

ANALISIS SENTIMEN APLIKASI BCA MOBILE MENGGUNAKAN ALGORITMA NAÏVE BAYES DAN SUPPORT VECTOR MACHINE

Dimas Wahyu Bhatara¹⁾, Ryan Randy Suryono^{*2)}.

1. Sistem Informasi, Fakultas Teknik dan Ilmu Komputer, Universitas Teknokrat Indonesia, Indonesia.
2. Sistem Informasi, Fakultas Teknik dan Ilmu Komputer, Universitas Teknokrat Indonesia, Indonesia.

Article Info

Kata Kunci: Analisis Sentimen; BCA Mobile; *Naïve Bayes*; SMOTE; *Support Vector Machine*

Keywords: *BCA Mobil*; *Naïve Bayes*; *Sentiment Analysis*; SMOTE; *Support Vector Machine*

Article history:

Received 27 September 2024

Revised 25 Oktober 2024

Accepted 14 November 2024

Available online 4 December 2024

DOI :

<https://doi.org/10.29100/jipi.v9i4.5536>

* Corresponding author.

Corresponding Author

E-mail address:

ryan@teknokrat.ac.id

ABSTRAK

Kemajuan teknologi telah mengubah banyak aspek terutama dalam hal transaksi, dengan aplikasi seperti BCA Mobile menjadi salah satu pilihan utama. Aplikasi ini memungkinkan pengguna untuk melakukan berbagai aktivitas finansial secara *online*. Dengan popularitasnya yang terus meningkat, mencapai lebih dari 5 juta unduhan di *Google Play Store*, penelitian ini bertujuan untuk mengevaluasi pandangan pengguna terhadap aplikasi ini, baik positif maupun negatif. Analisis dilakukan menggunakan dua metode utama, yaitu algoritma *Naïve Bayes* dan *Support Vector Machine* (SVM), yang kemudian diperbaiki kinerjanya dengan menggunakan *Synthetic Minority Over-Sampling Technique* (SMOTE). Hasil penelitian menunjukkan bahwa SVM mencapai akurasi 85%, sementara *Naïve Bayes* 83%. Meskipun keduanya memiliki tingkat akurasi yang hampir serupa terdapat perbedaan dalam kemampuan masing-masing model dalam mengklasifikasikan sentimen positif dan negatif. *Naïve Bayes* memiliki *recall* yang sedikit lebih rendah untuk ulasan positif sebesar 81% dibandingkan dengan SVM mencapai 85%, namun memiliki presisi yang sedikit lebih tinggi. Sebaliknya, SVM memiliki *recall* yang lebih rendah untuk ulasan negatif, namun memiliki presisi yang lebih tinggi. Ini menunjukkan kemampuan SVM dalam menangani distribusi fitur dan kelas yang kompleks, yang tidak dapat ditangani dengan baik oleh *Naïve Bayes*.

ABSTRACT

Advances in technology have changed many aspects, especially in terms of transactions, with applications such as BCA Mobile being one of the main choices. This application allows users to conduct various financial activities online. With its growing popularity, reaching more than 5 million downloads on the Google Play Store, this study aims to evaluate users' views on the app, both positive and negative. The analysis was conducted using two main methods, namely Naïve Bayes and Support Vector Machine (SVM) algorithms, which were then improved by using Synthetic Minority Over-Sampling Technique (SMOTE). The results showed that SVM achieved 85% accuracy, while Naïve Bayes 83%. Although both models have similar accuracy rates, there are differences in their ability to classify positive and negative sentiments. Naïve Bayes has a slightly lower recall for positive reviews at 81% compared to SVM at 85%, but has a slightly higher precision. In contrast, SVM has a lower recall for negative reviews, but a higher precision. This demonstrates SVM's ability to handle complex feature and class distributions, which Naïve Bayes cannot handle well.

I. PENDAHULUAN

PERKEMBANGAN teknologi saat ini dapat mempermudah aktifitas manusia salah satunya dalam hal bertransaksi. Pada perkembangan teknologi digital BCA Mobile menjadi salah satu aplikasi yang dapat dimanfaatkan untuk melakukan transaksi secara digital. BCA Mobile menawarkan pelayanan transaksi secara digital dengan aman, nyaman, dan praktis [1]. Pada aplikasi BCA Mobile masyarakat dapat melakukan aktifitas secara digital antara lain pengiriman uang (*transfer*), pembayaran (*payment*), cek mutasi rekening dan sebagainya tanpa perlu mendatangi Anjungan Tunai Mandiri (ATM). Dengan semua kemudahan yang di tawarkan,

aplikasi BCA Mobile telah berhasil di unduh pada *google play store* sebanyak lebih dari 5 juta pengguna. Setelah masyarakat merasakan penggunaan dari aplikasi BCA Mobile mereka dapat memberikan ulasan pada aplikasi tersebut, sehingga ulasan yang diberikan tersebut harus dilakukan analisis lebih lanjut untuk mengetahui apakah ulasan yang diberikan oleh pengguna pada aplikasi ini memberikan komentar dengan ulasan positif atau negatif. Hasil dari eksperimen analisis sentimen tersebut dapat dijadikan bahan evaluasi oleh para pengembang dalam memperbaiki aplikasi BCA Mobile agar dapat berjalan lebih optimal dan dapat membuat pengguna aplikasi BCA Mobile lebih aman dan nyaman dalam hal bertransaksi [2].

Analisis sentimen merupakan suatu cara pengumpulan data opini masyarakat melalui *platform* sosial media dimana topik yang dibahas mengandung isu-isu terkini [3]. Analisis sentimen adalah proses pengelompokan suatu kalimat agar mengetahui banyaknya opini yang ada untuk mengetahui suatu kelompok memiliki kategori positif atau negatif [4]. Pada eksperimen ini Teknik pengumpulan data yang akan digunakan melalui aplikasi *google play store*, dimana data yang diambil berupa opini pengguna pada aplikasi BCA Mobile.

Pada penelitian terdahulu yang berjudul Perbandingan Metode *Random Forest* dan *Naive Bayes* Pada Analisis *Sentiment Review* Aplikasi BCA Mobile mendapatkan hasil terbaik pada metode *Random Forest* yang mana nilai akurasi sebesar 93,93%, presisi 93,02%, *recall* 89,89% dan *F1-Score* 91,43% [5]. Penelitian selanjutnya yang berjudul Analisis Sentimen pada *review* Aplikasi Grab di *Google Play Store* Menggunakan *Support Vector Machine* dari penelitian yang dilakukan ada lebih dari 1.000 *review* pengguna dikumpulkan menghasilkan akurasi 85,54% *review* [6]. Adapun penelitian Perbandingan Metode Klasifikasi *Support Vector Machine* dan *Naive Bayes* untuk Analisis sentiment pada ulasan Tekstual di *Google Play Store* Hasil Pengujian yang didapat dari metode *3-folds cross validation* menghasilkan *SVM Classifier* memiliki nilai tinggi sebesar 81,46% jika dibandingkan dari akurasi *Naive Bayes classifier* sebesar 75,41% [7].

