

ANALISIS SENTIMEN ULASAN APLIKASI SIREKAP MENGGUNAKAN ALGORITMA SUPPORT VECTOR MACHINE

Muhammad Zainul Abidin^{1*}, Ronny Makhfuddin², Achmad Syaifuddin³

^{1,2,3} Universitas Islam Majapahit, Jl. Raya Jabon No.KM.0,7, Tambak Rejo, Gayaman, Kec. Mojoanyar, Kabupaten Mojokerto, Jawa Timur 61364, [\(0321\) 399474](tel:0321399474)

Keywords:

Analisis Sentimen, SIREKAP, Google Play Store, Support Vector Machine, Pemilu.

Correspondent Email:

mz.abidin@unim.ac.id

Abstrak. SIREKAP merupakan aplikasi resmi Komisi Pemilihan Umum (KPU) yang digunakan untuk rekapitulasi suara secara digital pada Pemilu 2024. Sejak dirilis di Google Play Store, aplikasi ini mendapat berbagai tanggapan dari pengguna, mulai dari pujian hingga keluhan. Penelitian ini bertujuan untuk menganalisis sentimen ulasan pengguna terhadap aplikasi SIREKAP dengan menerapkan algoritma *Support Vector Machine* (SVM). Data yang dianalisis sebanyak 15.000 ulasan berbahasa Indonesia yang diperoleh melalui teknik web scraping. Proses penelitian meliputi preprocessing teks (cleaning, tokenizing, dan stemming), pelabelan sentimen menjadi positif, negatif, dan netral, serta pelatihan model SVM dengan kernel linear. Evaluasi performa model dilakukan menggunakan *confusion matrix* berdasarkan metrik akurasi, presisi, dan recall. Hasil menunjukkan bahwa model SVM mampu mengklasifikasikan sentimen dengan akurasi tinggi mencapai 99%. Penelitian ini memberikan gambaran persepsi publik terhadap aplikasi SIREKAP dan dapat menjadi bahan evaluasi bagi pengembang maupun penyelenggara pemilu ke depan.

Abstract. SIREKAP is the official application of the General Election Commission (KPU) used for digital vote tabulation in the 2024 elections. Since its release on the Google Play Store, the application has received various responses from users, ranging from praise to complaints. This study aims to analyze user sentiment toward the SIREKAP application by applying the Support Vector Machine (SVM) algorithm. The data analyzed consists of 15,000 Indonesian-language reviews obtained through web scraping techniques. The research process includes text preprocessing (cleaning, tokenizing, and stemming), sentiment labeling into positive, negative, and neutral categories, and training the SVM model with a linear kernel. Model performance evaluation was conducted using a confusion matrix based on accuracy, precision, and recall metrics. The results showed that the SVM model was able to classify sentiment with high accuracy, reaching 99%. This study provides an overview of public perception of the SIREKAP application and can serve as evaluation material for developers and election organizers in the future.



Copyright © [JITET](http://www.jitet.org) (Jurnal Informatika dan Teknik Elektro Terapan). This article is an open access article distributed under terms and conditions of the Creative Commons Attribution (CC BY NC)

1. PENDAHULUAN

Pemilihan Umum (Pemilu) di Indonesia merupakan salah satu pesta demokrasi terbesar di dunia, melibatkan ratusan juta pemilih yang tersebar di ribuan Tempat Pemungutan Suara

(TPS). Kompleksitas proses rekapitulasi suara pada skala nasional menimbulkan tantangan besar, khususnya dalam aspek kecepatan, akurasi, transparansi, dan akuntabilitas hasil pemilu. Untuk menjawab tantangan tersebut,

Komisi Pemilihan Umum (KPU) meluncurkan Sistem Informasi Rekapitulasi Elektronik (SIREKAP) pada Pemilu 2024 sebagai solusi digitalisasi proses rekapitulasi suara.

SIREKAP dirancang untuk mempercepat proses pengumpulan dan perhitungan suara, sekaligus meminimalkan potensi kesalahan manusia (human error) yang dapat terjadi dalam proses manual. Namun, implementasi teknologi baru pada skala masif tidak lepas dari tantangan adopsi dan penerimaan publik. Sejak dirilis di Google Play Store, aplikasi ini memperoleh berbagai tanggapan dari pengguna, mulai dari apresiasi terhadap inovasi digital hingga keluhan terkait antarmuka pengguna, kecepatan respons sistem, dan kendala teknis lainnya.[1]

Ulasan-ulasan yang dipublikasikan di Google Play Store merupakan sumber data yang kaya untuk memahami persepsi publik. Data ini mencerminkan pengalaman langsung pengguna, baik dalam aspek fungsionalitas, kemudahan penggunaan, maupun tingkat kepuasan terhadap layanan. Oleh karena itu, diperlukan metode analisis yang mampu mengolah data teks dalam jumlah besar untuk memperoleh wawasan yang mendalam.[2]

Salah satu pendekatan yang umum digunakan dalam mengolah opini publik berbasis teks adalah analisis sentimen. Teknik ini bertujuan mengidentifikasi polaritas emosi dalam teks, yaitu apakah bersifat positif, negatif, atau netral. Dalam penelitian ini, metode Support Vector Machine (SVM) dipilih sebagai algoritma utama. SVM memiliki reputasi baik dalam menangani data berdimensi tinggi seperti teks dan dikenal memiliki kemampuan generalisasi yang kuat pada tugas klasifikasi.[3]

Tujuan penelitian ini adalah mengklasifikasikan sentimen ulasan pengguna aplikasi SIREKAP ke dalam tiga kategori utama positif, negatif, dan netral serta mengukur kinerja model menggunakan berbagai metrik evaluasi, termasuk akurasi, presisi, recall, dan F1-score.[4] Hasil analisis ini diharapkan dapat menjadi masukan berbasis data bagi KPU dan pengembang aplikasi untuk meningkatkan kualitas layanan SIREKAP, sekaligus menjadi acuan bagi penelitian selanjutnya dalam penerapan pembelajaran mesin pada analisis opini publik.[5]

2. TINJAUAN PUSTAKA

Bab ini menjelaskan konsep-konsep dasar yang menjadi fondasi dalam penelitian, seperti teori analisis sentimen, algoritma *Support Vector Machine (SVM)*, metode vektorisasi TF-IDF, dan teknik evaluasi menggunakan *confusion matrix*. Pemahaman terhadap teori-teori ini penting untuk mendukung validitas pendekatan yang digunakan dalam penelitian.

