

Aspect-Based Sentiment Analysis Pada Ulasan Aplikasi Access by KAI Menggunakan Metode TF-IDF Dan Algoritma Support Vector Machine

Muhammad Nur Rachman Nidhi Suryono^{*1}, Amalia Anjani Arifiyanti², Dhian Satria Yudha Kartika³

^{1,2,3}Sistem Informasi, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional “Veteran” Jawa Timur, Indonesia

Email: ¹naudhi153@gmail.com, ²amalia_anjani.fik@upnjatim.ac.id,

³dhian.satria@upnjatim.ac.id

Abstrak

Access by KAI merupakan aplikasi layanan transportasi digital dari PT Kereta Api Indonesia yang mempermudah pengguna dalam mengakses layanan perjalanan kereta api. Untuk meningkatkan kualitas layanan dan pengalaman pengguna, penelitian ini melakukan analisis sentimen terhadap ulasan pengguna aplikasi menggunakan algoritma Support Vector Machine (SVM). Tiga aspek utama yang dianalisis yaitu Financial Transactions, Technical Issues and Performance, serta User Experience and Interface. Penelitian menggunakan kombinasi metode sampling (SMOTE dan Non-SMOTE), kernel (Linear, RBF, Polynomial), dan pembagian data (80:20 dan 70:30) untuk menemukan model terbaik. Hasil terbaik untuk aspek Financial Transactions diperoleh dari model SMOTE dengan kernel RBF dan rasio 70:30 (akurasi 0.9270). Untuk Technical Issues and Performance, model terbaik adalah Non-SMOTE dengan kernel Linear dan rasio 70:30 (akurasi 0.8718). Sedangkan untuk User Experience and Interface, model Non-SMOTE dengan kernel Linear dan rasio 80:20 memberikan akurasi tertinggi sebesar 0.8825. Model terbaik ini diimplementasikan dalam aplikasi web berbasis Flask yang dapat memprediksi sentimen, mengeksport hasil dalam bentuk .csv, serta menampilkan visualisasi data. Hasil implementasi menunjukkan bahwa kombinasi model terpilih mampu memberikan pemetaan sentimen yang konsisten dan terstruktur terhadap ulasan pengguna, sehingga dapat digunakan sebagai dasar evaluasi berbasis data dalam pengembangan fitur aplikasi.

Kata kunci: Access by KAI, Analisis Sentimen Berbasis Aspek, BERTopic, Flask, Support Vector Machine

Abstract

Access by KAI is a digital transportation service application developed by PT Kereta Api Indonesia that facilitates users in accessing railway travel services. To enhance service quality and user experience, this study conducts sentiment analysis on user reviews of the application using the Support Vector Machine (SVM) algorithm. Three main aspects are analyzed: Financial Transactions, Technical Issues and Performance, and User Experience and Interface. The study employs a combination of sampling methods (SMOTE and Non-SMOTE), kernels (Linear, RBF, Polynomial), and data splits (80:20 and 70:30) to identify the best-performing model. The best results for the Financial Transactions aspect are obtained using the SMOTE method with the RBF kernel and a 70:30 ratio (accuracy 0.9270). For Technical Issues and Performance, the best model is Non-SMOTE with a Linear kernel and a 70:30 ratio (accuracy 0.8718). Meanwhile, for User Experience and Interface, the highest accuracy of 0.8825 is achieved by the Non-SMOTE model with a Linear kernel and an 80:20 ratio. The selected best models are implemented in a Flask-based web application capable of predicting sentiment, exporting results in .csv format, and presenting data visualizations. The implementation results indicate that the selected model combination provides consistent and structured sentiment mapping of user reviews, making it suitable as a data-driven basis for evaluating and improving application features.

Keywords: Access by KAI, Aspect-Based Sentiment Analysis, BERTopic, Flask, Support Vector Machine

This work is an open access article and licensed under a Creative Commons Attribution-NonCommercial ShareAlike 4.0 International (CC BY-NC-SA 4.0)



1. PENDAHULUAN

Transformasi digital telah menjadi faktor kunci dalam meningkatkan kualitas layanan publik, termasuk di sektor transportasi [1]. PT Kereta Api Indonesia (KAI) mengubah aplikasi KAI Access menjadi Access by KAI pada tahun 2023, menghadirkan antarmuka dan fitur modern. Meski diharapkan

meningkatkan kemudahan, perubahan ini menuai respons beragam dari pengguna di *Google Play Store*. Ada respons positif, namun banyak yang kecewa dengan penurunan kinerja, navigasi membingungkan, dan antarmuka kurang intuitif. Perbedaan sentimen ini memunculkan kebutuhan akan teknik analisis yang efisien untuk memahami dan mengklasifikasikan opini pengguna.

