

## **Comparative Study of SVM and Decision Tree Algorithms on the Effect of SMOTE Technique on LinkAja Application**

**Muhammad Kholfan Faruq<sup>1\*</sup>, Khothibul Umam<sup>2\*</sup>, Mokhamad Iklil Mustofa<sup>3\*</sup>, Adzhal Arwani Mahfudh<sup>4\*</sup>**

\* Teknologi Informasi, Fakultas Sains dan Teknologi, UIN Walisongo Semarang

[2208096102@student.walisongo.ac.id](mailto:2208096102@student.walisongo.ac.id)<sup>1</sup>, [khothibul\\_umam@walisongo.ac.id](mailto:khothibul_umam@walisongo.ac.id)<sup>2</sup>, [iklil@walisongo.ac.id](mailto:iklil@walisongo.ac.id)<sup>3</sup>, [adzhal@walisongo.ac.id](mailto:adzhal@walisongo.ac.id)<sup>4</sup>

---

### **Article Info**

#### **Article history:**

Received 2025-06-13

Revised 2025-10-16

Accepted 2025-11-05

---

#### **Keyword:**

*Decision Tree,  
LinkAja,  
Sentiment Analysis,  
SMOTE,  
Support Vector Machine.*

---

### **ABSTRACT**

The widespread adoption of digital wallets like LinkAja in Indonesia has led to a surge in user-generated reviews, which are valuable for assessing service quality. This study compares the classification performance of Support Vector Machine (SVM) and Decision Tree algorithms on user reviews from the LinkAja application. 7.000 reviews were gathered through web scraping and processed with standard text cleaning, tokenization, stopword removal, and stemming, resulting in 6,261 usable entries. These were divided into training and testing sets in a 70:30 ratio. The performance of each algorithm was evaluated both before and after the application of Synthetic Minority Oversampling Technique (SMOTE) to address class imbalance. Prior to SMOTE, SVM recorded an accuracy of 77.97%, precision of 0.74, recall of 0.33, and F1 score of 0.45, while Decision Tree reached 72.01% accuracy, 0.50 precision, 0.62 recall, and 0.55 F1 score. After SMOTE, SVM accuracy slightly improved to 78.29%, with notable increases in recall (0.74) and F1 score (0.60); Decision Tree also saw an accuracy rise to 74.56% but experienced a slight decline in F1 score to 0.52. These findings demonstrate that SVM, particularly when used with SMOTE, offers better overall performance and class balance in classifying reviews with imbalanced sentiment distribution, making it more suitable than Decision Tree for this application.



This is an open access article under the [CC-BY-SA](#) license.

---

### **I. PENDAHULUAN**

Perkembangan teknologi informasi yang pesat telah mendorong semakin banyaknya aplikasi digital yang digunakan untuk memudahkan transaksi dan interaksi sehari-hari. Salah satu aplikasi yang populer di Indonesia adalah LinkAja, sebuah platform pembayaran digital yang menyediakan berbagai layanan keuangan. Dalam penggunaan aplikasi tersebut, analisis sentimen pengguna menjadi aspek penting untuk memahami kepuasan dan permasalahan yang dialami oleh pengguna.

Teknik pengolahan data yang efektif sangat dibutuhkan untuk menghasilkan model klasifikasi yang akurat dalam menganalisis sentimen pengguna. Algoritma Support Vector Machine (SVM) dan Decision Tree merupakan metode pembelajaran mesin yang umum digunakan untuk tugas klasifikasi karena keunggulannya masing-masing dalam hal performa dan interpretabilitas. Namun, tantangan yang sering

dihadapi dalam pengklasifikasian data sentimen adalah ketidakseimbangan kelas, di mana jumlah data pada satu kelas jauh lebih sedikit dibandingkan kelas lainnya. Untuk mengatasi masalah tersebut, Synthetic Minority Over sampling Technique (SMOTE) telah banyak diterapkan sebagai teknik augmentasi data untuk menyeimbangkan distribusi kelas. Penerapan SMOTE dipercaya dapat meningkatkan performa algoritma klasifikasi dengan memperbaiki representasi minoritas pada dataset.

Dua algoritma yang digunakan dalam analisis sentimen ini adalah Support Vector Machine (SVM) dan Decision Tree. SVM dikenal memiliki kemampuan dalam memisahkan kelas data dengan margin optimal dan sering digunakan dalam berbagai studi analisis sentimen di Indonesia, seperti pada analisis sentimen petani milenial di media sosial X yang mencapai akurasi hingga 70% [1]. Sementara itu, Decision Tree menawarkan model yang mudah diinterpretasikan dan telah berhasil digunakan dalam analisis sentimen kebijakan

fiskal di media sosial dengan akurasi mencapai 81,34% [2]. Selain itu, penelitian lain yang membandingkan SVM dan Decision Tree pada ulasan aplikasi Microsoft Co-Pilot menunjukkan bahwa SVM memiliki performa lebih baik dengan akurasi 95,4%, sedangkan Decision Tree mencapai 92,1% [3].