2.1 Analisis Sentimen

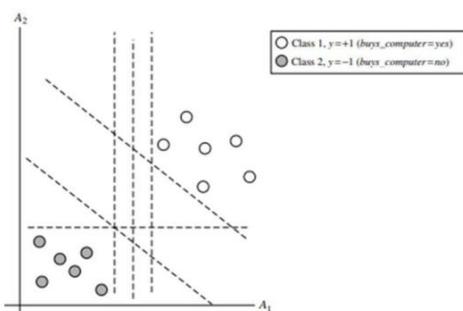
Analisis sentimen adalah proses otomatis untuk mengidentifikasi, mengekstraksi, dan mengkategorikan opini, penilaian, atau emosi yang diekspresikan dalam suatu teks, terutama untuk menentukan apakah sikap penulis terhadap suatu topik bersifat positif, negatif, atau netral. Dikenal juga dengan istilah *opinion mining*, teknik ini merupakan bagian dari *text mining* dan *Natural Language Processing (NLP)* yang berfokus pada pemrosesan data teks tidak terstruktur, seperti komentar di media sosial, forum diskusi, blog, atau situs ulasan.

Dalam praktiknya, analisis sentimen banyak dimanfaatkan untuk mengolah opini yang tersebar di berbagai platform digital seperti Facebook dan Twitter, lalu dikonversi menjadi informasi yang terstruktur dan berguna, seperti sistem penilaian atau peringkat. Hal ini penting terutama dalam dunia bisnis, di mana rating menjadi tolok ukur keberhasilan produk atau layanan. Namun, sistem penilaian konvensional yang sering kali dikelola oleh lembaga besar seperti Nielsen dapat memunculkan keraguan terhadap objektivitasnya. Oleh karena itu, analisis sentimen berbasis komentar pengguna hadir sebagai alternatif yang lebih transparan dan langsung, dengan menggali opini publik secara real time dari sumber-sumber digital.[6]

2.2 Support Vector Machine (SVM)

Support Vector Machine (SVM) adalah algoritma supervised learning yang digunakan secara luas dalam permasalahan klasifikasi maupun regresi. Dalam konteks klasifikasi, terutama klasifikasi teks, SVM dianggap sangat efektif karena mampu menghasilkan akurasi tinggi dan bekerja dengan baik pada data berdimensi tinggi.[7]

Tujuan utama dari SVM adalah mencari hyperplane terbaik yang dapat memisahkan data ke dalam dua kelas atau lebih secara optimal. Hyperplane adalah sebuah bidang dalam ruang berdimensi tinggi yang berfungsi sebagai pemisah antara kelas-kelas data. Optimalitas hyperplane ditentukan oleh sejauh mana ia memaksimalkan margin, yaitu jarak antara hyperplane dengan titik data terdekat dari masing-masing kelas. Semakin besar margin yang tercipta, semakin baik kemampuan generalisasi model terhadap data baru.



Gambar 2. 1 SVM memisahkan dua data dengan Hyperlane

Gambar 2.1 memperlihatkan ilustrasi pemisahan dua kelas data oleh algoritma Support Vector Machine (SVM) dalam ruang dua dimensi, dengan sumbu A_1 dan A_2 sebagai fitur.

Titik-titik pada gambar merepresentasikan dua kelas:

Class 1 ($y = +1$) ditunjukkan oleh lingkaran putih (\circ), yang dalam konteks ini merepresentasikan `buys_computer = yes`.

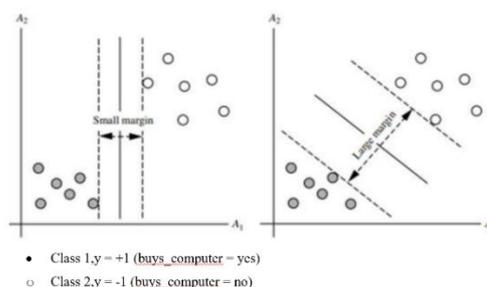
Class 2 ($y = -1$) ditunjukkan oleh lingkaran hitam (\bullet), yang mewakili `buys_computer = no`.

Garis hitam putus-putus di tengah merupakan hyperplane utama yang berfungsi sebagai batas pemisah antara kedua kelas. Dua garis sejajar di kedua sisi hyperplane menunjukkan margin, yaitu jarak antara hyperplane dan titik data terdekat dari masing-masing kelas, yang dikenal sebagai support vectors. SVM akan memilih hyperplane dengan margin terluas untuk meningkatkan kemampuan generalisasi model terhadap data baru.

Dengan kata lain, SVM tidak hanya mencari pemisah antar kelas, tetapi juga memastikan bahwa pemisahan tersebut dilakukan dengan jarak maksimum dari masing-masing kelas

untuk mengurangi kemungkinan kesalahan klasifikasi.

Namun, tidak semua data dapat dipisahkan secara linear. Dalam kasus di mana data memiliki distribusi non-linear, SVM menggunakan pendekatan yang disebut kernel trick. Teknik ini memetakan data ke ruang berdimensi lebih tinggi, di mana pemisahan linear antar kelas menjadi lebih memungkinkan. Dengan demikian, SVM tidak hanya fokus pada pemisahan kelas, tetapi juga pada pemilihan hyperplane dengan margin maksimum (*Maximum Margin Hyperplane/MMH*), yang memungkinkan model untuk melakukan generalisasi lebih baik pada data yang belum pernah dilihat sebelumnya.



Gambar 2. 2 Dua hyperlane dalam satu data

Gambar 2.2 memperlihatkan dua ilustrasi perbandingan antara hyperplane dengan margin sempit (*small margin*) dan margin lebar (*large margin*) dalam konteks klasifikasi menggunakan *Support Vector Machine (SVM)*.

Gambar sebelah kiri menunjukkan hyperplane dengan margin yang sempit, di mana garis pemisah sangat dekat dengan titik data dari kedua kelas. Meskipun data masih dapat dipisahkan, margin yang sempit meningkatkan risiko overfitting karena model terlalu sensitif terhadap data latih.

Gambar sebelah kanan menggambarkan hyperplane dengan margin yang lebih lebar, atau disebut juga *Maximum Margin Hyperplane (MMH)*. Ini adalah pendekatan ideal yang diupayakan oleh SVM, karena margin yang lebar mengindikasikan generalisasi yang lebih baik terhadap data baru, sekaligus mengurangi risiko overfitting.

Pada kedua ilustrasi, titik-titik mewakili dua kelas:

Lingkaran putih (\circ): Class 1 ($y = +1$)

Lingkaran hitam (\bullet): Class 2 ($y = -1$)

Kesimpulannya, pemilihan margin yang optimal sangat penting dalam algoritma SVM, karena memengaruhi kinerja model dalam memisahkan kelas serta kemampuannya untuk menangani data baru di luar data pelatihan.