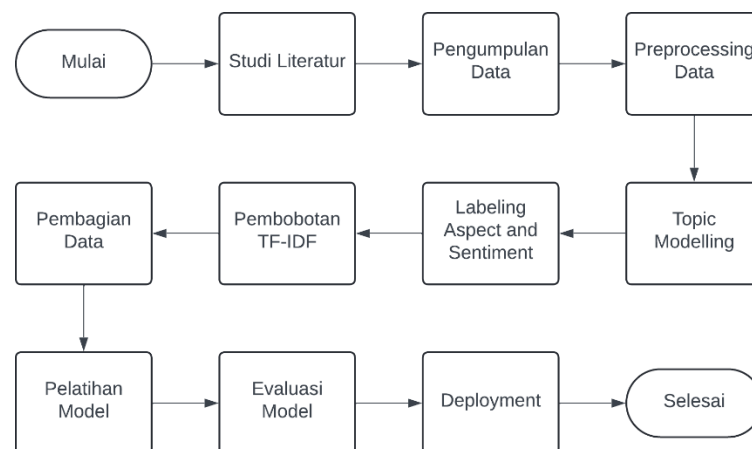
Analisis sentimen adalah teknik untuk memahami, menginterpretasikan, dan mengklasifikasikan opini dalam teks, seperti ulasan atau komentar di media sosial, dengan mengidentifikasi sentimen sebagai positif atau negatif [2]. Teknik ini menjadi semakin penting dalam era digital yang penuh data opini terkait produk, layanan, atau topik tertentu. Namun, untuk mendapatkan pemahaman yang lebih mendalam, diperlukan analisis yang lebih spesifik pada aspek-aspek tertentu dari produk atau layanan tersebut. *Aspect-Based Sentiment Analysis* (ABSA) adalah versi lebih mendalam dari analisis sentimen yang tidak hanya mengidentifikasi sentimen terhadap sebuah teks secara umum, tetapi juga mengekstrak aspek-aspek tertentu dari sebuah ulasan dan menilai sentimen yang terkait dengan masing-masing aspek tersebut [3]. Misalnya, ABSA dapat mengevaluasi sentimen terkait kinerja aplikasi, antarmuka pengguna, atau fungsionalitas fitur, sehingga memberikan gambaran lebih rinci tentang kekuatan dan kelemahan suatu produk.

Untuk klasifikasi sentimen, algoritma SVM bekerja dengan menciptakan fungsi *linear* dalam ruang fitur berdimensi tinggi untuk memisahkan data. Algoritma ini dilatih menggunakan teori optimasi dari pembelajaran statistik. Dalam analisis sentimen, SVM memproyeksikan teks ke dalam ruang berdimensi tinggi dan mencari *hyperplane* optimal yang mampu memisahkan data ke dalam dua kelas, misalnya sentimen positif dan negatif. Data yang paling dekat dengan *hyperplane*, yang disebut *support vectors*, sangat penting dalam menentukan posisi optimal dari *hyperplane* tersebut [4].

Beberapa penelitian terkait ABSA dan SVM telah dilakukan pada aplikasi lain. Penelitian yang dilakukan oleh Marchenda Fayza Madjid, Dian Eka Ratnawati, dan Bayu Rahayudi dari Universitas Brawijaya menganalisis sentimen terhadap ulasan pengguna aplikasi Allo Bank dengan menggunakan SVM dan *Naïve Bayes Classifier* (NBC). Hasilnya, SVM dengan *kernel* RBF menghasilkan akurasi tertinggi sebesar 92,86%, sementara NBC dengan *Multinomial Naïve Bayes* menghasilkan akurasi 92,54% [5]. Selanjutnya, penelitian oleh Huda Mustakim dan Sigit Priyanta dari Universitas Gadjah Mada melakukan *aspect-based sentiment analysis* pada ulasan pengguna aplikasi KAI Access menggunakan NBC dan SVM. Penelitian ini menguji tiga skenario model dan menunjukkan bahwa SVM dengan *hyperparameter tuning* menghasilkan akurasi sebesar 91,63%, *f1-score* 75,55%, presisi 77,60%, dan *recall* 74,47% [6].

Berdasarkan latar belakang tersebut, jurnal ini bertujuan untuk menganalisis sentimen pengguna terhadap perubahan aplikasi Access by KAI pada aspek-aspek spesifik dengan memanfaatkan metode TF-IDF untuk ekstraksi fitur dan algoritma SVM untuk klasifikasi sentimen.

2. METODE PENELITIAN



Gambar 1. Alur Metode Penelitian

2.1. Studi Literatur

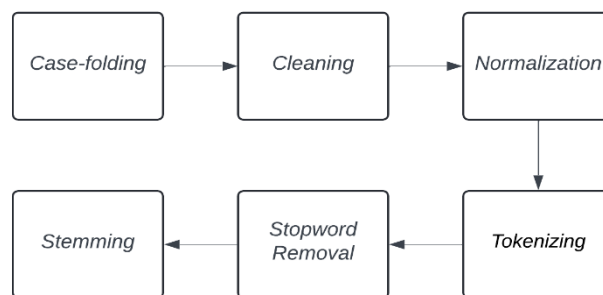
Studi literatur dalam skripsi ini bertujuan untuk menemukan teori yang relevan dan membangun pemahaman komprehensif mengenai topik yang dibahas, didukung oleh penelitian terdahulu. Proses ini

melibatkan penelusuran berbagai sumber terpercaya seperti jurnal, artikel ilmiah, hasil penelitian, serta sumber daring. Secara spesifik, skripsi ini merujuk pada dokumentasi dan referensi terkait *Aspect-Based Sentiment Analysis* (ABSA), *Topic Modelling BERTopic*, *Support Vector Machine* (SVM), *Data Mining*, dan TF-IDF, serta literatur lain yang berkaitan.