Penelitian-penelitian tersebut memberikan gambaran umum mengenai keunggulan dan keterbatasan masing-masing algoritma dalam konteks analisis sentimen, namun belum banyak yang fokus pada aplikasi keuangan seperti LinkAja. Penelitian ini bertujuan melakukan studi komparasi performa algoritma SVM dan Decision Tree dalam menganalisis sentimen ulasan pengguna aplikasi LinkAja di Google Play Store. Keunggulan penelitian ini terletak pada pemanfaatan dataset terkini yang relevan dengan konteks aplikasi keuangan digital di Indonesia serta evaluasi performa yang komprehensif menggunakan metrik akurasi, presisi, dan recall. Dengan demikian, penelitian ini diharapkan memberikan kontribusi baru dalam pemilihan algoritma yang tepat untuk analisis sentimen aplikasi keuangan, sekaligus mengisi kekosongan riset pada domain ini.

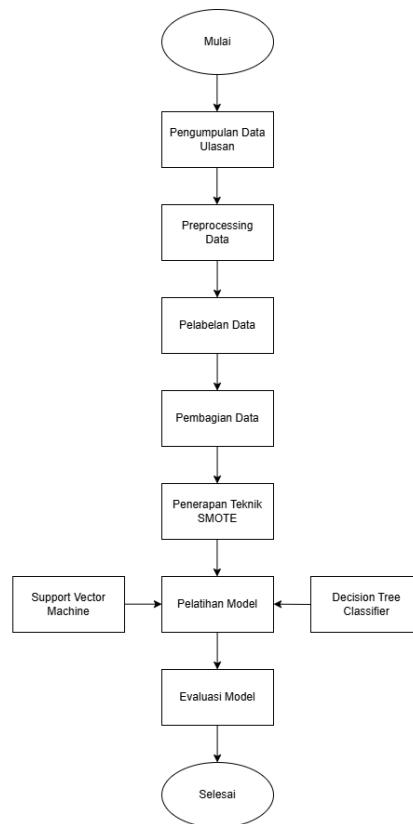
Penelitian ini bertujuan untuk melakukan studi komparatif antara algoritma SVM dan Decision Tree dalam pengaruh teknik SMOTE terhadap performa klasifikasi sentimen pada data ulasan aplikasi LinkAja. Hasil dari studi ini diharapkan dapat memberikan gambaran yang lebih jelas mengenai efektivitas kedua algoritma tersebut dalam mengelola data tidak seimbang serta memberikan rekomendasi untuk pengembangan sistem analisis sentimen yang lebih optimal. Salah satu batasan dalam penelitian ini adalah bahwa data yang digunakan hanya berasal dari satu sumber (Google Play Store) dan berfokus pada bahasa Indonesia, sehingga hasil yang didapat mungkin tidak dapat diterapkan secara luas pada konteks yang lain.

Beberapa studi komparatif menegaskan bahwa SVM dan Decision Tree secara konsisten memberikan hasil yang kompetitif dan optimal di banyak dataset sentiment analysis dibanding algoritma lain, termasuk Naïve Bayes, Logistic Regression, dan Random Forest, terutama dalam hal akurasi dan keseimbangan performa pada berbagai kelas sentimen [4]. SVM dipilih karena kemampuannya untuk mengklasifikasikan data dengan dimensi tinggi secara efektif melalui konsep hyperplane yang memisahkan kelas dengan margin maksimal. Keunggulan ini membuat SVM sangat akurat dan tahan terhadap data noise sehingga memberikan performa terbaik dalam klasifikasi sentimen, terutama untuk data teks yang memiliki banyak fitur. Selain itu, SVM memiliki kemampuan komputasi cepat dan dapat dioptimalkan dengan berbagai teknik tuning parameter [5].

Sedangkan Decision Tree dipilih karena sifatnya yang mudah diinterpretasikan dan cepat dalam proses klasifikasi. Decision Tree mampu memberikan hasil yang cukup baik dengan struktur model berupa pohon keputusan yang dapat memvisualisasikan proses klasifikasi, memudahkan analisis dan implementasi. Walaupun akurasinya kadang lebih rendah daripada SVM, Decision Tree menunjukkan performa yang

cukup seimbang dalam memproses data yang tidak linear dan distribusi kelas yang tidak seimbang [6].

## II. METODE



Gambar 1. Diagram Alur Penelitian

### A. Pengumpulan Data

Penelitian ini mengambil 7000 data ulasan aplikasi Kredivo yang dikumpulkan dengan metode scraping data melalui Google Play Store. Pengambilan data dilakukan dengan memanfaatkan library Python seperti google-play-scraper atau API SerpApi yang memungkinkan pengambilan ulasan secara otomatis dan efisien. Proses scraping dilakukan dengan menentukan app ID aplikasi LinkAja dan menggunakan parameter bahasa Indonesia untuk mendapatkan ulasan yang relevan [7].

