contohnya “dan”, “atau”, serta”, dan “tersebut.                |
| Proses <i>Stemming</i>                    | Mengonversi kata menjadi bentuk akar menggunakan <i>library</i> Sastrawi (contoh: “membeli” → “beli”).                                                                                    |

Tahapan *preprocessing* pada Tabel 1 dirancang untuk memastikan bahwa teks ulasan berada dalam kondisi yang siap diolah oleh algoritma klasifikasi. Proses *case folding* menjadi tahap awal untuk menyeragamkan format karakter sehingga variasi huruf besar dan kecil tidak mengganggu proses analisis. Selanjutnya, proses *cleaning* membantu menghilangkan elemen yang tidak relevan dan berpotensi menimbulkan noise, terutama pada data ulasan yang biasanya mengandung emotikon, tanda baca berlebihan, dan URL. Tahap *tokenizing* memungkinkan teks diproses sebagai kumpulan kata yang bisa diproses secara terpisah. Setelah itu, proses eliminasi *stopword* digunakan untuk mengurangi kata-kata yang tidak memiliki kontribusi signifikan terhadap penentuan sentimen, sehingga model lebih fokus pada kata-kata bermakna. Terakhir, proses *stemming* menstandarkan kata ke bentuk dasarnya, yang penting untuk menghindari duplikasi fitur (misalnya “membeli”, “pembelian”, dan “dibeli” dipetakan menjadi “beli”).

**3.1.1 Proses *Cleaning***

Proses *cleaning* dilakukan untuk meningkatkan kualitas teks ulasan dengan menghilangkan berbagai elemen yang dianggap sebagai noise dan tidak memiliki nilai informasi. Tahap ini mencakup penghapusan simbol tipografi, karakter numerik, emoji, entitas nonleksikal, serta tautan eksternal yang kerap muncul pada ulasan pengguna. Selain itu, perbaikan format teks turut dilakukan dengan menghapus spasi ganda dan karakter berulang agar struktur kalimat menjadi lebih rapi dan konsisten. Penerapan *cleaning* membantu model dalam mengenali pola linguistik dengan lebih efektif, mengurangi gangguan dari komponen yang tidak relevan, serta meningkatkan akurasi proses klasifikasi sentimen. Ilustrasi mengenai luaran tahapan pembersihan data tersaji dalam Tabel 2.

**Tabel 2.** Representasi Luaran Pembersihan Data

| Tabel <i>Cleaning</i>           | Penjelasan                                                                             |
|---------------------------------|----------------------------------------------------------------------------------------|
| Pembersihan Tanda Baca & Simbol | Membuang tanda baca, emoji, lalu karakter khusus yang tidak mempunyai nilai informasi. |
| Penghapusan Angka & URL         | Membersihkan angka, tautan eksternal, dan elemen digital lain yang tidak relevan.      |
| Perapian Spasi                  | Menghapus spasi ganda, karakter berulang, dan merapikan struktur teks.                 |

Berdasarkan Tabel 2, proses *cleaning* diterapkan guna menjamin integritas data ulasan berada pada kondisi yang reliabel dan seragam guna meningkatkan akurasi klasifikasi sebelum memasuki tahap analisis berikutnya. Tahap pertama adalah penghapusan tanda baca, simbol, dan emoji yang sering muncul pada teks informal seperti ulasan aplikasi, namun tidak memberikan kontribusi terhadap representasi sentimen. Selanjutnya, angka dan URL dihilangkan karena cenderung tidak berhubungan dengan informasi emosional atau opini pengguna.

Tahap terakhir adalah perapian spasi, yang bertujuan memperbaiki struktur teks dengan menghapus spasi ganda dan karakter berulang sehingga format kalimat menjadi lebih teratur. Rangkaian proses ini membantu mengurangi noise dalam data dan memastikan bahwa fitur linguistik yang relevan dapat diproses secara optimal oleh algoritma SVM dan CNN. Dengan demikian, hasil *preprocessing* yang bersih berkontribusi langsung pada peningkatan kualitas ekstraksi fitur serta akurasi model klasifikasi sentimen.