2.2. Pengumpulan Data

Data ulasan pengguna aplikasi Access By KAI dikumpulkan secara otomatis menggunakan teknik *scraping* dari *Google Play Store* dengan bantuan pustaka python *google_play_scraper*. Fokus pengumpulan data adalah pada ulasan yang diberikan setelah pembaruan aplikasi sejak Juli 2023 hingga Juli 2024, sehingga hanya mencakup pengalaman pengguna terhadap versi aplikasi yang terbaru. Data yang terkumpul nantinya akan dianalisis untuk mendapatkan gambaran umum mengenai persepsi dan pengalaman pengguna terhadap aplikasi tersebut.

2.3. Preprocessing Data



Gambar 2. Alur *Preprocessing Data*

Preprocessing Data merupakan serangkaian langkah penting dalam mengolah data teks sebelum analisis, yang dalam skripsi ini terdiri dari enam tahap: *case-folding*, *cleaning*, *normalization*, *tokenizing*, *stopword removal*, dan *stemming*. *Case-folding* adalah proses mengubah seluruh teks menjadi huruf kecil untuk konsistensi data dan menghindari perbedaan yang dapat mengganggu analisis [7]. Selanjutnya, *cleaning* bertujuan membersihkan teks dari karakter yang tidak relevan seperti tanda baca, angka, atau simbol khusus guna mengurangi *noise* data [8]. *Normalization* menyelaraskan bentuk kata-kata yang memiliki makna serupa namun ditulis berbeda menjadi bentuk standar atau seragam, mengatasi variasi bahasa seperti singkatan dan istilah tidak formal [9]. Setelah itu, *tokenizing* memecah teks menjadi unit-unit kecil seperti kata atau token, memudahkan algoritma menganalisis makna [10]. *Stopword removal* menghilangkan kata-kata umum yang tidak memiliki makna penting (misalnya, "dan", "di", "yang") untuk meningkatkan efisiensi pemrosesan [11]. Terakhir, *stemming* mengubah kata ke bentuk dasarnya dengan menghilangkan imbuhan, seperti mengubah "berjalan" menjadi "jalan", untuk mengurangi variasi kata yang memiliki arti sama [12].

2.4. Topic Modelling

BERTopic adalah metode untuk menemukan topik dalam teks secara otomatis. Metode ini menggunakan model BERT dan teknik c-TF-IDF untuk mengelompokkan teks ke dalam grup yang padat dan mudah dipahami. Dengan cara ini, BERTopic dapat menghasilkan topik yang jelas dan relevan sambil tetap mempertahankan kata-kata penting yang mendeskripsikan topik tersebut [13]. Meskipun aspek penelitian diekstraksi menggunakan BERTopic, tiga kategori utama—Financial Transactions, Technical Issues and Performance, serta User Experience and Interface—dipilih karena topik yang muncul paling banyak dan paling konsisten mengelompok pada tiga area tersebut, sehingga mewakili isu yang paling dominan dalam ulasan pengguna.

2.5. Labeling Aspect and Sentiment

Pelabelan dalam penelitian ini terdiri dari dua tahap: pelabelan aspek dan pelabelan sentimen. Pelabelan aspek dilakukan secara otomatis menggunakan hasil *topic modeling* dari BERTopic, yang mengidentifikasi kata kunci relevan untuk mengelompokkan ulasan pengguna ke dalam kategori aspek

tertentu. Sementara itu, pelabelan sentimen menetapkan label berdasarkan polaritas opini dalam teks: label 0 untuk tidak ada sentimen, label 1 untuk sentimen positif, dan label -1 untuk sentimen negatif, bertujuan untuk mengklasifikasikan data ulasan pengguna berdasarkan emosi atau opini yang terkandung.

2.6. Pembobotan TF-IDF

TF-IDF adalah metode pembobotan penting dalam analisis teks yang menilai signifikansi sebuah kata dalam suatu dokumen relatif terhadap keseluruhan koleksi dokumen (korpus). Umumnya digunakan dalam *information retrieval* dan *text mining*, TF-IDF membantu mengidentifikasi kata kunci yang relevan. Metode ini memadukan dua elemen: *Term Frequency* (TF), yang menghitung seberapa sering kata muncul dalam dokumen spesifik, dan *Inverse Document Frequency* (IDF), yang mengurangi bobot kata-kata yang sering muncul di banyak dokumen [14].

2.7. Pelatihan Model

Penelitian ini mengevaluasi kinerja tiga model klasifikasi SVM (*Linear*, *RBF*, dan *Polynomial*) menggunakan berbagai rasio pembagian data *training* dan *test* (80%:20% dan 70%:30%). Sebagaimana diuraikan dalam Tabel 1, terdapat 12 skenario pengujian, yang juga mempertimbangkan penerapan teknik SMOTE untuk menangani *imbalanced data*. Tujuannya adalah menilai kestabilan dan akurasi model dalam beragam kondisi. Melalui penggunaan *confusion matrix* pada setiap skenario, peneliti membandingkan performa model-model SVM untuk menentukan konfigurasi optimal berdasarkan nilai *accuracy*, *precision*, *recall*, dan *F1-score* tertinggi.