### B. Preprocessing Data

Langkah awal yang penting dalam pengolahan data adalah *preprocessing*, yang melibatkan transformasi data mentah ke dalam format yang dapat digunakan oleh prosedur penambangan. Tahap ini mempersiapkan data untuk digunakan pada tahap-tahap berikutnya dengan memproses data teks guna meningkatkan kualitasnya. Proses praprosesan meliputi pembersihan data (cleaning),

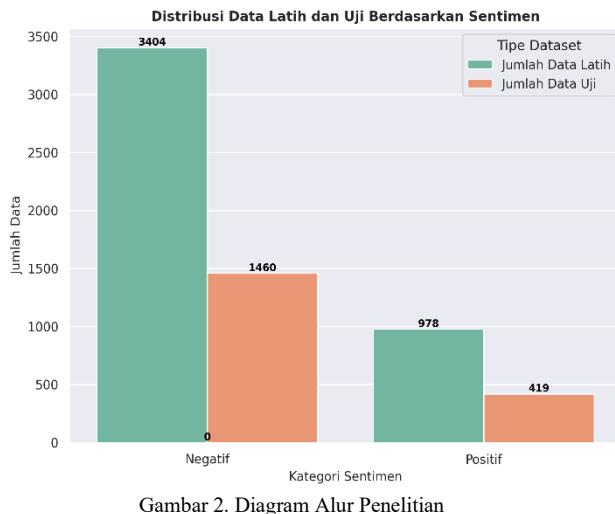
pengubahan huruf menjadi huruf kecil (case folding), penghilangan kata-kata yang tidak bermakna (stopword removal), stemming, dan tokenisasi [8].

### C. Pelabelan Data

Pelabelan data adalah proses memberikan tanda atau label pada sampel data untuk mengkategorikan data tersebut sesuai dengan tujuan analisis, terutama dalam konteks pelatihan model machine learning. Dalam penelitian ini, pelabelan dilakukan secara manual dengan aturan bahwa ulasan yang memiliki skor rating lebih dari 3 dikategorikan sebagai sentimen positif, sedangkan skor rating kurang dari atau sama dengan 3 dikategorikan sebagai sentimen negatif. Pelabelan manual ini penting untuk memastikan kualitas data pelatihan yang digunakan dalam model klasifikasi [9].

### D. Pembagian Data

Pembagian set data sangat krusial dalam proses pengembangan model machine learning agar model dapat belajar dan diuji dengan baik. Dalam studi ini, perbandingan pembagian data yang diterapkan adalah 70:30, di mana 70% digunakan untuk proses pelatihan dan sisanya 30% untuk pengujian. Dari total 7.000 data ulasan aplikasi LinkAja, terdapat 6.261 data yang tersisa setelah melalui tahap preprocessing, dengan 4.382 data dialokasikan sebagai data pelatihan dan 1.879 data yang lain sebagai data pengujian. Pembagian data ini dilakukan secara acak (*random split*) untuk menjaga representasi data yang seimbang dan menghindari bias dalam pelatihan dan pengujian model [10].



Gambar 2. Diagram Alur Penelitian

### E. Support Vector Machine

Support Vector Machine (SVM) merupakan salah satu algoritma machine learning yang banyak digunakan untuk tugas klasifikasi, termasuk dalam analisis sentimen teks. Algoritma ini bekerja dengan menentukan hyperplane terbaik yang memisahkan kelas-kelas data secara maksimal di ruang fitur, sehingga menghasilkan model yang mampu menggeneralisasi data baru dengan baik. Dalam analisis

sentimen, SVM terbukti efektif dalam menangani dataset yang besar dan kompleks serta sering kali memberikan tingkat akurasi yang lebih tinggi dibandingkan metode lain [11].

### F. Decision Tree Classifier

Decision Tree Classifier merupakan salah satu algoritma Decision Tree Classifier adalah salah satu jenis algoritma machine learning yang banyak dipakai dalam tugas-tugas klasifikasi, termasuk analisis sentimen. Algoritma ini menciptakan model berupa pohon keputusan dengan membagi set data berdasarkan atribut yang paling signifikan, sehingga membuat hasil klasifikasi lebih mudah untuk diinterpretasikan. Decision Tree Classifier terbukti efektif dalam mengklasifikasikan data teks seperti ulasan aplikasi dan postingan media sosial dengan tingkat akurasi yang bervariasi, mulai dari 79% hingga 96,5% [12].

## III. HASIL DAN PEMBAHASAN

### A. Pengumpulan Data

Pengambilan data dalam penelitian ini dilakukan dengan metode scraping data ulasan aplikasi LinkAja yang diunduh dari Google Play Store. Total data yang dikumpulkan sebanyak 7000 ulasan berbahasa Indonesia yang relevan dengan topik penelitian. Proses scraping dilakukan menggunakan library Python seperti google-play-scraper dan API SerpApi, yang memungkinkan pengambilan data secara otomatis dan efisien. Data yang diambil meliputi konten ulasan, tanggal ulasan, skor rating, dan metadata lainnya. Pengambilan data dilakukan dengan menentukan app ID LinkAja dan menggunakan parameter bahasa Indonesia untuk memastikan relevansi data.