2.3 Term Frequency-Inverse Document Frequency (TF-IDF)

Term Frequency-Inverse Document Frequency (TF-IDF) adalah salah satu metode yang umum digunakan untuk merepresentasikan data teks dalam bentuk vektor numerik. Metode ini berperan penting dalam pemrosesan bahasa alami (*Natural Language Processing/NLP*), khususnya dalam tahap ekstraksi fitur untuk analisis teks, seperti klasifikasi atau analisis sentimen.

TF-IDF bekerja dengan memberikan bobot pada setiap kata berdasarkan dua komponen utama:

Term Frequency (TF), yang mengukur seberapa sering suatu kata (term) muncul dalam sebuah dokumen. Semakin sering kata muncul, semakin besar nilai TF-nya, sehingga menunjukkan relevansi kata tersebut dalam konteks lokal dokumen tersebut.

Inverse Document Frequency (IDF), yang mengukur tingkat kekhususan suatu kata dengan menghitung seberapa jarang kata tersebut muncul di seluruh koleksi dokumen (corpus). Kata-kata yang jarang muncul di banyak dokumen dianggap lebih bermakna dan akan memiliki bobot IDF yang lebih tinggi.

Kombinasi kedua komponen tersebut menghasilkan nilai TF-IDF, yang dapat dihitung menggunakan rumus berikut:

$$TF - IDF(t, d, D) = TF(t, d) \times IDF(t, D) \quad 2.1$$

Keterangan :

TF-IDF(t, d, D) = Pembobotan TF-IDF
 TF(t,d) = Jumlah kemunculan term dalam dokumen

IDF(t,d) = Bobot inverse dalam nilai df

Dengan pendekatan ini, TF-IDF mampu menyeimbangkan antara kepentingan lokal suatu kata dalam sebuah dokumen dan distribusinya secara global dalam korpus. Kata-kata yang sering muncul dalam satu dokumen namun jarang ditemukan di dokumen lain akan

memiliki bobot yang tinggi, karena dianggap memiliki nilai informasi yang tinggi. Sebaliknya, kata-kata umum yang muncul hampir di semua dokumen (seperti "dan", "yang", "adalah") akan diberikan bobot rendah karena kurang bernilai dalam membedakan antar dokumen.[8]

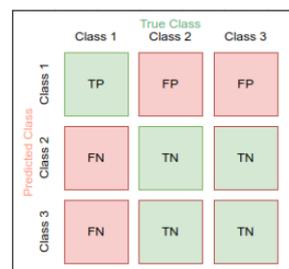
2.4 Confusion Matrix

Pada tahap akhir penelitian ini, efektivitas model dievaluasi dengan memanfaatkan *Confusion Matrix*. *Confusion Matrix* memiliki peranan penting dalam mengukur kinerja model klasifikasi dengan cara membandingkan hasil prediksi model terhadap label sebenarnya.[9] Matriks ini menyajikan informasi dalam bentuk empat komponen utama, yaitu:

1. *True Positive (TP)*: Kasus positif yang diprediksi dengan benar sebagai positif.
2. *True Negative (TN)*: Kasus negatif yang diprediksi dengan benar sebagai negatif.
3. *False Positive (FP)*: Kasus negatif yang salah diprediksi sebagai positif.
4. *False Negative (FN)*: Kasus positif yang salah diprediksi sebagai negatif.

Nilai-nilai ini selanjutnya digunakan untuk menghitung sejumlah metrik evaluasi, seperti akurasi, presisi, recall, dan F1-score, yang memberikan gambaran menyeluruh mengenai performa model klasifikasi. Evaluasi ini sangat penting karena dapat mengidentifikasi kekuatan dan kelemahan model, serta memberikan informasi yang berguna untuk perbaikan di masa mendatang.

Gambar 2.3 berikut menggambarkan konsep penggunaan *Confusion Matrix* secara visual, termasuk penerapannya dalam kasus klasifikasi multi-kelas.



Gambar 2. 3 Visualisasi Confusion Matrik

Rumus yang digunakan dalam menghitung performa klasifikasi dengan confusion matrix dapat dilihat pada persamaan 2.2,2.3,2.4,dan 2.5

1. *Accuracy* merupakan pengukuran jumlah prediksi yang benar sesuai data secara keseluruhan.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad 2.2$$

2. *Precision* merupakan persentase dokumen dengan teks yang relevan di antara semua dokumen yang telah dipilih oleh sistem.

$$Precision = \frac{TP}{TP + F} \quad 2.3$$

3. *Recall* merupakan ukuran yang menggambarkan sejauh mana sistem mampu mengidentifikasi seluruh dokumen teks yang relevan dalam dataset, yaitu dengan membandingkan jumlah dokumen relevan yang berhasil terdeteksi dengan total keseluruhan dokumen relevan yang ada (Husada & Paramita, 2021).

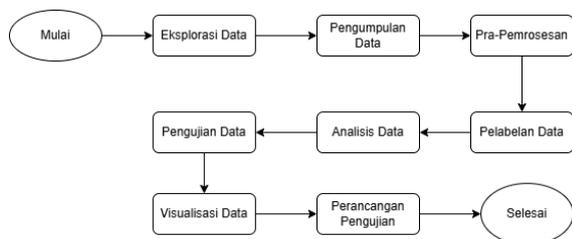
$$Recall = \frac{TP}{TP + FN} \quad 2.4$$

4. *F1-Score* merupakan proses penggabungan precision dan recall pada matriks evaluasi.

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad 2.5$$

3. METODE PENELITIAN

Bab ini memaparkan tahapan-tahapan penelitian yang dilakukan secara sistematis, mulai dari pengumpulan data, proses pra-pemrosesan, pelabelan sentimen, vektorisasi, pelatihan model, hingga evaluasi performa klasifikasi. Metodologi ini dirancang untuk memastikan hasil analisis yang akurat dan dapat direproduksi.



Gambar 3. 1 Metode Penelitian

Gambar tersebut menunjukkan alur metodologi penelitian yang terdiri dari beberapa tahapan penting dalam proses analisis

sentimen ulasan aplikasi SIREKAP. Proses dimulai dari tahap Eksplorasi Data, yaitu memahami karakteristik data serta meninjau sumber data yang akan digunakan. Setelah itu, dilakukan Pengumpulan Data melalui metode web scraping dari Google Play Store, khususnya pada ulasan pengguna aplikasi SIREKAP. Data yang telah diperoleh kemudian masuk ke tahap Pra-Pemrosesan, yang meliputi pembersihan teks, case folding, tokenisasi, penghapusan stopword, dan stemming untuk menghasilkan data yang bersih dan siap diproses lebih lanjut.