Tabel 1. Skenario Pengujian

No.	Data Handling	Classification Models	Training Data	Test Data
1.	Non-SMOTE	SVM Kernel Linear	80%	20%
2.			70%	30%
3.		SVM Kernel RBF	80%	20%
4.			70%	30%
5.		SVM Kernel Polynomial	80%	20%
6.			70%	30%
7.	SMOTE	SVM Kernel Linear	80%	20%
8.			70%	30%
9.		SVM Kernel RBF	80%	20%
10.			70%	30%
11.		SVM Kernel Polynomial	80%	20%
12.			70%	30%

2.8. Evaluasi Model

Dalam penelitian ini, performa model dievaluasi menggunakan *Confusion Matrix*. Ini adalah representasi tabel yang merangkum kinerja model klasifikasi dengan membandingkan hasil prediksinya

terhadap data sebenarnya. *Confusion matrix* memberikan gambaran detail tentang seberapa baik model dalam mengklasifikasikan setiap kelas, seperti ditunjukkan pada Tabel 2.

Tabel 2. *Confussion Matrix*

		<i>Predict</i>		
		<i>A</i>	<i>B</i>	<i>C</i>
<i>Actual</i>	<i>A</i>	<i>True Positive (TP)</i>	<i>False Negative (FN)</i>	<i>False Negative (FN)</i>
	<i>B</i>	<i>False Positive (FP)</i>	<i>True Neutral (TN)</i>	<i>False Negative (FN)</i>
	<i>C</i>	<i>False Positive (FP)</i>	<i>False Positive (FP)</i>	<i>True Negative (TN)</i>

2.9. Deployment

Dashboard ini dikembangkan menggunakan Flask untuk backend dan HTML/CSS untuk *frontend*. Backend Flask mengintegrasikan model terlatih untuk memproses input teks atau CSV dari pengguna. *Frontend* HTML/CSS menyediakan antarmuka interaktif dengan grafik visualisasi hasil analisis sentimen berbasis aspek (ABSA), membantu pengguna memahami data.

3. HASIL DAN PEMBAHASAN

3.1. Pengumpulan Data

Data ulasan aplikasi Access by KAI sejumlah 22.684 entri dikumpulkan dari *Google Play Store* menggunakan *scraping* Python (*google_play_scraper*). Penelitian fokus pada ulasan 7 Juli 2023 hingga 7 Juli 2024 (periode setelah pembaruan aplikasi), dengan hanya kolom *content* yang dianalisis karena memuat teks ulasan pengguna secara langsung.

3.2. Data Preprocessing

Tahap *data preprocessing* membersihkan dan menyiapkan data ulasan untuk analisis lebih lanjut. Proses ini melibatkan enam langkah utama: *case-folding* (mengubah teks menjadi huruf kecil), *cleaning* (menghapus karakter tidak relevan), *normalization* (menstandardisasi penulisan kata), *tokenizing* (memecah teks menjadi unit-unit kecil), *stopword removal* (menghilangkan kata-kata umum yang tidak bermakna), dan *stemming* (mengubah kata ke bentuk dasarnya).

3.3. Topic Modelling

Tabel 3. Pengelompokkan Topik

Label	Topik	Deskripsi	Kategori Utama
Label 1	24, 29	Masalah Pengembalian Dana dan Pemrosesan Dana	<i>Financial Transactions</i>
Label 2	6, 20, 21, 22, 25, 26, 27, 32, 34	Masalah Teknis dan Performa Aplikasi	<i>Technical Issues and Performance</i>
Label 3	18, 28	Antarmuka Pengguna	<i>User Experience and Interface</i>
Label 4	9, 11, 13, 30	Pengalaman dan Kepuasan Pengguna	<i>User Experience and Interface</i>
Label 5	5, 17, 31, 33	Aspek Terkait Pembayaran	<i>Financial Transactions</i>
Label 6	0, 1, 2, 3, 4, 7, 8, 10, 12, 14, 15, 16, 19, 23	Masalah Lain-lain dan Keluhan Umum	<i>Technical Issues and Performance</i>

Setelah mengetahui bahwa setiap label telah dikelompokkan ke dalam kategori utama yaitu *Financial Transactions*, *Technical Issues and Performance*, dan *User Experience and Interface* sesuai dengan Tabel 3, langkah berikutnya adalah menentukan kata kunci yang relevan untuk masing-masing kategori tersebut. Kata kunci tersebut digunakan untuk membuat pelabelan otomatis, sehingga setiap

komentar dapat diklasifikasikan secara tepat berdasarkan kemunculan kata-kata yang merepresentasikan aspek-aspek penting dalam kategori masing-masing, seperti yang ditunjukkan pada Tabel 4.