### 3.1.2 Proses Case folding

*Case folding* merupakan tahap awal dalam *preprocessing* yang bertujuan mengonversi seluruh teks pada ulasan menjadi huruf kecil (*lowercase*). Langkah ini dilakukan untuk memastikan bahwa kata yang sama namun ditulis dengan variasi kapitalisasi seperti “Bagus”, “bagus”, atau “BAGUS” dipandang sebagai entitas yang identik. Dengan menyeragamkan seluruh huruf menjadi *lowercase*, variasi kata yang tidak diperlukan dapat dikurangi sehingga memudahkan model dalam proses pembelajaran. Contoh penerapan *case folding* dan normalisasi ditampilkan pada Tabel 3.

**Tabel 3.** Contoh Proses *Case folding*

| Tahap <i>Case folding</i>           | Penjelasan                                                                                                      | Contoh Sebelum       | Contoh Sesudah       |
|-------------------------------------|-----------------------------------------------------------------------------------------------------------------|----------------------|----------------------|
| Konversi Huruf Besar ke Huruf Kecil | Menyeragamkan seluruh teks menjadi <i>lowercase</i> agar tidak ada perbedaan makna akibat variasi kapitalisasi. | “Aplikasi Ini Bagus” | “aplikasi ini bagus” |
| Normalisasi Kapitalisasi Berlebih   | Menghapus penggunaan huruf besar yang tidak konsisten di tengah kata atau kalimat.                              | “BaGuS BaNgEt”       | “bagus banget”       |
| Standardisasi Penulisan             | Menetapkan format teks menjadi <i>lowercase</i> secara menyeluruh untuk memudahkan pemrosesan model.            | “BSB MOBILE ERROR”   | “bsb Mobile error”   |

Tabel 3 menunjukkan bahwa *case folding* berfungsi untuk menyeragamkan bentuk teks sebelum diproses oleh algoritma klasifikasi. Tahap pertama, yaitu konversi seluruh huruf menjadi *lowercase*, penting untuk menghindari perlakuan berbeda terhadap kata yang secara semantik sama tetapi ditulis dalam format kapitalisasi yang berbeda. Normalisasi kapital berlebih juga diperlukan karena ulasan pengguna sering kali menggunakan huruf besar secara acak, yang dapat menyebabkan variasi kata yang tidak relevan bagi model.

Standardisasi penulisan secara keseluruhan membantu menstabilkan representasi teks sehingga fitur yang diekstraksi menjadi lebih konsisten. Tahapan ini memastikan bahwa model CNN dan SVM dapat mempelajari pola linguistik tanpa terganggu oleh perbedaan penulisan yang tidak memiliki makna informatif.

### 3.1.3 Tokenizing, Stopword Removal, and Stemming

Tahapan *preprocessing* bertujuan menyiapkan teks ulasan agar berada dalam kondisi yang bersih dan konsisten sebelum diproses oleh model. Setelah melalui *case folding* dan pembersihan teks untuk menghilangkan simbol, angka, URL, emotikon, dan spasi berlebih, proses dilanjutkan dengan tokenizing, yakni menguraikan kalimat ke dalam beberapa token kata tunggal. Tahap berikutnya merupakan proses penghapusan *stopword*, yang bertujuan menyingkirkan kata-kata sering muncul yang tidak memberikan nilai penting pada analisis semantik. Selanjutnya, stemming dilakukan menggunakan *library* Sastrawi guna mentransformasi setiap kalimat ke awal. Rangkaian proses menjamin bahwa *dataset* menunjukkan pola organisasi yang lebih rapi serta seragam sehingga siap diproses lebih lanjut melalui tahapan pembobotan fitur dan klasifikasi sentimen. Contoh *preprocessing* dapat dilihat pada Tabel 4.

