

Analisis Sentimen Ulasan Pengguna Aplikasi KitaLulus pada Google Play Store dengan menggunakan algoritma *Support Vector Machine (SVM)*

Sentiment Analysis of User Reviews of the KitaLulus Application on Google Play Store using the Support Vector Machine (SVM) Algorithm

¹Ahmad Syaifudin Agil Rafsanjani, ²Diana Laily Fithri, ³Supriyono

^{1,2,3}Program Studi Sistem Informasi, Fakultas Teknik, Universitas Muria Kudus

^{1,2,3}Jl. Lingkar Utara, Bae, Kota Kudus, Jawa Tengah, Indonesia

*e-mail: 202153155@std.umk.ac.id, diana.laily@umk.ac.id, supriyono.si@umk.ac.id

(received: 11 July 2025, revised: 15 July 2025, accepted: 16 July 2025)

Abstrak

Perkembangan teknologi digital mendorong meningkatnya penggunaan aplikasi pencarian kerja seperti KitaLulus. Ulasan pengguna di Google Play Store menjadi sumber penting untuk mengevaluasi kualitas layanan dan tingkat kepuasan pengguna. Tujuan dari penelitian ini untuk menganalisis sentimen ulasan pengguna terhadap aplikasi KitaLulus menggunakan algoritma Support Vector Machine (SVM) yang dikombinasikan dengan teknik Synthetic Minority Over-sampling Technique (SMOTE) guna mengatasi ketidakseimbangan data sentimen. Proses penelitian meliputi pengumpulan 1.000 ulasan melalui web scraping, pra-pemrosesan teks, pelabelan berbasis rating, transformasi data dengan TF-IDF, pembagian data menjadi 80% latih dan 20% uji, penerapan SMOTE, pelatihan model SVM, serta evaluasi kinerja model. Hasil menunjukkan bahwa SVM yang dilatih dengan data seimbang melalui SMOTE menghasilkan akurasi 89%, presisi 90%, recall 89%, F1-score 90%, dan AUC 0,93. Penelitian ini berkontribusi dalam memberikan implementasi praktis kombinasi SVM dan SMOTE yang terbukti efektif untuk klasifikasi sentimen berbasis teks, khususnya pada data ulasan aplikasi yang tidak seimbang.

Kata kunci: analisis sentimen, kitalulus, *support vector machine*, SMOTE, TF-IDF

Abstract

The advancement of digital technology has driven the increasing use of job search applications such as KitaLulus. User reviews on the Google Play Store serve as a crucial source for evaluating service quality and user satisfaction. This study aims to analyze user sentiment toward the KitaLulus application using the Support Vector Machine (SVM) algorithm, combined with the Synthetic Minority Over-sampling Technique (SMOTE) to address class imbalance in sentiment data. The research process includes collecting 1,000 user reviews through web scraping, text preprocessing, rating-based labeling, data transformation using TF-IDF, splitting the dataset into 80% training and 20% testing, applying SMOTE, training the SVM model, and evaluating its performance. The results show that SVM trained with SMOTE-balanced data achieved an accuracy of 89%, precision of 90%, recall of 89%, F1-score of 90%, and an AUC of 0.93. This study contributes a practical implementation of the SVM-SMOTE combination, demonstrating its effectiveness in text-based sentiment classification, particularly in handling imbalanced review data from mobile applications.

Keywords: sentiment analysis, kitalulus, *support vector machine*, SMOTE, TF-IDF.

1 Pendahuluan

Perkembangan teknologi digital telah mendorong peningkatan penggunaan aplikasi pencarian kerja berbasis mobile. Salah satu dari aplikasi populer di Indonesia adalah KitaLulus, yang memberikan informasi lowongan pekerjaan, fitur pembuatan curriculum vitae (CV), serta pelatihan wawancara. Meningkatnya penggunaan aplikasi ini juga diiringi dengan banyaknya ulasan dari

<http://sistemasi.ftik.unisi.ac.id>

pengguna yang mencerminkan pengalaman, kepuasan, dan keluhan mereka. Ulasan tersebut menjadi sumber data yang penting dalam mengevaluasi kualitas layanan dan pengalaman pengguna terhadap aplikasi KitaLulus [1].

Dalam konteks analisis ulasan, teknik analisis sentimen digunakan untuk mengelompokkan opini menjadi sentimen positif atau negatif secara otomatis. Namun, banyak penelitian sebelumnya menghadapi tantangan berupa ketidakseimbangan data, di mana ulasan positif mendominasi jumlah data negatif. Ketidakseimbangan ini berpotensi menurunkan performa model klasifikasi karena model menjadi bias terhadap kelas mayoritas. Beberapa studi sebelumnya telah menggunakan algoritma seperti Naïve Bayes, tetapi hasil klasifikasi pada data tidak seimbang menunjukkan performa yang kurang optimal [1]. Untuk mengatasi masalah ini, penelitian ini menggabungkan algoritma Support Vector Machine (SVM), yang dikenal efektif dalam klasifikasi berbasis margin, dengan teknik Synthetic Minority Over-sampling Technique (SMOTE), yang mampu menyeimbangkan distribusi kelas melalui pembuatan data sintetis [2][3].

Penelitian ini memberikan kontribusi pada penerapan metode SVM dan SMOTE secara terintegrasi dalam klasifikasi sentimen ulasan pengguna aplikasi KitaLulus, yang belum banyak dibahas dalam literatur sebelumnya. Selain itu, penelitian ini juga mengadopsi tahapan pra-pemrosesan teks yang lengkap, meliputi normalisasi kata tidak baku, stemming, dan TF-IDF. Evaluasi performa model dilakukan secara menyeluruh menggunakan metrik akurasi, presisi, recall, F1-score, dan AUC untuk memastikan kualitas klasifikasi yang dihasilkan. Inovasi penelitian ini terletak pada pemanfaatan teknik balancing berbasis SMOTE untuk meningkatkan akurasi klasifikasi model SVM terhadap data ulasan pengguna yang tidak seimbang pada ulasan aplikasi kitalulus.

Penelitian ini bertujuan untuk melakukan analisis sentimen terhadap ulasan yang diberikan oleh pengguna aplikasi KitaLulus di Google Play Store menggunakan Support Vector Machine (SVM) yang dikombinasikan dengan SMOTE, serta mengevaluasi performa model berdasarkan metrik klasifikasi supaya bisa memberikan hasil prediksi yang akurat meskipun pada data yang tidak seimbang.

2 Tinjauan Literatur

Sentiment analysis (SA) merupakan teknik yang terus mengalami perkembangan, yang bertujuan untuk menggali opini, perasaan, serta aspek subjektif dari suatu teks. Metode ini umum dimanfaatkan dalam mengevaluasi pandangan, penghargaan, sikap, maupun emosi dari setiap individu terhadap sebuah entitas, maupun karakteristiknya [4].