Tabel 4. Kata Kunci Topik

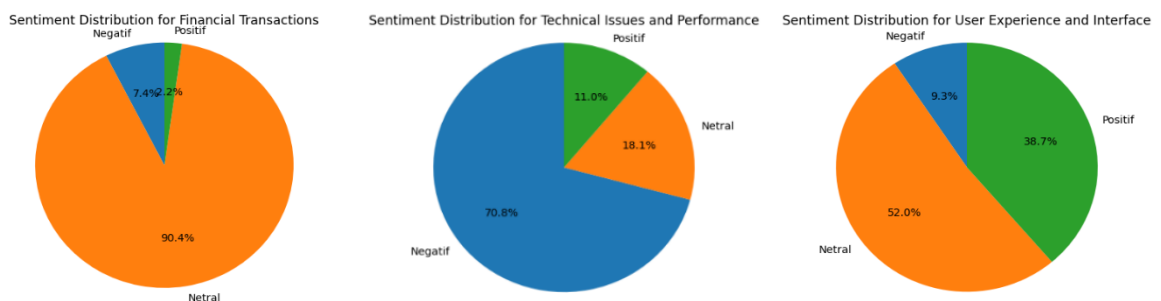
Kategori	Kata Kunci
<i>Financial Transactions</i>	<i>refund</i> , dana, uang, kembali, batal, biaya, cair, <i>pay</i> , <i>kai pay</i> , saldo, bayar, potong, <i>topup</i> , <i>wallet</i> , <i>transfer</i> , qris, ovo, <i>ewallet</i> , metode, transaksi, rekening
<i>Technical Issues and Performance</i>	aplikasi, <i>app</i> , versi, <i>update</i> , <i>upgrade</i> , makin, tambah, lambat, <i>loading</i> , <i>server</i> , <i>error</i> , <i>bug</i> , sering, dulu, sekarang, <i>login</i> , <i>logout</i> , susah, sulit, gagal, masalah, <i>trouble</i> , <i>crash</i> , data, isi, tumpang, <i>member</i> , daftar, <i>booking</i> , pesan, tiket, <i>seat</i> , kursi, <i>available</i> , waktu, jam, jadwal, <i>install</i> , <i>uninstall</i> , download, koneksi, wifi, internet, jaring, performa, kualitas, respon, ribet, jelek, buruk, parah
<i>User Experience and Interface</i>	sangat, bantu, mudah, lancar, mantap, praktis, baik, terima, kasih, guna, nyaman, cepat, pengalaman, puas, senang, manfaat, rekomendasi, tampilan, navigasi, menu, pilih, tiket, pesan
Sentimen Positif	mudah, cepat, rapi, menarik, bagus, stabil, lancar, aman, bantu, keren, mantap, praktis, baik, terima, puas, guna, canggih, sukses, normal, ok, oke, jempol
Sentimen Negatif	lemot, <i>bug</i> , lambat, kecil, <i>error</i> , <i>berat</i> , tidak, <i>crash</i> , hilang, susah, sulit, gagal, buruk, jelek, parah, ribet, bodoh, anjing, tolol, astagfirullah, melulu, sering, payah, ampas, nyebelin, ngecewain, masalah, jelas, kurang, <i>failed</i> , besar

3.4. Labeling Aspect and Sentiment

Dalam menganalisis ulasan pengguna aplikasi Access by KAI, setiap *trigram* dalam teks akan diperiksa untuk mengidentifikasi pasangan aspek dan sentimen (positif/negatif) berdasarkan kata kunci yang ada, mirip dengan pendekatan penambangan opini yang dilakukan oleh M. Hu [15]. Misalnya, *trigram* "sangat membantu untuk" mengidentifikasi sentimen positif terkait aspek *User Experience and Interface*, sementara "aplikasi error terus" akan menguatkan sentimen negatif pada aspek *Technical Issues and Performance*. Setelah semua *trigram* teranalisis, sentimen mayoritas untuk setiap aspek dihitung menggunakan fungsi *Counter*. Ini membantu menentukan sentimen dominan ketika ada beberapa indikator sentimen untuk satu aspek. Hasil akhirnya adalah daftar aspek yang terdeteksi, sentimen dominannya (diberi nilai 1 untuk positif, -1 untuk negatif, atau 0 jika tidak ada sentimen), dan salah satu *trigram* yang mendukung sentimen tersebut sebagai contoh representatif. Proses ini disajikan per kalimat untuk memberikan gambaran lengkap tentang umpan balik pengguna.

3.5. Exploratory Data Analysis (EDA)

1. Pie Chart



Gambar 3. Pie Chart

- Financial Transactions:** Mayoritas ulasan (90.4%) tidak mengandung sentimen jelas. Sentimen negatif (7.4%) lebih dominan dibandingkan positif (2.2%), mengindikasikan potensi masalah pada aspek transaksi keuangan, sebagaimana terlihat pada distribusi sentimen di Gambar 3.
- Technical Issues and Performance:** Seperti yang ditampilkan pada Gambar 3, aspek ini didominasi sentimen negatif (70.8%), menunjukkan banyaknya keluhan terkait performa dan masalah teknis.

Sentimen tidak bersentimen 18.1% dan positif hanya 11.0%, mengindikasikan pengalaman pengguna yang kurang memuaskan.

3. **User Experience and Interface:** Seperti yang ditampilkan pada Gambar 3, sebagian besar ulasan tidak bersentimen (52.0%). Namun, sentimen positif cukup tinggi (38.7%) dibandingkan negatif (9.3%), menunjukkan penerimaan yang baik terhadap pengalaman pengguna dan tampilan antarmuka aplikasi.