Berdasarkan penelitian yang berjudul Sentimen Analisis Terhadap pada *Google Playstore* Menggunakan Algoritma *Naive Bayes* dan Algoritma Genetika, hasil pada proses pengumpulan dataset, pengelompokan dataset, proses *oversampling*, proses *pre-processing*, proses algoritma *Naive Bayes* dan proses algoritma genetika eksperimen ini berupa perbandingan nilai akurasi Shoppe dengan nilai 96,53%, Ruang guru dengan akurasi 95,54%, Tokopedia dengan akurasi 96,68% dan Gojek dengan nilai akurasi 96,54% [8]. Selanjutnya penelitian yang berjudul Analisis Sentimen *Review* Aplikasi Ruangguru Menggunakan Algoritma *Support Vector Machine* melakukan uji coba dengan menggunakan kernel linier menghasilkan akurasi tertinggi dengan nilai 0.897. kombinasi data training 60% dan data testing 40% menghasilkan akurasi tertinggi sebesar 0,900 [9].

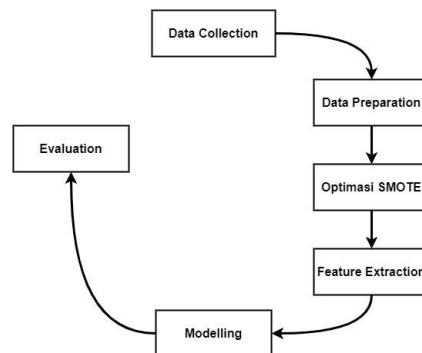
Dalam penelitian ini menggunakan Metode algoritma *Naive Bayes* (NB) dan *Support Vector Machine* (SVM) untuk melakukan analisis sentimen, sehingga dengan menerapkan metode algoritma tersebut mesin mampu belajar memahami dan membedakan komentar yang diberikan oleh masyarakat melalui layanan ulasan penggunaan Aplikasi BCA Mobile kedalam suatu sentimen. *Naive Bayes* merupakan algoritma klasifikasi yang digunakan dalam sentimen analisis untuk mengklasifikasikan teks berdasarkan probabilitas statistik dari suatu data [10], [11]. *Support Vector Machine* (SVM) merupakan suatu algoritma *supervised learning*, dimana algoritma ini hanya menangani klasifikasi biner (dua kelas) seperti ya atau tidak, 1 dan 0, *True* dan *False* dan sebagainya [12]. SVM akan mengklasifikasikan suatu data berdasarkan *hyperplane* yang terdekat dengan support vectors dari suatu data. Pada penelitian sebelumnya, berbagai metode analisis sentimen telah diterapkan, termasuk *Random Forest*, *Naive Bayes*, dan *Support Vector Machine* (SVM).

Namun, penelitian sebelumnya belum menerapkan *Synthetic Minority Over-Sampling Technique* (SMOTE), sebuah teknik yang dapat meningkatkan performa algoritma dengan menyamakan jumlah data sentimen positif dan negatif. Ini merupakan kontribusi baru dari penelitian ini yang diharapkan dapat meningkatkan akurasi dan konsistensi hasil analisis sentimen. Dalam penelitian ini, kami menggunakan metode algoritma *Naive Bayes* (NB) dan *Support Vector Machine* (SVM) untuk melakukan analisis sentimen terhadap ulasan pengguna BCA Mobile. Dengan menerapkan SMOTE pada penelitian ini, kami bertujuan untuk tidak hanya memahami ulasan pengguna, tetapi juga memberikan kontribusi pada pengembangan metodologi analisis sentimen yang lebih baik. Hasil dari penelitian ini diharapkan dapat memberikan wawasan berharga bagi pengembang aplikasi BCA Mobile dan masyarakat umum yang tertarik dengan penggunaannya.

II. METODE PENELITIAN

A. Tahapan Penelitian

Terdapat 6 tahapan penelitian yang dimulai dengan *data collection*, *data preparation*, optimasi SMOTE, *feature extraction*, *modelling*, dan *evaluation*. Berikut tahapan penelitian yang dapat dilihat pada gambar 1.



Gambar 1. Tahapan Eksperimen

Tahapan eksperimen penelitian ini mengarah pada pendekatan text mining. Tahapan awal pada penelitian ini adalah data collection (mengumpulkan data) dimana eksperimen ini akan menggunakan data dari google Play Store yang berisi ulasan dari aplikasi BCA Mobile Tahapan selanjutnya adalah *data preparation*, dimana tahapan ini akan mempersiapkan data agar bersih sebelum dilakukan *modelling*. Selanjutnya untuk memastikan data agar seimbang maka dilakukan optimasi data menggunakan SMOTE. Setelah data dipastikan bersih dan seimbang maka akan diproses pada tahapan *feature extraction*, sehingga data tersebut mampu dilakukan *modelling* dimana metode yang digunakan yaitu *Naïve Bayes* dan *Support Vector Machine (SVM)*, lalu peneliti akan melakukan evaluasi hasil dari setiap algoritma.

1) Data Collection

Data Collection merupakan langkah awal dalam pengumpulan data yang akan diolah pada tahap *text mining*. Data dikumpulkan tersebut bisa berasal dari ulasan pengguna aplikasi BCA Mobile yang berasal dari *Google Playstore* [13]. Teknik pengumpulan datanya menggunakan *package name* aplikasi yang ada pada domain di *Google Playstore*. Dalam eksperimen ini akan mengambil data ulasan yang relevan dan terbaru.

2) Data Preparation

Data preparation merupakan tahapan yang digunakan untuk membersihkan data agar lebih konsisten, dimana pada tahapan yang dilakukan pada *data preparation* dibagi menjadi beberapa tahapan sebagai berikut:

2.1 Cleansing

Data Cleansing merupakan suatu tahapan pertama yang digunakan untuk membersihkan data, dimana data akan dibersihkan dari data yang kosong, data yang memiliki duplikat, data yang terdapat HTML, angka, hastag dan emoji [14].

2.2 Casefolding

Casefolding yaitu teknik yang digunakan untuk mengubah kata yang memiliki huruf kapital menjadi huruf kecil sehingga teks akan menjadi selaras [15]. Bahasa pemrograman *python* merupakan salah satu bahasa pemrograman yang *case sensitive* sehingga fungsi adanya *casefolding* adalah agar kata lebih seragam sehingga dapat dikelompokkan.

2.3 Filtering and Stopword

Filtering digunakan untuk memfilter kata-kata yang tidak dibutuhkan data menjadi lebih berkualitas [16], sedangkan *Stopword* menghapus kata umum yang sering muncul dalam suatu teks dan tidak memberikan informasi penting. Tujuan dari *stopword* adalah untuk menghapus kata-kata yang tidak bermakna seperti penghapusan kata hubung “di”, “ke”, “yang”, dll [17].

2.4 Tokenize

Tokenize adalah proses yang berfungsi untuk mengubah atau memisahkan suatu kalimat menjadi potongan-potongan karakter kata. Bentuk kata tersebut akan dipisahkan oleh spasi sehingga akan lebih mudah dianalisis [18].

2.5 Stemming

Stemming adalah proses yang digunakan untuk mengubah atau mengganti kata yang memiliki akhiran atau imbuhan menjadi bentuk dasarnya [19]. Proses *stemming* ini menggunakan bantuan *package* sastra dengan *library stemmer* sehingga mesin akan mengubah kata yang memiliki imbuhan ke bentuk kata dasar.