Support Vector Machines dan teknik-teknik lainnya telah digunakan dalam sejumlah studi sebelumnya yang meneliti analisis sentimen. Menurut salah satu studi oleh Dina *et al.*, dari September hingga November 2024, Metode scraping web digunakan untuk mengumpulkan 1.000 ulasan pengguna aplikasi KitaLulus di Google Play Store, dan sentimen dari ulasan-ulasan ini dianalisis menggunakan algoritma Naive Bayes. Sebelum data dikategorikan dalam tiga sentimen yaitu positif, netral, dan negatif data tersebut melalui langkah-langkah pra-pemrosesan. Model mendapatkan nilai *accuracy* sebesar 88%, *precision* 87%, *recall* 88%, dan skor F1 85%. Temuan menunjukkan bahwa meskipun masih ada keluhan tentang fungsionalitas dan kinerja program, sebagian besar ulasan adalah positif [1].

Tujuan selanjutnya dari penelitian Idris *et al.* adalah untuk mengelompokkan ulasan dari pengguna aplikasi Shopee ke dua kategori sentimen positif dan negatif dan menilai seberapa baik teknik klasifikasi tersebut bekerja. Melalui web scraping, 3.000 ulasan secara keseluruhan dikumpulkan. Dengan akurasi 98% dan skor F1 sebesar 0,98, hasilnya menunjukkan kinerja luar biasa dari algoritma SVM, yang menunjukkan efektivitasnya dalam klasifikasi opini berbasis teks [5].

Kemudian, dengan memanfaatkan 2.000 data ulasan algoritma yang dipakai adalah *Naïve Bayes* dan *Support Vector Machine* (SVM), penelitian Dimas *et al* menggunakan analisis sentimen untuk membagi opini pengguna menjadi dua sentimen: positif dan negatif. Menurut hasil pengelompokan, 23,3% dari ulasan adalah negatif dan 76,7% adalah positif. Dibandingkan dengan Naïve Bayes, yang mencapai akurasi 79%, SVM lebih baik dengan 84%, menunjukkan efektivitas SVM dalam klasifikasi teks ulasan pengguna [6].

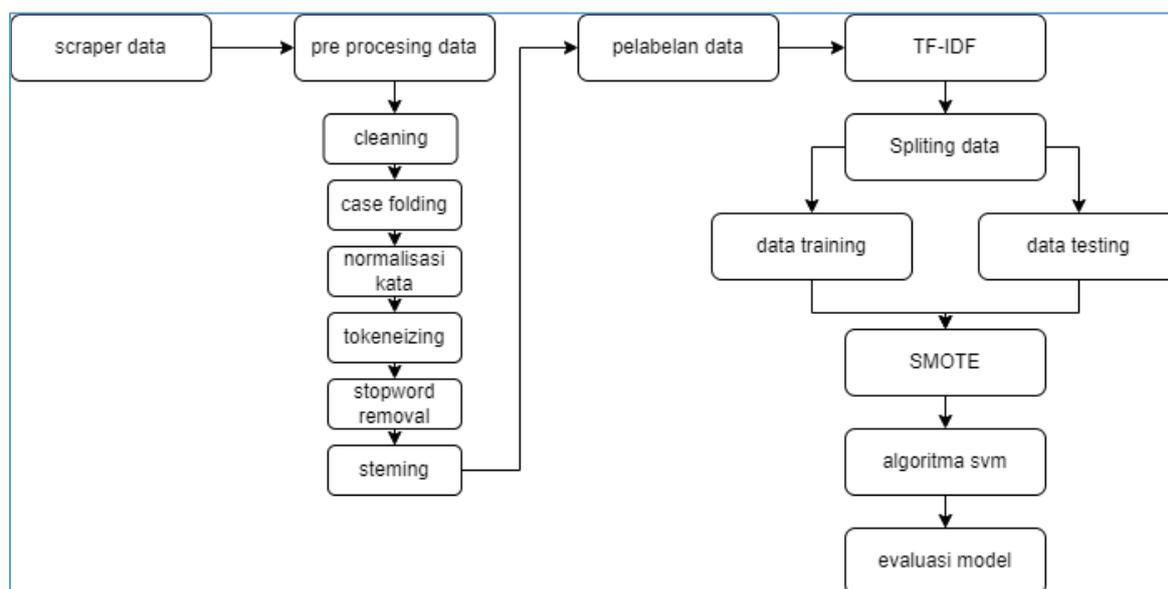
Terakhir adalah studi terbaru oleh Suryono dan Eskiyaturofikoh. 4.087 data ulasan 70% data pelatihan dan 30% data pengujian dianalisis dalam studi ini. Sebelum penggunaan SMOTE, SVM dan

Naïve Bayes masing-masing memiliki tingkat akurasi sebesar 75,5% dan 75%. Akurasi SVM meningkat menjadi 81% setelah SMOTE digunakan untuk mengatasi ketidakseimbangan kelas, namun akurasi Naïve Bayes hanya meningkat sedikit menjadi 75,5%. Temuan ini menunjukkan seberapa baik SMOTE bekerja untuk meningkatkan kinerja klasifikasi, terutama ketika diterapkan pada data sentimen yang tidak seimbang. Signifikansi analisis sentimen sebagai dasar untuk pengambilan keputusan strategis para pengembang aplikasi juga ditekankan oleh studi ini [3], dalam penelitian tersebut menunjukkan bahwa algoritma *Support Vector Machine* sangat baik untuk menganalisis sentimen.

Berdasarkan studi yang telah dikaji, algoritma SVM dikombinasikan dengan SMOTE sangat kuat dalam menganalisis data sehingga peneliti memilih algoritma tersebut untuk menganalisis aplikasi KitaLulus untuk mengatasi ketidakseimbangan kelas dalam data sentimen. Sebagian besar penelitian sebelumnya hanya menerapkan algoritma seperti Naïve Bayes tanpa penanganan khusus terhadap data yang tidak seimbang. Oleh sebab itu, tujuan dari penelitian ini untuk mengisi celah tersebut dengan mengimplementasikan integrasi SVM dan SMOTE secara spesifik pada data ulasan KitaLulus, serta memberikan bukti empiris terhadap peningkatan kinerja klasifikasi sentimen yang dihasilkan. Hasil penelitian ini diharapkan dapat menjadi rujukan metodologis bagi studi sejenis dalam konteks aplikasi digital lainnya yang menghadapi permasalahan serupa.

