

Analisis Komparatif Kinerja Algoritma Support Vector Machine, Random Forest, dan Naive Bayes untuk Klasifikasi Sentimen pada Komentar YouTube

Eustachius Dito Dewantoro¹, Sindhu Rakasiwi^{2*}

¹Fakultas Ilmu Komputer, Teknik Informatika, Universitas Dian Nuswantoro, Semarang, Indonesia

² Fakultas Ilmu Komputer, Teknik Informatika, Universitas Dian Nuswantoro, Semarang, Indonesia

Email: ¹111202214105@mhs.dinus.ac.id, ^{2,*} sindhu.rakasiwi@dsn.dinus.ac.id

Email Penulis Korespondensi: sindhu.rakasiwi@dsn.dinus.ac.id

Submitted: 99/99/9999; Accepted: 99/99/9999; Published: 99/99/9999

Abstrak—Perkembangan media sosial seperti YouTube telah menjadikannya platform utama bagi masyarakat untuk menyuarakan opini terhadap isu sosial-politik, salah satunya adalah peristiwa "demo 25 agustus". Volume data komentar yang masif menyulitkan analisis opini publik secara manual, sehingga diperlukan pendekatan komputasi seperti analisis sentimen. Penelitian ini bertujuan untuk mengklasifikasikan sentimen publik ke dalam dua kategori (positif dan negatif) serta membandingkan kinerja dari tiga algoritma machine learning: Naive Bayes, Random Forest, dan Support Vector Machine (SVM) untuk menemukan model klasifikasi yang paling efektif. Metode penelitian dimulai dari pengumpulan 2917 data komentar, diikuti oleh tahap pra-pemrosesan teks, pelabelan sentimen otomatis berbasis leksikon, dan pembobotan fitur menggunakan TF-IDF. Hasil penelitian menunjukkan bahwa sentimen negatif mendominasi opini publik sebesar 78.8%. Dalam evaluasi model, Support Vector Machine (SVM) menunjukkan performa terbaik dengan mencapai tingkat akurasi sebesar 88,2%, mengungguli Random Forest dengan akurasi 83,1% dan Naive Bayes sebesar 81,2%. Hasil ini mengindikasikan bahwa SVM merupakan model yang paling akurat untuk studi kasus klasifikasi sentimen pada komentar YouTube berbahasa Indonesia.

Kata Kunci: Analisis Sentimen; Komentar YouTube; Naive Bayes; Random Forest; Support Vector Machine (SVM); TF-IDF

Abstract—The development of social media such as YouTube has made it a primary platform for people to voice their opinions on socio-political issues, one of which is the "August 25 demo" event. The massive volume of comment data makes manual analysis of public opinion difficult, thus requiring a computational approach such as sentiment analysis. This research aims to classify public sentiment into two categories (positive and negative) as well as compare the performance of three machine learning algorithms: Naive Bayes, Random Forest, and Support Vector Machine (SVM) to find the most effective classification model. The research method begins with the collection of 2917 comment data, followed by a text preprocessing stage, automatic lexicon-based sentiment labeling, and feature weighting using TF-IDF. The research results show that negative sentiment dominates public opinion at 78.8%. In the model evaluation, Support Vector Machine (SVM) showed the best performance by achieving an accuracy rate of 88.2%, outperforming Random Forest with 83.1% accuracy and Naive Bayes at 81.2%. These results indicate that SVM is the most accurate model for the sentiment classification case study on Indonesian-language YouTube comments.

Keywords: Sentiment Analysis; YouTube Comments; Naive Bayes; Random Forest; Support Vector Machine (SVM); TF-IDF

1. PENDAHULUAN

Era transformasi digital telah secara fundamental mengubah lanskap komunikasi publik, menjadikan media sosial sebagai arena utama wacana sosial dan politik. Platform seperti YouTube telah terbukti menjadi media yang signifikan untuk pemasaran politik dan diskusi publik di Indonesia, yang memungkinkan jutaan pengguna untuk mengekspresikan opini mereka [1]. Salah satu isu yang memicu diskusi publik secara masif adalah peristiwa "demo 25 agustus", yang menarik perhatian luas dan menghasilkan ribuan respons dalam bentuk komentar. Volume data tekstual yang sangat besar ini (*big data*) menghadirkan tantangan signifikan. Karakteristik komentar di media sosial sangat unik, seringkali menggunakan bahasa yang sangat informal, penuh dengan bahasa gaul, singkatan non-standar, dan ekspresi emosional yang kuat, yang memerlukan penanganan khusus dalam analisis teks [2]. Analisis manual terhadap ribuan komentar ini tidak hanya memakan waktu dan tenaga, tetapi juga sangat rentan terhadap bias subjektif. Kompleksitas ini diperparah oleh adanya fenomena disinformasi dan operasi pengaruh terkoordinasi, seperti aktivitas "buzzer" yang telah diidentifikasi dalam analisis komentar Instagram [3]. Aktivitas semacam ini dapat mendistorsi opini publik yang sebenarnya, sehingga membedakan antara sentimen organik dan yang dimanipulasi menjadi sangat sulit. Oleh karena itu, untuk mendapatkan wawasan yang objektif dan skalabel dari data yang bising (*noisy*) ini, diperlukan pendekatan komputasi otomatis yang canggih.

Solusi teknologis untuk tantangan ini terletak pada bidang *Natural Language Processing* (NLP) dan *machine learning*, khususnya melalui teknik yang dikenal sebagai analisis sentimen. Analisis sentimen bertujuan untuk mengklasifikasikan sentimen (misalnya positif atau negatif) dari teks secara otomatis, seringkali memanfaatkan algoritma *machine learning* atau *deep learning* untuk menangani data bervolume besar [4]. Disiplin ilmu komputasi ini berfokus pada identifikasi, ekstraksi, dan kuantifikasi keadaan afektif atau opini subjektif yang diekspresikan dalam data teks. Dalam konteks penelitian ini, tujuannya adalah untuk mengklasifikasikan komentar publik ke dalam sentimen biner: Positif atau Negatif. Fokus biner ini sengaja dipilih untuk mempertajam pemahaman tentang polarisasi publik. Kemampuan untuk memetakan sentimen publik secara akurat sangat krusial. Bagi pembuat kebijakan dan lembaga pemerintah, ini berfungsi sebagai alat untuk memahami opini publik terhadap suatu isu atau peristiwa politik, seperti debat kandidat atau kebijakan yang sedang berjalan [5]. Bagi sosiolog dan ilmuwan politik, ini menyediakan

data empiris untuk memahami dinamika sosial. Bagi media dan jurnalis, ini membantu mengukur dampak pemberitaan. Keberhasilan analisis sentimen ini sangat bergantung pada pemilihan model klasifikasi yang tepat, yang mampu menangani kompleksitas bahasa informal Indonesia.