3.6. Pembobotan TF-IDF

Proses mengubah teks menjadi format numerik dimulai dengan menginisialisasi *TfidfVectorizer* dari pustaka *scikit-learn*. Alat ini bertugas menghitung skor TF-IDF, yang menunjukkan seberapa penting sebuah kata dalam suatu dokumen relatif terhadap keseluruhan koleksi dokumen. Selanjutnya, seluruh data teks akan diubah menjadi matriks TF-IDF, di mana setiap baris merepresentasikan satu dokumen (misalnya, sebuah komentar), dan setiap kolom mewakili kata unik dengan nilainya berupa skor TF-IDF.

3.7. Pembagian Data

Proses pembagian *train* dan *split* dilakukan dengan pustaka *Scikit-Learn* menggunakan fungsi *train_test_split*, yang memungkinkan pemisahan data menjadi *training data* dan *testing data*. Penggunaan parameter *random_state* berguna untuk memasukkan ke dalam data latih dan uji secara acak namun terkontrol.

3.8. Pelatihan Model

```
smote = SMOTE(random_state=42)
X_train_resampled, y_train_resampled = smote.fit_
resample(X_train, y_train)
```

Gambar 4. SMOTE

Model SVM dilatih dan dievaluasi dalam dua skenario utama, dibedakan pada penerapan SMOTE. Seperti yang ditunjukkan pada Gambar 4, skenario dengan SMOTE, data latih awalnya diseimbangkan (*X_train_resampled*, *y_train_resampled*) sebelum model SVM diinisialisasi dengan *kernel* ('linear', 'rbf', atau 'poly') dan dilatih. Sebaliknya, skenario tanpa SMOTE melatih model SVM secara langsung dengan data latih TF-IDF yang belum diseimbangkan. Untuk kedua skenario, model kemudian memprediksi data uji (*X_test*), dan performanya dievaluasi menggunakan *classification_report* serta *accuracy_score*. Pembagian data latih dan uji (80:20 atau 70:30) diatur via parameter *test_size* pada *train_test_split*, dan semua model yang telah dilatih disimpan menggunakan *library* *joblib*.

3.9. Evaluasi Model

Tabel 5. Akurasi Model

<i>Data Handling</i>	<i>Kernel</i>	<i>Split Data</i>	<i>Financial Transactions</i>	<i>Technical Issues and Performance</i>	<i>User Experience and Interface</i>	<i>Average Accuracy</i>
Non-SMOTE	Linear	80:20	0.9230	0.8691	0.8825	0.8915
		70:30	0.9243	0.8718	0.8800	0.8920
	RBF	80:20	0.9262	0.8529	0.8774	0.8855
		70:30	0.9229	0.8631	0.8754	0.8871
	Polynom	80:20	0.9103	0.7379	0.7552	0.8011
		70:30	0.9113	0.7411	0.7515	0.8013
SMOTE	Linear	80:20	0.9074	0.8225	0.8463	0.8587
		70:30	0.9096	0.8200	0.8510	0.8602
	RBF	80:20	0.9262	0.8525	0.8749	0.8845
		70:30	0.9270	0.8588	0.8754	0.8870
	Polynom	80:20	0.9197	0.7820	0.8044	0.8353
		70:30	0.9197	0.7886	0.8026	0.8369

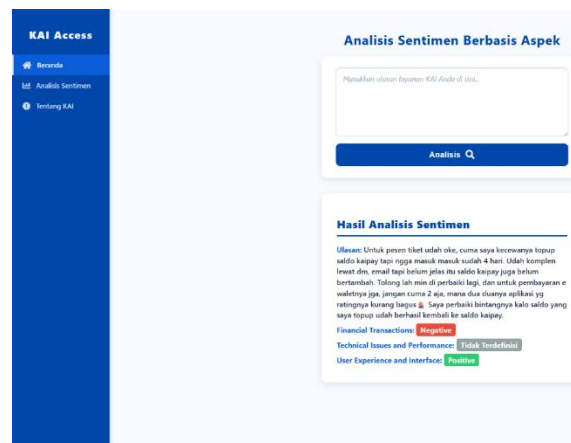
Pengujian model SVM pada Tabel 5. bahwa *kernel* RBF dengan SMOTE memberikan kinerja terbaik secara keseluruhan, terutama pada aspek *Financial Transactions* dengan akurasi 0.9270 (*split data* 70:30). Ini menegaskan efektivitas SMOTE untuk kategori tersebut. Sementara itu, *kernel Linear* tanpa SMOTE unggul pada aspek *Technical Issues and Performance* (akurasi 0.8718 pada *split data* 70:30) dan *User Experience and Interface* (akurasi 0.8825 pada *split data* 80:20).

Selain itu, analisis *Average Accuracy* yang mengagregasikan kinerja model di seluruh aspek *Financial Transactions*, *Technical Issues and Performance*, dan *User Experience and Interface*, memberikan gambaran komprehensif tentang efektivitas setiap konfigurasi. Berdasarkan rata-rata akurasi, konfigurasi *Kernel Linear* tanpa SMOTE dengan *split data* 70:30 menunjukkan performa terbaik secara keseluruhan dengan *Average Accuracy* 0.8920. Meskipun *Kernel RBF* dengan SMOTE menunjukkan akurasi tertinggi pada *Financial Transactions* secara spesifik, *Kernel Linear* tanpa SMOTE memberikan konsistensi kinerja yang kuat di berbagai aspek.