TABEL 1  
DATA ULASAN APLIKASI

No.	Text
1	Saya suka sekali dengan aplikasi ini
2	Semakin susah digunakan padahal jaringan stabil tapi disuruh cek koneksi, sedangkan apk lain lancar...
3	Dana ga bisa ditarik
...	...
7000	Biaya transfer ke Bank sangat tinggi

### B. Preprocessing Data

Setelah pengumpulan data selesai, tahapan selanjutnya adalah melakukan pra-pemrosesan data. Pra-pemrosesan adalah aktivitas yang melibatkan seleksi dan evaluasi teks, termasuk membersihkan teks, memperbaiki kesalahan, serta menyederhanakan teks agar siap untuk analisis lebih lanjut. Tujuan utama dari proses pra-pemrosesan adalah untuk meningkatkan mutu data teks yang akan dipergunakan. Proses ini terdiri dari beberapa langkah, yaitu pembersihan data, transformasi huruf menjadi huruf kecil, normalisasi teks,

penguraian menjadi token, penghilangan kata-kata yang tidak memiliki makna signifikan, serta penerapan stemming.

TABEL 2  
HASIL CLEANING

<b>Ulasan</b>	<b>Cleaning</b>
Semakin susah digunakan padahal jaringan stabil tapi disuruh cek koneksi, sedangkan apk lain lancar...	Semakin susah digunakan padahal jaringan stabil tapi disuruh cek koneksi sedangkan apk lain lancar

TABEL 3  
HASIL CASE FOLDING

<b>Ulasan</b>	<b>Case Folding</b>
Semakin susah digunakan padahal jaringan stabil tapi disuruh cek koneksi, sedangkan apk lain lancar...	semakin susah digunakan padahal jaringan stabil tapi disuruh cek koneksi sedangkan apk lain lancar

TABEL 4  
HASIL NORMALIZATION

<b>Ulasan</b>	<b>Normalization</b>
semakin susah digunakan padahal jaringan stabil tapi disuruh cek koneksi sedangkan apk lain lancar	semakin sulit digunakan padahal jaringan stabil tetapi diminta periksa koneksi sedangkan aplikasi lain lancar

TABEL 5  
HASIL TOKENIZING

<b>Ulasan</b>	<b>Tokenizing</b>
semakin sulit digunakan padahal jaringan stabil tetapi diminta periksa koneksi sedangkan aplikasi lain lancar	['semakin', 'sulit', 'digunakan', 'padahal', 'jaringan', 'stabil', 'tetapi', 'diminta', 'periksa', 'koneksi', 'sedangkan', 'aplikasi', 'lain', 'lancar']

TABEL 6  
HASIL STOPWORD REMOVAL

<b>Ulasan</b>	<b>Stopword Removal</b>
semakin sulit digunakan padahal jaringan stabil tetapi diminta periksa koneksi sedangkan aplikasi lain lancar	['sulit', 'digunakan', 'jaringan', 'stabil', 'diminta', 'periksa', 'koneksi', 'aplikasi', 'lancar']

TABEL 7  
HASIL STEMMING

<b>Stopword Removal</b>	<b>Stemming</b>
['sulit', 'digunakan', 'jaringan', 'stabil', 'diminta', 'periksa', 'koneksi', 'aplikasi', 'lancar']	['sulit', 'guna', 'jaring', 'stabil', 'minta', 'periksa', 'koneksi', 'aplikasi', 'lancar']

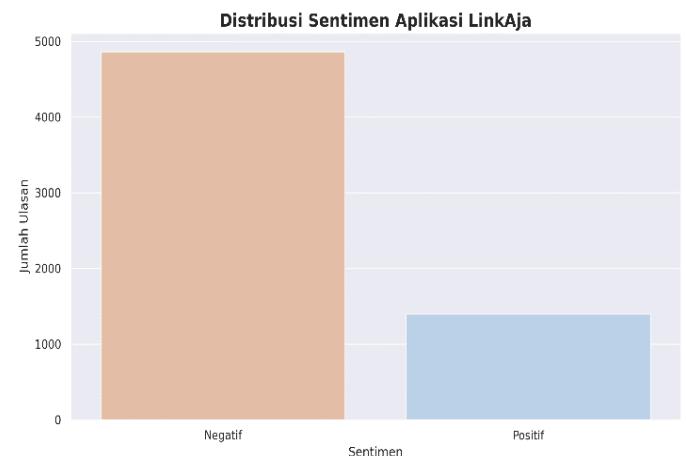
### C. Pelabelan Data

Pelabelan data berfokus pada pengelompokan data ulasan ke dalam kategori positif dan negatif, memungkinkan model machine learning untuk dengan efisien mempelajari pola-pola sentimen tersebut. Dalam studi ini, pelabelan dilakukan dengan dua metode utama, yaitu pendekatan yang berdasarkan rating. Pendekatan ini memanfaatkan skor rating yang diberikan pengguna pada ulasan aplikasi sebagai petunjuk untuk menentukan sentimen. Ulasan yang memiliki rating antara 1 hingga 3 dianggap sebagai sentimen negatif, sementara rating 4 dan 5 diakui sebagai sentimen positif. Pendekatan ini cukup sederhana namun efektif karena secara

langsung mencerminkan tingkat kepuasan pengguna terhadap aplikasi berdasarkan pengalaman mereka [13]. Hasil pelabelan dijelaskan dalam Tabel 8.