## 3.2 Hasil Pembobotan Menggunakan TF-IDF

Pembobotan fitur menggunakan TF-IDF digunakan untuk mengonversi teks ulasan menjadi bobot numerik yang menunjukkan tingkat kepentingan tiap kata dalam dokumen. Perhitungan TF-IDF mempertimbangkan frekuensi kemunculan kata pada sebuah ulasan serta tingkat kelangkaannya pada seluruh dokumen. Kata yang jarang muncul tetapi relevan cenderung memiliki bobot lebih tinggi, sedangkan kata yang muncul merata pada semua ulasan akan memiliki bobot lebih rendah. Matriks TF-IDF yang dihasilkan selanjutnya digunakan sebagai input bagi model SVM untuk meningkatkan akurasi dalam proses klasifikasi sentimen.

**Tabel 4.** Tokenizing, Stopword removal, dan Stemming

| Tahap Preprocessing       | Penjelasan                                                                                     | Contoh Sebelum                         | Contoh Sesudah                         |
|---------------------------|------------------------------------------------------------------------------------------------|----------------------------------------|----------------------------------------|
| Penyeragaman format huruf | Menyeragamkan seluruh karakter ke dalam format huruf kecil.                                    | “Aplikasi Ini Bagus”                   | “aplikasi ini bagus”                   |
| Cleaning                  | Menghapus tanda baca, angka, URL, simbol, serta spasi ganda.                                   | “Bagus!! cek di https://bsb.co.id”     | “bagus cek di”                         |
| Tokenizing                | Memecah kalimat menjadi daftar token.                                                          | “Aplikasi ini sangat bagus”            | [“aplikasi”, “ini”, “sangat”, “bagus”] |
| Eliminasi kata-kata umum  | Membuang istilah-istilah yang bersifat umum yang memengaruhi arti.                             | [“aplikasi”, “ini”, “sangat”, “bagus”] | [“aplikasi”, “bagus”]                  |
| Stemming                  | Melakukan proses stemming untuk mengembalikan kata ke bentuk dasarnya dengan bantuan Sastrawi. | “pembelian”, “membeli”, “dibeli”       | “beli”                                 |

**Tabel 5.** Contoh Istilah dengan Bobot TF-IDF Tertinggi

| Term    | Nilai TF- IDF | Keterangan                                           |
|---------|---------------|------------------------------------------------------|
| error   | 0.412         | Dominan pada ulasan negatif terkait kendala aplikasi |
| bagus   | 0.387         | Banyak muncul pada ulasan positif                    |
| loading | 0.355         | Muncul pada keluhan performa aplikasi                |
| cepat   | 0.331         | Umum pada ulasan positif terkait kecepatan layanan   |
| gagal   | 0.298         | Umum pada ulasan negatif terkait transaksi           |

Tabel 5 menunjukkan kata-kata dengan bobot TF-IDF tertinggi, yang dianggap paling informatif dalam membedakan sentimen ulasan. Kata seperti “error” dan “gagal” memiliki bobot tinggi karena sering muncul pada ulasan yang berisi keluhan, sehingga menjadi indikator kuat untuk sentimen negatif. Sebaliknya, kata seperti “bagus” dan “cepat” lebih dominan pada ulasan positif. Adapun istilah seperti “loading” menunjukkan keluhan spesifik terkait performa aplikasi. Pola bobot ini membantu model SVM mengenali kata-kata kunci yang berpengaruh dalam proses klasifikasi sentimen.

### 3.3 Pelabelan Data

Pelabelan data diimplementasikan melalui kategorisasi setiap poin ulasan ke dalam tiga kelas label, yakni positif, negatif, serta netral, yang ditentukan menurut isi dan arti kalimat. Proses ini memastikan bahwa model CNN dan SVM dapat belajar dari *dataset* yang telah teranotasi sesuai polaritas opini pengguna. Representasi hasil anotasi data disajikan secara mendalam pada Tabel 6.