Tahap berikutnya adalah Pelabelan Data, di mana setiap ulasan diklasifikasikan ke dalam sentimen positif, negatif, atau netral menggunakan pendekatan berbasis kamus (lexicon-based). Setelah data diberi label, dilakukan Analisis Data menggunakan algoritma *Support Vector Machine (SVM)*, di mana model dilatih untuk mengenali pola sentimen berdasarkan fitur yang dihasilkan dari proses vektorisasi. Model yang telah dilatih kemudian diuji pada tahap Pengujian Data, untuk mengevaluasi kinerjanya menggunakan metrik seperti akurasi, presisi, recall, dan F1-score.

Selanjutnya, hasil dari analisis dan pengujian disajikan dalam bentuk grafik pada tahap Visualisasi Data, sehingga lebih mudah dipahami dan dibandingkan. Terakhir, dilakukan Perancangan Pengujian, yaitu penyusunan interpretasi hasil dan penarikan kesimpulan dari seluruh proses yang telah dilakukan. Penelitian diakhiri pada tahap Selesai, dengan menyimpulkan temuan utama dan memberikan rekomendasi untuk pengembangan selanjutnya.

3.1. Pengumpulan Data

Data dalam penelitian ini dikumpulkan dari platform Google Play Store dengan menggunakan metode web scraping, yakni teknik ekstraksi data secara otomatis dari halaman web. Untuk keperluan ini, digunakan pustaka *google-play-scrapers* yang memungkinkan pengambilan informasi aplikasi secara sistematis, termasuk detail ulasan pengguna.

Proses scraping dilaksanakan di lingkungan Google Colaboratory (Google Colab), yang menyediakan kemampuan komputasi berbasis cloud serta dukungan

eksekusi kode Python secara efisien. Melalui proses ini, berhasil diperoleh sebanyak 15.000 ulasan pengguna terhadap aplikasi SIREKAP yang dikembangkan oleh Komisi Pemilihan Umum (KPU) Republik Indonesia. Seluruh ulasan dikumpulkan dalam Bahasa Indonesia, sehingga relevan untuk dianalisis dalam konteks linguistik dan sosial pengguna lokal.

3.2. Pra-Pemrosesan Data

Tahapan *preprocessing* bertujuan membersihkan dan menormalkan teks ulasan agar siap dianalisis. Proses ini meliputi penghapusan data kosong, duplikat, elemen tidak relevan, serta normalisasi huruf dan ejaan. URL, *mention*, angka, simbol, dan *emoji* ditangani sesuai relevansi, sedangkan kata tidak baku dan singkatan diseragamkan. Tokenisasi dilakukan untuk memecah teks, diikuti penanganan negasi, penghapusan *stopword* penting secara selektif, dan stemming untuk mengembalikan kata ke bentuk dasar.[10] Pembuatan *n-grams* digunakan untuk menangkap makna frasa, seluruh langkah diintegrasikan ke dalam *pipeline*, dan hasilnya diperiksa melalui laporan statistik sebelum dan sesudah *preprocessing*. Oleh karena itu, dilakukan serangkaian tahapan *preprocessing* sebagai berikut:

1. Case Folding

Proses ini mengubah seluruh karakter dalam teks menjadi huruf kecil (*lowercase*), sehingga kata-kata seperti "Baik", "baik", dan "BAIK" dianggap sama.[11] Langkah ini membantu menyederhanakan representasi kata dan mengurangi duplikasi yang tidak diperlukan.

Tabel 3. 1 Case Folding

| Sebelum | Sesudah |
|---|---|
| Sirekap pilkada Belum bisa Diupdate dr Versi Lama | sirekap pilkada belum bisa diupdate dr versi lama |

2. Pembersihan Teks (Cleaning)

Tahapan ini mencakup penghapusan karakter atau simbol yang tidak relevan seperti angka, tanda baca, emotikon, URL, dan karakter khusus lainnya. Tujuannya adalah untuk menyisakan hanya teks yang memiliki makna linguistik bagi analisis.[12]

Tabel 3. 2 Cleaning

| Sebelum | Sesudah |
|---|---|
| Sirekap pilkada belum bisa diupdate dr versi lama 🙏 | sirekap pilkada belum bisa diupdate dr versi lama |

3. Tokenisasi

Tokenisasi memecah teks menjadi unit-unit kata (*token*). Misalnya, kalimat "Aplikasi ini sangat berguna" akan diubah menjadi token: ["aplikasi", "ini", "sangat", "berguna"]. Langkah ini penting untuk memudahkan analisis berdasarkan frekuensi dan struktur kata.

Tabel 3. 3 Tokenisasi

| Sebelum | Sesudah |
|--|--|
| banyak penambahan fitur, terutama progres, | [banyak, penambahan, fitur, terutama, progres] |
| sirekap pilkada belum bisa diupdate | [sirekap, pilkada, belum, bisa, diupdate] |
| sangat membantu sekali | [sangat, membantu, sekali] |

4. Stopword Removal

Stopword adalah kata-kata umum yang tidak memiliki kontribusi signifikan terhadap makna utama teks, seperti "yang", "dan", "di", "adalah", dll. Dengan menghapus *stopword*, proses analisis dapat lebih fokus pada kata-kata kunci yang memiliki nilai informasi lebih tinggi.[13]

Tabel 3. 4 Stopword Removal

| Sebelum | Sebelum |
|--|---|
| aplikasi,sudah,lebih,baik ,daripada,sirekap,pilpres, saran,saya,untuk,kamera, lebih,baik,menggunakan, setting,mode,kamera,biasa,saja,daripada,kamera, mode,malam,karena,di,banyak, device,lebih bagus, hasil, potret, menggunakan, kamera, biasa,walaupun ,saat,malam,hari | aplikasi,sirekap,pilpres ,saran,kamera, setting,mode,kamera,kamera,mode, malam,device,bagus,hasil,potret,kamera,malam |

5. Stemming

Stemming bertujuan untuk mengembalikan setiap kata ke bentuk dasarnya (*root word*). Contohnya, kata "bermain", "bermainlah", dan

“memainkan” akan distem menjadi “main”. Proses ini berguna untuk mengurangi variasi kata dan meningkatkan efisiensi pencocokan istilah selama proses analisis.[14]

Tabel 3. 5 Steaming

| Sebelum | Sesudah |
|--|--|
| banyak penambahan fitur, terutama progres, sungguh sangat membantu di pilkada, semoga pemilu kedepan semakin canggih ... tengkyu kpu | 'tambah', 'fitur', 'progres', 'sungguh', 'bantu', 'pilkada', 'moga', 'milu', 'depan', 'canggih', 'tengkyu', 'kpu'] |

3.3. Pelabelan Sentimen

Proses pelabelan sentimen dilakukan secara otomatis menggunakan pendekatan lexicon-based, yaitu metode berbasis kamus polaritas kata yang telah dikategorikan sebelumnya ke dalam sentimen positif maupun negatif. Setiap ulasan dianalisis dengan menghitung skor polaritas berdasarkan jumlah dan intensitas kata-kata yang memiliki makna sentimen sesuai dengan daftar yang terdapat dalam kamus (lexicon) tersebut.