TABEL 8  
JUMLAH KELAS SENTIMEN

<b>Label</b>	<b>Jumlah</b>
Positif	4.864
Negatif	1.397
Total	6.261



Gambar 3. Hasil Pelabelan Sentimen

Berdasarkan hasil dari pelabelan tersebut, terdapat total 6.261 data sentimen yang telah dibagi menjadi dua kelas sentimen, yaitu 4.864 data berlabel positif dan 1.397 data berlabel negatif. Oleh karena itu, mayoritas pengguna aplikasi memberikan tanggapan positif terhadap layanan yang disediakan. Dataset ini tidak seimbang secara signifikan, dengan jumlah data positif jauh lebih banyak dibandingkan data negatif. Namun, perbedaan jumlah yang cukup besar ini perlu diperhatikan karena dapat mempengaruhi proses analisis dan pelatihan model.

Sentimen positif yang dominan menunjukkan bahwa banyak pengguna merasa puas dengan fitur dan performa aplikasi. Namun, sentimen negatif yang muncul sebagian besar disebabkan oleh keluhan pengguna terkait transaksi yang gagal namun tetap dikenakan tagihan, layanan customer service yang lambat merespons, serta kesulitan dalam mendaftarkan kartu di dalam e-wallet LinkAja. Untuk mengurangi sentimen negatif tersebut, pengembang aplikasi disarankan untuk memperbaiki sistem transaksi agar lebih andal dan transparan, meningkatkan kecepatan serta kualitas layanan customer service, serta menyederhanakan proses pendaftaran kartu. Dengan perbaikan ini, diharapkan tingkat kepuasan dan kepercayaan pengguna terhadap aplikasi dapat meningkat secara signifikan.



Gambar 4. WordCloud Kata Positif



Gambar 5. WordCloud Kata Negatif

#### *D. Pelatihan Model*

Dalam tahap pelatihan model, algoritma Support Vector Machine (SVM) yang menggunakan kernel linear diaplikasikan bersamaan dengan metode ekstraksi fitur TF-IDF (TfidfVectorizer) agar dapat mengkonversi data teks menjadi representasi angka yang dapat dianalisis oleh model. TF menghitung seberapa sering sebuah kata muncul dalam suatu dokumen, sementara IDF mengevaluasi seberapa jarang kata itu muncul di semua dokumen, sehingga kata-kata yang muncul sering dalam dokumen tertentu tetapi jarang dalam dokumen lainnya diberikan bobot yang lebih tinggi. Metode ini banyak digunakan dalam pengambilan informasi dan analisis teks untuk mengekstrak fitur penting dalam klasifikasi sentimen dan teks [14].

Pemilihan kernel linear didasari oleh karakteristik data teks yang biasanya memiliki dimensi tinggi dan bersifat linier, sehingga kernel ini memberikan hasil yang optimal dan lebih cepat dibandingkan kernel nonlinier (RBF, polynomial) dalam banyak kasus analisis sentimen [15]. Algoritma Decision Tree juga dilatih menggunakan hasil ekstraksi fitur TF-IDF yang sama, di mana proses konversi teks ke bentuk vektor dilakukan terlebih dahulu sebelum model diklasifikasikan. Pendekatan ini memastikan alur kerja yang sistematis dan konsisten dalam pemrosesan data teks serta pelatihan kedua algoritma. Decision Tree dipilih karena kemampuannya dalam menangani data kategorikal dan

numerik serta kemudahan interpretasi model melalui pohon keputusan yang dihasilkan

#### *E. Penyeimbangan Data Menggunakan SMOTE*

Dataset *review* pengguna menunjukkan distribusi kelas yang tidak seimbang, di mana jumlah data dengan sentimen negatif jauh lebih banyak daripada yang positif. Ketidakseimbangan ini bisa mengakibatkan model memiliki kecenderungan bias terhadap kelas yang lebih besar. Untuk mengatasi masalah ini, studi ini menggunakan metode Synthetic Minority Over-sampling Technique (SMOTE) pada data pelatihan. Teknik ini memperbaiki representasi kelas minoritas agar model machine learning dapat belajar dengan lebih efektif tanpa adanya bias terhadap kelas mayoritas. Banyak studi menunjukkan bahwa SMOTE dapat meningkatkan akurasi model klasifikasi dengan memperbaiki keseimbangan data [16].

```
Distribusi sebelum SMOTE: {'Negatif': 4864, 'Positif': 1384}
Distribusi sesudah SMOTE: {'Negatif': np.int64(4864), 'Positif': np.int64(4864)}
```