2.6 Labeling

Labeling yang digunakan dalam penelitian ini menggunakan bantuan *libraly textblob* dimana sebelum dilakukan *labeling* maka mesin akan menentukan nilai *score* polaritas suatu teks, dimana *score* tersebut dapat dijadikan acuan untuk menentukan suatu teks masuk kedalam sentiment dengan label positif atau negatif [20].

3) Optimasi SMOTE

Optimasi SMOTE (Synthetic Minority Over-sampling Technique) adalah proses untuk mengatasi dataset yang mengalami ketidak seimbangan data. Metode SMOTE merupakan teknik yang digunakan untuk menghasilkan sebuah metode baru melalui sampel kelas minoritas [21]. Metode ini dilakukan dengan cara mengambil ulang contoh data yang tersedia sehingga kumpulan data tersebut menjadi lebih proporsional. Tujuannya adalah untuk meningkatkan kinerja dan keefektifan agar data yang dikelola lebih optimal [22]. SMOTE, atau *Synthetic Minority Over-sampling Technique*, menjadi langkah penting dalam menangani ketidakseimbangan dataset pada analisis sentimen. Langkah-langkah penggunaannya mencakup identifikasi kelas minoritas, pemilihan sampel dari kelas minoritas, pembentukan data sintetis baru dengan memilih sampel yang berdekatan dalam ruang fitur, penambahan sampel sintetis ke dataset asli, dan evaluasi hasil untuk memastikan keseimbangan dataset. Dalam analisis yang lebih mendalam, SMOTE tidak hanya menciptakan sampel sintetis untuk kelas minoritas, tetapi juga meminimalisir risiko *overfitting* dengan mempertimbangkan ruang fitur dan distribusi data. Dengan dataset yang seimbang, algoritma *Naive Bayes* dan *Support Vector Machine* diharapkan dapat mengenali pola sentimen dengan lebih baik, meningkatkan interpretasi yang lebih akurat dari ulasan pengguna BCA Mobile. Hasilnya diharapkan memberikan wawasan yang lebih mendalam bagi pengembang aplikasi BCA Mobile dan manfaat bagi pengguna yang akan mendapatkan layanan yang lebih baik dan sesuai dengan kebutuhan mereka.

4) Feature Extraction

Feature Extraction adalah proses pengubahan data teks menjadi suatu angka berbentuk *matrix* dua dimensi [23]. Pada *machine learning*, mesin tidak mampu belajar dengan bentuk data teks sehingga perlu dilakukan transformasi data teks menjadi data berbentuk *matrix* 2 dimensi. Pada eksperimen ini menggunakan *feature extraction* TF-IDF yang akan mengelompokkan suatu data berdasarkan frekuensi kemunculan data yang ada didalam suatu *corpus*. Berikut ini penjelasan cara kerja dan rumus yang digunakan pada TF-IDF:

4.1 Term Frequency (TF)

Term frequency digunakan untuk mengukur seberapa sering suatu kata muncul dalam sebuah dokumen yang di dasarkan dari kata kunci yang telah ditetapkan sebelumnya. Terdapat rumus yang digunakan pada tahapan TF sebagai berikut:

$$tf_{t,d} = \frac{n_{t,d}}{\text{(Total Number of tern in document)}} \quad (1)$$

$tf_{t,d}$ akan menghitung jumlah kemunculan kata didalam suatu dokumen, dimana jumlah kemunculan suatu kata tersebut dilambangkan dengan $n_{t,d}$ dan akan dibagi dengan jumlah dokumen yang ada didalam *corpus* (*Total Number of tern in document*).

4.2 Inverse Document Frequency (IDF)

Inverse Document Frequency digunakan untuk mengetahui jumlah dokumen yang terdapat dalam suatu data. IDF akan menunjukkan seberapa sering suatu kata digunakan, oleh sebab itu IDF berfungsi untuk mengurangi nilai suatu *term* jika banyak tersebar di dalam dokumen.

$$idf_d = \log \frac{\text{Number of document}}{\text{(Total Number of tern in document)}} \quad (2)$$

Pada proses perhitungan IDF menggunakan logaritma, dimana akan menghitung jumlah kemunculan kata didalam semua dokumen yang ada atau didalam *corpus* yang ada. Total dari dokumen tersebut akan dibagi dengan jumlah *term* atau kata yang terdapat didalam suatu dokumen lalu akan dikalikan dengan nilai logaritma.

$$TF - IDF = tf_{t,d} \times idf_d \quad (3)$$

Hasil dari $tf_{t,d}$ dan idf_d selanjutnya akan dikalikan, sehingga output dari hasil perkalian tersebut berupa angka dalam bentuk *matrix* 2 dimensi.

5) Modeling

Paparan data yang digunakan dalam melakukan penelitian ini yaitu menggunakan model algoritma *Naive Bayes* dan *Support Vector Machine* (SVM), model akan melatih dan menguji data agar mendapatkan performa pada setiap model.

5.1 Naive Bayes

Naive Bayes merupakan algoritma yang berguna dalam mengklasifikasikan data, mengusung konsep dasar dari *Teorema Bayes* yang disusun oleh ilmuwan Inggris, Thomas Bayes. Dengan menggabungkan pengalaman masa

lalu untuk meramalkan peristiwa di masa depan, klasifikasi *Bayesian* memberikan kerangka statistik yang dapat memperkirakan probabilitas keanggotaan suatu kelas. Pemilihan istilah "*naive*" menunjukkan bahwa setiap atribut atau variabel dianggap mandiri satu sama lain. Sementara menurut perspektif frekuentis, model ini memaknai representasi melalui dua skenario kejadian yang berbeda. *Naive Bayes* menjadi salah satu metode dalam pembelajaran mesin yang memanfaatkan perhitungan probabilitas, menggunakan pendekatan statistik yang sederhana dengan anggapan bahwa kelas-kelas yang berbeda tidak bergantung satu sama lain secara langsung [24]. Rumus *Naive Bayes* dapat dilihat pada persamaan 4 berikut ini :

$$P(X|Y) = \frac{P(Y|X) \times P(X)}{P(Y)} \quad (4)$$

$P(X|Y)$ menggambarkan probabilitas bahwa peristiwa X terjadi jika peristiwa Y telah terjadi, sedangkan $P(Y|X)P(Y|X)$ adalah probabilitas peristiwa Y terjadi jika peristiwa X telah terjadi, yang merupakan informasi awal yang kita ketahui. $P(X)P(X)$ adalah probabilitas murni dari peristiwa X terjadi, tanpa mempertimbangkan peristiwa Y, sedangkan $P(Y)P(Y)$ adalah probabilitas murni dari peristiwa Y terjadi, tanpa mempertimbangkan peristiwa X. Dengan menggunakan rumus ini, kita dapat memperbarui perkiraan probabilitas peristiwa X berdasarkan informasi baru yang diberikan tentang peristiwa Y, yang berguna dalam mengambil keputusan dan analisis data di berbagai konteks.