Skor polaritas dihitung dengan menjumlahkan bobot kata-kata bernuansa positif dan negatif dalam setiap ulasan. Adapun aturan klasifikasi yang digunakan adalah sebagai berikut:

1. Nilai mendekati -1 menunjukkan bahwa ulasan memiliki kecenderungan sentimen negatif yang kuat, ditandai oleh dominasi kata-kata bernuansa negatif.
2. Nilai mendekati 1 mengindikasikan sentimen positif yang kuat, di mana kata-kata positif lebih dominan dalam teks.
3. Nilai mendekati 0 mencerminkan sentimen netral, karena jumlah kata positif dan negatif relatif seimbang atau tidak menunjukkan kecenderungan emosi yang jelas.

Penentuan polaritas ini mempertimbangkan frekuensi kemunculan serta tingkat intensitas dari masing-masing kata berdasarkan kamus lexicon yang digunakan. Dengan pendekatan ini, proses pelabelan dapat dilakukan secara efisien dan konsisten pada seluruh data ulasan, tanpa keterlibatan manual dalam interpretasi sentimen.

3.4. Vektorasi Data

Setelah proses pelabelan sentimen dilakukan, tahap selanjutnya adalah transformasi data teks ke dalam format numerik agar dapat diolah oleh algoritma pembelajaran mesin. Dalam penelitian ini, digunakan metode *Term Frequency–Inverse Document Frequency (TF-IDF)* sebagai teknik representasi fitur. TF-IDF merupakan salah satu pendekatan populer dalam pemrosesan bahasa alami (*Natural Language Processing/NLP*) yang digunakan untuk mengukur pentingnya suatu kata dalam dokumen tertentu relatif terhadap kumpulan dokumen (korpus) secara keseluruhan.

Secara teknis, Term Frequency (TF) menghitung seberapa sering sebuah kata muncul dalam satu dokumen, yang mencerminkan relevansi kata tersebut secara lokal. Sementara itu, Inverse Document Frequency (IDF) mengukur seberapa jarang kata tersebut muncul di seluruh korpus. Kombinasi keduanya memberikan bobot yang lebih tinggi pada kata-kata yang sering muncul dalam satu dokumen tetapi jarang muncul di dokumen lain, sehingga dianggap lebih bermakna dan khas dalam konteks analisis.

Proses ini menghasilkan matriks fitur numerik berdimensi tinggi, di mana setiap baris merepresentasikan satu ulasan, dan setiap kolom mewakili kata unik dalam korpus. Bobot TF-IDF yang dihasilkan mencerminkan kekuatan representasi kata terhadap makna keseluruhan ulasan, yang kemudian digunakan sebagai input dalam pelatihan model klasifikasi sentimen.

3.5. Pelatihan Model SVM

Setelah proses preprocessing, pelabelan sentimen, dan transformasi data menggunakan TF-IDF selesai dilakukan, tahap selanjutnya adalah pelatihan model klasifikasi. Dalam penelitian ini, digunakan algoritma Support Vector Machine (SVM), yang dikenal efektif dalam menangani permasalahan klasifikasi, khususnya pada data teks berdimensi tinggi.

Model SVM dilatih menggunakan kernel linear, karena berdasarkan hasil vektorisasi fitur, data ulasan menunjukkan pola distribusi yang cenderung linier separable. Pemilihan kernel linear juga mempertimbangkan efisiensi komputasi dan stabilitas performa dalam pengolahan data berbasis teks.

Untuk mengevaluasi performa model secara menyeluruh, dilakukan pelatihan dan pengujian dengan lima skenario pembagian data, yaitu:

1. 50% : 50%
2. 60% : 40%
3. 70% : 30%
4. 80% : 20%
5. 90% : 10%

Setiap skenario dirancang untuk menguji kestabilan, ketahanan, dan akurasi model terhadap variasi jumlah data latih dan data uji. Proses pembagian dilakukan secara acak (randomized split) guna menjaga proporsi distribusi kelas yang seimbang dan menghindari bias pada subset data.[15]

Model SVM dilatih menggunakan parameter default dari pustaka scikit-learn, dengan melakukan penyesuaian terhadap nilai parameter regularisasi (C). Parameter ini mengontrol keseimbangan antara margin maksimum dan tingkat kesalahan klasifikasi. Nilai C yang terlalu tinggi dapat menyebabkan model overfitting, sementara nilai yang terlalu rendah dapat menghasilkan margin yang terlalu lebar dan meningkatkan risiko underfitting. Oleh karena itu, pemilihan nilai C dilakukan secara empiris berdasarkan hasil pengujian awal untuk memperoleh performa model yang optimal dan generalisasi yang baik.

Proses pelatihan dijalankan di lingkungan Google Colaboratory dengan memanfaatkan pustaka Python seperti scikit-learn, yang menyediakan implementasi SVM secara efisien, fleksibel, dan terintegrasi dengan pipeline pemrosesan data teks.

4. HASIL DAN PEMBAHASAN

Bab ini menyajikan hasil implementasi algoritma Support Vector Machine (SVM) terhadap data ulasan pengguna aplikasi SIREKAP, yang diperoleh melalui proses pengumpulan, preprocessing, pelabelan sentimen, hingga transformasi fitur dengan metode TF-IDF. Fokus utama dari bab ini adalah untuk mengevaluasi kemampuan model dalam mengklasifikasikan sentimen dan memahami persepsi pengguna berdasarkan distribusi kelas sentimen dalam data.

Model SVM diimplementasikan untuk mengklasifikasikan data ke dalam tiga kategori

sentimen, yaitu positif, negatif, dan netral. Evaluasi dilakukan dengan menggunakan confusion matrix dan metrik evaluasi standar berupa akurasi, presisi, recall, dan F1-score. Nilai-nilai tersebut digunakan untuk menilai efektivitas model dalam mengenali pola sentimen dari ulasan berbasis teks.