Berbagai penelitian sebelumnya telah banyak menerapkan algoritma *machine learning* klasik untuk tugas klasifikasi teks dan analisis sentimen di media sosial Indonesia. Di platform Twitter, misalnya, algoritma Multinomial Naïve Bayes telah berhasil diterapkan untuk mendeteksi ujaran kebencian terhadap PSSI dengan performa klasifikasi yang kuat, mencapai F1-Score sebesar 91% [6]. Masih di platform yang sama, sebuah studi komparatif yang sangat relevan untuk analisis sentimen program PPPK secara langsung membandingkan Naïve Bayes dengan Support Vector Machine (SVM). Temuan kunci dari studi tersebut menunjukkan bahwa SVM secara signifikan lebih unggul, dengan perolehan akurasi 93,31% berbanding 89,24% untuk Naïve Bayes [7]. Platform Instagram juga telah menjadi objek penelitian populer, di mana SVM telah digunakan secara spesifik untuk menganalisis sentimen ujaran kebencian pada kolom komentar [8]. Kegunaan SVM pada Instagram juga diperkuat dalam konteks yang berbeda sebagai salah satu klasifikator kunci dalam mendeteksi akun palsu, di mana model tersebut mampu mencapai akurasi hingga 96,77% ketika dioptimalkan [9]. Selain itu, model berbasis pohon (*tree-based*) juga telah dieksplorasi; algoritma Decision Tree telah diimplementasikan untuk mengklasifikasikan ujaran kebencian di berbagai media sosial, yang menunjukkan potensinya sebagai dasar untuk metode *ensemble* yang lebih kompleks [10].

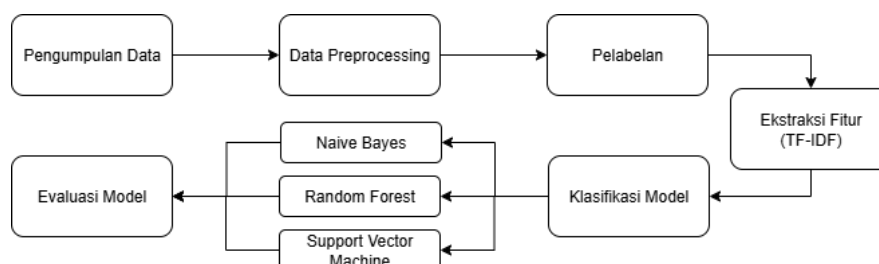
Berdasarkan tinjauan pustaka yang ekstensif tersebut, dapat diidentifikasi beberapa kesenjangan penelitian (*research gap*) yang signifikan. Kesenjangan pertama adalah fokus platform. Mayoritas literatur yang ada terkonsentrasi pada data dari Twitter (X) atau Instagram. Platform YouTube masih sangat kurang terwakili (*under-researched*) dalam konteks analisis sentimen berbahasa Indonesia. Padahal, karakteristik komentar YouTube secara unik berbeda: teksnya bisa jauh lebih panjang daripada *tweet* atau komentar IG, namun tetap mempertahankan tingkat informalitas yang tinggi, dan konteksnya sangat terikat pada konten audio-visual yang ditonton, yang dapat mempengaruhi pilihan kata dan ekspresi sentimen. Kesenjangan kedua adalah metodologi perbandingan. Banyak studi komparatif yang ada cenderung hanya membandingkan dua model (misalnya, Naïve Bayes vs SVM) atau berfokus pada implementasi satu model saja. Sementara itu, penelitian yang mulai mengeksplorasi *deep learning* seperti LSTM seringkali membutuhkan sumber daya komputasi yang besar [11]. Oleh karena itu, sebuah studi komparatif yang mengadu secara langsung tiga pilar algoritma *machine learning* klasik Naive Bayes (sebagai *baseline* probabilistik), Random Forest (sebagai metode *ensemble* yang kuat), dan SVM (sebagai klasifikator *margin-based* berperforma tinggi) pada *dataset* komentar YouTube berbahasa Indonesia yang sama masih jarang dilakukan dan sangat dibutuhkan.

Berdasarkan latar belakang dan *gap analysis* yang telah diuraikan, penelitian ini mengusulkan sebuah kerangka kerja untuk mengklasifikasikan sentimen publik pada isu "demo 25 agustus" dari komentar YouTube. Dengan fokus pada klasifikasi biner (positif dan negatif), penelitian ini bertujuan untuk secara sistematis menerapkan pipeline analisis sentimen dan mengevaluasi kinerja tiga model klasifikasi yang berbeda: Naive Bayes, Random Forest, dan SVM. Hasil dari perbandingan ini akan digunakan untuk menyimpulkan model mana yang paling akurat dan andal untuk studi kasus komentar YouTube, sehingga dapat memberikan kontribusi metodologis untuk penelitian di masa depan.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Metodologi penelitian ini mengadopsi sebuah kerangka kerja (*framework*) penelitian yang sistematis dan telah divalidasi, yang umum diterapkan dalam studi analisis sentimen berbasis *machine learning*. Sebuah tinjauan literatur sistematis terhadap metode analisis sentimen mengonfirmasi bahwa alur kerja yang paling umum digunakan terdiri dari pengumpulan data, pra-pemrosesan, ekstraksi fitur, pemodelan, dan evaluasi [12]. Kerangka kerja ini, yang seringkali membandingkan berbagai algoritma untuk menemukan yang paling optimal, telah terbukti efektif dalam berbagai penelitian untuk mengolah data teks mentah menjadi wawasan yang terstruktur [13].



Gambar 1. Tahapan Penelitian

Secara spesifik, alur kerja yang diimplementasikan dalam penelitian ini diilustrasikan pada Gambar 1. Tahapan diawali dengan Pengumpulan Data mentah dari YouTube API, diikuti oleh tahap Preprocessing yang komprehensif. Data bersih kemudian melalui proses Pelabelan sentimen otomatis. Teks berlabel ditransformasi menggunakan Ekstraksi Fitur TF-IDF. Penggunaan TF-IDF sebagai metode pembobotan kata yang dilanjutkan dengan klasifikasi *machine learning* juga merupakan pendekatan yang telah terbukti efektif dalam studi sejenis [7], [10]. Dataset yang siap pakai ini kemudian dibagi menjadi data latih dan data uji untuk tahap Klasifikasi, di mana tiga model (Naive Bayes, Random Forest, dan SVM) dilatih. Model-model yang telah dilatih kemudian diuji performanya pada tahap Evaluasi. Terakhir, hasil akurasi dari ketiga model tersebut Dibandingkan untuk mengidentifikasi model yang paling unggul. Detail dari setiap tahapan ini akan dijelaskan pada sub-bab berikutnya.

2.2 Pengumpulan Data

Tahap pengumpulan data merupakan langkah awal dalam penelitian ini, di mana data primer berupa komentar publik diekstraksi dari platform YouTube. Proses akuisisi data dilakukan secara otomatis (*crawling*) dengan memanfaatkan YouTube Data API v3. Penggunaan Antarmuka Pemrograman Aplikasi (API) untuk mengumpulkan dataset komentar YouTube merupakan metode yang lazim dan tervalidasi dalam berbagai penelitian analisis sentimen [14]. Dengan menggunakan API tersebut dan kata kunci pencarian spesifik yaitu "demo 25 agustus", berhasil dikumpulkan dataset mentah sebanyak 2.917 komentar. Dataset ini kemudian disimpan dalam format file .csv untuk diproses pada tahap selanjutnya.