Gambar 6. SMOTE

#### *F. Evaluasi Model*

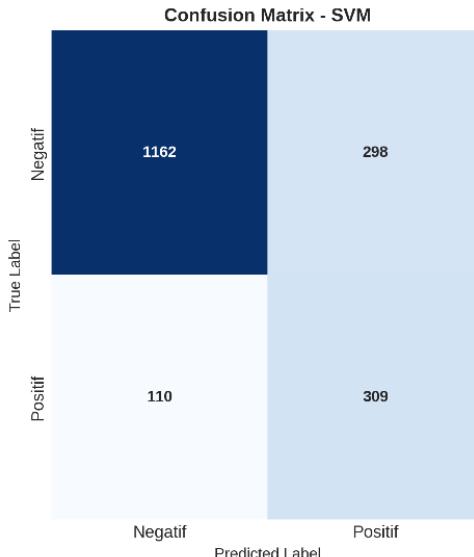
Evaluasi hasil dari penelitian ini dilaksanakan dengan memanfaatkan Confusion Matrix sebagai alat pengukuran. Dengan menggunakan Confusion Matrix, dihitunglah metrik seperti akurasi, presisi, recall, dan F1-score untuk menilai kinerja model klasifikasi yang menerapkan metode Support Vector Machine (SVM) serta Decision Tree [10]. Confusion Matrix merupakan sebuah tabel yang berfungsi untuk menilai kinerja model klasifikasi dengan cara membandingkan prediksi model dengan data yang sebenarnya. Dalam konteks klasifikasi sentimen yang menggunakan algoritma SVM, confusion matrix mengklasifikasikan hasil prediksi dalam empat kategori utama.

**TABEL 9**  
**HASIL PERBANDINGAN EVALUASI**

Algoritma	Metode	Akurasi	Precision	Recall	F-1 Score
SVM	Sebelum SMOTE	77.97%	0.74	0.33	0.45
	Setelah SMOTE	78.29%	0.51	0.74	0.60
Decision Tree	Sebelum SMOTE	72.01%	0.50	0.62	0.55
	Setelah SMOTE	74.56%	0.45	0.63	0.52

Tabel 9 menampilkan hasil perbandingan evaluasi algoritma SVM dan Decision Tree sebelum dan sesudah penerapan SMOTE, menggunakan metrik akurasi, presisi, recall, dan F1 score. SVM menunjukkan peningkatan akurasi tipis dari 77,97% menjadi 78,29% setelah SMOTE, dengan presisi menurun namun recall dan F1 score meningkat secara signifikan, mengindikasikan model dapat menangkap lebih banyak data positif sesungguhnya. Decision Tree juga mengalami kenaikan akurasi dari 72,01% menjadi 74,56% setelah SMOTE, namun perubahan presisi dan F1 score

cenderung menurun, menunjukkan performa model yang kurang optimal dalam keseimbangan presisi dan recall setelah oversampling.



Gambar 7. Confusion Matrix SVM

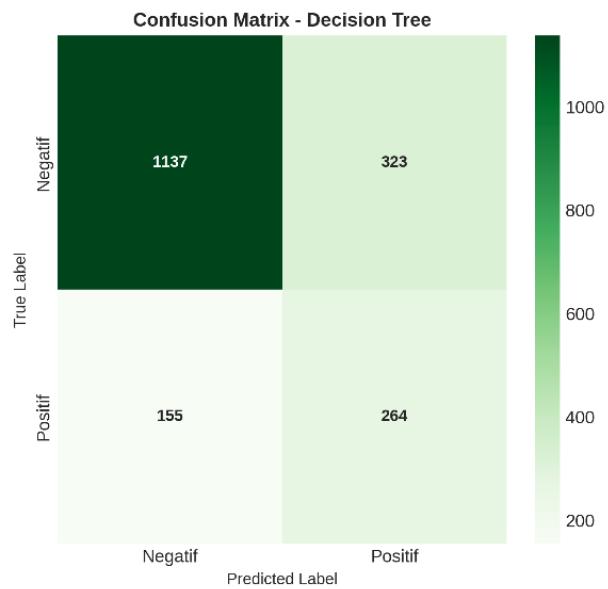
Gambar 7 memperlihatkan bahwa True Positive (TP) adalah total data yang sejatinya positif dan dapat diprediksi dengan tepat sebagai positif oleh model, yaitu 309 ulasan. False Negative (FN) merupakan data positif yang justru keliru diprediksi negatif, yang mencapai 110 ulasan, menunjukkan bahwa model gagal dalam mendeteksi sentimen positif. False Positive (FP) adalah data yang seharusnya negatif namun diprediksi sebagai positif oleh model, dengan jumlah 298 ulasan, menunjukkan bahwa model membuat prediksi positif yang salah. Sementara True Negative (TN) adalah data yang benar-benar negatif dan diprediksi dengan akurat sebagai negatif oleh model, sebanyak 1.162 ulasan, mencerminkan kemampuan model dalam mengenali sentimen negatif dengan tepat.