3 Metode Penelitian

Dalam menulis penelitian ini peneliti membuat alur penelitian, ada beberapa alur penelitian, yang pertama ada scraping, preprocessing, pelabelan data, TF-IDF, splitting dataset, SMOTE dilanjut melatih model dan terakhir melakukan evaluasi proses penelitian analisis sentimen dilampirkan pada Gambar 1 dibawah.



Gambar 1 Proses analisis sentimen

3.1. Scrapper data

Web scraping merupakan teknik otomatis guna mendapatkan sebuah informasi dari situs dengan mengekstraksi data yang tersedia di dalamnya [7]. dalam kasus ini data ulasan di aplikasi kitalulus di play store

3.2. Pre processing data

Preprocessing berfungsi untuk menyusun ulang data mentah yang belum terstruktur menjadi data yang lebih terorganisir dan dapat digunakan pada tahap-tahap berikutnya ada beberapa tahap dalam pre processing.

1. **Cleaning:** Selama fase cleaning, teks bebas dari komponen yang mengganggu seperti tag HTML, tautan, dan simbol-simbol berlebihan yang tidak ada hubungannya dengan konten utama dokumen.

2. **Case folding:** Pada *case folding* huruf yang asalnya adalah huruf kapital di ubah menjadi huruf kecil seperti kata "Senang" dan "senang" dianggap sama setelah case folding
3. **Normalisasi kata:** Normalisasi merupakan langkah untuk membenahi kata-kata yang tidak baku, kesalahan ejaan, serta bentuk singkatan dengan mencocokkannya pada kamus. Jika ditemukan kecocokan, kata tersebut diganti dengan bentuk yang benar dan sesuai makna aslinya.
4. **Tokenizing:** Pada proses tokenisasi, kalimat diuraikan menjadi elemen-elemen kata terpisah guna memudahkan analisis lebih lanjut.
5. **Stopword removal:** Dalam prosedur penghapusan *stopword*, kata-kata umum dan tidak perlu dari analisis seperti konjungsi dan artikel dihapus dari teks.
6. **Stemming:** Proses *stemming* berfungsi untuk mengkonversi kata-kata berimbuhan ke bentuk dasar katanya dengan menghilangkan awalan, akhiran, atau sisipan seperti "berlari", "lari-lari" menjadi "lari".

Itu adalah tahapan dalam pre processing dalam analisis sentimen [8]

3.3. Pelabelan data

Pelabelan yaitu Prosedur yang dilakukan untuk mengklasifikasi data ke kelas sentimen, seperti positif ataukah negatif, dalam analisis sentimen sehingga model pembelajaran mesin dapat mengidentifikasi dan mengelompokkan data ke dalam kategori tersebut. Pelabelan data dapat dilakukan dengan berbagai cara, termasuk manual, berbasis rating, dan berbasis lexicon [9] tapi dalam penelitian ini peneliti mengimplementasikan kelas positif dan negatif saja tidak menambahkan kelas netral karena klasifikasi dua kelas cenderung lebih mudah dan model lebih stabil karena tidak ada ambiguitas kelas netral dan juga menggunakan rating based yaitu mengelompokan sentimen sesuai rating yang telah diperoleh dari *scraping data*

3.4. Word cloud

WordCloud salah satu visualisasi dalam mengungkap pola serta tren di dalam volume teks yang besar. Selain itu, *Word Cloud* dapat dimanfaatkan untuk menggambarkan persebaran sentimen dalam suatu teks dengan menampilkan kata-kata yang memiliki makna positif maupun negatif secara mencolok [10]

3.5. TF-IDF

Metode (TF-IDF) digunakan pada pengembangan model *natural language* agar dapat mengetahui Seberapa penting sebuah kata muncul dalam dokumen tertentu dibandingkan dengan kemunculannya di seluruh dokumen lainnya.

1. **Term Frequency** TF adalah metode untuk menghitung frekuensi dari kata yang muncul dalam satu dokumen. Nilainya didapat dari jumlah kemunculan kata dibagi dengan total seluruh kata. Dalam kondisi tertentu, TF dapat dimodifikasi dengan sistem pembobotan yang lebih rumit untuk hasil yang lebih akurat.

$$tf(t, d) = \frac{\text{jumlah kemunculan kata } t \text{ dalam dokumen } d}{\text{jumlah total kata dalam dokumen } d} \quad (1)$$

2. **Inverse Document Frequency** IDF adalah metode yang mengukur tingkat kepentingan dari suatu kata berdasarkan distribusinya dalam sekumpulan dokumen. Kata yang frekuensinya rendah dalam keseluruhan dokumen biasanya memiliki nilai IDF yang tinggi. Untuk memodifikasi skala, logaritma diambil setelah membagi total jumlah dokumen dengan jumlah dokumen yang mengandung kata tersebut.

$$idf(t) = \log \frac{\text{total jumlah dokumen dalam dokumen } d}{\text{jumlah dokumen dimana kata } t \text{ muncul}} \quad (2)$$

Proses selanjutnya yaitu menghitung bobot Term Frequency-Inverse Document Frequency (TF-IDF), yang didapatkan dari hasil perkalian TF dan IDF [11], sesuai rumus yang tercantum di bawah

$$tf - idf(t, d) = tf \times idf \quad (3)$$

3.6. Splitting dataset

Dalam proses pemisahan dataset, dataset dipisahkan kedalam dua subset dataset, yakni dataset untuk latih dan data untuk uji. Penelitian ini menerapkan skema pembagian 80:20, 80% data dimanfaatkan sebagai pelatihan model, sementara 20% sisanya dialokasikan untuk menguji kinerja model pada data baru.

Proporsi ini dipilih karena dianggap mampu memberikan keseimbangan antara kemampuan model dalam mempelajari pola dari data yang cukup besar serta mengukur generalisasi model dengan

data uji yang memadai. Berdasarkan studi yang dilakukan oleh Prasetyo et al., pendekatan ini terbukti efektif dalam meningkatkan performa SVM, khususnya dalam konteks klasifikasi data berbasis teks [12].

3.7. SMOTE

SMOTE (*Synthetic Minority Over-sampling Technique*) adalah suatu teknik peningkatan data yang berasal dari metode oversampling. Prosesnya melibatkan pembuatan data baru berdasarkan sampel kelas minoritas agar distribusi data antar kelas menjadi lebih seimbang [13].