**Tabel 6.** Representasi Hasil Anotasi Data

| No | Sentimen | Ringkasan Ulasan           |
|----|----------|----------------------------|
| 1  | Positif  | bagus                      |
| 2  | Negatif  | aplikasi tidak bisa dibuka |
| 3  | Netral   | Selalu disuruh update      |

### 3.4 Algoritma Klasifikasi Berbasis Hyperplane

Kategorisasi polaritas opini menggunakan SVM diimplementasikan melalui pemanfaatan hasil pembobotan TF-IDF sebagai input untuk memisahkan dan mengelompokkan ulasan ke dalam tiga spektrum polaritas, yakni positif, negatif, serta netral. SVM bekerja serta mencari batas pemisah terbaik berdasarkan pola kata dalam ulasan sehingga setiap data dapat diklasifikasikan sesuai sentimennya secara optimal.

**Tabel 7.** Classification SVM

| Kelas        | Precision | Recall | F1-Score | Support |
|--------------|-----------|--------|----------|---------|
| Negatif      | 0.79      | 0.88   | 0.83     | 440     |
| Netral       | 0.19      | 0.10   | 0.13     | 59      |
| Positif      | 0.88      | 0.82   | 0.85     | 390     |
| Accuracy     | -         | -      | 0.80     | 889     |
| Macro Avg    | 0.62      | 0.60   | 0.61     | 889     |
| Weighted Avg | 0.79      | 0.80   | 0.79     | 889     |

Berdasarkan hasil evaluasi pada Tabel 7, model SVM memperoleh akurasi sebesar 0.80 dari total 889 data uji. Nilai ini menunjukkan bahwa model mampu mengklasifikasikan sekitar 80% ulasan dengan benar. Pada kelas negatif, performa model tergolong baik mencatatkan perolehan nilai *precision* sebesar 0.79,

sensitivitas 0.88, serta skor F1 0.83, mengindikasikan bahwasanya SVM cukup efektif di dalam mengenali ulasan bernada negatif dan jarang melakukan kesalahan prediksi.

Kinerja pada kelas positif juga menunjukkan hasil yang kuat, dengan *precision* 0.88, *recall* 0.82, serta F1-Score 0.85. Hal ini berarti model mampu mengidentifikasi sebagian besar ulasan positif secara konsisten dan akurat. Namun demikian, performa pada kelas netral masih rendah, ditunjukkan oleh nilai F1-Score sebesar 0.13 yang disebabkan oleh jumlah data netral yang jauh lebih sedikit (hanya 59 sampel). Ketidakseimbangan ini membuat model sulit mempelajari pola yang merepresentasikan sentimen netral secara optimal. Secara keseluruhan, nilai weighted average F1-Score sebesar 0.79 menunjukkan bahwa SVM bekerja cukup baik dalam tugas klasifikasi sentimen, terutama pada kelas positif dan negatif yang mendominasi *dataset*.

### 3.4.1 Convolutional Neural Networks (CNN)

Pada penelitian ini, CNN diterapkan sebagai salah satu model klasifikasi sentimen untuk mengidentifikasi opini pengguna aplikasi BSB *Mobile* secara otomatis. Arsitektur CNN yang digunakan terdiri dari beberapa komponen utama, yaitu penggunaan *embedding layer* guna mengonversi setiap leksikon menjadi representasi vektor numerik, Convolutional Layer dengan sejumlah filter untuk menangkap pola lokal seperti *n-gram*, *Pooling Layer* untuk mereduksi dimensi fitur, serta *Dense Layer* dan *Dropout* yang berfungsi mengurangi risiko overfitting. Walaupun CNN memiliki kelebihan dalam mengekstraksi fitur secara otomatis tanpa memerlukan proses pembobotan manual seperti TF-IDF, hasil pengujian menunjukkan bahwa performanya pada *dataset* ini masih berada di bawah SVM. Salah satu faktor penyebabnya adalah karakteristik ulasan yang cenderung singkat sehingga memerlukan jumlah data yang jauh lebih besar agar model mampu melakukan generalisasi dengan baik.