5.2 Support Vector Machine

Support Vector Machine (SVM) merupakan sebuah algoritma yang sangat berguna dalam melakukan klasifikasi data, baik yang memiliki pola linear maupun non-linear. Algoritma ini berhasil mengatasi tantangan yang terkait dengan pola non-linear dengan memanfaatkan konsep kernel untuk melakukan pemetaan data ke dalam ruang dimensi yang lebih kompleks. SVM mampu membagi berbagai jenis objek ke dalam kelompok-kelompok yang berbeda dengan sangat efektif. Sebagai model *supervised*, pendekatan klasifikasi SVM menggunakan data sampel yang sudah diklasifikasikan sebelumnya untuk melakukan prediksi terhadap kelas yang mungkin ada berdasarkan sampel data yang telah ada. Rumus SVM dijelaskan pada persamaan 5 berikut ini :

$$f(x) = \text{sign}(\sum_{i=1}^n a_i y_i K(x_i, x) + b) \quad (5)$$

Rumus diatas menggambarkan fungsi $f(x)$ yang digunakan dalam proses klasifikasi dengan *Support Vector Machine* (SVM). Fungsi ini menerima data yang akan diklasifikasikan, dilambangkan dengan x , dan mengembalikan label kelas yang sesuai. Variabel x_i merujuk pada data latih, sedangkan y_i adalah label kelas yang terkait dengan data latih tersebut. Bobot dari setiap data latih direpresentasikan oleh a_i . Fungsi kernel, yang dilambangkan dengan $K(x_i, x)$, proses ini dilakukan untuk setiap data latih. Parameter b merupakan bias yang digunakan dalam proses klasifikasi. Dengan menggunakan fungsi ini, SVM dapat memprediksi label kelas yang sesuai untuk data yang akan diklasifikasikan berdasarkan informasi yang diperoleh dari data latih dan bobot yang dihitung [25].

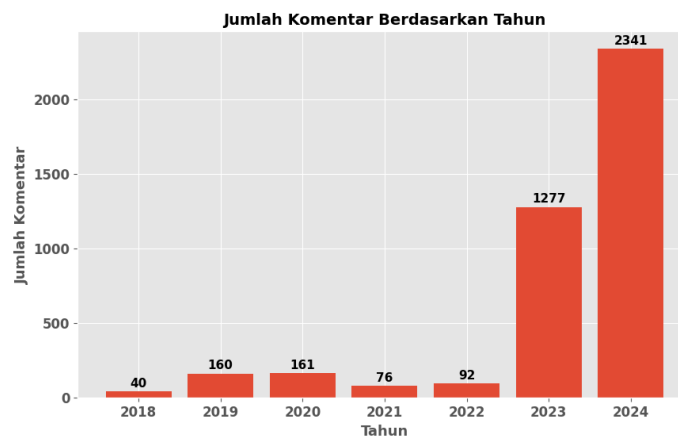
6) Evaluation

Evaluasi model *machine learning* adalah langkah penting dalam memahami kinerja model yang telah dibangun. Salah satu pendekatan yang umum digunakan adalah melalui metode *confusion matrix*. Metode ini telah terbukti efektif dalam mengukur performa suatu sistem klasifikasi dengan membandingkan hasil klasifikasi yang sebenarnya dengan hasil yang diprediksi oleh sistem tersebut. *Confusion matrix* menyediakan gambaran yang jelas tentang seberapa baik atau buruk kinerja sistem dalam mengklasifikasikan data. Dalam proses evaluasi ini, terdapat empat istilah penting yang digunakan untuk mengukur kinerja klasifikasi, yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN) [26].

III. HASIL DAN PEMBAHASAN

A. Dataset

Proses pengambilan data dilakukan dengan cara *crawling* ulasan pengguna aplikasi BCA Mobile dari *Google Playstore*. Total data yang terhimpun sebanyak 4147 ulasan, yang dikumpulkan dalam rentang waktu September 2018 hingga Maret 2024. Tahap ini memiliki peran penting karena kualitas dan representasi *dataset* mempengaruhi hasil analisis selanjutnya. Dengan membatasi rentang waktu tertentu, analisis dapat mencerminkan perkembangan aplikasi selama periode yang relevan. Pada gambar 2 menampilkan jumlah *dataset* komentar aplikasi BCA Mobile bervariasi dari tahun 2018 hingga 2024. Pada tahun 2018 terdapat 40 data, meningkat menjadi 160 data pada 2019, dan stabil di 161 data pada 2020. Jumlah data kemudian turun menjadi 76 pada 2021, naik menjadi 92 pada 2022, dan melonjak tajam menjadi 1277 pada 2023. Puncaknya terjadi pada tahun 2024 dengan 2341 data ulasan.



Gambar 2. Jumlah Komentar Berdasarkan Tahun

B. Hasil Preprocessing

Data ulasan yang dianalisis dalam penelitian ini merupakan bagian dari teks yang tidak terstruktur, sehingga diperlukan langkah-langkah pra-pemrosesan untuk membersihkan dan mengelola data tersebut. Tahapan *preprocessing* meliputi proses *cleansing*, *case folding*, *filtering and stopword removal*, *tokenize*, dan *stemming*, dengan tujuan menghasilkan *dataset* yang sesuai untuk analisis dan meningkatkan kualitas hasil klasifikasi. Setelah melalui proses *preprocessing*, data ulasan aplikasi BCA Mobile telah diolah untuk menjadi lebih sederhana dan bersih. Analisis menunjukkan bahwa setelah *preprocessing*, data menjadi lebih terstruktur dan relevan untuk analisis lebih lanjut. Hasil dari *Preprocessing* dapat dilihat pada Tabel 1.