3.8. Support Vector Machine (SVM)

SVM unggul dalam mengklasifikasikan sentimen negatif dan juga sentimen positif serta mampu memberikan hasil akurasi tinggi, khususnya saat data dapat dipisahkan secara linear. Model ini membentuk garis batas antar kelas dengan margin terluas dari data yang paling dekat, Inti dari SVM adalah mentransformasi data ke dimensi yang lebih tinggi untuk menemukan pemisah yang optimal, menggunakan teknik non-linear untuk menemukan garis pemisah linear terbaik. Penelitian ini menggunakan kernel linear [14]

Kemudian akan dilanjutkan dengan tahap evaluasi dalam analisis sentimen tahapan evaluasi berperan penting dalam mengukur performa model klasifikasi sentimen. Proses ini diterapkan untuk menguji sejauh mana model dapat mengidentifikasi sentimen yang tepat baik positif maupun negatif terhadap data uji yang belum dikenali sebelumnya, ada beberapa tahap dalam evaluasi [15].

1. Accuracy

Accuracy adalah suatu prediksi yang tepat dari seluruh jumlah prediksi yang telah dijalankan.

$$accuracy = \frac{TP+TN}{TP+FP+TN+FN} \quad (4)$$

2. Precision

merupakan rorsori prediksi dari kata positif yang benar-benar positif.

$$precision = \frac{TP}{TP+FP} \quad (5)$$

3. Recall

Fungsi recall mengukur sebaik apa model dalam menemukan seluruh data yang sebenarnya positif, dengan cara membandingkan dalam kisaran jumlah dari prediksi positif yang benar dengan jumlah total data positif.

$$recall = \frac{TP}{TP+FN} \quad (6)$$

4. F1-Score

F1-score digunakan sebagai indikator performa yang memadukan precision dan recall secara seimbang melalui rata-rata harmonik, sehingga memberikan penilaian menyeluruh terhadap performa model.

$$F1 = 2 \times \frac{precision \times recall}{precision + recall} \quad (7)$$

5. Confusion matrix

Confusion Matrix dipilih sebagai evaluasi, di mana hasil prediksi model dibandingkan dengan kelas sebenarnya. Matriks ini menyajikan informasi mengenai akurasi model berdasarkan kecocokan antara label prediksi dan data aktual setelah proses klasifikasi selesai [16]

6. AUC

AUC atau *Area Under Curve* merupakan salah satu metrik evaluasi yang dipakai sebagai acuan untuk menghitung performa dari suatu model klasifikasi dalam mengidentifikasi perbedaan antara positif dan negatif dalam kelas. Nilai dari AUC berada dari 0 hingga 1, dan jika semakin dekat dengan 1, maka semakin baik kemampuan model dalam melakukan prediksi, nilai AUC [17], Untuk mengetahui sejauh mana model mampu membedakan antara dua kategori sentimen, digunakan pendekatan melalui Kurva ROC dan nilai AUC sebagai indikator evaluatif, nilai interpretasi dari AUC bisa dilihat pada Tabel 1 di bawah.

Tabel 1 Nilai AUC

	Nilai AUC	Penjelasan
1.	0.91-1.00	Sangat Baik
2.	0.81-0.90	Baik
3.	0.71-0.80	Cukup
4.	0.60-0.70	Kurang
5.	<0.60	Gagal

4 Hasil dan Pembahasan

Setelah peneliti melakukan penelitian dan analisa dari model terkait analisis sentimen terhadap aplikasi kitalulus, dalam penelitan yang memiliki beberapa tahapan seperti scraping data,preprocessing data,pelabelan data,TF-IDF,spliting data,SMOTE,pelatihan model serta evaluasi, setelah melakukan tahapan tersebut peneliti akan menjelaskan hasil dari analisa yang telah dijalankan.

4.1. Scraping data

Setalah Peneliti mengumpulkan beberapa ulasan data dalam analisis sentimen ini diperoleh melalui proses scraping data dari play store yang memanfaatkan pustaka *google-play-scraper* berbasis Python. Teknik ini digunakan untuk mengekstraksi data dari review di aplikasi "Kita Lulus" yang beraada di Google Play Store.

Pengambilan dataset difokuskan pada ulasan yang dipublikasikan antara September 2024 hingga November 2024. Dalam rentang waktu tersebut, berhasil dikumpulkan sebanyak sekitar 1.000 ulasan. Informasi yang diambil meliputi teks ulasan, tanggal ulasan, serta ratingpengguna. Data selanjutnya diterapkan sebagai bahan dalam proses analisis sentimen data data *scraping* bisa dilihat pada Gambar 2 berikut.

	Review ID	Username	Rating	Review Text	Date
0	9d6106ec-e053-4523-a109-246831bd5152	Pengguna Google	1	Apk cacat, daftar nya terlalu ribet. Kalo penga...	10/28/2024 10:10
1	d6bc2255-5fa0-4096-8a9a-de08762cbd20	Pengguna Google	5	bagus	10/28/2024 8:45
2	5e1e546d-76dc-4ffb-88b5-e93453919a8b	Pengguna Google	5	aplikasi nya mantap, sangat membantu para rekr...	10/28/2024 8:25
3	bce838ed-cceb-4a77-9318-6c0ea7ccd3e8	Pengguna Google	5	propesional	10/28/2024 7:38
4	415e5dff-efce-4da8-af1a-a7db35d0f803	Pengguna Google	5	Pelayanan cepat dan supportif	10/28/2024 7:04
5	33dfed02-e26c-411f-b118-08523b0749d9	Pengguna Google	5	mntp apk nya	10/28/2024 5:51
6	bf4b5901-63a1-4d5e-9267-7f6c5fafa693	Pengguna Google	5	Aplikasi membantu buat cari pekerjaan terimaka...	10/28/2024 4:41
7	7597c72c-97e9-4051-97d6-42d0694b332a	Pengguna Google	5	sangat membantu	10/28/2024 4:37
8	b21946fa-5587-4be9-8394-ad1da11a4b68	Pengguna Google	5	sangat puas	10/28/2024 3:52
9	729c4599-0997-46b4-8263-6f475389ff27	Pengguna Google	5	trabaik	10/27/2024 16:46