2.3 Data Preprocessing

Data komentar mentah yang diperoleh dari YouTube pada dasarnya bersifat tidak terstruktur dan mengandung banyak *noise* (derau). Oleh karena itu, tahap pra-pemrosesan data sangat krusial untuk membersihkan dan mentransformasi data teks ini menjadi format yang bersih dan terstruktur. Penelitian sebelumnya telah membuktikan secara empiris bahwa penerapan tahapan *preprocessing* yang komprehensif memiliki dampak yang signifikan terhadap performa dan akurasi model klasifikasi sentimen [15]. Proses ini meliputi *Cleaning* (menghapus URL/symbol), *Case Folding* (penyeragaman huruf kecil), *Normalisasi* (pembakuan kata), *Stopword Removal* (penghapusan kata henti), dan *Stemming* (pencarian kata dasar).

2.4 Pelabelan Data

Setelah data teks bersih, setiap komentar diberi label sentimen untuk dapat digunakan dalam pelatihan model klasifikasi. Proses pelabelan dilakukan secara otomatis dengan pendekatan berbasis leksikon (*lexicon-based*), yang merupakan metode analisis sentimen yang umum digunakan dengan memanfaatkan fitur leksiko-semantik dari sebuah kamus [16]. Secara spesifik, penelitian ini menggunakan InSet Lexicon, sebuah kamus sentimen berbahasa Indonesia yang telah terbukti efektif ketika diimplementasikan dalam penelitian analisis sentimen baru-baru ini [17]. Proses pelabelan bekerja dengan menghitung skor sentimen total untuk setiap komentar berdasarkan kata-kata di dalamnya. Sesuai fokus penelitian pada klasifikasi biner, komentar dengan total skor > 0 diberi label Positif dan komentar dengan total skor ≤ 0 diberi label Negatif.

2.5 Ekstraksi Fitur (TF-IDF)

Model *machine learning* tidak dapat memproses data teks mentah secara langsung, sehingga data yang telah melalui tahap pra-pemrosesan perlu ditransformasi menjadi representasi vektor numerik. Penelitian ini menggunakan metode Term Frequency-Inverse Document Frequency (TF-IDF) untuk ekstraksi fitur dan pembobotan kata. TF-IDF adalah sebuah teknik statistik yang mengubah teks menjadi representasi numerik dengan menilai frekuensi sebuah kata dalam dokumen dan pentingnya kata tersebut dalam seluruh koleksi teks [5]. Metode ini bekerja dengan memberikan bobot tinggi pada kata-kata yang sering muncul dalam satu komentar tetapi jarang muncul di komentar-komentar lainnya, sehingga menjadikannya fitur pembeda yang kuat. Penggunaan TF-IDF sebagai metode pembobotan telah terbukti memberikan hasil akurasi yang baik ketika dikombinasikan dengan model klasifikasi seperti SVM untuk analisis sentimen berbahasa Indonesia [18].

2.6 Klasifikasi Model

Pada tahap ini, dataset yang telah berbentuk vektor TF-IDF dibagi menjadi dua subset data yang independen: 80% data latih (*training data*) dan 20% data uji (*testing data*). Pembagian ini penting untuk memastikan bahwa evaluasi model dilakukan pada data yang belum pernah dilihat oleh model selama proses pelatihan. Data latih digunakan untuk "mengajari" model mengenali pola sentimen positif dan negatif berdasarkan fitur TF-IDF, sedangkan data uji digunakan untuk mengukur seberapa baik model dapat menggeneralisasi pengetahuannya pada data baru. Penelitian ini secara spesifik membandingkan kinerja dari tiga algoritma klasifikasi yang berbeda, yaitu Naive Bayes, Random Forest, dan Support Vector Machine (SVM), yang masing-masing dijelaskan di bawah ini.

2.6.1 Naive Bayes

Naive Bayes adalah salah satu algoritma klasifikasi probabilistik yang populer dan sering digunakan sebagai model baseline dalam tugas klasifikasi teks karena kesederhanaan dan efisiensinya [19]. Algoritma ini didasarkan pada

Teorema Bayes, yang secara matematis mendeskripsikan probabilitas suatu kejadian berdasarkan pengetahuan sebelumnya tentang kondisi yang mungkin terkait dengan kejadian tersebut. Algoritma ini disebut "naif" karena mengasumsikan independensi antar fitur (kata), yang memungkinkan penyederhanaan perhitungan likelihood menjadi perkalian probabilitas setiap kata muncul di kelas tersebut [20]. Meskipun asumsi ini seringkali tidak sepenuhnya akurat untuk data teks, Naive Bayes dikenal efisien secara komputasi dan telah terbukti efektif dalam analisis sentimen opini publik berbahasa Indonesia [21]. Klasifikasi dilakukan dengan memilih kelas C yang menghasilkan nilai probabilitas posterior $P(C|x)$ tertinggi. Variasi Multinomial Naive Bayes sering digunakan untuk klasifikasi teks karena cocok untuk data dengan frekuensi kata [20].

$$P(C|x) = \frac{P(x|C)P(C)}{P(x)} \quad (1)$$

Dalam rumus tersebut, $P(C|x)$ merupakan probabilitas posterior, yaitu probabilitas kelas C (misalnya, Positif atau Negatif) terjadi setelah melihat data komentar x . $P(x|C)$ adalah probabilitas likelihood, yaitu probabilitas data komentar x muncul jika diketahui kelasnya adalah C . $P(C)$ adalah probabilitas prior, yaitu probabilitas awal bahwa suatu komentar termasuk dalam kelas C . Terakhir, $P(x)$ adalah probabilitas evidence, yaitu probabilitas data komentar x muncul secara keseluruhan; nilai ini sering diabaikan dalam klasifikasi karena konstan untuk semua kelas [19]. Dengan asumsi independensi Naive Bayes, $P(x|C)$ disederhanakan menjadi $\prod_{i=1}^n P(x_i|C)$ [19], [20].

2.6.2 Random Forest

Random Forest adalah algoritma *ensemble learning* yang efektif untuk klasifikasi, yang bekerja dengan membangun banyak Decision Tree (pohon keputusan) [19], [20]. Prediksi dari semua pohon ini kemudian digabungkan untuk menghasilkan keputusan akhir yang lebih akurat dan stabil, mengandalkan "kebijaksanaan kolektif" yang lebih baik daripada satu pohon tunggal [19], [20]. Metode ini telah terbukti efektif untuk analisis sentimen berbahasa Indonesia (Ardiansyah et al., 2023). Untuk mencegah *overfitting* dan menjaga keragaman antar pohon, Random Forest menggunakan dua teknik utama: Bagging (melatih pohon pada subset data acak) dan Random Feature Selection (memilih fitur acak pada tiap *node* pemisah) [19], [20]. Setelah sejumlah N pohon (T_1, T_2, \dots, T_N) dibangun, proses klasifikasi untuk data baru x dilakukan dengan *majority voting*. Prediksi akhir dari Random Forest, $C_{RF}(x)$, adalah kelas yang paling sering diprediksi oleh mayoritas pohon [20].

$$C_{RF}(x) = \text{mode}\{C_1(x), C_2(x), \dots, C_N(x)\} \quad (2)$$

Dalam rumus tersebut, $C_{RF}(x)$ adalah prediksi kelas akhir dari Random Forest untuk input data x , sedangkan $C_i(x)$ adalah prediksi kelas dari pohon ke- i (dari total N pohon). Fungsi *mode* $\{\dots\}$ memilih kelas yang paling banyak muncul (mayoritas) dari semua prediksi pohon individual. Penggunaan Decision Tree sebagai dasar dari Random Forest relevan dalam konteks klasifikasi teks ujaran kebencian [22].