Sebaliknya, *kernel Polynomial* menunjukkan performa yang lebih rendah pada semua metrik, mengindikasikan kompleksitasnya tidak memberikan nilai tambah signifikan. Berdasarkan hasil ini, untuk *deployment model*, *kernel RBF* dengan SMOTE akan tetap digunakan secara spesifik untuk aspek *Financial Transactions* karena akurasinya yang superior pada kategori tersebut, sedangkan *kernel Linear* tanpa SMOTE akan diterapkan untuk aspek *Technical Issues and Performance* dan *User Experience and Interface* mengingat kinerjanya yang solid dan rata-rata akurasi keseluruhan yang tinggi.

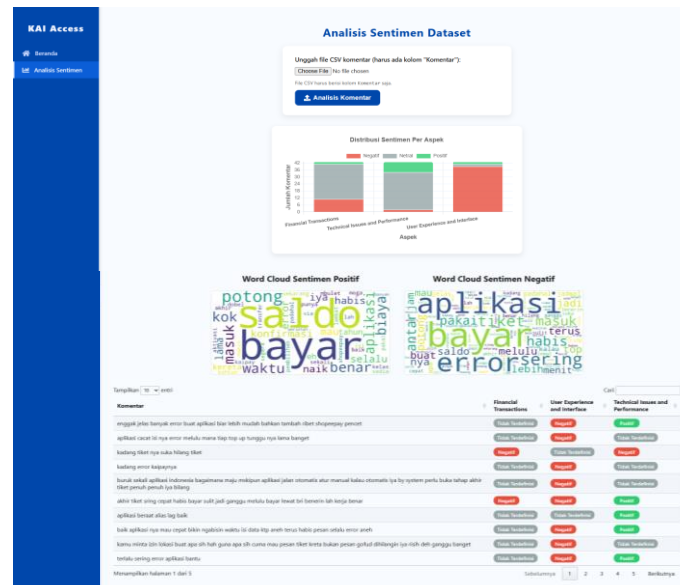
Perbedaan hasil antar model terjadi karena setiap aspek memiliki karakteristik data yang berbeda. Pada aspek *Financial Transactions*, variasi keluhan pengguna lebih beragam sehingga pola datanya tidak *linear*, membuat *kernel RBF* yang mampu menangani pola kompleks bekerja lebih baik, apalagi setelah dibantu SMOTE untuk menyeimbangkan jumlah kelas. Sementara itu, pada aspek *Technical Issues and Performance* serta *User Experience and Interface*, jenis kata yang digunakan pengguna cenderung lebih konsisten dan memiliki pola yang lebih mudah dipisahkan secara *linear*, sehingga *Kernel Linear* tanpa SMOTE memberikan hasil yang lebih stabil. Perbedaan inilah yang membuat setiap aspek memiliki konfigurasi model terbaik yang berbeda.

3.10. Model Deployment



Gambar 5. Tampilan Halaman Indeks

Gambar 5. menunjukkan halaman utama *dashboard* dengan kotak teks "Masukkan ulasan layanan KAI Anda di sini..." untuk *input* ulasan manual dan tombol "Analisis" untuk klasifikasi. Bagian "Hasil Analisis Sentimen" menampilkan analisis per aspek: contoh ulasan menunjukkan "*Financial Transactions*" **Negatif** (merah), "*Technical Issues and Performance*" **Tidak Terdefinisi** (abu-abu), dan "*User Experience and Interface*" **Positif** (hijau).



Gambar 6. Tampilan Halaman Submit File

Gambar 6. menunjukkan halaman "Analisis Sentimen *Dataset*", di mana pengguna dapat mengunggah *file* CSV untuk dianalisis. Halaman ini memvisualisasikan hasil analisis melalui tiga komponen utama: sebuah *bar chart* yang menampilkan jumlah sentimen positif, negatif, dan tidak terdefinisi untuk "*Financial Transactions*," "*Technical Issues and Performance*," dan "*User Experience and Interface*"; dua *wordcloud* yang menyoroti kata-kata dominan; serta sebuah tabel "Komentar" yang menyajikan komentar asli pengguna dengan *tag* sentimen teridentifikasi per aspek.

4. KESIMPULAN

Hasil penelitian menunjukkan bahwa perbedaan karakteristik linguistik pada tiap aspek menghasilkan konfigurasi model terbaik yang berbeda, di mana *kernel RBF* dengan *SMOTE* memberikan performa tertinggi pada aspek *Financial Transactions* dengan akurasi 0.9270 (*split* 70:30) karena mampu menangani pola data yang lebih kompleks dan non-linear, sedangkan *kernel Linear* tanpa *SMOTE* lebih unggul pada aspek *Technical Issues and Performance* (akurasi 0.8718, *split* 70:30) dan *User Experience and Interface* (akurasi 0.8825, *split* 80:20) yang memiliki pola kosakata lebih linear dan stabil. Temuan ini mengindikasikan bahwa analisis sentimen berbasis aspek sebaiknya dirancang secara modular, dengan model yang dipilih berdasarkan struktur linguistik masing-masing kategori agar hasil pemetaan sentimen lebih akurat dan dapat digunakan secara efektif untuk evaluasi kualitas layanan pada aplikasi *Access by KAI*. Untuk penelitian selanjutnya, penggunaan model *transformer* seperti *BERT*, *IndoBERT*, atau *RoBERTa-ID* berpotensi meningkatkan kualitas representasi teks dibandingkan pendekatan *TF-IDF* + *SVM*, sementara perluasan *dataset* atau lintas platform dapat meningkatkan generalisasi model serta memungkinkan eksplorasi metode lanjutan seperti *fine-grained sentiment analysis* atau *multitask learning* untuk menggabungkan analisis aspek dan sentimen secara simultan.