Gambar 2 Scraping ulasan

4.2 Preprocessing data

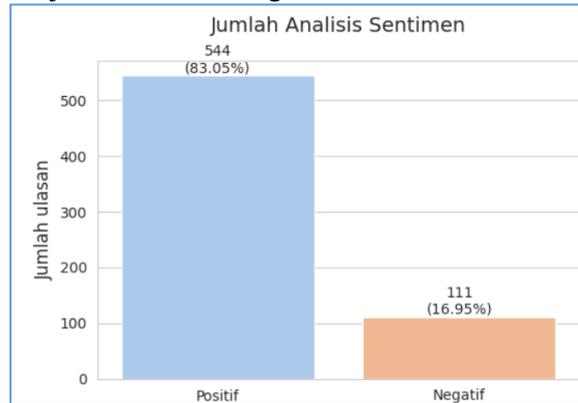
Dari total 1.000 ulasan yang berhasil diambil, setelah melewati proses prapemrosesan, jumlah data yang layak digunakan dalam analisis berkurang menjadi 655 ulasan data berkurang karena menggunakan fungsi `df.drop_duplicates(subset="Review Text")` untuk menghapus baris ulasan yang duplikat untuk mencegah bias dan juga `data.dropna()` menghapus baris dengan data kosong. Ulasan yang tersisa inilah yang kemudian digunakan dalam tahap pelabelan dan analisis sentimen, berikut hasil dari preprocessing data bisa dilihat pada Gambar 3 di bawah

```
Data columns (total 10 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Date                   655 non-null   object
1   Username                655 non-null   object
2   Rating                 655 non-null   int64
3   Review Text            655 non-null   object
4   cleaning                655 non-null   object
5   case_folding           655 non-null   object
6   normalisasi            655 non-null   object
7   tokenize                655 non-null   object
8   stopword removal       655 non-null   object
9   stemming_data          655 non-null   object
dtypes: int64(1), object(9)
memory usage: 56.3+ KB
```

Gambar 3 Data setelah preprocessing

4.3 Pelabelan data

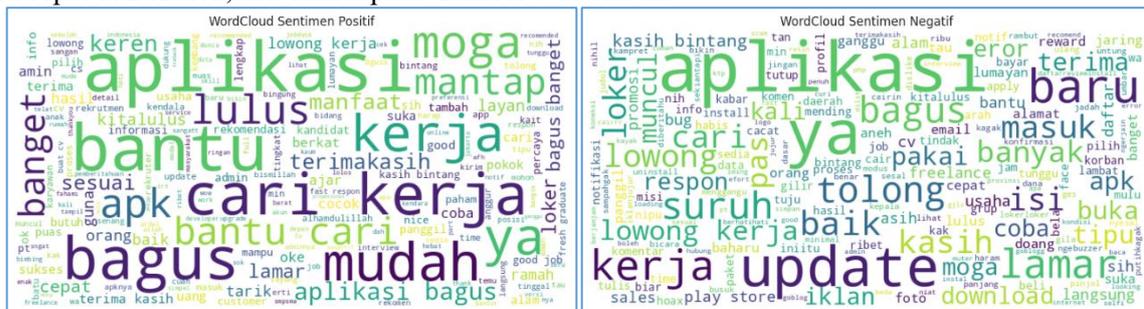
Pelabelan data dilakukan dengan cara membedakan ulasan menjadi dua klasifikasi sentimen, yaitu positif dan negatif. Kriteria pelabelan didasarkan pada nilai rating, di mana ulasan yang memiliki rating 4 dan 5 bintang dikategorikan sebagai sentimen positif, sedangkan ulasan yang memiliki rating 1 hingga 3 bintang dikategorikan menjadi sentimen negatif, ulasan didominasi oleh ulasan positif sehingga data menjadi tidak seimbang, hasil bisa dilihat di Gambar 4 di bawah



Gambar 4 Pelabelan data

4.4 Word Cloud

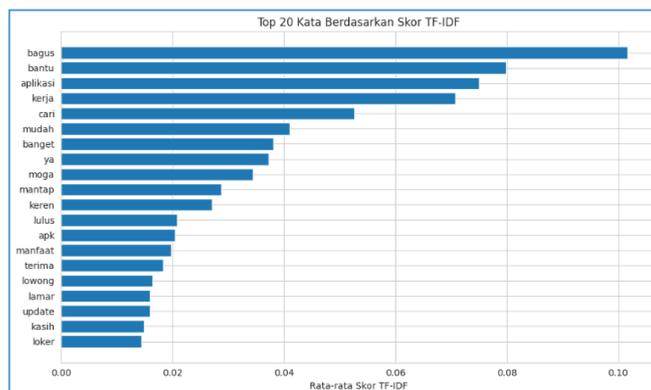
visualisasi dari *Word Cloud* berdasarkan dua jenis sentimen, yaitu positif dan juga negatif. *Word Cloud* ini menggambarkan ungkapan apa yang sering sekali keluar dalam review dari pengguna aplikasi *Kita Lulus*. Pada *Word Cloud* ulasan positif, kata-kata seperti, "*mantap*", dan "*bagus*" tampak dominan, mencerminkan kepuasan pengguna. Sementara itu, *Word Cloud* dari ulasan negatif didominasi oleh kosakata seperti "*tolong*", "*iklan*", dan "*error*", yang menunjukkan adanya keluhan terhadap kinerja aplikasi. Visualisasi ini membantu peneliti memahami pola umum opini pengguna secara cepat dan efisien, bisa dilihat pada Gambar 5 di bawah.



Gambar 5 Word cloud sentimen

4.5 TF IDF

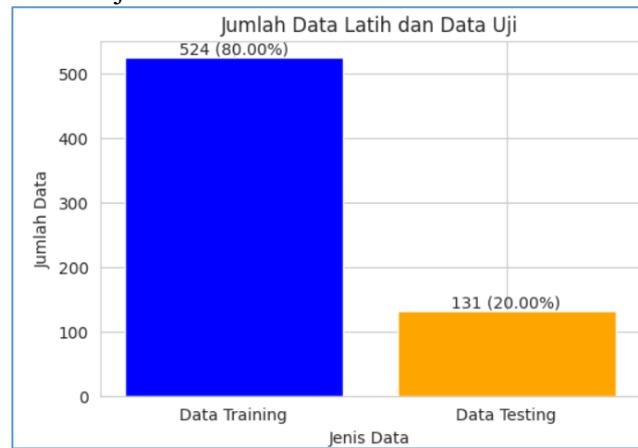
Hasil visualisasi TF-IDF. Gambar tersebut menampilkan 20 kata teratas yang memiliki nilai TF-IDF rata-rata tertinggi dalam data pelatihan. Visualisasi ini membantu untuk melihat kata-kata apa saja yang paling menonjol dan berpengaruh dalam proses pembelajaran model data bisa dilihat pada Gambar 6 di bawah



Gambar 6 Hasil TF IDF

4.6 Splitting dataset

Proses pemisahan atau *splitting* dataset merupakan langkah esensial dalam pengembangan model machine learning. Pada tahap ini, data dipisahkan kedalam dua dataset, menjadi dataset Training dan juga dataset Testing. Dataset Tes digunakan untuk membentuk model dengan mengenali pola dalam data, sebaliknya data train dimanfaatkan untuk mengukur keandalan model saat memprediksi data yang belum pernah dikenalnya. studi ini, proporsi 80% data dialokasikan sebagai Training dan 20% sebagai Testing, sebagaimana ditunjukkan Gambar 7.