Penerapan CNN pada penelitian ini menunjukkan bahwa jaringan saraf tiruan mampu mempelajari hubungan semantik antar kata dalam ulasan layanan perbankan digital. Namun, tingkat efektivitas model sangat dipengaruhi oleh konfigurasi hiperparameter, seperti jumlah *epoch*, ukuran batch, dan penggunaan *early stopping* untuk mencegah *overfitting* dan memastikan model berhenti pada performa terbaik. Secara keseluruhan, CNN berfungsi sebagai model pembanding yang memberikan gambaran tambahan mengenai kompleksitas representasi data teks ulasan pengguna Google Play Store dalam mengukur tingkat kepuasan terhadap aplikasi BSB *Mobile*. Tabel 8 merupakan hasil evaluasi kinerja model CNN.

**Tabel 8.** Hasil Evaluasi Kinerja Model CNN

| Kelas Sentimen       | <i>Precision</i> | <i>Recall</i> | F1-Score | Jumlah Sampel |
|----------------------|------------------|---------------|----------|---------------|
| Negatif              | 0.80             | 0.87          | 0.83     | 438           |
| Netral               | 0.29             | 0.11          | 0.16     | 62            |
| Positif              | 0.87             | 0.86          | 0.87     | 481           |
| Akurasi Keseluruhan  | -                | -             | 0.82     | 981           |
| Rata-rata Makro      | 0.65             | 0.62          | 0.62     | 981           |
| Rata-rata Tertimbang | 0.80             | 0.82          | 0.81     | 981           |

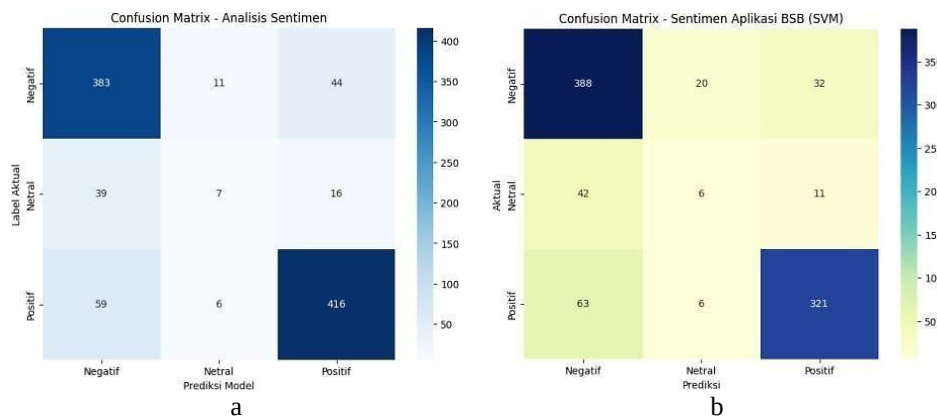
## 3.5 Analisis Kinerja Model

### 3.5.1 Evaluasi Confusion Matrix

Visualisasi confusion matrix mengindikasikan bahwa model SVM memperoleh akurasi sebesar 80,70% melalui performa yang stabil, terutama dalam mengenali ulasan dengan sentimen positif, yang terlihat dari prediksi benar pada diagonal utama. Penerapan parameter *class\_weight* = '*balanced*' membantu mengurangi kesalahan prediksi pada kelas netral yang cenderung memiliki pola kata yang kurang jelas. Secara keseluruhan, hasil ini memperlihatkan bahwa SVM mampu memetakan sentimen ulasan aplikasi BSB *Mobile* dengan lebih konsisten dibandingkan dengan CNN.

Gambar 2 menampilkan perbandingan Confusion Matrix antara model CNN dan SVM. Berdasarkan visualisasi tersebut, terlihat bahwa model SVM dengan kernel linear dan representasi fitur TF-IDF menunjukkan performa paling efektif dalam mengklasifikasikan sentimen ulasan BSB *Mobile*. Nilai prediksi yang terkonsentrasi pada diagonal utama merefleksikan tingkat presisi yang tinggi, menandakan bahwa sebagian besar ulasan berhasil dipetakan sesuai dengan label sebenarnya. Peningkatan akurasi hingga 80,70% menunjukkan bahwa penerapan *class weight* untuk menangani ketidakseimbangan data berhasil memperbaiki batas keputusan model, sehingga menghasilkan klasifikasi yang lebih stabil dibandingkan dengan model pembanding lainnya.