$$\text{Accuracy} = \frac{309 + 1162}{309 + 1162 + 298 + 110} = 0.78$$

$$\text{Precision} = \frac{309}{309 + 298} = 0.51$$

$$\text{Recall} = \frac{309}{309 + 110} = 0.74$$

$$\text{F1 - Score} = \frac{2 \times (0.51 \times 0.74)}{0.51 + 0.74} = 0.60$$



Gambar 8. Confusion Matrix Decision Tree

Untuk model Decision Tree, confusion matrix menunjukkan bahwa True Positive (TP) berjumlah 264 ulasan yang benar-benar positif dan diprediksi positif, False Negative (FN) sebanyak 155 ulasan positif yang salah diprediksi negatif, False Positive (FP) sejumlah 323 ulasan negatif yang salah diprediksi positif, dan True Negative (TN) sebanyak 1.137 ulasan negatif yang diprediksi dengan benar.

$$\text{Accuracy} = \frac{264 + 1137}{264 + 1137 + 323 + 155} = 0.75$$

$$\text{Precision} = \frac{264}{264 + 323} = 0.45$$

$$\text{Recall} = \frac{264}{264 + 155} = 0.63$$

$$\text{F1 - Score} = \frac{2 \times (0.45 \times 0.63)}{0.45 + 0.63} = 0.52$$

Model SVM menunjukkan akurasi yang lebih tinggi dan presisi yang lebih baik dibandingkan Decision Tree. Hal ini disebabkan oleh kemampuan SVM dalam memaksimalkan margin pemisah antar kelas, sehingga mampu mengurangi jumlah False Positives (FP) dan menghasilkan prediksi positif yang lebih tepat. Namun, recall SVM relatif rendah, yang berarti model ini kurang mampu mengenali seluruh sampel positif secara menyeluruh dan masih banyak False Negatives (FN). Kondisi ini menunjukkan bahwa SVM lebih selektif dalam mengklasifikasikan data sebagai positif, sehingga lebih sedikit kesalahan prediksi positif palsu, tetapi ada beberapa data positif yang terlewati.

Sebaliknya, model Decision Tree menunjukkan tingkat recall yang lebih baik daripada SVM, yang menunjukkan

bahwa model ini lebih efektif dalam mendeteksi ulasan positif secara umum. Namun, tingkat presisi dari Pohon Keputusan lebih rendah karena jumlah False Positives yang lebih besar, sehingga banyak data negatif yang salah teridentifikasi sebagai positif. Hal ini dapat terjadi karena Decision Tree cenderung melakukan pembelahan data berdasarkan fitur tertentu tanpa margin optimal, sehingga lebih rentan terhadap overfitting dan kesalahan klasifikasi pada data negatif. Recall yang tinggi pada Decision Tree menunjukkan model ini kurang selektif sehingga mampu menangkap lebih banyak data positif, tetapi dengan risiko meningkatkan kesalahan prediksi positif palsu

#### IV. KESIMPULAN

Model klasifikasi yang diuji dalam penelitian ini adalah Support Vector Machine (SVM) dan Decision Tree untuk analisis sentimen ulasan pengguna aplikasi LinkAja yang diperoleh dari Google Play Store. Dari total 7.000 data ulasan yang dikumpulkan, sebanyak 6.261 data layak digunakan setelah melalui proses preprocessing. Pelabelan dilakukan menggunakan metode rating-based, di mana ulasan dengan skor di atas 3 dikategorikan sebagai sentimen positif, sedangkan skor 3 ke bawah dikategorikan sebagai sentimen negatif. Dataset dibagi dengan rasio 70:30 menjadi 4.328 data latih dan 1.879 data uji, kemudian dilakukan penyeimbangan kelas menggunakan metode SMOTE pada data latih untuk mengatasi ketidakseimbangan jumlah kelas.

Berdasarkan hasil evaluasi, dapat disimpulkan bahwa penerapan SMOTE lebih efektif pada algoritma SVM karena mampu meningkatkan recall dan F1 score secara signifikan, sementara Decision Tree hanya menunjukkan kenaikan akurasi tanpa perbaikan keseimbangan performa. Dengan demikian, menunjukkan bahwa SVM unggul secara keseluruhan dengan nilai akurasi, presisi, dan F1-score yang lebih tinggi dibandingkan Decision Tree. Meskipun keduanya memiliki kemampuan yang baik dalam mengenali kelas negatif, SVM menunjukkan performa klasifikasi yang lebih stabil dan efisien dalam menangani variasi teks ulasan pengguna LinkAja setelah penerapan SMOTE.