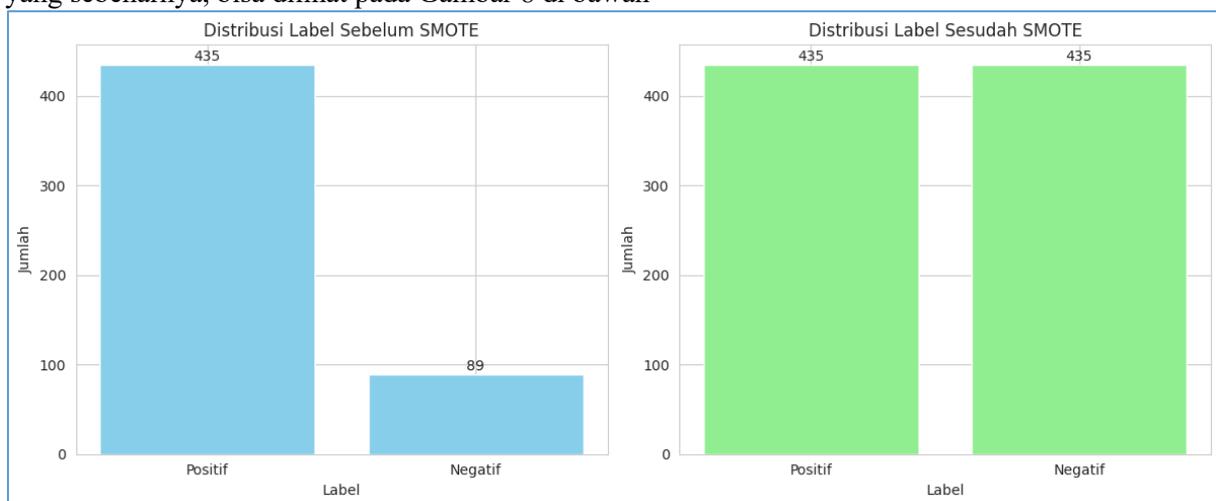


Gambar 7 Hasil Split data

4.7 SMOTE

Distribusi label dalam data latih, yang menunjukkan perbandingan sebelum dan sesudah diterapkannya teknik SMOTE. Sebelum dilakukan SMOTE, terlihat bahwa Jumlah data pada setiap kelas tidak seimbang, di mana salah satu kelas mendominasi jumlah data latih. Ketidakseimbangan Karena model cenderung lebih akurat dalam memprediksi kelas mayoritas berpotensi memunculkan bias dalam model.

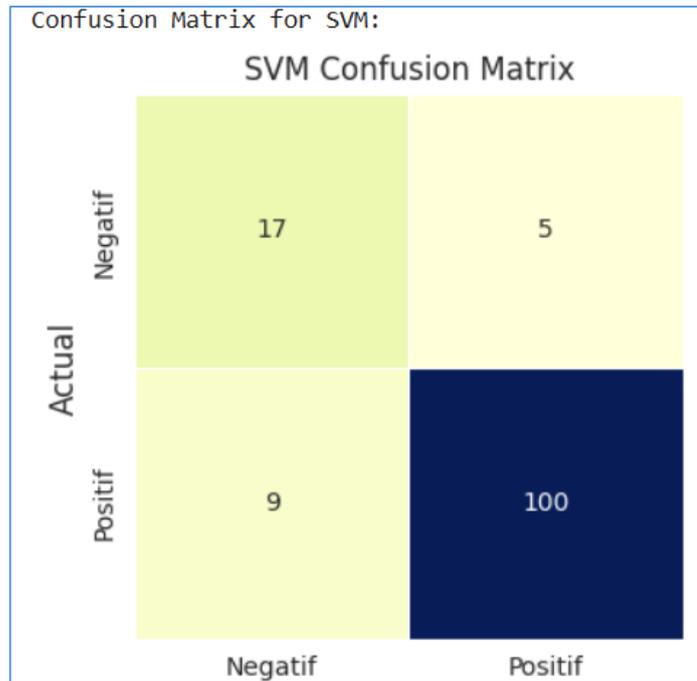
Setelah dilakukan SMOTE, distribusi data menjadi seimbang, dengan jumlah data pada setiap kelas yang setara. SMOTE berfungsi dengan cara membuat sampel sintetis untuk kelas yang lebih sedikit atau minoritas, bukan hanya menduplikasinya. Ini membantu model untuk mempelajari secara adil terhadap dua kelas, sehingga meningkatkan akurasi dan performa dalam mengklasifikasikan data yang sebenarnya, bisa dilihat pada Gambar 8 di bawah



Gambar 8 Hasil SMOTE

4.8 Support Vector Machine

Setelah dilakukan proses penyeimbangan data menggunakan teknik SMOTE, Tahap berikutnya adalah menilai kinerja model *Support Vector Machine* (SVM). terhadap data uji. Dalam proses evaluasi ini bermaksud untuk mengetahui tingkat keakuratan model dalam mengklasifikasikan data secara tepat setelah permasalahan ketidakseimbangan kelas telah ditangani, Gambar Hasil evaluasi ditunjukkan pada Gambar 9 di bawah



Gambar 9 Hasil confusion matrix

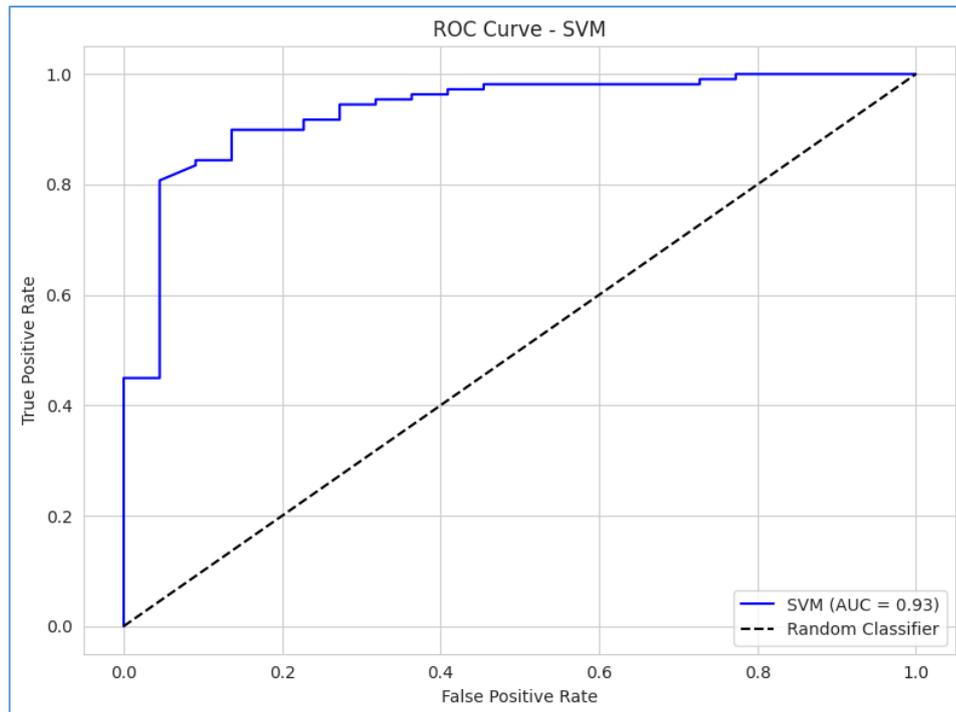
Dalam 100 data tersebut, ada kasus positif yang bekerja dengan baik dan diklasifikasikan dengan benar sebagai kelas positif (True Positives) dan 17 kasus negatif yang diklasifikasikan dengan benar sebagai kelas negatif (True Negative). Namun, ada beberapa kesalahan dalam klasifikasi, yaitu 5 data negatif yang salah dianggap positif dan 9 data positif yang salah diklasifikasikan sebagai negatif (False Negative).