2.6.3 Support Vector Machine (SVM)

Support Vector Machine (SVM) adalah algoritma *supervised learning* yang efektif untuk data berdimensi tinggi seperti teks TF-IDF [19], [20]. SVM bekerja dengan menemukan *hyperplane* (batas keputusan) optimal yang memaksimalkan *margin*—yaitu, jarak ke titik data terdekat (*support vectors*) dari setiap kelas [19]. Maksimasi *margin* ini bertujuan meningkatkan kemampuan generalisasi model pada data baru [20]. Untuk menangani data yang tidak dapat dipisahkan secara linier, SVM dapat menggunakan *kernel trick* (seperti kernel RBF) [19], [20]. SVM telah terbukti memberikan akurasi tinggi untuk analisis sentimen pada data media sosial Indonesia, dan penelitian menunjukkan performanya seringkali kompetitif atau mengungguli Naive [23]. Proses pelatihan SVM melibatkan penyelesaian masalah optimasi kuadratik untuk menemukan parameter *hyperplane* (W dan b) yang memaksimalkan *margin* [19].

$$\min_{w,b} \frac{1}{2} \|W\|^2 \quad (3)$$

Dengan Kendala (subject to): $y_i(W \cdot X_i + b) \geq 1$, untuk semua $i = 1, \dots, N$

Tujuannya adalah meminimalkan $\frac{1}{2} \|W\|^2$ (yang ekuivalen dengan memaksimalkan *margin* $2/\|W\|$). Kendala $y_i(W \cdot X_i + b) \geq 1$ memastikan setiap titik data X_i dengan label y_i (misal, +1 atau -1) berada di sisi *margin* yang benar [20]. Titik-titik data di mana kendala ini terpenuhi secara tepat (nilainya = 1) adalah *support vectors* yang menentukan posisi akhir *hyperplane* [19].

2.7 Evaluasi Model

Kinerja dari model klasifikasi (Naive Bayes, Random Forest, dan SVM) dievaluasi untuk mengukur performa dan kemampuan generalisasinya. Proses ini menggunakan *hold-out validation*, di mana data uji (*test set*) sebesar 20% dari total dataset digunakan untuk pengujian. Pengukuran kinerja didasarkan pada empat metrik standar yang diturunkan dari *confusion matrix*, yaitu Akurasi, Presisi, Recall, dan F1-Score [24]. Hasil dari metrik-metrik ini akan dibandingkan secara komprehensif untuk mengidentifikasi model dengan kinerja paling optimal, dengan F1-Score menjadi metrik penting untuk menangani potensi ketidakseimbangan kelas data [25].

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

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

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

$$F1 - Score = 2 \times \frac{Presisi \times Recall}{Presisi + Recall} \quad (7)$$

Penjelasan untuk komponen rumus di atas adalah sebagai berikut: TP (*True Positive*) adalah data positif yang diprediksi benar, TN (*True Negative*) adalah data negatif yang diprediksi benar, FP (*False Positive*) adalah data negatif yang salah diprediksi positif, dan FN (*False Negative*) adalah data positif yang salah diprediksi negatif [24]. Akurasi mengukur proporsi total prediksi yang benar. Presisi mengukur ketepatan prediksi positif, sementara Recall mengukur kemampuan model mengidentifikasi kasus positif aktual. F1-Score adalah rata-rata harmonik dari Presisi dan Recall, yang ideal untuk dataset yang tidak seimbang [25].

3. HASIL DAN PEMBAHASAN

3.1 Pengumpulan Data

Tahap penelitian diawali dengan pengumpulan data primer berupa komentar publik dari platform YouTube. Sesuai dengan metode yang dijelaskan pada sub-bab 2.2, proses *crawling* data dilakukan menggunakan YouTube Data API v3 dengan kata kunci pencarian "demo 25 agustus".

Tabel 1. Hasil Pengumpulan Data

Username	Tweet
@SechudinUdin-s1c	Sudah saatnya rakyat melawan sebelum indonesia hancur dn bubar karna ulah pejabat2 maling rakyat dn dpr yg justru mengkhianati rakyat...bersatulah rakyat indonesia mari kita bersatu lawan ketidakadilan ini hancurkan mereka bila perlu kita angkat senjata habisi mereka org2 yg telah menzalimi rakyat
@Sunda-c4s2b	Seandainya kota jakarta ada di Papua, DPR siap kena busur panah dari anak Papua.

Hasil dari proses pengumpulan data ini adalah sebuah dataset mentah (*raw dataset*) yang berhasil menghimpun total 2.917 komentar. Dataset mentah ini menjadi input utama untuk tahap selanjutnya.

3.2 Data Preprocessing

Data mentah yang telah dikumpulkan (2.917 komentar) masih bersifat tidak terstruktur (*unstructured*) dan mengandung banyak *noise* (derau). Oleh karena itu, tahap *preprocessing* sangat penting untuk membersihkan dan mentransformasi data ini menjadi format yang terstruktur dan siap dianalisis. Proses ini mencakup beberapa tahapan sistematis yang berurutan. Pertama, tahap *Cleaning* dilakukan untuk menghapus elemen-elemen yang tidak relevan seperti URL, tag HTML, emoji, angka, dan simbol atau tanda baca. Selanjutnya, *Case Folding* diterapkan untuk menyeragamkan semua teks menjadi huruf kecil (*lowercase*). Setelah itu, proses *Normalisasi* mengubah kata-kata tidak baku atau bersifat informal (bahasa gaul, singkatan) menjadi bentuk bakunya menggunakan kamus normalisasi. Teks yang sudah bersih kemudian melalui *Tokenization*, di mana kalimat dipecah menjadi unit-unit kata individual (token). Kata-kata umum dalam bahasa Indonesia yang tidak memiliki makna sentimen (seperti "di", "dan", "yang") dihapus dari daftar token melalui *Stopword Removal*. Terakhir, proses *Stemming* mengubah setiap kata yang tersisa ke dalam bentuk kata dasarnya (misalnya, "melawan" menjadi "lawan") menggunakan *library* Sastrawi.

Tabel 2. Hasil Preprocessing

Tahapan	Tweet
<i>Tweet</i>	Seandainya kota jakarta ada di Papua, DPR siap kena busur panah dari anak Papua.
<i>Cleaning</i>	Seandainya kota jakarta ada di Papua DPR siap kena busur panah dari anak Papua
<i>Case Folding</i>	seandainya kota jakarta ada di papua dpr siap kena busur panah dari anak papua
<i>Normalisasi</i>	seandainya kota jakarta ada di papua dpr siap kena busur panah dari anak papua
<i>Tokenization</i>	['seandainya', 'kota', 'jakarta', 'ada', 'di', 'papua', 'dpr', 'siap', 'kena', 'busur', 'panah', 'dari', 'anak', 'papua']
<i>Stopword Removal</i>	['seandainya', 'kota', 'jakarta', 'papua', 'dpr', 'kena', 'busur', 'panah', 'anak', 'papua']
<i>Stemming</i>	anda kota jakarta papua dpr kena busur panah anak papua