Tabel 1.
 Hasil Preprocessing

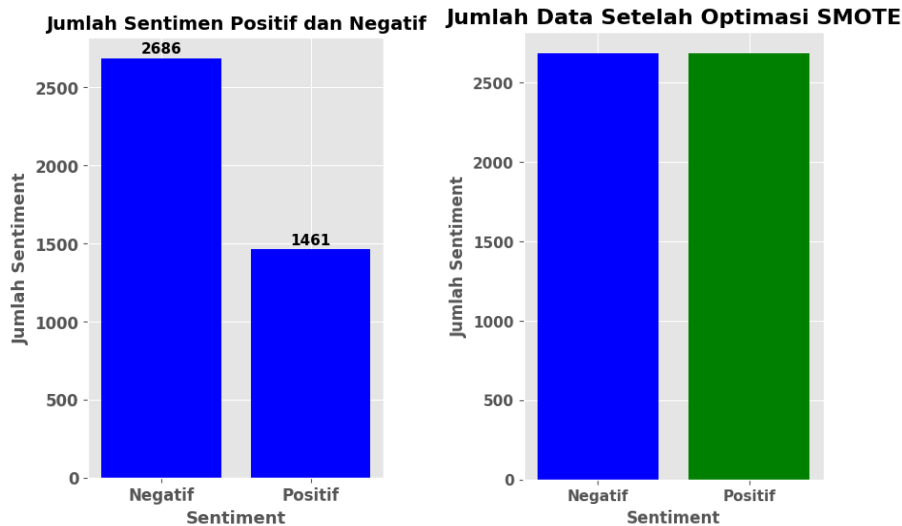
No	Proses	Hasil Preprocessing
1	<i>Dataset</i>	Sejak menggunakan BCA Mobile.. transaksi keuangan jd lbh mudah dan praktis.. bisa dilakukan dimana sana dan kpn saja.. tdk perlu hrs waktu ke Bank dan mengantri. Aplikasi BCA mobile tdk ribet dari segi bahasa dan fungsinya, jd gampang dimengerti.
2	<i>Cleansing</i>	Sejak BCA Mobile transaksi keuangan jd lbh mudah dan praktis bisa dilakukan dimana sana dan kpn saja tdk perlu hrs waktu ke Bank dan mengantri Aplikasi BCA mobile tdk ribet dari segi bahasa dan fungsinya jd gampang
3	<i>Case Folding</i>	sejak bca mobile transaksi keuangan jd lbh mudah dan praktis bisa dilakukan dimana sana dan kpn saja tdk perlu hrs waktu ke bank dan mengantri aplikasi bca mobile tdk ribet dari segi bahasa dan fungsinya jd gampang
4	<i>Tokenize</i>	['sejak', 'bca', 'mobile', 'transaksi', 'keuangan', 'jd', 'lbh', 'mudah', 'dan', 'praktis', 'bisa', 'dilakukan', 'dimana', 'sana', 'dan', 'kpn', 'saja', 'tdk', 'perlu', 'hrs', 'waktu', 'ke', 'bank', 'dan', 'mengantri', 'aplikasi', 'bca', 'mobile', 'tdk', 'ribet', 'dari', 'segi', 'bahasa', 'dan', 'fungsinya', 'jd', 'gampang']
5	<i>Filtering</i>	['bca', 'mobile', 'transaksi', 'keuangan', 'jd', 'lbh', 'mudah', 'praktis', 'dimana', 'kpn', 'tdk', 'hrs', 'bank', 'mengantri', 'aplikasi', 'bca', 'mobile', 'tdk', 'ribet', 'segi', 'bahasa', 'fungsinya', 'jd', 'gampang']
6	<i>Stemming</i>	bca mobile transaksi uang jd lbh mudah praktis mana kpn tdk hrs bank antri aplikasi bca mobile tdk ribet segi bahasa fungsi jd gampang

Setelah tahap *preprocessing* data selesai, langkah selanjutnya yaitu memberi label pada kumpulan data untuk membedakan setiap ulasan yang diposting oleh pengguna BCA Mobile di *Google PlayStore*, apakah itu positif atau negatif. Peneliti menentukan skor polaritas untuk setiap kata dengan menggunakan *Valence Aware Dictionary* dan *Sentiment Reasoner* (VADER), dua alat dari *Natural Language Toolkit* (NLTK). Ulasan dengan skor lebih dari 3 dianggap positif, sementara ulasan dengan skor 3 atau kurang dianggap negatif. Sebagai hasilnya, *dataset* terdiri dari 1461 ulasan positif dan 2686 ulasan negatif. Hasil proses *labelling* ditampilkan pada tabel 2.

Tabel 2.
 Hasil Labelling

No	Ulasan	Hasil Labelling
1	bca mobile transaksi uang jd lbh mudah praktis mana kpn tdk hrs bank antri aplikasi bca mobile tdk ribet segi bahasa fungsi jd gampang	Positif
2	aduuhhh parah banget bca mobile signal nya merah kuota jaring bagus indikator nya merah android restart ulang hasil nya transaksi susah	Negatif

C. Tahap Pengujian



Gambar 3. Jumlah data sebelum dan sesudah optimasi SMOTE

Gambar 3 menampilkan perbandingan antara data sebelum dan sesudah pengoptimalan menggunakan metode SMOTE. Sebelum proses pengoptimalan, jumlah data positif adalah 1461 dan jumlah data negatif adalah 2686. Tujuan dari pengoptimalan ini untuk mencapai keseimbangan antara kedua kelas. Dengan menerapkan SMOTE, jumlah data pada setiap sentimen disamakan, sehingga kedua kelas memiliki jumlah data yang sama, yaitu 2686. Hal ini dilakukan untuk mengatasi masalah ketidakseimbangan dalam pembelajaran model algoritma, di mana keberadaan jumlah data yang sedikit (*minority*) atau jumlah data dengan sentimen mayoritas (*majority*) dapat memengaruhi kinerja model. Dengan demikian, pengoptimalan menggunakan SMOTE bertujuan untuk meningkatkan kemampuan model dalam memahami dan memprediksi sentimen dengan lebih baik.

Setelah mencapai keseimbangan data, langkah selanjutnya adalah melakukan pembagian data menjadi dua *subset* yaitu data latih (*training*) dan data uji (*testing*). Dalam penelitian ini, kedua model algoritma, yaitu *Naïve Bayes* dan *Support Vector Machine*, menerapkan strategi yang serupa dalam proses pembagian data, dimana 80% dari total data digunakan untuk melatih model dan 20% sisanya digunakan untuk menguji performa model yang telah dilatih. Dari total dataset yang terdiri dari 4147 data, 80% atau sekitar 3317 data dipilih secara acak untuk proses pelatihan, sementara sisanya, sekitar 20% atau 830 data, digunakan untuk menguji model yang telah dilatih. Tahapan ini merupakan prasyarat penting sebelum melanjutkan ke proses analisis dan pembahasan performa klasifikasi dari kedua model tersebut.

D. Klasifikasi Model

Langkah berikutnya adalah menampilkan hasil klasifikasi dari model *Naïve Bayes* dan *Support Vector Machine*. Hasil akurasi menjadi indikator awal untuk mengevaluasi performa model. Perbedaan terlihat antara model *Support Vector Machine* (SVM) dan *Naïve Bayes*, walaupun hanya terpaut 2%. SVM mencapai tingkat akurasi 85%, sedangkan *Naïve Bayes* 83%. Hal ini menandakan keunggulan relatif SVM dalam klasifikasi. Faktor yang memengaruhi perbedaan ini termasuk distribusi fitur dan kelas dalam *dataset*, yang lebih teratasi oleh SVM karena kemampuannya menangani data yang tidak linier. Parameter model seperti kernel dalam SVM atau distribusi probabilitas dalam *Naïve Bayes* juga berperan.

Tabel 3.
 Hasil Accuracy

No	Model	Accuracy
1	<i>Naïve Bayes</i>	83 %
2	<i>Support Vector Machine</i>	85 %

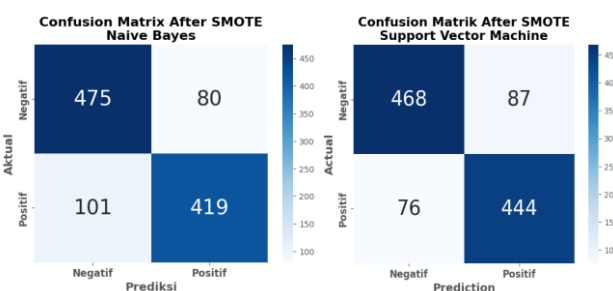
Pada tabel 4 menampilkan hasil klasifikasi bahwa *Naïve Bayes* memiliki *recall* 81% untuk ulasan positif dan 86% untuk ulasan negatif, dengan *precision* masing-masing sebesar 84% dan 82%, serta *F1-Score* sebesar 82% untuk ulasan positif dan 84% untuk ulasan negatif. Sementara itu, SVM memiliki *recall* 85% untuk ulasan positif dan 86% untuk ulasan negatif, dengan *precision* masing-masing sebesar 84% dan 86%, serta *F1-Score* sebesar 84% untuk ulasan positif dan 85% untuk ulasan negatif. Meskipun keduanya menunjukkan performa yang serupa, SVM cenderung sedikit lebih baik dalam mengklasifikasikan ulasan positif, sementara *Naïve Bayes* lebih baik dalam mengklasifikasikan ulasan negatif.