Berdasarkan hasil pengujian akhir dengan konfigurasi pembagian data 80% data latih dan 20% data uji, model SVM menunjukkan performa klasifikasi yang sangat tinggi dan stabil, dengan hasil sebagai berikut:

1. Akurasi: 99%
2. Presisi (Precision): 0,99
3. Recall: 0,99
4. F1-Score: 0,99

Hasil ini mencerminkan bahwa model memiliki kemampuan yang sangat baik dalam membedakan masing-masing kelas sentimen, serta mampu menggeneralisasi terhadap data uji secara efektif. Selain itu, nilai metrik yang seimbang antara presisi dan recall menunjukkan bahwa model tidak hanya tepat dalam memberikan prediksi, tetapi juga cukup sensitif dalam mendeteksi semua data relevan dalam masing-masing kelas.

Temuan ini memperkuat bahwa algoritma SVM, terutama dengan pendekatan kernel linear dan fitur TF-IDF, merupakan metode yang efektif dan efisien dalam menangani masalah klasifikasi teks berbasis sentimen dalam konteks aplikasi pemerintahan seperti SIREKAP.

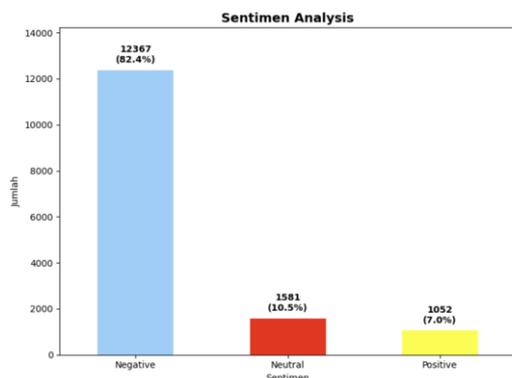
4.1 Distribusi Sentimen

Analisis distribusi sentimen dilakukan terhadap 15.000 ulasan pengguna aplikasi SIREKAP yang dikumpulkan dari Google Play Store. Hasil klasifikasi menunjukkan bahwa mayoritas ulasan pengguna mengandung sentimen negatif, dengan rincian sebagai berikut:

1. Sentimen Negatif: 12.367 ulasan (82,4%)
2. Sentimen Netral: 1.581 ulasan (10,5%)
3. Sentimen Positif: 1.052 ulasan (7%)

Persentase yang sangat tinggi pada sentimen negatif mencerminkan bahwa pengguna banyak mengalami kendala teknis, terutama yang berkaitan dengan kesulitan saat

login, lambatnya respon sistem, dan desain antarmuka yang dinilai kurang intuitif. Hal ini mengindikasikan perlunya perbaikan dan pengembangan lebih lanjut dari sisi teknis maupun pengalaman pengguna (user experience) pada aplikasi. Visualisasi distribusi sentimen dapat dilihat pada Gambar 4.1, yang menunjukkan dominasi ulasan negatif secara jelas dibanding dua kategori sentimen lainnya.



Gambar 4. 1 Distribusi Label Sentimen.

4.2 Evaluasi Kinerja Mode

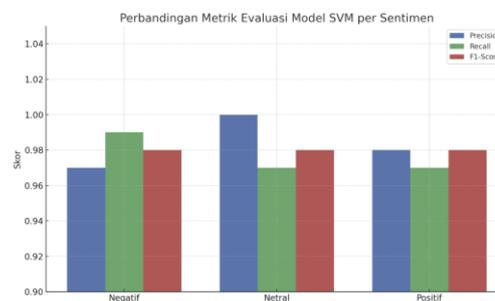
Setelah model SVM dilatih dan diuji, langkah selanjutnya adalah mengevaluasi sejauh mana model mampu mengklasifikasikan sentimen dari ulasan pengguna aplikasi SIREKAP. Evaluasi ini dilakukan menggunakan empat metrik utama: akurasi, presisi, recall, dan F1-score, yang masing-masing memberikan gambaran berbeda tentang kinerja model.

Hasil evaluasi menunjukkan bahwa model bekerja sangat baik dan stabil di semua kategori sentimen. Berikut adalah nilai metrik untuk masing-masing kelas:

1. Negatif → Precision: 0,97 | Recall: 0,99 | F1-Score: 0,98
2. Netral → Precision: 1,00 | Recall: 0,97 | F1-Score: 0,98
3. Positif → Precision: 0,98 | Recall: 0,98 | F1-Score: 0,98
4. Akurasi Keseluruhan: 99%

Dari hasil tersebut, terlihat bahwa model tidak hanya mampu mengenali ulasan dengan sangat baik, tapi juga konsisten dalam memetakan sentimen, baik itu negatif, netral, maupun positif. Nilai F1-score yang nyaris identik di ketiga kelas juga menunjukkan bahwa model tidak berat sebelah dan mampu menangani perbedaan distribusi data dengan cukup adil.

Visualisasi lengkap dari hasil evaluasi ini ditampilkan pada Gambar 4.2. Grafik tersebut memperlihatkan perbandingan metrik antar kelas, dan menegaskan bahwa performa model memang tinggi di seluruh aspek evaluasi.



Gambar 4. 2 Grafik Perbandingan Evaluasi Model.

Hasil ini menunjukkan bahwa pendekatan yang digunakan dalam penelitian ini mulai dari proses pembersihan data (preprocessing), transformasi teks menggunakan TF-IDF, hingga pemilihan algoritma SVM dengan kernel linear terbukti efektif dalam memahami dan mengklasifikasikan opini pengguna dalam bahasa Indonesia.

Dengan tingkat akurasi dan konsistensi yang tinggi, model ini sangat layak untuk diterapkan dalam konteks serupa, seperti analisis komentar di platform digital lainnya, khususnya yang berkaitan dengan pelayanan publik atau aplikasi pemerintahan.

4.3 Implementasi Hasil

Evaluasi terhadap performa model klasifikasi dilakukan dengan menggunakan metrik yang diperoleh dari *Confusion Matrix*, yaitu Akurasi, Presisi (Precision), Recall, dan F1-Score. Metrik-metrik ini bertujuan untuk memberikan gambaran menyeluruh mengenai kemampuan model dalam mengklasifikasikan data secara tepat dan konsisten.

1. Akurasi adalah proporsi prediksi yang benar dibandingkan dengan jumlah total data, yang mencerminkan ketepatan keseluruhan model.
2. Presisi mengukur ketepatan model dalam memprediksi kelas positif, yaitu seberapa banyak dari hasil prediksi positif yang benar-benar relevan.
3. Recall mengukur sensitivitas model, yaitu seberapa banyak data positif yang berhasil diidentifikasi dengan benar dari seluruh data positif yang tersedia.

4. F1-Score adalah rata-rata harmonik dari presisi dan recall, yang berguna dalam situasi di mana terdapat ketidakseimbangan kelas.