Tabel 2 di atas ini menyajikan contoh transformasi sebuah komentar mentah melalui setiap tahapan *preprocessing* yang telah dijelaskan. Pada contoh tersebut, teks awal (*Tweet*) "*Seandainya kota jakarta ada di Papua, DPR siap kena busur panah dari anak Papua.*" pertama-tama melalui tahap *Cleaning*, yang menghapus elemen *noise* seperti tanda baca, dalam hal ini adalah tanda koma (.). Selanjutnya, tahap *Case Folding* menyeragamkan seluruh teks menjadi huruf kecil, mengubah "*Seandainya*", "*Papua*", dan "*DPR*" menjadi "*seandainya*", "*papua*", dan "*dpr*". Tahap *Normalisasi* kemudian diterapkan untuk mengoreksi kata-kata tidak baku, meskipun pada contoh spesifik ini tidak ada kata yang berubah, menunjukkan bahwa tidak ada kata tidak baku yang terdeteksi dalam kalimat tersebut. Setelah itu, proses *Tokenization* memecah kalimat yang telah bersih menjadi daftar kata-kata individual (*token*). Dari daftar *token* ini, tahap *Stopword Removal* bekerja dengan menghapus kata-kata umum yang tidak memiliki makna sentimen, seperti kata '*ada*', '*di*', dan '*dari*'. Terakhir, proses *Stemming* mengubah kata-kata yang tersisa ke bentuk kata dasarnya, di mana kata '*seandainya*' diubah menjadi '*anda*'. Hasil dari semua tahapan ini adalah teks akhir yang bersih dan terstruktur: "*anda kota jakarta papua dpr kena busur panah anak papua*". Setelah proses ini, data duplikat dan baris yang mengandung nilai kosong (*NaN*) yang mungkin timbul selama proses pembersihan dihapus. Hasil akhir dari tahap ini adalah dataset bersih yang terdiri dari 2.839 komentar, yang siap untuk dilanjutkan ke tahap pelabelan sentimen.

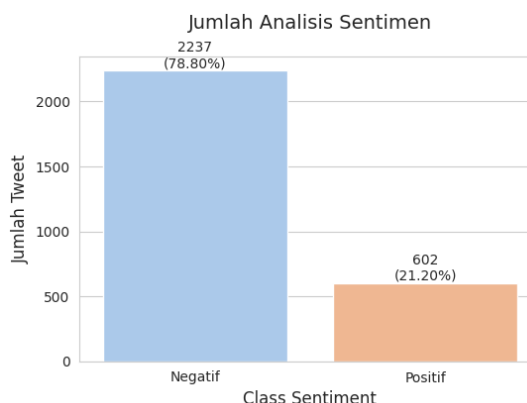
3.3 Pelabelan Data

Setelah data bersih dan terstruktur, 2.839 komentar yang tersisa dari tahap *preprocessing* siap untuk dilanjutkan ke tahap pelabelan sentimen. Proses pelabelan ini bertujuan untuk mengklasifikasikan setiap komentar ke dalam kategori sentimen yang telah ditentukan. Tentu, mari kita lanjutkan ke sub-bab 3.3. Penelitian ini menggunakan pendekatan pelabelan otomatis berbasis leksikon (*lexicon-based*). Metode ini memanfaatkan kamus sentimen berbahasa Indonesia yang telah tervalidasi, yaitu *InSet Lexicon*. Proses pelabelan bekerja dengan menghitung skor sentimen total untuk setiap komentar. Setiap kata dalam komentar dicocokkan dengan kamus; kata-kata positif akan menambah skor, sementara kata-kata negatif akan mengurangnya. Sesuai dengan fokus penelitian pada klasifikasi biner, komentar dengan total skor > 0 diberi label "*Positif*", sedangkan komentar dengan total skor ≤ 0 (termasuk 0) diberi label "*Negatif*".

Tabel 3. Hasil Proses Pelabelan

<i>Stemming</i>	Score	Sentiment
rakyat lawan indonesia hancur bubar ulah jabat maling rakyat dpr menghianati rakyatbersatulah rakyat indonesia mari satu lawan tidak adil hancur angkut senjata habis orang menzhalimi rakyat	-3	Negatif
terima kasih juang sungguh wakil rakyat indonesia	1	Positif
pulisi sikat dukung masatakat antek dpr	0	Negatif

Hasil dari proses pelabelan ini menunjukkan distribusi sentimen yang sangat tidak seimbang (*imbalanced*) pada keseluruhan dataset. Gambar 2 berikut ini menyajikan visualisasi dari distribusi kedua kelas sentimen tersebut.



Gambar 2. Visualisasi Hasil Pelabelan

Dari total 2.839 komentar yang dianalisis, ditemukan bahwa 2.237 komentar (78.80%) diklasifikasikan sebagai *Negatif*, sementara hanya 602 komentar (21.20%) yang diklasifikasikan sebagai *Positif*. Distribusi data yang

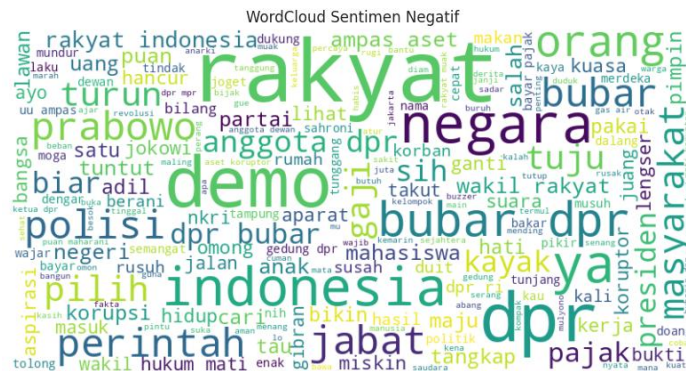


timbang ini mengindikasikan bahwa opini publik pada video YouTube terkait topik "demo 25 agustus" didominasi oleh sentimen negatif. Dataset berlabel inilah yang akan digunakan untuk tahap ekstraksi fitur dan klasifikasi model.

3.4 Visualisasi Word Cloud

Setelah data berhasil dilabeli, *word cloud* (awan kata) dibuat untuk mendapatkan wawasan kualitatif mengenai kata-kata yang paling dominan pada masing-masing kelas sentimen. *Word cloud* ini dihasilkan dari kolom *steming_data* yang telah bersih, di mana ukuran kata merepresentasikan frekuensi kemunculannya.

Gambar 3 menampilkan *word cloud* untuk komentar yang telah diklasifikasikan sebagai Sentimen Negatif.



Gambar 3. WordCloud Sentimen Negatif

Pada Gambar 3, terlihat jelas bahwa kata-kata yang paling dominan dalam sentimen negatif adalah "rakyat", "dpr", "negara", "demo", "indonesia", "bubar", dan "jabat" (jabatan) . Hal ini menunjukkan bahwa fokus utama dari sentimen negatif adalah ketidakpuasan yang ditujukan kepada DPR dan pemerintah (jabatan/perintah), dengan "bubar" menjadi salah satu kata kunci yang sering muncul sebagai bentuk protes. Selanjutnya,

Gambar 4 menampilkan *word cloud* untuk komentar yang diklasifikasikan sebagai Sentimen Positif.