#### DAFTAR PUSTAKA

- [1] M. Ma'rufudin and A. Yudhistira, "Analisis Sentimen Petani Milenial Pada Media Sosial X Menggunakan Algoritma Support Vector Machine (SVM)," *J. Pendidik. dan Teknol. Indones.*, vol. 5, no. 3, pp. 845–857, 2025, doi: 10.52436/1.jpti.717.
- [2] I. Diah Hardyatman and F. Noor Hasan, "Analisis Sentimen Masyarakat Terhadap Rencana Kenaikan PPN 12% Di Indonesia Pada Media Sosial X Menggunakan Metode Decision Tree," *J. Inf. Syst. Res.*, vol. 6, no. 2, pp. 1126–1134, 2025, doi: 10.47065/josh.v6i2.6573.
- [3] V. No, I. K. Najibullah, D. Intan, and S. Saputra, "Edumatic : Jurnal Pendidikan Informatika Analisis Sentimen Ulasan Co-Pilot Google Play dengan SVM , Neural Network , dan Decision Tree," vol. 9, no. 1, pp. 275–283, 2025, doi: 10.29408/edumatic.v9i1.29673.
- [4] Y. A. Singgalen, "KLIK: Kajian Ilmiah Informatika dan Komputer Implementation of SVM and DT for Sentiment Classification: Tempel Hamlet Content Reviews," *Media Online*, vol. 4, no. 5, pp. 2571–2579, 2024, doi: 10.30865/klirk.v4i5.1826.
- [5] K. A. Rokhman, B. Beriliana, and P. Arsi, "Perbandingan Metode Support Vector Machine Dan Decision Tree Untuk Analisis Sentimen Review Komentar Pada Aplikasi Transportasi Online," *J. Inf. Syst. Manag.*, vol. 3, no. 1, pp. 1–7, 2021, doi: 10.24076/joism.2021v3i1.341.
- [6] F. Panjaitan, W. Ce, H. Oktafiandi, G. Kanugrahan, Y. Ramdhani, and V. H. C. Putra, "Evaluation of Machine Learning Models for Sentiment Analysis in the South Sumatra Governor Election Using Data Balancing Techniques," *J. Inf. Syst. Informatics*, vol. 7, no. 1, pp. 461–478, 2025, doi: 10.51519/journalisi.v7i1.1019.
- [7] K. Ahmad, "Analisis Sentimen Pinjaman Online Akulaku dan Kredivo dengan metode Support Vector Machine (SVM)," *J. Mandalika Lit.*, vol. 4, no. 4, pp. 323–332, 2023, doi: 10.36312/jml.v4i4.2045.
- [8] M. Hasanudin, S. Dwiasnati, and W. Gunawan, "Pelatihan Datascience pada Pra-Pemrosesan Data untuk Siswa SMK Media Informatika - Jakarta," *J. Pengabd. Pada Masy.*, vol. 9, no. 4, pp. 882–888, 2024, doi: 10.30653/jppm.v9i4.921.
- [9] H. Firda *et al.*, "Perbandingan Pelabelan Rating - based dan Inset Lexicon - based dalam Analisis Sentimen Menggunakan SVM ( Studi Kasus : Ulasan Aplikasi GoBiz di Google Play Store ) Comparison of Rating - based and Inset Lexicon - based Labeling in Sentiment Analysis usin," vol. 14, pp. 516–528, 2025.
- [10] D. Haliza and M. Ihsan, "Sentiment Analysis on Public Perception of the Nusantara Capital on Social Media X Using Support Vector Machine (SVM) and K-Nearest Neighbor (K-NN) Methods," *J. Appl. Informatics Comput.*, vol. 9, no. 3, pp. 716–723, 2025, doi: 10.30871/jaic.v9i3.9318.
- [11] M. G. R. Lubis, D. S. Sitompul, T. M. Giovanni, F. Ramadhani, and S. Dewi, "Evaluasi Kinerja Algoritma Support Vector Machine (SVM) Dalam Analisis Sentimen Publik Terhadap Naturalisasi Timnas Indonesia di Twitter," *J. Account. Law Commun. Technol.*, vol. 2, no. 1, pp. 81–89, 2024, doi: 10.57235/jalakotek.v2i1.4180.
- [12] I. L. Kharisma, D. A. Septiani, A. Fergina, and K. Kamdan, "Penerapan Algoritma Decision Tree untuk Ulasan Aplikasi Vidio di Google Play," *J. Nas. Teknol. dan Sist. Inf.*, vol. 9, no. 2, pp. 218–226, 2023, doi: 10.25077/teknosi.v9i2.2023.218-226.
- [13] R. Pohan, D. Ratnawati, and I. Arwani, "Implementasi Algoritma Support Vector Machine dan Model Bag-of-Words dalam Analisis Sentimen mengenai PILKADA 2020 pada Pengguna Twitter," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 6, no. 10, pp. 4924–4931, 2022, [Online]. Available: <http://j-ptik.ub.ac.id>
- [14] Y. Nuri and E. Senyurek, "Research Abstracts Similarity Implementation By Using TF-IDF Algorithm," vol. 27, no. 1, pp. 4–10, 2025, doi: 10.9790/0661-2701040410.
- [15] S. Christianto and J. Amanda Ginting, "Social Network Analysis Peringatan Darurat Ruu Pemilihan Kepala Daerah," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 9, no. 3, pp. 5366–5373, 2025, doi: 10.36040/jati.v9i3.13676.
- [16] W. Rahayu *et al.*, "Synthetic Minority Oversampling Technique (SMOTE) for Boosting the Accuracy of C4.5 Algorithm Model," *J. Artif. Intell. Eng. Appl.*, vol. 3, no. 3, pp. 624–630, 2024, doi: 10.59934/jaiea.v3i3.469.