Tabel 4.
 Hasil Klasifikasi

<i>Naïve Bayes</i>				<i>Support Vector Machine</i>			
No	Parameter	Nilai		No	Parameter	Nilai	
		Positif	Negatif			Positif	Negatif
1	<i>Recall</i>	81 %	86 %	1	<i>Recall</i>	85 %	84 %
2	<i>Precision</i>	84 %	82%	2	<i>Precision</i>	84 %	86 %
3	<i>F1-Score</i>	82 %	84 %	3	<i>F1-Score</i>	84 %	85 %

Perbandingan antara model klasifikasi *Naïve Bayes* dan *Support Vector Machine* (SVM) berdasarkan hasil *confusion matrix* yang dapat dilihat pada gambar 4 menunjukkan bahwa *Naïve Bayes* memiliki 419 *true positives* (TP) dan 475 *true negatives* (TN), menunjukkan kemampuan yang baik dalam mengklasifikasikan ulasan positif dan negatif dengan benar. Namun, terdapat 101 *false positives* (FP) dan 80 *false negatives* (FN), yang menunjukkan kecenderungan untuk salah mengklasifikasikan ulasan negatif sebagai positif, dan sebaliknya. Di sisi lain, SVM memiliki 444 TP dan 468 TN, dengan 76 FP dan 87 FN. SVM menunjukkan kecenderungan yang berbeda, lebih sering salah mengklasifikasikan ulasan negatif sebagai positif. Berdasarkan hasil penelitian ini menunjukkan bahwa meskipun keduanya memiliki kinerja yang baik, mereka cenderung melakukan kesalahan yang berbeda. *Naïve Bayes* lebih berhati-hati dalam mengklasifikasikan ulasan positif, sementara SVM lebih berhati-hati dalam mengklasifikasikan ulasan negatif.

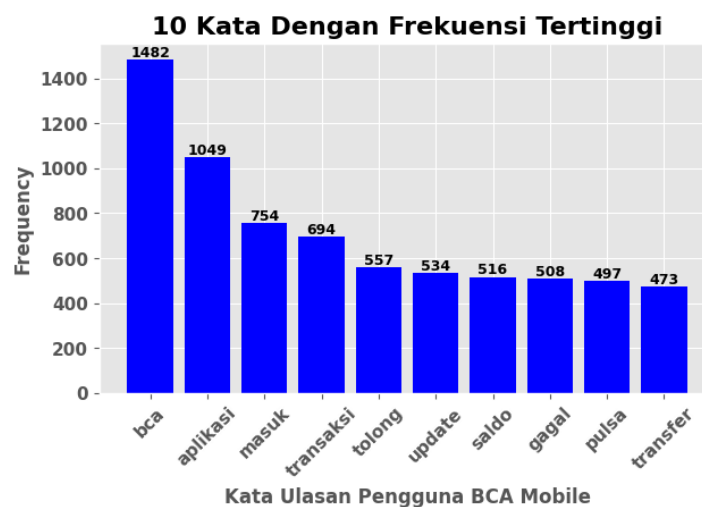
Meskipun telah dilakukan analisis perbandingan kinerja antara model *Naïve Bayes* dan *Support Vector Machine* (SVM), diperlukan pemahaman yang lebih mendalam terhadap faktor-faktor yang memengaruhi perbedaan tersebut, seperti distribusi fitur dan parameter model. Faktor-faktor ini dapat memengaruhi kinerja keduanya tergantung pada karakteristik dataset yang digunakan. Distribusi fitur mempengaruhi kinerja model, dengan *Naïve Bayes* lebih cocok untuk dataset dengan fitur yang saling bebas, sementara SVM lebih cocok untuk dataset kompleks dengan hubungan non-linear antara fitur-fiturnya. Selain itu, parameter model yang disesuaikan seperti *smoothing* parameter untuk *Naïve Bayes* dan parameter C serta kernel untuk SVM juga memainkan peran penting. Penyesuaian parameter ini dapat mempengaruhi kinerja model secara signifikan, dengan penyesuaian yang tepat dapat meningkatkan akurasi dan generalisasi model, sedangkan penyesuaian yang tidak tepat dapat mengakibatkan kinerja yang buruk. Karakteristik dataset seperti jumlah sampel, sebaran kelas, dan keberadaan atau ketiadaan fitur yang penting juga dapat memengaruhi kinerja model. Dengan memahami faktor-faktor ini, peneliti dapat melakukan analisis yang lebih mendalam untuk memilih model yang paling sesuai dengan karakteristik dataset dan tujuan analisis sentimen yang diinginkan.



Gambar 4. Hasil *Confusion Matrix*

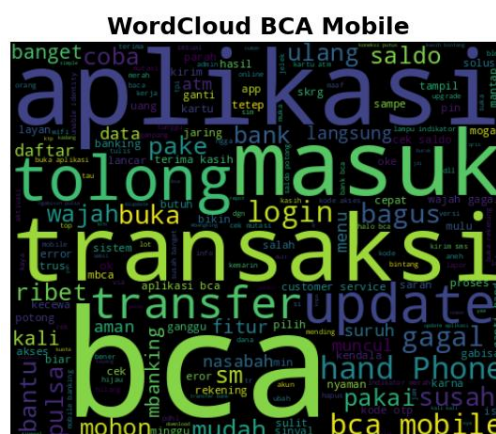
E. Visualisasi Data

Dalam penelitian ini, visualisasi data dalam bentuk diagram batang (*barchart*) untuk memperkuat hasil dan pembahasan. Diagram batang digunakan untuk memvisualisasikan frekuensi kemunculan kata-kata dalam ulasan pelanggan BCA Mobile. Sepuluh kata dengan frekuensi tertinggi dalam ulasan pelanggan BCA Mobile memberikan gambaran jelas tentang fokus dan prioritas pengguna terhadap aplikasi ini. "BCA" menjadi pusat perhatian, menunjukkan keterkaitan langsung dengan merek atau lembaga tersebut. "Aplikasi" mendominasi, menyoroti interaksi aktif pengguna dengan fitur dan layanan yang disediakan. "Masuk" mengindikasikan pentingnya proses login dalam pengalaman pengguna. "Transaksi" dan "saldo" menyoroti penggunaan aplikasi untuk kegiatan finansial. Permintaan "tolong" menunjukkan kebutuhan akan layanan pelanggan yang responsif. "Update" menandakan kesadaran pengguna terhadap pembaruan aplikasi. "Gagal" dan "error" menunjukkan kemungkinan frustrasi pengguna terkait kesalahan teknis. "Pulsa" dan "transfer" mencerminkan variasi penggunaan aplikasi untuk transaksi keuangan.