Classification Report for SVM:

	precision	recall	f1-score	support
Negatif	0.65	0.77	0.71	22.00
Positif	0.95	0.92	0.93	109.00
accuracy	0.89	0.89	0.89	0.89
macro avg	0.80	0.85	0.82	131.00
weighted avg	0.90	0.89	0.90	131.00

Gambar 10 Hasil skor

Gambar 10 di atas menunjukkan hasil evaluasi model SVM yang memiliki akurasi 89%, precision 90%, recall 89%, dan F1-score 90%. Performa model pada kelas positif sangat baik (precision 95%, recall 92%), namun pada kelas negatif terlihat lebih rendah (precision 65%, recall 77%). Meskipun SMOTE telah diterapkan, ketimpangan ini tetap muncul. Hal ini mungkin disebabkan oleh variasi kata-kata dalam ulasan negatif atau adanya sarkasme dan makna implisit yang sulit dikenali model berbasis TF-IDF. Temuan ini menunjukkan perlunya eksplorasi metode lain seperti word embeddings atau pendekatan berbasis deep learning untuk menangani kompleksitas tersebut..



Gambar 11 Visualisasi Roc curve dan AUC score

Pada Gambar 11 di atas menunjukkan visualisasi Model SVM dalam grafik ini menunjukkan performa yang sangat baik, ditunjukkan dengan garis biru yang jauh berada di atas garis diagonal hitam (Random Classifier). Garis diagonal ini merepresentasikan model acak yang tidak memiliki kemampuan klasifikasi yang lebih baik dari tebakan acak.

Nilai AUC (Area Under Curve) sebesar 0.93 memperlihatkan bahwa model dapat membedakan kelas mendapatkan akurasi yang sangat tinggi. Artinya, ada peluang 93% bahwa model akan memberikan skor yang lebih tinggi untuk observasi positif dibandingkan dengan observasi negatif secara acak. Secara keseluruhan, ROC Curve ini memperlihatkan bahwa model SVM yang digunakan cukup andal dan akurat dalam melakukan klasifikasi data berdasarkan sentimen.

4.9 Perbandingan Model

Dalam Analisis sentimen ini difokuskan pada penggunaan model SVM dalam analisa sentimen aplikasi kitalulus. Sebagai pembandingan, digunakan pula model KNN dan juga Naïve Bayes mengenai aplikasi kita lulus untuk mengukur efektivitas dan keandalan model SVM dalam klasifikasi data sentimen, berikut adalah Tabel 2 menunjukkan keseluruhan performa dari perbandingan skor dari model yang diteliti.

Tabel 2 Perbandingan metode

Model	Akurasi	Precision	Recall	F1 Score	AUC
Naive Bayes + SMOTE	85%	90%	85%	87%	0.93
SVM + SMOTE	89%	90%	89%	90%	0.93
KNN + SMOTE	75%	90%	75%	78%	0.90

Dari Tabel di atas peneliti melakukan pengembangan dengan menerapkan metode SMOTE untuk menangani data *imbalance*, serta menggunakan dua algoritma klasifikasi: SVM dan KNN. Model SVM + SMOTE memperlihatkan performa yang unggul, dengan akurasi 89%, precision 90%, recall 89%, F1-score 90%, dan nilai AUC sebesar 0.93. Nilai ini mencerminkan kemampuan model dalam mengklasifikasikan data dengan lebih baik dibandingkan pendekatan sebelumnya.

Model KNN + SMOTE memiliki precision yang sama tinggi (90%), namun secara keseluruhan performanya masih di bawah SVM, dengan akurasi 75%, recall 75%, F1-score 78%, dan AUC 0.90. Secara umum, dapat disimpulkan bahwa integrasi metode SMOTE dan pemilihan algoritma yang tepat, seperti SVM, dapat meningkatkan efektivitas klasifikasi sentimen secara signifikan.

Sementara Model Naïve Bayes memiliki skor rata rata dari dengan akurasi 85%, recall 85%, F1-score 87% dan precision 90%, dan skor AUC sebesar 0.93 dari skor rata rata keseluruhan model ini mengungguli KNN tapi masih dibawah SVM.

5 Kesimpulan

Berdasarkan dari hasil penelitian ini, bisa disimpulkan bahwa penerapan algoritma SVM yang dikombinasikan menggunakan teknik SMOTE mampu memperlihatkan performa klasifikasi sentimen yang unggul terhadap ulasan pengguna aplikasi KitaLulus di Google Play Store. Model SVM + SMOTE menunjukkan hasil evaluasi yang lebih baik dibandingkan dengan algoritma K-Nearest Neighbor (KNN) dan Naïve Bayes, dengan akurasi sebesar 89%, precision 90%, recall 89%, F1-score 90%, dan nilai AUC sebesar 0,93. SMOTE terbukti efektif dalam mengatasi data *imbalance*, sehingga mampu meningkatkan kemampuan model dalam mengenali sentimen negatif yang jumlahnya lebih sedikit. Dengan demikian, pendekatan ini dapat menjadi solusi optimal dalam pengembangan sistem klasifikasi sentimen untuk aplikasi digital. Saran dari peneliti untuk penelitian selanjutnya, untuk menganalisis teknik deep learning seperti LSTM atau BERT untuk menangani kompleksitas bahasa alami secara lebih mendalam. Selain itu, penggunaan data ulasan dalam jumlah yang lebih besar dan pengelompokan sentimen ke dalam lebih dari dua kelas (positif, netral, negatif) dapat memberikan wawasan yang lebih kaya terhadap persepsi pengguna. Integrasi fitur lain seperti analisis emosi dan konteks kalimat juga dapat menjadi pendekatan yang menarik untuk dikembangkan pada studi-studi berikutnya.