Gambar 4. WordCloud Sentimen Positif

Pada Gambar 4, kata-kata seperti "rakyat", "demo", "dpr", dan "indonesia" masih dominan, menunjukkan bahwa topik pembicaraannya sama. Namun, kata-kata ini didampingi oleh istilah-istilah yang bersifat suportif dan penuh harapan, seperti "semangat", "dukung", "adil", "juang" (berjuang), dan "moga" (semoga). Ini mengindikasikan bahwa sentimen positif lebih bersifat memberi dukungan moral dan harapan agar gerakan (demo) tersebut dapat berhasil membawa perubahan atau keadilan.

3.5 Ekstraksi Fitur (TF-IDF)

Sesuai metodologi pada sub-bab 2.5, data teks diubah menjadi representasi vektor numerik menggunakan Term Frequency-Inverse Document Frequency (TF-IDF). Metode ini menghitung bobot kata dengan mengalikan frekuensinya dalam satu komentar (TF) dengan kelangkaannya di seluruh dataset (IDF), yang bertujuan untuk

mengidentifikasi fitur paling penting. Kata-kata yang sering muncul di komentar tertentu namun relatif langka di komentar lainnya akan mendapat skor tinggi. Seperti terlihat pada Tabel 4, kata "dpr" (0.054244) dan "rakyat" (0.046141) mendapat skor tertinggi karena merupakan kata kunci yang sering menjadi fokus utama (TF tinggi) namun tidak muncul di *setiap* komentar (menjaga IDF tetap tinggi). Demikian pula, "bubar" (0.037240) memiliki skor tinggi karena merupakan kata kunci sentimen yang sangat spesifik dan diskriminatif. Matriks vektor TF-IDF yang dihasilkan dari bobot kata-kata inilah yang kemudian digunakan sebagai *input* untuk melatih model-model klasifikasi.

Tabel 4. Skor TF-IDF

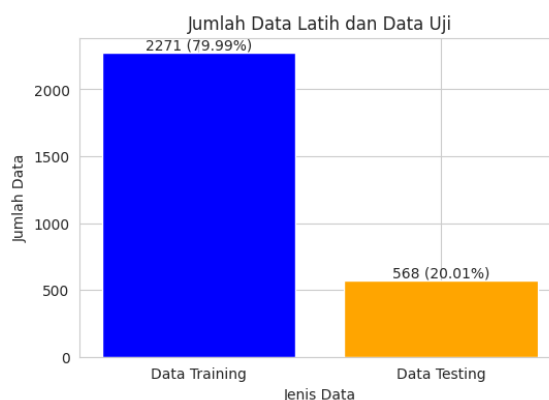
Kata	Average TF-IDF Score
dpr	0.054244
rakyat	0.046141
bubar	0.037240

3.6 Hasil Pengujian Model

Tahap ini berfokus pada implementasi dan evaluasi model *machine learning* untuk mengklasifikasikan sentimen. Proses ini dimulai dengan pembagian data yang telah dibobot, diikuti dengan pelatihan dan pengujian model.

3.6.1 Pembagian Data

Dataset yang telah berlabel (2.839 data) dan telah ditransformasi menjadi vektor numerik menggunakan TF-IDF kemudian dibagi menjadi dua subset yang independen: 80% data latih (*training data*) dan 20% data uji (*testing data*). Pembagian ini dilakukan secara *stratified* (*stratify=y*) untuk memastikan proporsi kelas sentimen (Positif dan Negatif) yang tidak seimbang tetap terjaga baik di data latih maupun data uji.



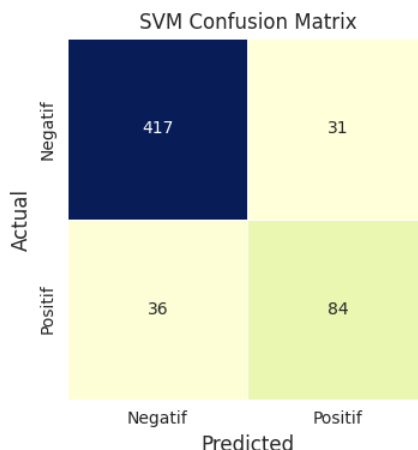
Gambar 5. Distribusi Data Latih dan Data Uji

Proses ini menghasilkan 2.271 data latih (79.99%) dan 568 data uji (20.01%). Data latih digunakan untuk melatih model, sementara data uji digunakan untuk mengevaluasi kinerjanya pada data yang belum pernah dilihat sebelumnya.

3.6.2 Kinerja Model Klasifikasi

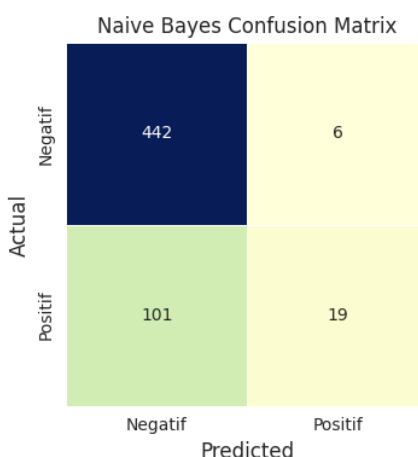
Ketiga model yang telah ditentukan Naive Bayes, Random Forest, dan Support Vector Machine (SVM) dilatih menggunakan 2.271 data latih dan diuji kinerjanya menggunakan 568 data uji. Kinerja setiap model dievaluasi secara mendalam menggunakan *confusion matrix*, yang memetakan jumlah prediksi yang benar dan salah untuk setiap kelas (Positif dan Negatif). Analisis *confusion matrix* ini sangat krusial, karena metrik akurasi saja tidak cukup untuk menilai kinerja model pada dataset yang sangat tidak seimbang (78.8% Negatif).

Gambar 6 menyajikan *confusion matrix* untuk model Support Vector Machine (SVM). Dalam pengujian terhadap 568 data, model SVM menunjukkan kinerja yang paling seimbang. Ia berhasil memprediksi 417 data Negatif dengan benar (True Negative) dan 84 data Positif dengan benar (True Positive). Yang terpenting, jumlah kesalahannya relatif rendah dan tersebar di kedua tipe. Model ini membuat 36 kesalahan prediksi data Negatif sebagai Positif (False Positive) dan 31 kesalahan prediksi data Positif sebagai Negatif (False Negative). Kemampuan SVM untuk mengidentifikasi 84 dari total 115 (31+84) data Positif menunjukkan bahwa model ini tidak hanya "menebak" kelas mayoritas, tetapi benar-benar berhasil mempelajari pola pembeda untuk kelas minoritas.



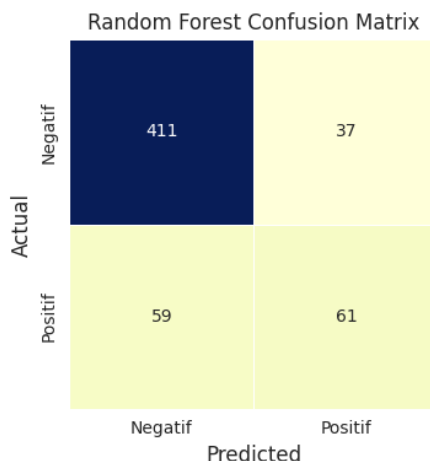
Gambar 6. Confusion Matrix Model SVM

Gambar 7 menyajikan *confusion matrix* untuk model Naive Bayes, yang menyoroti kelemahan signifikan dalam menangani data tidak seimbang. Model ini pada pandangan pertama terlihat sangat baik dalam memprediksi kelas Negatif, dengan 442 True Negative dan hanya 6 False Positive. Namun, performa ini dicapai dengan mengorbankan kelas Positif secara total. Model ini menghasilkan 101 kesalahan False Negative, yang berarti 101 data Positif salah diklasifikasikan sebagai Negatif. Model ini hanya mampu mengidentifikasi 19 data Positif dengan benar (True Positive). Ini adalah indikasi bias yang ekstrem, di mana model pada dasarnya "belajar" bahwa cara teraman untuk mendapatkan akurasi tinggi adalah dengan menebak hampir semua data sebagai "Negatif".