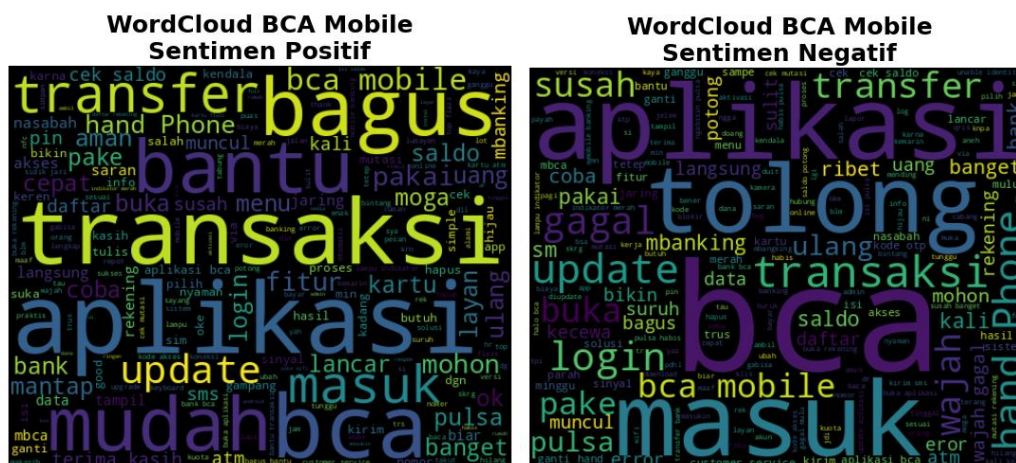
Gambar 5. Frekuensi kata yang sering muncul

Selain itu penelitian ini juga menggunakan teknik visualisasi *Wordcloud* untuk menganalisis teks data. *Wordcloud* akan menampilkan semua kata yang terdapat dalam teks tersebut, sehingga kata-kata yang sering muncul akan ditampilkan dengan ukuran font yang lebih besar dalam visualisasi *Wordcloud*. Ini memungkinkan untuk secara visual mengidentifikasi kata-kata yang paling umum atau sering muncul dalam teks. Gambar 6 menampilkan hasil *Wordcloud* ulasan BCA Mobile semua sentimen menunjukkan kata "BCA" dominan, diikuti "aplikasi", "transaksi", "masuk", "tolong", "transfer", dan "update". Pengguna menyoroti interaksi dan pengalaman mereka dengan aplikasi. Kata "aplikasi" mencerminkan fokus pada *platform*, "transaksi" dan "transfer" pada aspek keuangan. "Masuk" menunjukkan pengalaman akses, "tolong" permintaan bantuan, "update" sensitivitas terhadap perbaikan aplikasi. Ini menegaskan fokus pengguna pada fitur utama BCA Mobile dan respons terhadap pengalaman mereka.



Gambar 6. Hasil *Wordcloud* semua sentimen

Dari visualisasi *Wordcloud* yang mencakup semua sentimen dalam ulasan BCA Mobile, kita beralih ke visualisasi khusus untuk sentimen positif dan negatif. Dapat terlihat pada gambar 7 terdapat perbedaan antara pengalaman yang memuaskan dan masalah yang dihadapi pengguna, memberikan pemahaman yang lebih mendalam tentang pengalaman pengguna secara keseluruhan. Dalam penelitian *Wordcloud* terhadap ulasan BCA Mobile, hasilnya memperlihatkan adanya beragam sentimen yang tercermin melalui kata-kata yang mendominasi dalam ulasan positif dan negatif. Kata-kata seperti "transaksi", "aplikasi", "bagus", dan "update" mencerminkan pengalaman positif pengguna, menunjukkan kepuasan terhadap kemudahan penggunaan dan fungsi aplikasi. Di sisi lain, kata-kata seperti "ribet", "login", dan "eror" menunjukkan adanya ketidakpuasan atau masalah yang dihadapi, mengisyaratkan adanya tantangan dalam penggunaan aplikasi tersebut. Temuan ini menggambarkan variasi pengalaman pengguna BCA Mobile, dimana beberapa merasa puas dengan fitur dan kemudahan aplikasi, sementara yang lain menghadapi tantangan terkait dengan masalah teknis atau kompleksitas dalam penggunaannya.



Gambar 7. Hasil *Wordcloud* sentiment positif dan negatif

IV. KESIMPULAN

Berdasarkan hasil penelitian, kedua model klasifikasi, yaitu *Naïve Bayes* dan *Support Vector Machine* (SVM), menunjukkan kinerja yang baik dalam mengklasifikasikan sentimen ulasan pelanggan terhadap aplikasi BCA Mobile. Meskipun keduanya memiliki tingkat akurasi yang hampir serupa, yaitu 83% untuk *Naïve Bayes* dan 85% untuk SVM, terdapat perbedaan dalam kemampuan masing-masing model dalam mengklasifikasikan sentimen positif dan negatif. *Naïve Bayes* memiliki *recall* yang sedikit lebih rendah untuk ulasan positif sebesar 81% dibandingkan dengan SVM mencapai 85%, namun memiliki *precision* yang sedikit lebih tinggi. Sebaliknya, SVM memiliki *recall* yang lebih rendah untuk ulasan negatif, namun memiliki *precision* yang lebih tinggi. Ini menunjukkan kemampuan SVM dalam menangani distribusi fitur dan kelas yang kompleks, yang tidak dapat ditangani dengan baik oleh *Naïve Bayes*. Selain itu, SVM juga cenderung lebih baik dalam mengidentifikasi pola yang kompleks dan tidak linier dalam data, yang merupakan keunggulan dalam konteks analisis sentimen yang melibatkan data teks yang kompleks dan bervariasi.