Referensi

- [1] C. L. R. Dina Siti Nurrochmah, Nining Rahaningsih, Raditya Danar Dana, “Penerapan Algoritma *Naive Bayes* dalam Analisis Sentimen Ulasan Aplikasi Kitalulus di *Google Play Store*,” *J. Inform. Terpadu*, Vol. 9, No. 1, hal. 34–39, 2023, [Daring]. Tersedia pada: <https://journal.nurulfikri.ac.id/index.php/JIT> doi : <https://doi.org/10.54914/jit.v1i1.1544>
- [2] I. A. Ropikoh, R. Abdulhakim, U. Enri, dan N. Sulistiyowati, “Penerapan Algoritma *Support Vector Machine (SVM)* untuk Klasifikasi Berita Hoax Covid-19,” *J. Appl. Informatics Comput.*, Vol. 5, No. 1, hal. 64–73, 2021, doi: 10.30871/jaic.v5i1.3167.
- [3] Eskiyaturrofikoh, dan R. R. 'Suryono, “Analisis Sentimen Aplikasi X pada *Google Play Store* menggunakan *Algoritma Naïve Bayes* dan *Support Vector Machine (SVM)*,” *JUPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, Vol. 9, No. 3, hal. 1408–1419, 2024, [Daring]. Tersedia pada: <https://www.jurnal.stkipgritulungagung.ac.id/index.php/jipi/article/view/5392> doi : <https://doi.org/10.29100/jipi.v9i3.5392>
- [4] X. Yu, S. Wu, W. Chen, dan M. Huang, “*Sentiment Analysis of Public Opinions on the Higher Education Expansion Policy in China*,” *SAGE Open*, Vol. 11, No. 3, 2021, doi: 10.1177/21582440211040778.
- [5] I. S. K. Idris, Y. A. Mustofa, dan I. A. Salihi, “*Analisis Sentimen terhadap Penggunaan Aplikasi Shopee menggunakan Algoritma Support Vector Machine (SVM)*,” *Jambura J. Electr. Electron. Eng.*, Vol. 5, No. 1, hal. 32–35, 2023, doi: 10.37905/jjee.v5i1.16830.
- [6] R. R. S. Dimas Wahyu Bhatara, “Analisis Sentimen Aplikasi BCA Mobile menggunakan *Algoritma Naïve Bayes* dan *Support Vector Machine*,” *JUPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, Vol. 10, No. 2, hal. 176–184, 2024, doi: 10.37373/tekno.v10i2.419.
- [7] S. A. R. Rizaldi, S. Alam, dan I. Kurniawan, “Analisis Sentimen Pengguna Aplikasi JMO (Jamsostek Mobile) pada *Google Play Store* menggunakan *Metode Naive Bayes*,” *STORAGE J. Ilm. Tek. dan Ilmu Komput.*, Vol. 2, No. 3, hal. 109–117, 2023, doi: 10.55123/storage.v2i3.2334.
- [8] R. Ulgasesa, A. B. P. Negara, dan T. Tursina, “Pengaruh *Stemming* terhadap Performa Klasifikasi Sentimen Masyarakat tentang Kebijakan New Normal,” *J. Sist. dan Teknol. Inf.*, Vol. 10, No. 3, hal. 286, 2022, doi: 10.26418/justin.v10i3.53880.
- [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, hal. 516–528, 2025doi : <https://doi.org/10.32520/stmsi.v14i2.4795>

- [10] J. A. Wibowo, V. C. Mawardi, dan T. Sutrisno, “Visualisasi *Word Cloud* Hasil Analisis Sentimen berbasis Fitur Layanan Aplikasi Gojek dengan *Support Vector Machine*,” *J. Serina Sains, Tek. dan Kedokt.*, Vol. 2, No. 1, hal. 61–70, 2024, doi: 10.24912/jsstk.v2i1.32058.
- [11] C. H. Yutika, A. Adiwijaya, dan S. Al Faraby, “Analisis Sentimen berbasis Aspek pada *Review Female Daily* menggunakan TF-IDF dan *Naïve Bayes*,” *J. Media Inform. Budidarma*, Vol. 5, No. 2, hal. 422, 2021, doi: 10.30865/mib.v5i2.2845.
- [12] Y. A. Prasetyo, E. Utami, dan A. Yaqin, “Pengaruh Komposisi Split Data Terhadap Performa Akurasi Analisis Sentimen Algoritma *Naïve Bayes* dan SVM,” Vol. 6, No. 2, hal. 382–390, 2024, doi: 10.33650/jeeecom.v4i2.
- [13] S. Sofyan dan A. Prasetyo, “Penerapan *Synthetic Minority Oversampling Technique (SMOTE)* Terhadap Data Tidak Seimbang pada Tingkat Pendapatan Pekerja Informal di Provinsi D.I. Yogyakarta Tahun 2019,” *Semin. Nas. Off. Stat.*, Vol. 2021, No. 1, hal. 868–877, 2021, doi: 10.34123/semnasoffstat.v2021i1.1081.
- [14] A. M. Yolanda dan R. T. Mulya, “Implementasi Metode *Support Vector Machine* untuk Analisis Sentimen pada Ulasan Aplikasi Sayurbox di *Google Play Store*,” Vol. 6, No. 2, hal. 76–83, 2024, doi: 10.35580/variansiunm258.
- [15] E. Hokijuliandy, H. Napitupulu, dan F. Firdaniza, “Analisis Sentimen menggunakan Metode Klasifikasi *Support Vector Machine (SVM)* dan Seleksi Fitur *Chi-Square*,” *SisInfo J. Sist. Inf. dan Inform.*, Vol. 5, No. 2, hal. 40–49, 2023, doi: 10.37278/sisinfo.v5i2.670.
- [16] F. Kurniawati dan D. Brahma Arianto, “Analisis Implementasi Seleksi Fitur pada Klasifikasi Diabetes dengan Metode *Corellation Matrix* dan *Algoritma Logistic Regression*,” *Inform. J. Ilmu Komput.*, Vol. 19, No. 3, hal. 157–164, 2023, doi: 10.52958/iftk.v19i3.6019.
- [17] S. F. A. W. Ditha Lozera Devi, Amalia Anjani Arifiyanti, “Analisis Sentimen Ulasan Pengguna *Access By*,” Vol. 12, No. 3, 2024 doi : <https://doi.org/10.23960/jitet.v12i3.4892>