Gambar 7. Confusion Matrix Model Naive Bayes

Gambar 8 menyajikan *confusion matrix* untuk model Random Forest, yang kinerjanya berada di antara Naive Bayes dan SVM. Model ini berhasil memprediksi 411 data Negatif dengan benar (True Negative) dan 61 data Positif dengan benar (True Positive). Meskipun kemampuannya mengidentifikasi data Positif (61 TP) jauh lebih baik daripada Naive Bayes (19 TP), ia masih kalah signifikan dibandingkan SVM (84 TP). Selain itu, model ini mencatatkan jumlah kesalahan False Negative yang cukup tinggi, yaitu 59 data Positif yang salah diprediksi sebagai Negatif, dan 37 data Negatif yang salah diprediksi sebagai Positif (False Positive). Ini menunjukkan bahwa Random Forest mampu mempelajari pola dari kedua kelas, tetapi tidak seefektif SVM dalam menentukan batas keputusan yang optimal.

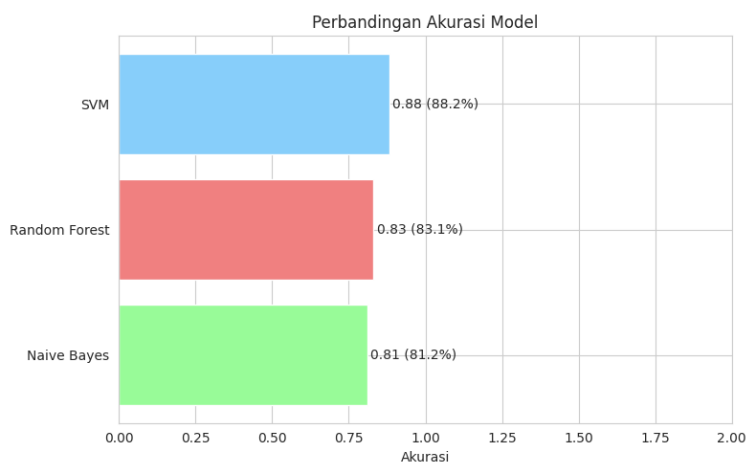


Gambar 8. Confusion Matrix Model Random Forest

Metrik evaluasi utama yang digunakan untuk membandingkan ketiga model adalah akurasi, yang dihitung dari nilai-nilai *confusion matrix* tersebut ($Akurasi = \frac{TP + TN}{TP + TN + FP + FN}$). Hasil perbandingan akurasi disajikan pada Tabel 5 dan divisualisasikan pada Gambar 9.

Tabel 5. Hasil Perbandingan Akurasi Model

Model Klasifikasi	Akurasi
Support Vector Machine (SVM)	88.2%
Random Forest	83.1%
Naive Bayes	81.2%



Gambar 9. Visualisasi Perbandingan Akurasi Model

Berdasarkan hasil pengujian pada Tabel 5 dan Gambar 9, Support Vector Machine (SVM) menunjukkan performa terbaik dengan mencapai tingkat akurasi sebesar 88.2%. Model ini mengungguli Random Forest yang memperoleh akurasi 83.1% dan Naive Bayes dengan akurasi 81.2%. Hasil ini mengindikasikan bahwa SVM merupakan model yang paling efektif untuk klasifikasi sentimen pada studi kasus ini.

4. KESIMPULAN

Penelitian ini bertujuan untuk mengklasifikasikan sentimen publik pada komentar YouTube terkait isu "demo 25 agustus" ke dalam kategori positif dan negatif, serta membandingkan kinerja tiga algoritma *machine learning*: Naive Bayes, Random Forest, dan Support Vector Machine (SVM). Berdasarkan hasil analisis terhadap 2.839 komentar yang telah melalui tahap *preprocessing*, ditemukan bahwa opini publik didominasi secara signifikan oleh sentimen negatif, yang mencakup 78.80% dari total dataset, sementara sentimen positif hanya sebesar 21.20%. Temuan ini

mengindikasikan bahwa persepsi publik terhadap topik tersebut di platform YouTube cenderung sangat negatif. Dalam evaluasi model klasifikasi, Support Vector Machine (SVM) menunjukkan kinerja terbaik dan paling unggul dengan mencapai tingkat akurasi sebesar 88.2%. Performa ini mengungguli Random Forest yang memperoleh akurasi 83.1% dan Naive Bayes dengan akurasi 81.2%. Hasil ini mengonfirmasi bahwa SVM merupakan model yang paling akurat dan andal untuk studi kasus klasifikasi sentimen pada data teks YouTube berbahasa Indonesia, terutama dalam menangani dataset yang bersifat tidak seimbang (*imbalanced*).

Meskipun demikian, penelitian ini memiliki beberapa keterbatasan yang dapat menjadi bahan perbaikan untuk penelitian selanjutnya. Pertama, proses pelabelan sentimen dilakukan secara otomatis menggunakan pendekatan berbasis leksikon (*lexicon-based*). Metode ini mungkin tidak sepenuhnya mampu menangkap nuansa, sarkasme, atau konteks bahasa informal yang kompleks yang sering ditemukan di komentar media sosial. Kedua, pengumpulan data hanya terbatas pada platform YouTube dan berfokus pada satu kata kunci pencarian spesifik, sehingga temuan ini mungkin tidak merepresentasikan keseluruhan opini publik yang ada di platform lain. Oleh karena itu, penelitian selanjutnya disarankan untuk menggunakan metode pelabelan yang lebih canggih, seperti pelabelan manual oleh ahli (*expert labeling*) untuk menghasilkan *ground truth* yang lebih akurat. Selain itu, penerapan model *deep learning* seperti LSTM atau BERT serta perluasan pengumpulan data ke platform media sosial lain seperti X (Twitter) atau Instagram dapat dilakukan untuk mendapatkan pemahaman sentimen publik yang lebih komprehensif dan mendalam.