Untuk menilai kestabilan dan ketahanan model, dilakukan evaluasi dengan lima skenario pembagian data latih dan uji, yakni: 50:50, 60:40, 70:30, 80:20, dan 90:10. Evaluasi dilakukan pada setiap rasio untuk mengetahui sejauh mana perubahan proporsi data mempengaruhi kinerja model. Hasil evaluasi disajikan dalam Tabel 3.6 berikut:

Tabel 4. 1 Hasil Evaluasi

| Rasio Train : Test | Jumlah Data Uji | Akurasi | Negatif (P/R/F1) | Netral (P/R/F1) | Positif (P/R/F1) |
|--------------------|-----------------|------------|---------------------------|---------------------------|---------------------------|
| 50% : 50% | 19.046 | 98% | 0.95 / 0.99 / 0.97 | 1.00 / 1.00 / 1.00 | 0.99 / 0.94 / 0.97 |
| 60% : 40% | 15.237 | 98% | 0.96 / 0.99 / 0.98 | 1.00 / 1.00 / 1.00 | 0.99 / 0.96 / 0.98 |
| 70% : 30% | 11.428 | 99% | 0.97 / 0.99 / 0.98 | 1.00 / 1.00 / 1.00 | 0.99 / 0.97 / 0.98 |
| 80% : 20% | 7.619 | 99% | 0.98 / 0.99 / 0.98 | 1.00 / 1.00 / 1.00 | 0.99 / 0.97 / 0.98 |
| 90% : 10% | 3.810 | 99% | 0.98 / 0.99 / 0.99 | 1.00 / 1.00 / 1.00 | 0.99 / 0.98 / 0.99 |

Berdasarkan hasil evaluasi tersebut, dapat dilihat bahwa skenario terbaik adalah 90:10 yang mana memberikan performa yang tinggi, dengan akurasi berkisar antara 99% hingga 99% dan skor F1 yang konsisten di atas 0.99 untuk seluruh kelas sentimen (negatif, netral, dan positif). Namun, rasio 80:20 dipilih sebagai konfigurasi utama dalam pelatihan model karena beberapa pertimbangan berikut:

1. Keseimbangan antara data latih dan data uji: Rasio 80:20 memberikan proporsi yang cukup besar untuk pelatihan model, tanpa mengorbankan jumlah data uji yang diperlukan untuk evaluasi.
2. Kinerja yang optimal dan stabil: Pada rasio ini, model mencapai akurasi 99%, dengan nilai F1-score tinggi untuk seluruh kelas. Ini menunjukkan bahwa model tidak hanya akurat, tetapi juga konsisten dalam mengenali masing-masing kategori sentimen.

3. Efisiensi komputasi: Dibandingkan dengan rasio 90:10, rasio 80:20 memberikan efisiensi yang lebih baik dalam pelatihan dan evaluasi, karena data uji yang digunakan masih cukup besar untuk validasi tanpa memperpanjang waktu pelatihan secara signifikan.

4. Standar umum dalam penelitian klasifikasi: Rasio 80:20 merupakan konfigurasi umum dan banyak digunakan dalam berbagai penelitian klasifikasi karena dianggap representatif untuk mengevaluasi performa model terhadap data yang belum dikenali.

Dengan demikian, meskipun semua rasio menunjukkan performa baik, pemilihan rasio 80:20 dalam pelatihan akhir didasarkan pada keseimbangan antara efektivitas, efisiensi, dan validitas hasil evaluasi model.

4.4 Visualisasi Data

Visualisasi data merupakan teknik penyajian informasi dalam bentuk grafis atau visual, seperti grafik, diagram, peta, atau infografis, yang bertujuan untuk mempermudah interpretasi pola, tren, serta hubungan antarvariabel. Melalui visualisasi, data yang kompleks dapat direpresentasikan secara ringkas dan informatif, sehingga mendukung proses analisis serta pengambilan keputusan berbasis data.

Wordcloud Positif



Gambar 4. 3 Wordcloud Positif

Gambar di atas menampilkan *word cloud* yang dihasilkan dari ulasan pengguna yang diprediksi memiliki sentimen positif menggunakan algoritma Support Vector Machine (SVM). Ukuran kata pada *word cloud* merepresentasikan frekuensi kemunculan kata tersebut dalam korpus ulasan, di mana kata dengan ukuran lebih besar muncul lebih sering. Beberapa kata yang dominan, seperti

mengklasifikasikan sentimen ulasan pengguna ke dalam tiga kategori, yaitu positif, netral, dan negatif. Pada matriks ini, nilai diagonal (2474, 2598, dan 2464) menunjukkan jumlah prediksi yang benar untuk masing-masing kelas, sedangkan nilai di luar diagonal merepresentasikan kesalahan klasifikasi.

Model berhasil mengklasifikasikan 2.474 ulasan positif dengan benar, meskipun terdapat 18 ulasan positif yang salah diprediksi sebagai negatif. Untuk kelas netral, model mencapai akurasi sempurna dengan 2.598 ulasan yang diprediksi tepat tanpa kesalahan klasifikasi yang signifikan. Sementara itu, pada kelas negatif, terdapat 64 ulasan yang keliru diprediksi sebagai positif dan 1 ulasan yang diprediksi sebagai netral.

Hasil ini menunjukkan bahwa model memiliki performa yang tinggi dalam membedakan ketiga kategori sentimen, dengan tingkat kesalahan relatif rendah, terutama pada kelas netral yang memiliki prediksi paling konsisten.

5. KESIMPULAN

Penelitian ini menunjukkan bahwa algoritma Support Vector Machine (SVM) dengan kernel linear merupakan pendekatan yang sangat efektif untuk melakukan klasifikasi sentimen terhadap ulasan pengguna aplikasi SIREKAP. Dengan tingkat akurasi mencapai 99%, model mampu mengenali pola opini dalam data teks berbahasa Indonesia secara konsisten dan akurat.

Hasil analisis juga mengungkap bahwa sebagian besar ulasan pengguna bersentimen negatif, yang mengindikasikan adanya permasalahan umum yang dihadapi oleh pengguna, seperti kendala teknis, kecepatan respon, dan antarmuka aplikasi. Temuan ini memberikan wawasan yang penting bagi pihak Komisi Pemilihan Umum (KPU) maupun pengembang aplikasi, sebagai bahan evaluasi untuk meningkatkan kualitas layanan digital di masa mendatang.

Sebagai tindak lanjut, penelitian ini menyarankan beberapa arah pengembangan ke depan, antara lain:

- a. Menggunakan pendekatan berbasis deep learning, seperti LSTM atau BERT, untuk menangkap konteks bahasa yang lebih kompleks dan meningkatkan akurasi klasifikasi.