DAFTAR PUSTAKA

- [1] A. L. Marshal and A. N. Fajar, "Analisis Faktor Kesalahan Dalam Proses Pengisian Saldo Kedalam Dompot Digital Yang Dituju Dengan Aplikasi BCA Mobile," *Metta J. Ilmu Multidisiplin*, vol. 3, no. 3, pp. 257–267, 2023, doi: 10.37329/metta.v3i3.2695.
- [2] F. Gunawan, M. A. Fauzi, and P. P. Adikara, "Analisis Sentimen Pada Ulasan Aplikasi Mobile Menggunakan Naive Bayes dan Normalisasi Kata Berbasis Levenshtein Distance (Studi Kasus Aplikasi BCA Mobile)," *Syst. Inf. Syst. Informatics J.*, vol. 3, no. 2, pp. 1–6, 2017, doi: 10.29080/systemic.v3i2.234.
- [3] M. Hariyanto, M. Kholiq, A. Yani, and Narti, "Inti nusa mandiri," *Inti Nusa Mandiri*, vol. 14, no. 2, pp. 133–138, 2020.
- [4] A. Z. Amrullah, A. Sofyan Anas, and M. A. J. Hidayat, "Analisis Sentimen Movie Review Menggunakan Naive Bayes Classifier Dengan Seleksi Fitur Chi Square," *Jurnal*, vol. 2, no. 1, pp. 40–44, 2020, doi: 10.30812/bite.v2i1.804.
- [5] D. Ahmad Dzulhijjah, H. Sanjaya, A. Said Wahyudi Hidayat, A. Yulistia Alwanda, and E. Utami, "Perbandingan Metode Random Forest dan KNN pada Analisis Sentimen Twitter," *Smart Comp Jurnalnya Orang Pint. Komput.*, vol. 12, no. 3, pp. 767–772, 2023, doi: 10.30591/smartcomp.v12i3.5106.
- [6] R. Wahyudi and G. Kusumawardana, "Analisis Sentimen pada Aplikasi Grab di Google Play Store Menggunakan Support Vector Machine," *J. Inform.*, vol. 8, no. 2, pp. 200–207, 2021, doi: 10.31294/ji.v8i2.9681.
- [7] L. B. Ilmawan and M. A. Mude, "Perbandingan Metode Klasifikasi Support Vector Machine dan Naïve Bayes untuk Analisis Sentimen pada Ulasan Tekstual di Google Play Store," *Ilk. J. Ilm.*, vol. 12, no. 2, pp. 154–161, 2020, doi: 10.33096/ilkom.v12i2.597.154-161.
- [8] A. Rahman, E. Utami, and S. Sudarmawan, "Sentimen Analisis Terhadap Aplikasi pada Google Playstore Menggunakan Algoritma Naïve Bayes dan Algoritma Genetika," *J. Komtika (Komputasi dan Inform.)*, vol. 5, no. 1, pp. 60–71, 2021, doi: 10.31603/komtika.v5i1.5188.
- [9] F. F. Irfani, "Analisis Sentimen Review Aplikasi Ruangguru Menggunakan Algoritma Support Vector Machine," *JBMI (Jurnal Bisnis, Manajemen, dan Inform.)*, vol. 16, no. 3, pp. 258–266, 2020, doi: 10.26487/jbmi.v16i3.8607.

- [10] Syahril Dwi Prasetyo, Shofa Shofiah Hilabi, and Fitri Nurapriani, "Analisis Sentimen Relokasi Ibukota Nusantara Menggunakan Algoritma Naïve Bayes dan KNN," *J. KomtekInfo*, vol. 10, pp. 1–7, 2023, doi: 10.35134/komtekinfo.v10i1.330.
- [11] A. Novriandy, "Implementasi Algoritma Naive Bayes dan Algoritma C4. 5 dalam Klasifikasi Kelayakan Bantuan UMKM," *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 4, no. 1, pp. 208–217, 2023, doi: 10.30865/klik.v4i1.1099.
- [12] Dhina Nur Fitriana and Yuliant Sibaroni, "Sentiment Analysis on KAI Twitter Post Using Multiclass Support Vector Machine (SVM)," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 4, no. 5, pp. 846–853, 2020, doi: 10.29207/resti.v4i5.2231.
- [13] J. Informasi, D. K. Pramudito, M. Tubagus, A. Ismail, and F. D. Anggraini, "Application of Support Vector Machine and Naive Bayes Method to Analyze Mobile Banking User Behavior Through social media Twitter," vol. 5, pp. 325–330, 2024, doi: 10.60083/jidt.v5i4.462.
- [14] S. Wijanarko and S. A. Santoso, "Penerapan Fungsi Mid Dan Find pada Pembersihan Data Alamat," vol. X, no. 1, pp. 14–18, 2024.
- [15] Darussalam and G. Arief, "Jurnal Resti," *Resti*, vol. 1, no. 1, pp. 19–25, 2018.
- [16] A. N. Khusna, K. P. Delasano, and D. C. E. Saputra, "Penerapan User-Based Collaborative Filtering Algorithm," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 20, no. 2, pp. 293–304, 2021, doi: 10.30812/matrik.v20i2.1124.
- [17] M. S. Anwar, I. M. I. Subroto, and S. Mulyono, "Sistem Pencarian E-Journal Menggunakan Metode Stopword Removal Dan Stemming," *Pros. Konf. Ilm. Mhs. UNISSULA* 2, pp. 58–70, 2019, [Online]. Available: <http://ppm-unissula.com/jurnal.unissula.ac.id/index.php/kimueng/article/viewFile/8420/3887>
- [18] G. A. Trianto, T. Y. Sihotang, M. F. Marzuki, and H. Irsyad, "Klasifikasi Opini Terhadap Resesi Indonesia 2023 pada Twitter Menggunakan Algoritma Decesion Tree," *MDP Student Conf.*, vol. 2, no. 1, pp. 1–9, 2023, doi: 10.35957/mdp-sc.v2i1.3997.
- [19] R. K. Dinata, S. Safwandi, N. Hasdyna, and R. Mahendra, "Kombinasi Algoritma Brute Force dan Stemming pada Sistem Pencarian Mashdar," *CESS (Journal Comput. Eng. Syst. Sci.)*, vol. 5, no. 2, p. 273, 2020, doi: 10.24114/cess.v5i2.17989.
- [20] S. Zhang, O. Jafari, and P. Nagarkar, "A Survey on Machine Learning Techniques for Auto Labeling of Video, Audio, and Text Data," pp. 1–13, 2021, [Online]. Available: <http://arxiv.org/abs/2109.03784>
- [21] V. Alapjân-, "濟無No Title No Title No Title," vol. 5, no. 3, pp. 1–23, 2016.
- [22] R. Nurul Ikhsani and F. Fauzi Abdulloh, "Optimasi SVM dan Decision Tree Menggunakan SMOTE Untuk Mengklasifikasi Sentimen Masyarakat Mengenai Pinjaman Online," *J. Media Inform. Budidarma*, vol. 7, no. 4, pp. 1667–1677, 2023, doi: 10.30865/mib.v7i4.6809.
- [23] Adrian Pirtama, Yuda Prasetia, Redho Irmindo Saputra, and Eko Arip Winanto, "Improvement Attack Detection on Internet of Thinks Using Principal Component Analysis and Random Forest," *Media J. Gen. Comput. Sci.*, vol. 1, no. 1, pp. 14–19, 2023, doi: 10.62205/mjgcs.v1i1.8.
- [24] M. Afriansyah, J. Saputra, V. Yoga, P. Ardhana, and Y. Sa, "Algoritma Naive Bayes Yang Efisien Untuk Klasifikasi Buah," vol. 1, no. 2, pp. 236–248, 2024.
- [25] Sarip. K. And Suryonor. R., "Komparasi Algoritma Support Vector Machine Dan Random Forest Untuk Analisis Sentimen Metaverse", *Mnemonic*, Vol. 7, No. 1, Pp. 31-39, Feb. 2024.
- [26] F. M. Fathoni, C. A. Putra, and A. L. Nurlaili, "Klasifikasi Penyakit Daun Anggur Menggunakan Metode K-Nearest Neighbor Berdasarkan Gray Level Co-Occurrence Matrix," *Biner J. Ilm. Inform. dan Komput.*, vol. 3, no. 1, pp. 8–15, 2024.