REFERENCES

- [1] M. Dube, "When the Challenges of Widowhood Extend to Childcare: Essential Considerations for Social Work Practice," *Soc Sci*, vol. 11, no. 5, hlm. 225, Mei 2022, doi: 10.3390/socsci11050225.
- [2] R. Vindua dan A. U. Zailani, "Analisis Sentimen Pemilu Indonesia Tahun 2024 Dari Media Sosial Twitter Menggunakan Python," *JURIKOM (Jurnal Riset Komputer)*, vol. 10, no. 2, hlm. 479, Apr 2023, doi: 10.30865/jurikom.v10i2.5945.
- [3] * Tantiara, F. Sinaga, I. Gavrilu, B. Sembiring, dan R. Rangkuti, "A Comparative Analysis of Buzzer and Non-Buzzer Comment on Prabowo's Instagram Posts," *Print) Journal of English Language and Education*, vol. 10, hlm. 2502–4132, 2025, doi: 10.31004/jele.v10i4.913.
- [4] M. Musfiroh, A. Tholib, dan Z. Arifin, "Analisis Sentimen Terhadap Ulasan Aplikasi Shopee di Google Play Store Menggunakan Metode TF-IDF dan Long Short-Term Memory)," *Journal of Electrical Engineering and Computer (JEECOM)*, vol. 6, no. 2, hlm. 371–381, Okt 2024, doi: 10.33650/jeeecom.v6i2.8713.
- [5] R. R. Harahap dan Mhd. Furqan, "Sentiment Analysis towards the 2024 Vice Presidential Candidate Debate Using the Support Vector Machine Algorithm," *sinkron*, vol. 8, no. 3, hlm. 1783–1794, Jul 2024, doi: 10.33395/sinkron.v8i3.13903.
- [6] R. Wulan dan I. Hertanto, "IMPLEMENTASI ALGORITMA MULTINOMIAL NAÏVE BAYES UNTUK MENDETEKSI TWEET UJARAN KEBENCIAN BAHASA INDONESIA TERHADAP PSSI," *SKANIKA: Sistem Komputer dan Teknik Informatika*, vol. 8, no. 1, hlm. 193–203, Jan 2025, doi: 10.36080/skanika.v8i1.3355.
- [7] L. H. Sarumpaet dan R. R. Suryono, "Analisis Sentimen Publik Program PPPK di Media Sosial X menggunakan Naïve Bayes dan SVM," *Edumatic: Jurnal Pendidikan Informatika*, vol. 9, no. 2, hlm. 362–371, Agu 2025, doi: 10.29408/edumatic.v9i2.30065.
- [8] E. A. Salsabila, M. Pratama, P. Wahyuni, V. Purwaningrum, dan M. Faruq Aziz, "Analisis Sentimen Ujaran Kebencian Pada Kolom Komentar Di Instagram."
- [9] P. Azami dan K. Passi, "Detecting Fake Accounts on Instagram Using Machine Learning and Hybrid Optimization Algorithms," *Algorithms*, vol. 17, no. 10, Okt 2024, doi: 10.3390/a17100425.
- [10] M. Ghianza dan A. Ghifari, "HATE SPEECH CLASSIFICATION IN INDONESIAN SOCIAL MEDIA COMMENTS USING THE DECISION TREE ALGORITHM," doi: 10.13140/RG.2.2.35551.57762.
- [11] Pawit Widiyantoro, Paradise Paradise, dan Yogo Dwi Prasetyo, "Deteksi Cyberbullying pada Pemain Sepak Bola di Platform Media Sosial 'X' Menggunakan Metode Long Short-Term Memory (LSTM)," *Repeater: Publikasi Teknik Informatika dan Jaringan*, vol. 3, no. 1, hlm. 201–217, Jan 2025, doi: 10.62951/repeater.v3i1.382.
- [12] F. Fitroh dan F. Hudaya, "Systematic Literature Review: Analisis Sentimen Berbasis Deep Learning," *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 9, no. 2, hlm. 132–140, Agu 2023, doi: 10.25077/TEKNOSI.v9i2.2023.132-140.
- [13] J. W. Iskandar dan Y. Nataliani, "Perbandingan Naïve Bayes, SVM, dan k-NN untuk Analisis Sentimen Gadget Berbasis Aspek," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 6, hlm. 1120–1126, Des 2021, doi: 10.29207/resti.v5i6.3588.
- [14] Chely Aulia Misrun, E. Haerani, M. Fikry, dan E. Budianita, "Analisis sentimen komentar youtube terhadap Anies Baswedan sebagai bakal calon presiden 2024 menggunakan metode naive bayes classifier," *Jurnal CoSciTech (Computer Science and Information Technology)*, vol. 4, no. 1, hlm. 207–215, Apr 2023, doi: 10.37859/coscitech.v4i1.4790.
- [15] R. E. Saudy, A. E. D. M. El-Ghazaly, E. S. Nasr, dan M. H. Gheith, "A Novel Hybrid Sentiment Analysis Classification Approach for Mobile Applications Arabic Slang Reviews," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 8, hlm. 435–443, 2022, doi: 10.14569/IJACSA.2022.0130849.
- [16] M. Mohd, S. Javeed, Nowsheena, M. A. Wani, dan H. A. Khanday, "Sentiment analysis using lexico-semantic features," *J Inf Sci*, vol. 50, no. 6, hlm. 1449–1470, Des 2024, doi: 10.1177/01655515221124016.
- [17] R. H. Muhammadi, T. G. Laksana, dan A. B. Arifa, "Combination of Support Vector Machine and Lexicon-Based Algorithm in Twitter Sentiment Analysis," *Khazanah Informatika : Jurnal Ilmu Komputer dan Informatika*, vol. 8, no. 1, hlm. 59–71, Mar 2022, doi: 10.23917/khif.v8i1.15213.



- [18] V. W. D. Thomas dan F. Rumaisa, “Analisis Sentimen Ulasan Hotel Bahasa Indonesia Menggunakan Support Vector Machine dan TF-IDF,” *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 6, no. 3, hlm. 1767, Jul 2022, doi: 10.30865/mib.v6i3.4218.
- [19] E. Alpaydin, *Introduction to machine learning*, 4th ed. MIT press, 2020.
- [20] J. Han, M. Kamber, dan J. Pei, *Data mining: concepts and techniques*, 4th ed. Morgan Kaufmann, 2022.
- [21] F. N. Rahman dan S. Lestari, “Analisis Sentimen Masyarakat Terhadap Pemerintah di Era Kabinet Joko Widodo Berdasarkan Sosial Media X Menggunakan Naïve bayes dan K-Nearest Neighbor (KNN),” *INTECOMS: Journal of Information Technology and Computer Science*, vol. 7, no. 5, hlm. 1537–1544, Sep 2024, doi: 10.31539/intecom.v7i5.11823.
- [22] Ismail. B. Mustapha, S. Hasan, H. S. Y. Nabbus, M. M. A. Montaser, S. O. Olatunji, dan S. M. Shamsuddin, “Investigating Group Distributionally Robust Optimization for Deep Imbalanced Learning: A Case Study of Binary Tabular Data Classification,” *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 2, hlm. 726–731, 2023, doi: 10.14569/IJACSA.2023.0140286.
- [23] K. S. Kodoati dan K. D. Hartomo, “Evaluasi Keberhasilan F-Learn Menggunakan Human Organization Technology (HOT) Fit Model pada Universitas Kristen Satya Wacana,” *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, vol. 9, no. 3, hlm. 2096–2111, Sep 2022, doi: 10.35957/jatisi.v9i3.2201.
- [24] I. Jahan, M. N. Islam, M. M. Hasan, dan M. R. Siddiky, “Comparative analysis of machine learning algorithms for sentiment classification in social media text,” *World Journal of Advanced Research and Reviews*, vol. 23, no. 3, hlm. 2842–2852, 2024, doi: 10.30574/wjarr.2024.23.3.2983.
- [25] P. Vickers, M. Bohn, Y. He, dan R. McConville, “We Need to Talk About Classification Evaluation Metrics in NLP,” *arXiv preprint arXiv:2401.03831 (Proceedings of ACL)*, 2024, doi: 10.48550/arXiv.2401.03831.