Secara keseluruhan, *Confusion Matrix* menegaskan bahwa performa SVM sudah berada pada tingkat yang layak untuk menganalisis persepsi pengguna. Meski masih terdapat kesalahan pada kelas netral akibat keterbatasan konteks linguistik dalam ulasan singkat, model tetap mampu memberikan gambaran sentimen yang akurat dan konsisten terhadap ulasan pengguna di Google Play Store.



Gambar 2. (a) Confusion Matrix CNN (b) Confusion Matrix SVM

### 3.5.2 Evaluasi Menggunakan Parameter Evaluasi Kinerja Model

Evaluasi performa model dalam mengklasifikasikan sentimen pada *dataset* ulasan BSB *Mobile* dilakukan menggunakan empat parameter utama, yaitu *Accuracy*, *Precision*, *Recall*, dan *F1-score*. Akurasi digunakan untuk mengukur proporsi ulasan yang berhasil diprediksi dengan benar oleh model, dan dalam penelitian ini nilai akurasi mencapai 80,70%. Namun, mengingat *dataset* memiliki distribusi yang tidak seimbang antarkelas sentimen, metrik *precision* menjadi penting karena mampu menunjukkan ketepatan model dalam menghasilkan prediksi yang benar dan meminimalkan terjadinya false positive.

Sementara itu, *recall* digunakan untuk mengevaluasi sejauh mana model mampu mengenali seluruh sampel yang benar-benar termasuk ke dalam suatu kelas. Hal ini khususnya penting pada kelas negatif, karena keluhan pengguna perlu teridentifikasi dengan baik untuk kepentingan analisis layanan. Sebagai pelengkap, *F1-score* digunakan untuk menilai keseimbangan antara *precision* dan *Recall*. Metrik ini memberikan gambaran yang lebih stabil terhadap performa model, terutama ketika distribusi data antarkelas tidak seimbang.

Hasil diskusi ini adalah bagian yang menyajikan interpretasi mendalam terhadap hasil evaluasi. Kinerja SVM dan CNN menunjukkan bahwa akurasi tinggi yang dicapai SVM terutama dipengaruhi oleh kemampuan TF-IDF dalam merepresentasikan fitur penting pada teks pendek. Hal ini sejalan dengan temuan penelitian sebelumnya yang menyebutkan bahwa SVM cenderung unggul pada *dataset* ulasan berukuran kecil hingga menengah karena margin maximization memungkinkan model bekerja stabil meskipun informasi kontekstual terbatas. Sebaliknya, CNN membutuhkan jumlah data yang lebih besar agar dapat belajar pola semantik secara optimal, sehingga performanya menjadi kurang kompetitif pada *dataset* ulasan pendek seperti yang digunakan dalam studi ini. Ketidakseimbangan data juga berdampak pada penurunan kinerja pada kelas netral. Studi terdahulu dalam analisis sentimen menyatakan bahwa kelas netral sering menjadi tantangan karena pola leksikalnya tidak sejelas kelas positif dan negatif. Hal ini turut tercermin pada hasil *Recall* dan *F1-score* yang rendah pada kelas tersebut. Penggunaan *class weight* pada SVM terbukti membantu meningkatkan sensitivitas model, namun upaya ini belum sepenuhnya mengatasi keterbatasan pola linguistik yang minim pada teks netral.

Secara praktis, temuan ini mengindikasikan bahwa algoritma berbasis pembobotan fitur seperti SVM lebih efektif digunakan untuk konteks analisis ulasan aplikasi perbankan digital yang didominasi oleh teks pendek. Namun, CNN tetap relevan sebagai model pembanding karena memberikan wawasan mengenai kemampuan jaringan saraf dalam menangkap pola semantik yang lebih kompleks. Implikasi penelitian ini dapat dijadikan dasar untuk pengembangan sistem analisis sentimen pada layanan *mobile banking*, terutama dengan menambah jumlah *dataset*, melakukan balancing data, atau mengadopsi arsitektur deep learning yang lebih adaptif seperti Bi-LSTM atau Transformer pada penelitian berikut.