- b. Memperluas sumber data dengan menyertakan platform media sosial lain seperti Twitter dan Instagram, agar cakupan analisis lebih representatif terhadap opini publik secara luas.
- c. Mengintegrasikan fitur visualisasi data real-time, untuk membantu pemangku kebijakan memantau dan merespons opini masyarakat dengan lebih cepat dan adaptif.
- d. Dengan demikian, penelitian ini tidak hanya memberikan solusi teknis dalam pengolahan data teks, tetapi juga menawarkan kontribusi nyata dalam upaya peningkatan layanan publik berbasis digital.

UCAPAN TERIMA KASIH

Kami mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang telah memberikan dukungan dan kontribusi dalam kelancaran penelitian ini, mulai dari tahap perencanaan, penulisan, hingga proses penerbitan. Ucapan terima kasih secara khusus kami tujukan kepada orang tua tercinta atas doa, motivasi, dan dukungannya yang tiada henti, serta kepada rekan-rekan dan sahabat yang senantiasa memberikan semangat dan masukan berharga. Kami juga menyampaikan apresiasi yang tulus kepada tim Jurnal Informatika dan Teknik Elektro Terapan yang telah meluangkan waktu untuk membaca, mengevaluasi, serta memberikan kesempatan bagi kami mempublikasikan karya ilmiah ini. Semoga hasil penelitian ini dapat menjadi referensi yang bermanfaat bagi para pembaca, memberikan kontribusi positif dalam pengembangan ilmu pengetahuan, serta menjadi pijakan bagi penelitian-penelitian selanjutnya.

DAFTAR PUSTAKA

- [1] Akhsan Firly Saetrian, Intan Syahrani, Melisa Nurdiana, Muhamad Rizal Fauzan, Novira Putri Rustandi, and Salwa Yurisha, "Analisis Yuridis Tentang Sengketa Pemilu Terhadap Pelaksanaan Sistem Informasi Rekapitulasi (SIREKAP) Yang Terindikasi 'Defect' Pada Pemilu Tahun 2024 Yang Berpotensi Merugikan Bakal Calon Presiden," *Demokr. J. Ris. Ilmu Hukum, Sos. dan Polit.*, vol. 1, no. 3, pp. 224–240, 2024, doi: 10.62383/demokrasi.v1i3.283.
- [2] N. Inzana, A. A. Maulana, and P. M. Sari, "Inovasi Sirekap dalam Meningkatkan Partisipasi Politik," *J. Adm. Pemerintah. Desa*, vol. 5, no. 2, pp. 1–13, 2024, doi: 10.47134/villages.v5i2.106.
- [3] M. Rizky Hanafi and R. Kurniawan, "Sentiment Analysis on Sirekap App Reviews on Google Play Using Naive Bayes Algorithm," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 4, no. October, pp. 1578–1586, 2024.
- [4] F. Rahmasari, N. Rahaningsih, R. D. Dana, and C. L. Rohmat, "Optimasi Analisis Sentimen Aplikasi Glints Menggunakan Algoritma Support Vector Machine (Svm)," *J. Inform. dan Tek. Elektro Terap.*, vol. 13, no. 1, 2025, doi: 10.23960/jitet.v13i1.5681.
- [5] İ. Tarimer, A. Çoban, and A. E. Kocaman, "Sentiment Analysis on IMDB Movie Comments and Twitter Data by Machine Learning and Vector Space Techniques," pp. 1–8, 2019, [Online]. Available: <http://arxiv.org/abs/1903.11983>
- [6] G. R. Ditami, E. F. Ripanti, and H. Sujaini, "Implementasi Support Vector Machine untuk Analisis Sentimen Terhadap Pengaruh Program Promosi Event Belanja pada Marketplace," *J. Edukasi dan Penelit. Inform.*, vol. 8, no. 3, p. 508, 2022, doi: 10.26418/jp.v8i3.56478.
- [7] R. W. Pratiwi, S. F. H. D. Dairoh, D. I. Af'idah, Q. R. A, and A. G. F, "Analisis Sentimen Pada Review Skincare Female Daily Menggunakan Metode Support Vector Machine (SVM)," *J. Informatics, Inf. Syst. Softw. Eng. Appl.*, vol. 4, no. 1, pp. 40–46, 2021, doi: 10.20895/inista.v4i1.387.
- [8] R. Septian, S. Irianto, and A. Andriani, "Matematika Berbasis Model Realistic Mathematics," *J. Educ. FKIP UNMA*, vol. 5, no. 1, pp. 59–67, 2019.
- [9] H. C. Husada and A. S. Paramita, "Analisis Sentimen Pada Maskapai Penerbangan di Platform Twitter Menggunakan Algoritma Support Vector Machine (SVM)," *Teknika*, vol. 10, no. 1, pp. 18–26, 2021, doi: 10.34148/teknika.v10i1.311.
- [10] M. R. Manoppo, I. C. Kolang, D. N. Fiat, R. Michelly, and C. Mawara, "Analisis Sentimen Publik Di Media Sosial Terhadap Kenaikan Ppn 12 % Di Indonesia Menggunakan Indobert Analysis of Public Sentiment on Social Media Towards the 12 % Ppn Increase in Indonesia Using Indobert," vol. 4, no. 2, pp. 152–163, 2025, [Online]. Available: <https://repository.uinjkt.ac.id/dspace/handle/123456789/76873>
- [11] M. Alfin, Z. Abidin, and P. M. N. S. A. Basid, "Peringkasan Multi Dokumen Berbahasa Indonesia Menggunakan Metode Recurrent Neural Network," *Techno.Com*, vol. 23, no. 1, pp. 187–197, 2024, doi: 10.62411/tc.v23i1.9605.
- [12] A. R. Lubis and M. K. M. Nasution, "Twitter Data Analysis and Text Normalization in Collecting Standard Word," *J. Appl. Eng. Technol. Sci.*, vol. 4, no. 2, pp. 855–863, 2023, doi: 10.37385/jaets.v4i2.1991.
- [13] M. S. F. Fayaza and F. F. Farhath, "Towards Stopwords Identification in Tamil Text Clustering," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 12, pp. 524–529, 2021, doi: 10.14569/IJACSA.2021.0121267.
- [14] R. I. Agustin, "Komparasi Algoritma Naïve Bayes Dan Svm Untuk Analisis Sentimen Twitter Korupsi Bansos Beras Masa Pandemi," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 2, pp. 912–918, 2024, doi: 10.23960/jitet.v12i2.4020.
- [15] E. E. Pratiwi, A. R. Aisy, R. Rahmaddeni, and N. Ananta, "Klasifikasi Kesehatan Mental Pada Usia Remaja Menggunakan Metode Svm," *J. Inform. dan Tek. Elektro Terap.*, vol. 13, no. 2, 2025, doi: 10.23960/jitet.v13i2.6232.