DAFTAR PUSTAKA

- [1] B. Susarianto, "Analisis Peran Kepemimpinan Digital Dalam Transformasi Digital Di Sektor Publik," *Jutisi J. Ilm. Tek. Inform. dan Sist. Inf.*, pp. 1530–1537, 2024, [Online]. Available: <http://ojs.stmik-banjarbaru.ac.id/index.php/jutisi/article/view/2227%0Ahttp://ojs.stmik-banjarbaru.ac.id/index.php/jutisi/article/download/2227/1170>
- [2] B. Brahimi, M. Touahria, and A. Tari, "Improving sentiment analysis in Arabic: A combined approach," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 33, no. 10, pp. 1242–1250, 2021, doi: 10.1016/j.jksuci.2019.07.011.
- [3] R. A. Rahman, V. H. Pranatawijaya, and N. N. K. Sari, "Analisis Sentimen Berbasis Aspek pada Ulasan Aplikasi Gojek," *KONSTELASI Konvergensi Teknol. dan Sist. Inf.*, vol. 4, no. 1, 2024, doi: 10.24002/konstelasi.v4i1.8922.
- [4] I. Monika Parapat and M. Tanzil Furqon, "Penerapan Metode Support Vector Machine (SVM)

- Pada Klasifikasi Penyimpangan Tumbuh Kembang Anak,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 10, pp. 3163–3169, 2018, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [5] F. M. Madjid, Ratnawati Eka D, and D. Rahayudi, “Sentimental Analysis on Product Reviews Using Support Vector Machine and Nave Bayes,” *Appl. Comput. Eng.*, vol. 2, no. 1, pp. 66–72, 2023, doi: 10.54254/2755-2721/2/20220586.
- [6] H. Mustakim and S. Priyanta, “Aspect-Based Sentiment Analysis of KAI Access Reviews Using NBC and SVM,” *IJCCS (Indonesian J. Comput. Cybern. Syst.*, vol. 16, no. 2, p. 113, 2022, doi: 10.22146/ijccs.68903.
- [7] F. S. Jumeilah, “Penerapan Support Vector Machine (SVM) untuk Pengkategorian Penelitian,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 1, no. 1, pp. 19–25, 2017, doi: 10.29207/resti.v1i1.11.
- [8] A. Kusuma, E. Ermatita, and H. N. Irmanda, “Analisis Sentimen Pada Ulasan Aplikasi Indodax di Google Play Store Menggunakan Metode Support Vector Machine,” *Pros. Semin. Nas. Mhs. Bid. Ilmu Komput. dan Apl.*, vol. 3, no. 2, pp. 773–782, 2022.
- [9] A. A. Aliero, B. S. Adebayo, H. O. Aliyu, A. G. Tafida, B. U. Kangiwa, and N. M. Dankolo, “Systematic Review on Text Normalization Techniques and its Approach to Non-Standard Words,” *Int. J. Comput. Appl.*, vol. 185, no. 33, pp. 44–55, 2023, doi: 10.5120/ijca2023923106.
- [10] T. Setiawan, S. Liem, and D. M. R. Pribadi, “Perbandingan Algoritma SVM dan Naïve Bayes dalam Analisis Sentimen Komentar Tiktok pada Produk Skincare,” *Appl. Inf. Technol. Comput. Sci.*, vol. 3, no. 2, pp. 28–32, 2024, [Online]. Available: <https://jurnal.politap.ac.id/index.php/aicoms>
- [11] A. F. Rochim, K. Widyaningrum, and D. Eridani, “Comparison of Kernels Function between of Linear, Radial Base and Polynomial of Support Vector Machine Method Towards COVID-19 Sentiment Analysis,” pp. 224–228, 2021.
- [12] Yeni Anistyasari and Eko Hariadi, “Algoritma baru pembentukan kata dasar,” *Pros. SNRT (Seminar Nas. Ris. Ter.*, vol. 5662, no. November, pp. 70–76, 2019.
- [13] I. K. T. Mertayasa and I. D. M. B. A. Darmawan, “Pemodelan Topik Pada Ulasan Hotel Menggunakan Metode BERTopic Dengan Prosedur c-TF-IDF,” *J. Nas. Teknol. Inf. dan Apl.*, vol. 1, no. 1, pp. 307–316, 2022.
- [14] D. Septiani and I. Isabela, “Analisis Term Frequency Inverse Document Frequency (TF-IDF) Dalam Temu Kembali Informasi Pada Dokumen Teks,” *SINTESIA J. Sist. dan Teknol. Inf. Indones.*, vol. 1, no. 2, pp. 81–88, 2023.
- [15] M. Hu and B. Liu, “Mining and summarizing customer reviews,” *KDD-2004 - Proc. Tenth ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, pp. 168–177, 2004, doi: 10.1145/1014052.1014073.