## 4. KESIMPULAN

Penelitian ini berhasil mengimplementasikan pendekatan *Convolutional Neural Networks* (CNN) serta *Support Vector Machine* untuk mengklasifikasikan polaritas opini pengguna pada platform aplikasi BSB layanan perbankan digital tersebut melalui platform Google Play Store. Temuan penelitian eksperimen mengindikasikan bahwa kedua metode komputasi memiliki kinerja model yang sangat baik dalam mengenali pola opini masyarakat, namun terdapat perbedaan signifikan pada tingkat akurasi akhir. Metode SVM dengan kernel linear dan fitur TF-IDF mampu mencapai akurasi sebesar 80,70%, mengungguli metode CNN yang menghasilkan akurasi sebesar 80,10%. Keunggulan SVM pada *dataset* ini dipengaruhi oleh karakteristik ulasan teks pendek yang lebih optimal dipetakan melalui ekstraksi fitur statistik dibandingkan dengan pendekatan deep learning yang membutuhkan skala data lebih besar untuk proses generalisasi. Penggunaan parameter *class weight* terbukti efektif dalam menangani ketidakseimbangan data sehingga nilai metrik

*precision*, *recall*, dan *F1-score* tetap stabil dalam kelas positif, negatif, dan netral. Secara komprehensif, penelitian tersebut menyimpulkan bahwa capaian algoritma metode SVM didefinisikan sebagai model yang lebih baik, konsisten, serta andal untuk digunakan sebagai alat analisis kepuasan pelanggan pada aplikasi perbankan digital.

## REFERENSI

- [1] M. R. Sanjaya *et al.*, “Rancang Bangun Aplikasi Pengenalan Huruf Hijaiyah Dan 10 Surat Pendek Alqurâ€™an Pendekatan UML,” *INTECOMS: Journal of Information Technology and Computer Science*, vol. 5, no. 1, 2022, doi: 10.31539/intecom.v5i1.3463.
- [2] A. I. Faiq, M. R. S. Sanjaya, Dwi Rosa Indah, and Endang Lestari Ruskan, “Hybrid Fine-Tuning Indobert Dan Hybrid Fine-Tuning Indobert Dan Ensemble Tf-Idf Logistic Regression Untuk Analisis Sentimen Ulasan Aplikasi Zalora,” *Rabit : Jurnal Teknologi dan Sistem Informasi Univrab*, vol. 11, no. 1, 2026, doi: 10.36341/rabit.v11i1.6924.
- [3] Jasmalizal, Junadhi, Rahmadden, and M. Khairul Anam, “Penerapan Metode Support Vector Machine Untuk Analisis Sentimen Terhadap Produk Skincare,” *Indonesian Journal of Computer Science*, vol. 13, no. 1, 2024, doi: 10.33022/ijcs.v13i1.3654.
- [4] A. Muzaki, V. Febriana, and W. N. Cholifah, “Analisis Sentimen Pada Ulasan Produk di E-commerce dengan Metode Naive Bayes,” *Jurnal Riset dan Aplikasi Mahasiswa Informatika (JRAMI)*, vol. 5, no. 4, 2024, doi: 10.30998/jrami.v5i4.9647.
- [5] R. Ramdani and R. Cahyana, “Analisis Sentimen Ulasan Wisata Budaya Menggunakan Metode Support Vector Machine dan Long Short-Term Memory,” *Jurnal Algoritma*, vol. 22, no. 2, 2025, doi: 10.33364/algoritma/v.22-2.2585.
- [6] H. Hairani, A. Anggrawan, M. R. Akbar, K. Hidjah, and M. Innuddin, “Optimizing Sentiment Analysis for Lombok Tourism Using SMOTE and Chi-Square with Machine learning,” *Jurnal RESTI*, vol. 9, no. 4, 2025, doi: 10.29207/resti.v9i4.6623.
- [7] Y. U. Ladopurab and Y. R. Kaesmetan, “Analisis Perbandingan Convolutional Neural Network Dan Support Vector Machine Pada Analisis Sentimen Ruu Pilkada Di Aplikasi X,” *Journal of Information System, Applied, Management, Accounting and Research*, vol. 9, no. 2, 2025, doi: 10.52362/jisamar.v9i2.1887.
- [8] V. B. Lestari and C. A. Hutagalung, “Evaluation of TF-IDF Extraction Techniques in Sentiment Analysis of Indonesian-Language Marketplaces Using SVM, Logistic Regression, and Naive Bayes,” *Journal of Computer Science and Applications*, vol. 8, no. 1, 2025.
- [9] Muh Zainal Altim, Faisal, Salmiah, Kasman, Andi Yudhistira, And Rita Amalia Syamsu, “Pengklasifikasi Beras Menggunakan Metode Cnn (Convolutional Neural Network),” *Jurnal INSTEK (Informatika Sains dan Teknologi)*, vol. 7, no. 1, 2022, doi: 10.24252/instek.v7i1.28922.
- [10] M. F. Naufal, “Analisis Perbandingan Algoritma SVM, KNN, dan CNN untuk Klasifikasi Citra Cuaca,” *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 8, no. 2, 2021, doi: 10.25126/jtiik.2021824553.
- [11] A. D. M. Putri, N. Sulistianingsih, and R. Rismayati, “Pengaruh Teknik Representasi Teks Bag-of-Words dan TF-IDF terhadap Akurasi Klasifikasi Sentimen Teks Multi-Domain,” *JTIM : Jurnal Teknologi Informasi dan Multimedia*, vol. 7, no. 4, 2025, doi: 10.35746/jtim.v7i4.756.
- [12] A. Nurian, M. S. Ma’arif, I. N. Amalia, and C. Rozikin, “Analisis Sentimen Pengguna Aplikasi Shopee Pada Situs Google Play Menggunakan Naive Bayes Classifier,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 1, 2024, doi: 10.23960/jitet.v12i1.3631.
- [13] F. H. Hawari, F. Fadillah, M. R. Alviandi, and T. Arifin, “Klasifikasi Penyakit Tanaman Padi Menggunakan Algoritma Cnn (Convolutional Neural Network),” *Jurnal Responsif: Riset Sains dan Informatika*, vol. 4, no. 2, 2022, doi: 10.51977/jti.v4i2.856.
- [14] A. Saini, “Guide on Support Vector Machine (SVM) Algorithm,” *Analytics Vidhya*, 2024.
- [15] Siti Sarah Sobariah Lestari, Gina Purnama Insany, Dede Sukmawan, and Faiz Dzulfikar Yusuf, “Mengatasi Permasalahan High Dimensional Space dalam Klasifikasi Multikelas Big Data pada Data Gambar dengan DCSVM,” *Jurnal RESTIKOM: Riset Teknik Informatika dan Komputer*, vol. 5, no. 3, 2023, doi: 10.52005/restikom.v5i3.259.
- [16] F. Satria and Edhy Poerwandono, “Implementasi Yolov8 Untuk Deteksi Dan Klasifikasi Tingkat Kematangan Buah Mangga Berdasarkan Citra Digital,” *STORAGE: Jurnal Ilmiah Teknik dan Ilmu Komputer*, vol. 4, no. 4, 2025, doi: 10.55123/storage.v4i4.6381.
- [17] Amrin Amrin, Omar Pahlevi, and Sismadi Sismadi, “Optimasi Algoritma Random Forest Dengan Fitur Seleksi Backward Elimination Untuk Penilaian Kelayakan Kredit,” *Jurnal Ticom: Technology of Information and Communication*, vol. 13, no. 3, 2025, doi: 10.70309/ticom.v13i3.151.
- [18] St. F. Fattah and Purnawansyah, “Analisis sentimen terhadap Body Shaming pada Twitter menggunakan Metode Naïve Bayes Classifier,” *Indonesian Journal of Data and Science*, vol. 3, no. 2, 2022, doi: 10.56705/ijodas.v3i2.46.